TY DIPLOMA - VIMP QUESTIONS
ADVANCE JAVA PROGRAMMING

Unit :01
Abstract Windowing Toolkit (AWT)

1) The various controls supported by AWT are
   a) Labels, push buttons
   b) Checkboxes, choice, list
   c) Scroll bars, text area, text field
   d) **All of these**

2) The concept of the menu bar can be implemented by using three java classes—
   a) MenuBar
   b) Menu
   c) MenuItem
   d) **All of these**

3) The constructor which the Text Event class defines.
   a) **TextEvent(Object source, int event_type)**
   b) textevent (Object source, int event_type)
   c) textevent (object Source, float event_type)
   d) textevent (Object source, string event_type)

4) In Java an event is an _____which specifies the change of state in the source.
   a) Class
   b) Object
   c) Int
   d) String

5) The classes and interfaces defined in AWT are contained within the _____ package.

a) java.awt.*

b) java.sql.*

c) java.io.*

d) java.int*

6) The general form to set a specific type of layout manager is

a) void setLayout(LayoutManager lm)

b) Void setLayout(LayoutManager lm)

c) void setLayout(layoutManager lm)

d) Void setLayout(Layoutmanager lm)

7) The AWT container is an instance of the _____ class which holds various components and other containers

a) Graphics

b) **Container**

c) Eventobj

d) None of these

8) A checkbox is a control that consists of a

a) Combination of a small box

b) A label

c) Combination of a large box and a label

d) **Both a & b**

9) Java applets are used to create _____ applications

a) Graphical

b) User interactive

c) **Both a & b**

d) None of these

10) AWT means

a) **Abstract Window Toolkit**

b) Abstract Window Toollayout

c) Abstract Withdraw Tools

d) Abstract Window Title

11) An event is generated when the internal state of the event source is_____

a) Not changed

b) Changed

c) Either changed or not

d) None of these

12) Positions the components into five regions:east, west, north, south, center

a) **BorderLayout**

b) CardLayout

c) GridLayout

d) FlowLayout

13) Arranges the components as a deck of cards such that only one component is visible at a time

a) BorderLayout

b) **CardLayout**

c) GridLayout

d) FlowLayout

14) Arranges the components horizontally

a) BorderLayout

b) CardLayout

c) GridLayout

d) **FlowLayout**

15) Arranges the componemnts into grid

a) BorderLayout

b) CardLayout

c)GridLayout

d) FlowLayout

16) _____ creates a dropdown list of textual entries

a) Choice

b) Checkbox

c) Textbox

d) TextComponent

17) The Component class is an abstract class and so its _____ are used to create components.

a) Subclasses

b)Superclasses

c) Both a & b

d)None of these

18) The AWT classes can be roughly categorized into the following groups:

a) GUI Components

b) Layouts

c) Graphics Tools

d) Event Handlers

e) **All of these**

19) An Applet is a _____ of Panel:

a) Subclass

b) Superclass

c) Both a & b

d) None of these

20) The subclasses of Window are

a) Dialog

b) Frame

c) **Both a & b**

d) None of these

21) A menu bar represents

a) A list of menus which can be added to the top of a top-level window

b) A list of menus which can be deleted to the top of a top-level window

c) A list of menus which can be added to the bottom of a bottom-level window d) None of these

22) Each menu is associated with a _____ list of menu items:

a) Checkbox

b) **Drop-down**

c) Choice

d) None of these

23) The two types of menus which are given as follows:

a) Pop-up menus

b) Regular menus

c) **Both a & b**

d) None of these

24) Regular menus are placed at the _____ of the application window within a menu bar

a) Top

b) Bottom

c) Top-down

d) Botttom-up

25) The text field and text area controls create a _____ area respectively

a) Single-line text

b) Multi-line text

c) **Both a & b**

d) None of these

26) A push button is an active control that has a _____ appearance

a) One dimensional

b) Two dimensional

c) **Three dimensional**

d) None of these

27) _____ is a superclass of TextField and TextArea classes that is used to create single-line or multiline textfields respectively:

a)TextBox

b)CheckBox

c)TextComponent

d)Choice

28) A label is a simple control which is used to display_____ on the window:

a) Text(non-editable)

b) Text(editable)

c) Both a & b

d) None of these

29) _____ is an abstract class that encapsulates all of the attributes of a visual component.

a)Component

b)Window

c)Frame

d) Panel

30) A ____object is responsible for remembering the current foreground and background colors

a)Window

b)Component

c)None

d)Both

31) Which class is a subclass of Component?

a) Container

b) Window

c) Frame

d) none

32)The _____ class is a concrete subclass of Container.

a) Container

b) Window

c) **Panel**

d) None

33)Panel is a superclass for _____.

a) Window

b) Frame

c) **Applet**

d) None

34) Which is a container that does not contain a title bar, menu bar, or border?

a)Window

b)Frame

c)Panel

d)Container

35)Name the class used to represent GUI component that has a title bar, menu bar, borders, and resizing corners.

a) Window

b) Container

c) **Frame**

d) Panel

36) How many Frame constructor are present?

a) 1

b) 2

c) 3

d) 4

37) Which method is used to set title for the Frame window?

a)Frame()

b)Frame(String title)

c) Frame(String title,int x,int y)

d) None

38) Which method is used to set the dimensions of the window?

a)setSize()

b)void setSize(int newWidth, int newHeight)

c)void setSize(Dimension newSize)

d)**All of the above**

39) The ____ method is used to obtain the current size of a window.

a) getSize()

b)setSize()

c)None

d)Both

40)Which method is used for hiding and showing a window in the frame window.

a)setVisible()

b)getVisible( )

c)both

d)None

41) To implement the windowClosing() method which interface is used?

a) ActionListener

b) ItemListener

c) WindowListener

c) None

42) Which of these packages contains all the classes and methods required for event handling

a) java.awt.Applet;

b) **java.awt.event;**

c) java.awt;

d) java.event;

43) On invoking repaint() method for a Component the method invoked by AWT is:

a)draw( )

b)show( )

c)update( )

d) paint( )

44) Which of these events will be generated if we close the applet?

a)ActionListener

b)ItemListener

c)MouseListener

d)WindowListener

45) _____ encapsulates a semantics-free window.

a)Frame

b)Panel

c)Canvas

d)None

46)The class at the top of the AWT hierarchy

a)Component

b)Frame

c)Window

d)Panel

47)Which class dispatches events to multiple listener

a)AWTEvent

b)AWTEventMulticaster

c)AWTEventManager

d) All of the Above

48) Add instance of the desired control to the window by calling ____ method, defined by ____ class.

a)add() , Container

b)add() , Component

c)addTo() , Container

d)addTo(), Component

49)To remove a control from a window ___ method is used.

a)removeControl()

b)remove()

c)removeAll()

d)None of the above

50)You can remove all controls by calling method.

a)remove()

b)removeFrom()

c)Remove()

d)removeAll( )

51)Label defines the following constructors:

a)Label( )

b)Label(String str)

c)Label(String str, int how)

d)All of the above

52)In the Label(String str, int how) constructor value of how will be-

a)Label.LEFT, Label.RIGHT, or Label.CENTER

b)Label.LEFT, Label.RIGHT, or Label.DOWN

c)Label.TOP, Label.BOTTOM, or Label.CENTER

d)Label.TOP, Label.BOTTOM, or Label.DOWN

53)To set or change the text in a label __ method and to obtain the current label by calling ___ method is used.

a)setTxt() , getTxt()

b)settext() , gettext()

c)setText() , getText()

d)getText() , setText()

54)To set the alignment of the string within the label by calling ___ method ,to obtain the current alignment ___ method is used.

a)getAlignment( ) ,setAlignment( )

b)setAlign( ) ,getAlign( )

c)setAlignment( ) ,getAlignment( )

d)getAlign( ) ,setAlign( )

55) Which packages will use for the following code:

/*

<applet code="LabelDemo" width=300 height=200>

</applet>

```
*/
public class LabelDemo extends Applet { public void
init() {
Label one = new Label("One");
Label two = new Label("Two");
Label three = new Label("Three"); //
add labels to applet window
add(one); add(two); add(three);
}
}
```

a) **import java.awt.*; import java.applet.*;**

b) import javax.swing.*; import java.awt.*;

c) import javax.swing.*; import java.applet.*;

d) d)import java.applet.*; import java.awt.event.*;

56) Button defines these two constructors:

a) Button( ), Button(Boolean str)

b) Button( ) ,Button(int str)

c) **Button( ), Button(String str)**

d) Button( ) ,Button(String str , int name)

57) Set label of a button by calling __ method and can retrieve its label by calling __ method.

a) getLabel( ) , setLabel( )

b) **setLabel( ) , getLabel( )**

c) getlabel( ) , setlabel( )

d) d)setlabel( ) , getlabel( )

58) Each time a button is pressed, an ___ is generated.

a) mouse event

b) Keyboard event

c) item event

d) action event

59) For button __ interface is implemented.

a)ActionListener

b)ItemListener

c)MouseListener

d)FocusListener

60) actionPerformed() method is used in ___ interface.

a)ActionListener

b)ItemListener

c)MouseListener

d)FocusListener

61)Which class can be used to represent a checkbox with a textual label that can appear in a menu.

a)MenuBar

b) MenuItem

c)CheckboxMenuItem

d)Menu

62) To retrieve the current state of a check box, call __ method , to set its state, call __ method.

a) setState( ) , getState( )

b) **getState( ) , setState( )**

c) setstate( ) , getstate( )

d) getstate( ) , setstate( )

63) Event handling in checkbox is done by __ listener and __ object is used.

a)ItemListener , ItemEvent

b)MouseListener , MouseEvent

c)ActionListener , ActionEvent

d)KeyListener , KeyEvent

64) Which method is defined by the ItemListener interface?

a)actionPerformed()

b)ActionPerformed()

c)itemStateChanged()

d)ItemstateChanged()

65) Choose the correct:

a)public class CheckboxDemo implement Applet extend ItemListener

b)public class CheckboxDemo extends Applet implements ItemListener

c)public class CheckboxDemo implements Applet extends ItemListener

d)public class CheckboxDemo extend Applet implement ItemListener

66) .It is possible to create a set of mutually exclusive check boxes in which one and only one check box in the group can be checked at any one time by using __ component.

a) **CheckboxGroup**

b) Radio Button

c) Checkbox

d) Choice

67) You can determine which checkbox in a group is currently selected by calling __ method.

a)getselectedCheckbox( )

b)GetSelectedCheckbox( )

c)getSelectedCheckbox( )

d)None of the above

68)To set a checkbox which method is used-

a)setselectedCheckbox( )

b)setSelectedCheckbox( )

c)getselectedCheckbox( )

d)getSelectedCheckbox( )

69)In which of the following the only one checkbox will be selected.

a)Checkbox Win98 = new Checkbox("Windows 98/XP", cbg, false);

b)Checkbox Win98 = new Checkbox("Windows 98/XP", cbg, true);

c)Checkbox Win98 = new Checkbox("Windows 98/XP", true);

d)Checkbox Win98 = new Checkbox("Windows 98/XP", false);

70) Which class is used to create a pop-up list of items from which the user may choose.

a) **Choice**

b) List

c) Checkbox

d) CheckboxGroup

71) Method used to add items in a choice-

a)addItem()

b)additem()

c)Add()

d)add()

72)To determine which item is currently selected, you may call either ___ or ____ method.

a)setSelectedItem( ),setSelectedIndex( )

b)GetSelectedItem( ),GetSelectedIndex( )

c)getSelectedItem( ),getSelectedIndex( )

d)getselectedItem( ),getselectedIndex( )

73)The getItemCount( ) method is used to-

a)To obtain the value of items in the list

b)To obtain the number of items in the list

c)Both a & b

d)None of the above

74) Constuctors of scrollbar are-

a) Scrollbar( ) ,Scrollbar(int style) ,Scrollbar(int style, int initialValue, int thumbSize)

b) Scrollbar( ), Scrollbar(int style) ,Scrollbar(int style, int initialValue, int thumbSize, int min

c) Scrollbar( ),Scrollbar(int style),Scrollbar(int style, int initialValue, int thumbSize, int max)

d) **Scrollbar( ),Scrollbar(int style),Scrollbar(int style, int initialValue, int thumbSize, int min, int max)**

75) Scrollbar uses which two constants to create horizontal and vertical scrollbar.

a)Scrollbar.Vertical , Scrollbar.Horizontal

b)Vertical.SCROLLBAR , Horizontal.SCROLLBAR

c)Scrollbar.VERTICAL , Scrollbar.HORIZONTAL

d)None of the above

76) To obtain the current value of the scroll bar, call ___, to set the current value, call __ method.

a) setvalue( ), getvalue( )

b) setValue( ) , getValue( )

c) getvalue( ). setvalue( )

d) **getValue( ) , setValue( )**

77) You can retrieve the minimum and maximum values of scrollbar by __ and___ method

a)getMinimum( ), getMaximum( )

b)getMax(), getMin()

c)setMinimum(), setMaximum()

d)setMax(), setMin()

78) Which interface is implemented for handling scrollbars. a)ActionListener

b)AdjustmentListener

c)MouseMotionListener

d)ItemListener

79) The_____ class implements a single-line text-entry area

a)TextArea class

b)TextField class

c)both a & b

d)none of the above

80) Which of these is not aTextField Constructor

a)TextField( )

b)TextField(intnumChars)

c)TextField(int rows)

d)TextField(Stringstr)

81) To obtain the text currently in the text field, which method is used?

a) getWord()

b) getString()

c)**getText()**

d) getRow()

82) Program can obtain the currently selected text by calling _____

a) **getSelectedText()**

b) getText()

c) getSelected()

d) getEdit()

83) Contents of a text field may be modified by the user by calling

a) **setEditable( )**

b) getEditable()

c) isEditable()

d) None of the above

84) The echoing of the characters as they are typed by calling

a) setPassword()

b) **setEchoChar( )**

c) hideText()

d) setChar()

85) The AWT includes a simple multiline editor called

a)TextField

b)TextArea

c)Editor

d)Label

86)Which of the following is not a constructor of TextArea

a)TextArea(String str)

b)TextArea(intnumLines, intnumChars)

c)TextArea(Stringstr, intnumLines, intnumChars, intsBars)

d)TextArea(Stringstr, intsrows)

87) Which methods is not supported by TextArea

a) getText()

b) **setFormat()**

c) setText()

d) SetEditable()

88) The_____method appends the string specified by str to the end of the current text.

a)append()

b)insertText()

c)attach()

d)editText()

89) ____ method inserts the string passed in str at the specified index.

a)append()

b)attachText()

c)insert()

d)join()

90) To replace a text, which method is called?

a) replaceText()

b) changeText()

c) editText()

d) replaceRange()

91) Each _____ object has a layout manager associated with it.

a) Applet

b) Frame

c) Panel

d) **Container**

92) A layout manager is an instance of any class that implements the _____ interface.

a) **LayoutManager**

b) ActionListener

c) ItemListener

d) MouseListener

93) The layout manager is set by which method.

a) setText()

b) getText()

c) **setLayout( )**

d) setVisible()

94) Which method is used to determine position and shape of a component manually

a) setBounds()

b) setPosition()

c) **Both a and b**

d) None

95) Which of the following LayouManager is/are consulted whenever the container needs to be resized

a) minimumLayoutSize( )

b) preferredLayoutSize( )

c) **Both**

d) None

96) _____ and _____ are contained by each Layout manager

a) getPrefferedSize(),getMinimumSize( )

b) getPrefferedsize(),getMinimumsize( )

c) getprefferedSize(),getminimumSize( )

d) None

97) Which of these is the default Layout Manager

a) FlowLayout()

b) BorderLayout()

c) GridLayout()

d) CardLayout()

98) Constructors of FlowLayout

a) FlowLayout( )

b) FlowLayout(int how)

c) FlowLayout(int how, int horz, int vert)

d) **All of the above**

99) In the constructor FlowLayout(int how, int horz, int vert) what is the value of how

a) FlowLayout.LEFT,FlowLayout.CENTER,FlowLayout.RIGHT

b) FlowLayout.TOP,FlowLayout.BOTTOM,FlowLayout.CENTE

c) FlowLayout.EAST,FlowLayout.WEST, FlowLayout.CENTER

d) None

100) Constructors of BorderLayout

a) BorderLayout( ),

b) BorderLayout(int horz, int vert)

c) BorderLayout(int how, int horz, int vert)

d) Both a and b

101) Constants of BorderLayout

a) BorderLayout.CENTER,BorderLayout.SOUTH,BorderLayout.EAST, BorderLayout.WEST, BorderLayout.NORTH

b) BorderLayout.CENTER, BorderLayout.TOP, BorderLayout.BOTTOM, BorderLayout.LEFT, BorderLayout.RIGHT

c) BorderLayout.CENTER, BorderLayout.LEFT, BorderLayout.RIGHT,

d) None of the above

102) GridLayout lays out components in a _____ grid.

a) One-dimensional

b) Three-dimensional

c) Multi-dimensional

d) **Two-dimensional**

103) Construcors of GridLayout

a) GridLayout( )

b) GridLayout(int numRow,numColumns)

c) GridLayout(int numRows, int numColumns, int horz, int vert)

d) **All of the Above**

104) The CardLayout class is _____ among the other layout managers in that it stores several different _____.

a) unique,classes

b) **unique,layout**

c) antique,methods

d) special,packages

105) Constructors of CardLayout are:

a) CardLayout( ), CardLayout(int horz, int vert)

b) CardLayout( ),CardLayout(inthorz,intvert), CardLayout(int numRows, int numColumns, int horz, int vert)

c)Both

d)None

106) The cards are held in an object of type _____

a) Frame

b) Applet

c) **Panel**

d) Container

107) Methods of CardLayout void

first(Container deck) void

last(Container deck) void

next(Container deck) void

previous(Container deck)

void show(Container deck, String cardName)

a)All

b)only first 2

c)Both

d)None

108) Dialog box maybe _____ or

_____

a) Fixed,Variable

b) static,dynamic

c) manual,automated

d) **modal,modeless**

109) Which of these is true of modal dialog box

a) **You cannot access other parts of your progra**m until you have closed the dialog box.

b) Input focus can be directed to another window in your program.

c) Both a and b

d) None

110) Which of these is true of modeless dialog box

a) You cannot access other parts of your program until you have closed the dialog box.

b) **Input focus can be directe**d to another window in your program.

c) Both a and b

d) None

111) Constructors of dialog box

a) Dialog(Frame parentWindow, boolean mode), Dialog(Frame parentWindow, String title, boolean mode)

b) Dialog(Frame parentWindow, boolean mode) ,Dialog(Frame parentWindow, String title, boolean mode, int horz, int vert)

c) Dialog(Frame parentWindow, boolean mode), Dialog(Frame parentWindow, String title, boolean mode,int rows,int column)

d) Dialog(Frame parentWindow, boolean mode) ,Dialog(Frame parentWindow, String title)

112) To create a file dialog box, instantiate an _____ of type FileDialog.

a) Method

b) Class

c) **Object**

d) Package

113) Constructor of FileDialog

a) FileDialog(Frame parent, String boxName), FileDialog(Frame parent, String boxName, int how,int horz,int vert), FileDialog(Frame parent) b) FileDialog(Frame parent, String boxName),

c) Fi**leDialog(Frame parent, String boxName) ,FileDialog(Frame parent, String boxName, int how) FileDialog(Frame parent**)

d)FileDialog(Frame parent, String boxName),FileDialog(Frame parent, String boxName, int how)

114) For the file to be in reading mode which method is used

a) FileDialog.SAVE

b) FileDialog.WRITE

c) **FileDialog.LOAD**

d) FileDialog.READ

115) For the file to be in writing mode which method is used

a) **FileDialog.SAVE**

b) FileDialog.WRITE

c) FileDialog.LOAD

d) FileDialog.READ

116) Select the correct Menubar classes from the following options

a) MenuBar, Menu, MenuItem

b) menubar, menu, menuitem

c) Both

d) None of these

117) Which menu option of types will have a checkmark next to them when they are selected?

a) CheckboxMenuItem

b) CheckedItem

c) MenuItem

d) None of these

118) You can disable or enable a menu item by using the _____method.

a) setStatus()

b)**setEnabled( )**

c) setMenuStatus()

d)None of these

119) Which of these constructor throws HeadlessException?

a) CheckboxMenuItem()

b) CheckboxMenuItem(String itemName)

c) CheckboxMenuItem(String itemName, boolean on)

d) **All of these**

120) Which sets the command name of the action event that is fired by this menu item?

a) **setActionCommand()**

b) setMenuCommand()

c) Both

d) None of these

121) To check an item, pass ____ to _____.

a) **true, setState()**

b) check, setItemStatus()

c) check, setItemState()

d) None of these

122) Constuctors of checkbox are -

a) Checkbox( )

b) Checkbox(String str)

c) Checkbox(String str, boolean on, CheckboxGroup cbGroup)

d) **All of the above**

123) import
java.awt.*; import
java.applet.*; import
java.util.*;

```
/*
<applet code="BorderLayoutDemo" width=400 height=200>
</applet>
*/
public class BorderLayoutDemo extends Applet
{ public void init() {
add(new Button("north."),BorderLayout.NORTH);
add(new Button("south"),BorderLayout.SOUTH);
add(new Button("Right"), BorderLayout.EAST);
add(new Button("Left"), BorderLayout.WEST); String
msg = "this is in center"; add(new TextArea(msg),
BorderLayout.CENTER); }
}
```

a) SetLayout(new BorderLayout());
b) **setLayout(new BorderLayout());**
c) setLayout(new Borderlayout());
d) setLayout(new borderLayout());

124) What should be written in blank space.

```
import java.awt.*; import
java.awt.event.*; import
java.applet.*;
/*
<applet code="CBGroup" width=250 height=200>
</applet>
*/
public class CBGroup extends Applet
{
String msg = "";
Checkbox Win98, winNT; CheckboxGroup
cbg;
public void init()
{
cbg = new CheckboxGroup();
Win98 = new Checkbox("Windows 98/XP",      , true); winNT = new
Checkbox("Windows NT/2000",      , false);
add(Win98);
add(winNT);
```

```
Win98.addItemListener(this); winNT.addItemListener(this);
}
}
```

a) Win98
b) winNT
c)cbg
d)this

125) Find error in following code.

```
import java.awt.*; import
java.awt.event.*;
import java.applet.*;

public class ChoiceDemo extends Applet
{
Choice os; String
msg = ""; public
void init()
{
os = new Choice();

// add items to os list os.add("Windows
98/XP");
os.add("Windows NT/2000");

add(os);

}
```

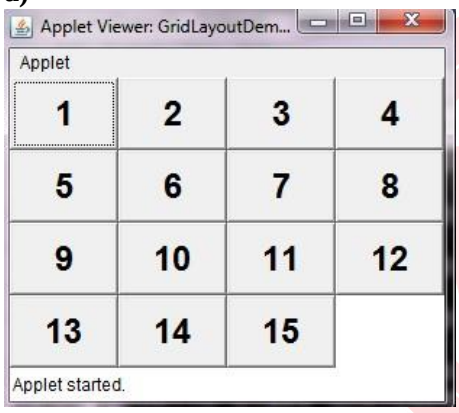
a)Listener missing
b)applet code is missing
c)package missing
d)All

126) **What will be the output for following code?**

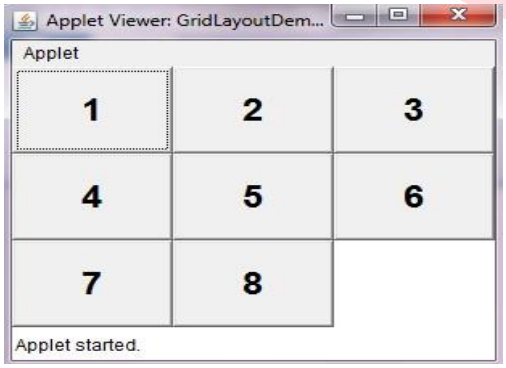
```
import java.awt.*; import java.applet.*;
/*
<applet code="GridLayoutDemo11" width=300 height=200>
</applet>
*/
```

```
public class GridLayoutDemo11 extends Applet {
static final int n = 4; public void init() {
setLayout(new GridLayout(n, n)); setFont(new
Font("SansSerif", Font.BOLD, 24));
for(int i = 0; i < n; i++) {
for(int j = 0; j < n; j++) {
int k = i * n + j; if(k > 0)
add(new Button("" + k));
}
}
}
}
```

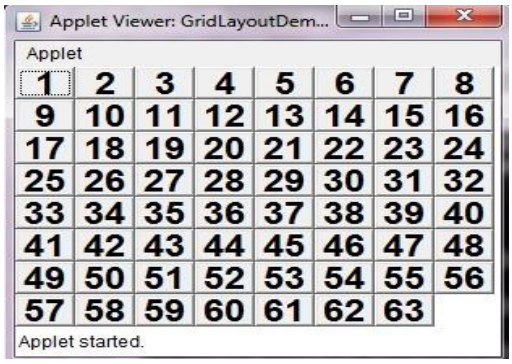
a)



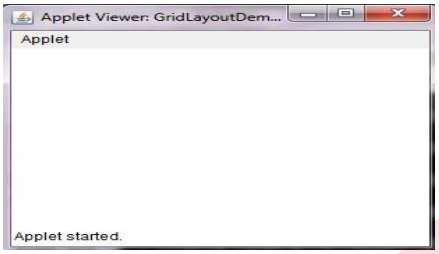
b)



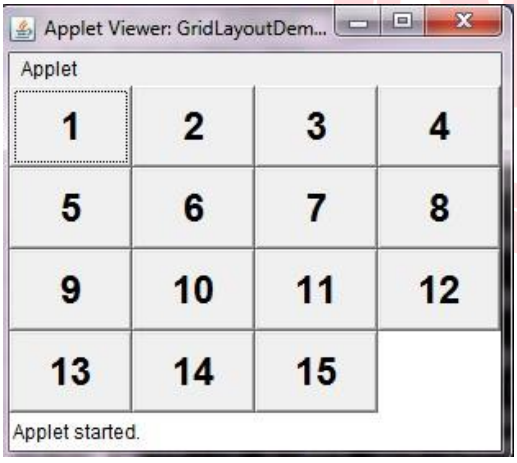
c)



d)



Answer: a)



127) Consider the following program. Find the error.

```
immport java.awt.*; import java.applet.*; import
java.awt.event.*;
/*<applet code=demo width=100 height=100>
</applet>  */

public class demo extends Applet
{
```

```
public void init()
{ firstlabel =new Label("Label 1");
secondlabel =new Label("Label 2");
b1=new Button("Enter");
add(l1); add(l2);
add(b1);

} }
```

a) Firstlabel object is not declared
b) Secondlabel object is not declared
c) b1 object is not declared
d) **All of above**

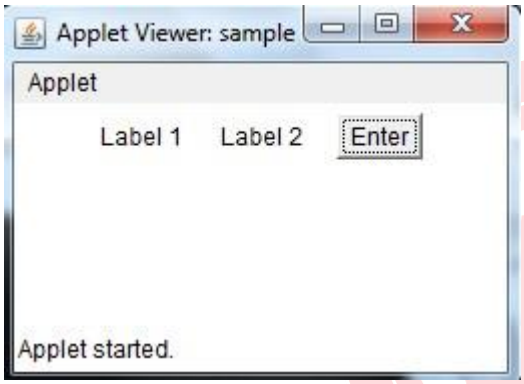
128) What will be the missing statement in java to get following output:


```
import java.awt.*; import
java.applet.*;
/*<applet code=fontc width=500 height=500>
</applet> */ public class fontc
extends Applet
{
public void init()
{
Font f=new Font("Times New Roman",Font.ITALIC,30)
setFont(f); }
```

```
public void paint(Graphics g)
{
g.setColor(Color.red);
g.drawString("SAGAR",10,50);
}
}
```

A. Missing {
B. Missing }
C. **Missing semicolon**
D. Missing ()

129) What is the code to get the following output:



a)
```
import java.awt.*;
import java.applet.*; import
java.awt.event.*;
/*<applet code=sample width=100 height=100>
</applet>
*/
public class sample extends Applet
{
Label l1,l2;
Button b1;
String msg="";
public void init()
{
l1=new Label("Label 1"); l2=new
Label("Label 2"); b1=new
Button("Enter");
add(l1); add(l2);
add(b1);
}
}
```

b) import java.awt.*;
import java.applet.*; import
java.awt.event.*;
/*<applet code=sample width=100 height=100>
</applet>
*/
public class sample extends Applet
{
Label l1;
Button b1;
String msg=""; public void
init() { l1=new
Label("Label 1"); b1=new
Button("Enter"); add(l1);

add(b1); }
}

c) import java.awt.*;
import java.applet.*; import
java.awt.event.*;
/*<applet code=sample width=100 height=100>
</applet>
*/
public class sample extends Applet
{
Label l1,l2;
Button b1;
String msg=""; public void
init() { l1=new
Label("Label 1"); l2=new
Label("Label 2");

add(l1);
add(l2); }
}

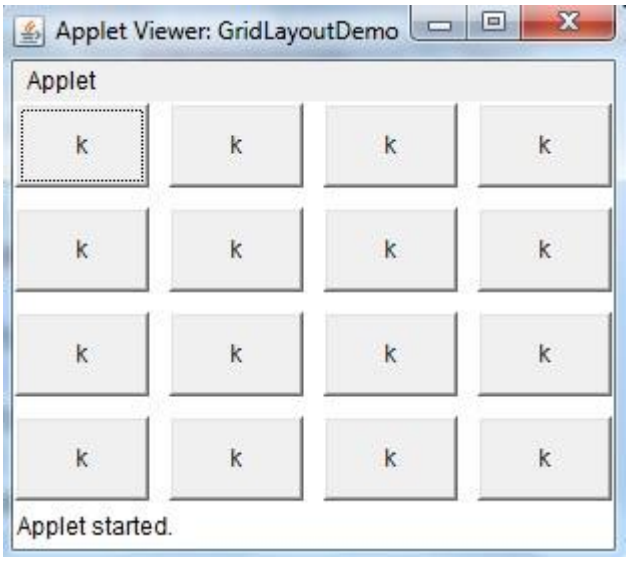
d) import java.awt.*;
import java.applet.*; import
java.awt.event.*;

```
/*<applet code=sample width=100 height=100>
</applet>
*/
public class sample extends Applet
{
Label l1,l2;
Button b1;
String msg=""; public void
init() { l1=new
Label("Label 1"); l2=new
Label("Label 2"); b1=new
Button("Enter"); add(l1);
add(l2);
add(b1);
}
}
```

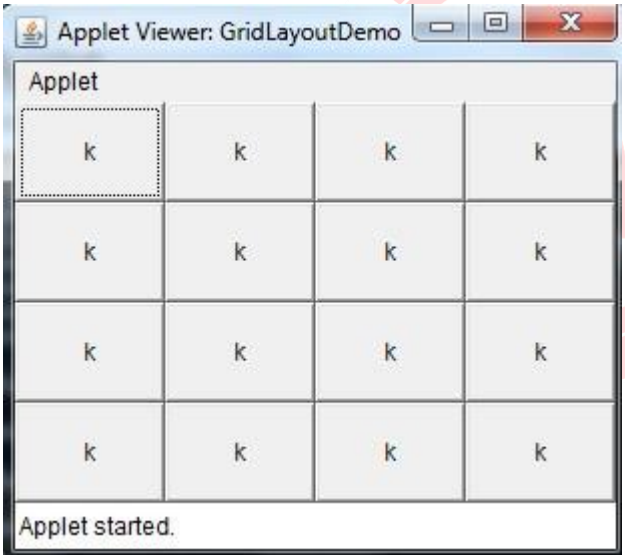
130) What is the output of the following code:

```
import java.awt.*; import
java.applet.*;
/*
<applet code="GridLayoutDemo" width=300 height=200></applet>
*/
public class GridLayoutDemo extends Applet
 {  int n = 4;
public void init()
{
setLayout(new GridLayout(n,n,10,10));

for(int i = 0; i <n; i++)
{
for(int j = 0; j <n; j++)
{
add(new Button("k"));
}
}
} }
```

a)



b)



c)none of the above
d)all of these

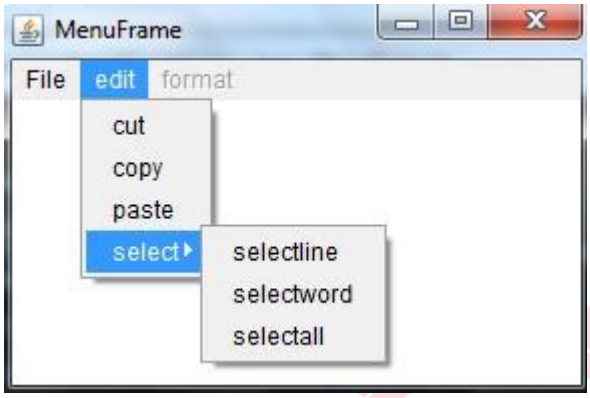
131) What will be the output of following code

```
import java.awt.*; import
java.awt.event.*; import
java.applet.*;
```

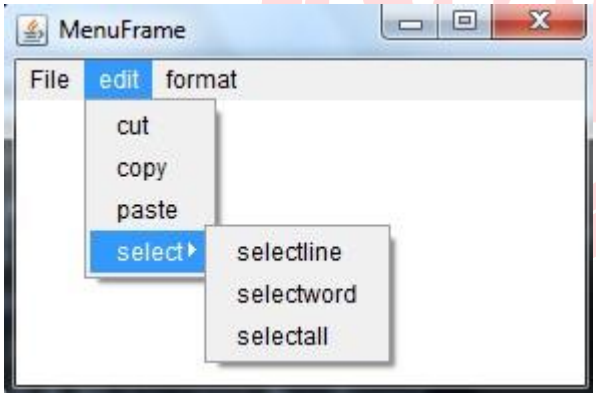
```
class MenuFrame1 extends Frame
{
String msg = "";
MenuBar mbar;
MenuItem copy,paste,selectline,selectword,selectall;
CheckboxMenuItem open,cut;
TextField t1; Menu
file,edit,select,format;
public MenuFrame1()
 {  mbar =new MenuBar();
setMenuBar(mbar); file =
new Menu("File");
 open = new CheckboxMenuItem("open");
 file.add(open); mbar.add(file); edit=
 new Menu("edit"); cut = new
 CheckboxMenuItem("cut"); copy = new
 MenuItem("copy");
paste = new MenuItem("paste");     edit.add(cut);
 edit.add(copy);  edit.add(paste);  select =new
 Menu("select");  selectline = new
 MenuItem("selectline");   selectword = new
 MenuItem("selectword");  selectall = new
 MenuItem("selectall");
 selectall.setEnabled(false);
 select.add(selectline);  select.add(selectword);
 select.add(selectall);  edit.add(select);
 mbar.add(edit);  format =new
 Menu("format");  format.setEnabled(false);
 mbar.add(format);
 addWindowListener(new MyWindowAdapter1());

 }
 public static void main(String[] args)
 {
 MenuFrame1 mf =new MenuFrame1();
 mf.setTitle("MenuFrame");
 mf.setSize(300,200); mf.setVisible(true);
 }

 class MyWindowAdapter1 extends WindowAdapter {
   public void windowClosing(WindowEvent we) {
```

```
    System.exit(0);
  }
}
}
```

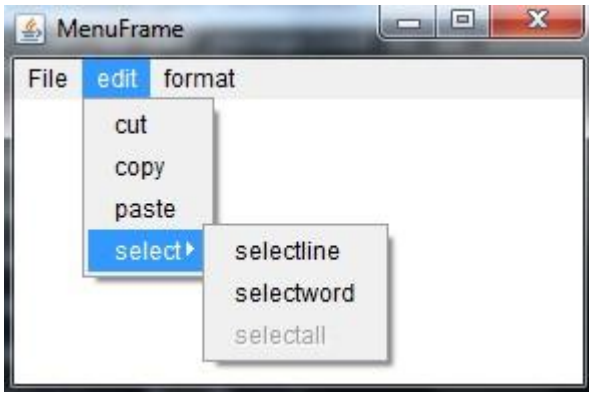
a)



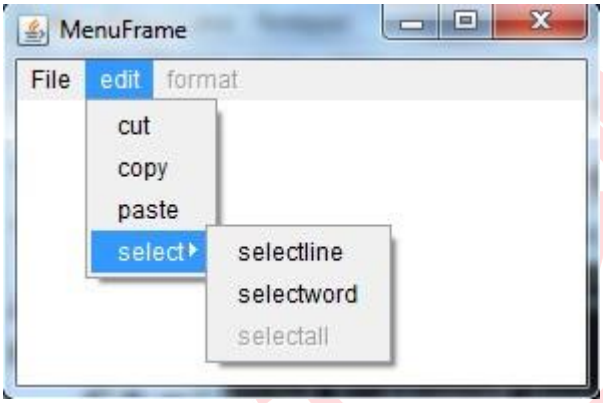
b)



c)



d)



132) Which constructor creates a TextArea with 10 rows and 20 columns?

a) **new TextArea(10, 20)**

b) new TextArea(20, 10)

c) new TextArea(new Rows(10), new columns(20))

d) new TextArea(200)

133) Which of the following creates a List with 5 visible items and multiple selection enabled?

a) **new List(5, true)**

b) new List(true, 5)

c) new List(5, false)

d) new List(false,5)

134) Which method will cause a Frame to be displayed?

a) show( )

b) setVisible( )

c) display( )

d) displayFrame( )

e) **both a and b**

135) The Choice component allows multiple selection.

a) True

b) **False**

136) The List component does not generate any events.

a) True

b) **False**

137) Which of the following components allow multiple selections?

a) Non-exclusive Checkboxes

b) Radio buttons

c) Choice

d) List

e) **Both a and d**

138) Which containers use a BorderLayout as their default layout?

a) Window

b) Frame

c) Dialog

d) **All of above**

139) Which containers use a FlowLayout as their default layout?

a) Panel

b) Applet

c) **both a and b**

d) only d

140) Which method returns the preferred size of a component?

a) **getPreferredSize( )**

b) getPreferred( )

c) getRequiredSize( )

d) getLayout( )

141) Which layout should you use to organize the components of a container in a tabular form?

a) CardLayout

b) BorederLayout

c) FlowLayout

d) **GridLayout**

142) What is the default layouts for a applet, a frame and a panel?

a)**Flow layout, Border layout, Flow layout**

b)Flow layout, Flow layout, Border layout

c)Border layout, Flow layout, Flow layout

d)Border layout,Border layout, Flow layout

143) An Applet has its Layout Manager set to the default of FlowLayout. What code would be the correct to change to another Layout Manager?

a)setLayoutManager(new GridLayout());

b)setLayout(new GridLayout(2,2));

c)setGridLayout(2,2,))

d)setBorderLayout();

144) Which is a dual state menu item?

a) CheckboxMenuItem

b)Menu

c)MenuItem

d) All of above

145) Which method can be used to enable/diable a checkbox menu item?

a)setState(boolean)

b) setstate(boolean)

c)setEnabled(boolean)

d)setenabled(boolean)

146) Which of the following may a menu contain?

a)A separator

b) A check box

c) A menu

d) A button

e)both a and c

Answer Key

Unit : 02

Swings

1) Swing is a set of classes that provides more _____ and _____components than are possible with the AWT

a) Powerful , Flexible

b) Great, Awesome

c) Better , Powerful

d) Stable, customizable

2) Swing components are not implemented by ____ code.

a) **platform-Specific**

b) Platform- independant

c) Both

d) none

3) Swing Components are _____ Weight

a) **Light**

b) Moderate

c) Heavy

d) d)All of the Above

4) Super Class For Swing Buttons

a)AbstractButton b)Button c)Jbutton d)none

5) Which class encapsulates a mutually exclusive set of buttons.

a) MutualButtonGroup

b) **ButtonGroup**

c) JButtonGroup

d) ButtonsGroup

6) Which class encapsulates an icon.

a) Image

b) Icon

c) IconImage

d) ImageIcon

7) Which class represents the Swing version of Applet.

a) SwingApplet

b) SApplet

c) **JApplet**

d) JSApplet

8) Which is Swing push button class.

a) JPushButton

b) PushButton

c) Jbutton

d) None

9) Which is the Swing check box class.

a) JCheckbox

b) JSCheckbox

c) **JCheckBox**

d) None

10) JApplet is ___ with functionality when compared with Applet

a) **Rich**

b) poor

c) Same

d) different

11) Which class encapsulates a Swing combo box

a) JCombobox

b) jCombobox

c) **JComboBox**

d) none

12) Which package needs to be imported for using Swing Classes

a) java.swing

b) java.applet.swing

c) **javax.swing**

d) java.lang.swing

13) The Swing version of a label.

a) SwingLabel

b) Label

c) Jlabel

d) none

14) The method used to add components to a Container

a) **add()**

b) Insert()

c) addComponent

d) InsertComponent

15) The Swing version of a radio button.

a) **JRadioButton**

b) JRadiobutton

c) JradioButton

d) none

16) Which class Encapsulates a scrollable window in Swing

a) JScrollableWindow

b) JScrolllablewindow

c) JscrollableWindow

d) JScrollPane

17) The Components in JApplet are added to

a) Current JApplet Class

b) **ContentPane**

c) both

d) none

18) Which class encapsulates a tabbed window.

a) JTabbedWindow

b) jTabbedWindow

c) JtabbedWindow

d) none

19) Class which Encapsulates a table-based control

a) JControlTable

b) **JTable**

c) JcontrolTable

d) JControltable

20) The Swing classes of a text field.

a) **JTextField**

b) JInputField

c) Both

d) None

21) For using JTree one has to import javax.swing.tree seperately as only javax.swing.* does not works

a) **True**

b) Partially True

c) False

d) Partially False

22) Constructors for using Icon and Label In Swing

a) ImageIcon(String filename)

b) ImageIcon(URL url)

c) **Both**

d) None

23) int getIconHeight( ) Returns the height of the icon as

a) centimetres

b) millimetres

c) pixels

d) none

24) The Constructor used to create a JTextField with predefined Text

a) **JTextField(String s, int cols)**

b) void setText(String)

c) both

d) none

25) JRadioButton class, which is a concrete implementation of _____

a) Container

b) **AbstractButton**

c) Both

d) none

26) Items are added to the JComboBox using _____ method

a) add()

b) **addItem()**

c) addOption()

d) none

27) For a JComboBox Constructor which can be the arguments passed

a) Array

b) **Vector**

c) Both

d) None

28) Which are the methods uesd for JLabel

Icon getIcon( )

String getText( )

void setIcon(Icon i)

void setText(String s)

a) **All**

b) Only Icon getIcon( ) and String getText( )

c) Both

d) None

29) Constructors for JTextField

a)

JTextField( )

JTextField(int cols)

JTextField(String s, int cols)

JTextField(String s)

b)Only JTextField( ) and JTextField(int cols)

c) both

d) none

30) Constructors for JCheckBox

a)

JCheckBox(Icon i)

JCheckBox(Icon i, boolean state)

JCheckBox(String s)

JCheckBox(String s, boolean state)

JCheckBox(String s, Icon i)

JCheckBox(String s, Icon i, boolean state)

b)Only JCheckBox(Icon i) and JCheckBox(Icon i, boolean state)

c)both

d)none

31) For JLabel alignment Constants are

a) LEFT, RIGHT, CENTER, LEADING, TRAILING

b) Only LEFT, and RIGHT,

c) Both

d) None

32) Constructors for JLabel

a) JLabel(Icon i) ,Label(String s),JLabel(String s, Icon i, int align)

b) Only JLabel(Icon i) and Label(String s)

c) Both

d) done

33) Components such as buttons have _____ capabilities in Swing.

a) **More**

b) Less

c) Equal

d) No

34) An Image can be changed as the state of the button changes. True or False?

a) **True**

b) False

35) What is an AbstractButton class?

a) Abstract Superclass for Swing

b) Abstract subclass for Swing

c) Abstract Superclass for AWT

d) Abstract Subclass for AWT

36) Which class encapsulates a mutually exclusive set of buttons?

a) GroupButton

b) ButtonGroup

c) GroupingButton

d) ButtonGrouping

37) Which class is used to manage buttons in Swing?

a) SwingButton

b) Button

c) **JButton**

d) JSwingButton

38) AbstractButton class extends which class?

a) **JComponent**

b) AbstractComponent

c) ButtonComponent

d) Component

39) void setDisabledIcon(Icon di) - in this method, what is the work of "Icon di"?

a) This is the component that will be disabled

b) It is the icon that will be disabled

c) **It is an icon that is displayed when a component is disabled**

d) It deletes the icon displayed on the screen

40) Which Listener is used with subclasses of AbstractButton to generate events?

a) **ActionListener**

b) ItemListener

c) AdjustmentListener

d) KeyListener

41) AbstractButton is a superclass for which components?

a) Combo boxes, check boxes, and lists

b) push buttons, check boxes, and radio buttons.

c) Text fields, Text areas

d) None of the above

42) Which of the following is the correct constructor for JButton?

a) **JButton(String s, Icon i)**

b) JButton(Icon i, String s)

c) JButton(String s, Icon i, JButton.LEFT)

d) None of the above

43) Which are the constructors of JButton class?

a) JButton(Icon i) , JButton(String s) , JButton(String s, Icon i)

b) Only JButton(String s) ,JButton(String s, Icon i)

c) Only JButton(String s, Icon i)

d) Jbutton(icon i), Jbutton(string s), Jbutton(string s, icon i)

44) Which class is used to add Radio Buttons in swing?

a) RadioButtonSwing

b) RadioButton

c) SwingRadioButton

d) **JRadioButton**

45) What is the immediate superclass of JRadioButton class?

a) **JToggleButton**

b) JButton

c) JAbstractButton

d) JButtonMain

46) Which is the correct syntax for constructor of JRadioButton?

a) JRadioButton(Icon i, String s, boolean state)

b) JRadioButton(String s, Icon i, boolean state)

c) JRadioButton(Icon i,boolean state, String s,)

d) d)JRadioButton(boolean state, String s, Icon i)

47) How many options can be selected at a time in JRadioButton?

a) **Only one**

b) Multiple

c) It is defined in the program

d) Only a maximum of six

48) Which methods is used to get the text associated with a Radio Button?

a) getText()

b) getActionCommand()

c) getradioButtonText()

d) getAction()

49) How many options can be selected at a time in a JCheckBox?

a) Only one

b) It is defined in the program

c) **As many as you want**

d) JCheckBox isn't a Swing component

50) What will be the correct statement if you want a checkbox to be checked by default?

a) JCheckBox jc= new JCheckBox(String s, boolean checked)

b) JCheckBox jc= new JCheckBox(String s, boolean false)

c) JCheckBox jc= new JCheckBox(String s, boolean unchecked)

d) JCheckBox jc= new JCheckBox(String s, boolean true)

51) Item Event for JCheckBox is handled by which method?

a) ItemStateChanged()

b) itemChanged()

c) itemStateChange()

d) None of the above

52) Which method is used to get the text of a check box?

a) GetText()

b) getCheckboxText()

c) getText( )

d) GetCheckboxText()

53) Which method can be used to change the state of the JCheckBox?

a) void setSelecteor(boolean state)

b) **void setSelected(boolean state)**

c) void Setselected(boolean state)

d) void setSelectedItem(boolean state)

54) JTabbed Pane manages a set of _____ by linking them with ___

a) links,GUI

b) **components,tabs**

c) data,constructors

d) controls,GUI

55) JTabbed pane uses

a) **SingleSelectionModel**

b) MutipleSelectionModel

c) MVC Architecture

d) None of these

56) Form of adding Tab

a) **void addTab(String name, Component comp)**

b) void addtab(String name, Component comp)

c) void addTab()

d) void AddTab(Stringname, Componentcomp)

57) Tabs are added by calling

a) getTab()

b) setTab()

c) **addTab()**

d) all of these

58) Here ,the component is added to a tab is

a) **JPanel**

b) JFrame

c) JTextField

d) All of these

59) Constructor(s) of JScrollPane

a) **JScrollPane(Component comp)**

b) ScrollPane(Componentcomp)

c) JScrollpane(Component string)

d) All of these

60) A tree presents _____ view of data

a) Structural

b) **Hierchical**

c) Multiple

d) Single

61) Contructors of jtree are

a)JTree(Object obj[])

JTree(Vec)

JTree(TreeNode tn)

b)JTree(Objectobj[])

JTree(Vector<?>v)

JTree(Treenode tn)

c)JTree(Objectobj[])

JTree(Vector<?> v)

JTree(TreenNode tn)

d)None of these

62) The tree event classes and listener interfaces are packaged in

a) JAva.Swing.*;

b) javaSwing.;

c) java;

d) javax.Swing.event

63) You can obtain the path to the selected object by calling _____ shown here, on the event object _____

a) getPath(),path p

b) getpath(),TreePath getPath( )

c) setpath,TreePath

d) None of these

64) The _____ interface extends TreeNode

a) **MutableTreeNode**

b) DefaultMutableTreeNode

c) Both a & b

d) None of these

65) _____ is a component that displays rows and columns of data.

a) JPanel

b) JFrame

c) **JTable**

d) None of these

66) Constructor used for JTable is

a) JTable(Objectdata[ ][ ], ObjectcolHeads[ ])

b) JTable(Objectdata[ ][ ], ObjectcolHeads[ ]

c) JTable()

d) All of these

67) MVC stands for

a) Model-View-Control

b) Menu-View-Control

c) Menu-Visual-Control

d) **Model-View-Controller**

68) In MVC terminology, the model corresponds to the

a) Components and links

b) **state information associated with the component.**

c) Data

d) All of these

69) The _____ determines how the component reacts to the user

a) View

b) **Controller**

c) User

d) Model

70) In which package Swing components are defined?

a) javax.applet.swing

b) javax.swing

c) java.javax.swing

d) javax.java.swing

71) The super class of all swing buttons is –

a) Button

b) ButtonGroup

c) JButton

d) AbstractButton

72) Which of the following alignment is not possible for JLabel?

a) **TOP**

b) LEFT

c) CENTER

d) LEADING

73) Alignment constants of JLabel are the part of –

a) **SwingConstants interface**

b) SwingConstants class

c) Swing class

d) Graphics class

74) How will you set icon for the Jlabel?

a) Using Icon class directly

b) **Using setIcon( ) method**

c) Using makeIcon( ) method

d) It is not possible to set icon for JLabel

75) Swing's text field is encapsulated by –

a) Component class

b) JComponent class

c) Container class

d) JTextComponent class

76) How to give number of columns for JTextField?

a) Use setColumns( ) method

b) **Use the value directly in the constructor**

c) Using applyColumn( ) method.

d) We have to use JTextArea class

77) What is the return type of getText( ) method of JButton class?

a) void

b) **String**

c) Character array

d) There is no such method

78) How will you assign the string and icon both to the JButton?

a) It is not possible

b) Use the setTextIcon( ) method

c) Use the setIconText( ) method

d) **Initialize them directly in the constructor**

79) Which event is generated when a JButton is pushed?

a) ItemEvent b) TextEvent c) PushEvent d**) ActionEvent**

80) Immediate super class of JCheckBox is –

a) JComponent b) JApplet c) JCkeckBoxGroup **d) JToggleButton**

81) The constructor JCheckBox(true, "YES") suggests that –

a) Checkbox is selected and displays the string "YES" on it.

b) Checkbox is deselected and displays the string "YES" on it.

c) Checkbox is selected and it shows the tick always on it.

d) There is no such constructor.

82) When JCheckBox is clicked the _____ event is generated.

a) **ItemEvent** b) ActionEvent c) TextEvent d) MouseEvent

83) How can we create Radio buttons in Swing?

a) Using ButtonGroup class

b) Using JCheckboxGroup class

c) **Using JRadioButton class**

d) Using JButton class

84) How to make the group of Radio buttons?

a) **Using ButtonGroup class**

b) Using JButtonGroup class

c) Using JRadioButton

d) Using AbstractButton class

85) How to contents of whole vector into the JComboBox?

a) Use the constructor of JComboBox

b) Use method addItem( )

c) Use method addVector( )

d) Use method addValues( )

86) Which method is used to add the tabs in the tabbed pane?

a) add( )

b) addItem( )

c) addPane( )

d) **addTab( )**

87) The scroll bar constants for scroll pane are defined in –

a) Scrollbar class

b) ScrollPane class

c) **ScrollPaneConstants class**

d) Component class

88) Which of the following constants shows scroll bars always?

a) **HORIZONTAL_SCROLLBAR_ALWAYS**

b) HORIZONTAL_SCROLLBAR_AS_NEEDED

c) HORIZONTAL_SCROLLBARS

d) HORIZONTAL_ALWAYS

89) JScrollPane is an immediate sub-class of –

a) JContainer

b) JApplet

c) JComponent

d) ScrollPane

90) Is it possible to add array of objects to trees? How?

a) Not possible

b) Yes, using its one of the forms of constructor

c) Yes, using the add( ) method

d) Yes, using the addItem( ) method

91) When tree is expanded, which event is generated?

a) ExpansionEvent

b) **TreeExpansionEvent**

c) ItemEvent

d) ActionEvent

92) When the user selects or deselects a node within the tree which event is generated.

a) ExpansionEvent

b) TreeExpansionEvent

c) ItemEvent

d) TreeSelectionEvent

93) When the data or structure of the tree changes which event generates:

a) ExpansionEvent

b) TreeExpansionEvent

c) **TreeModelEvent**

d) TreeSelectionEvent

94) We can obtain the path to the selected object in tree by calling _____ method.

a) translatePoint( )

b) getLocation( )

c) getPathForLocation( )

d) **getPath( )**

95) The TreeNode is –

a) A class

b) **An interface**

c) A variable

d) Nothing

96) The TreeExpansionEvent class is defined in –

a) java.awt package

b) javax.swing package

c) java.awt.event package

d) javax.swing.event package

97) TreeExpansionListener interface provides following method –

a) getExpanded( ) **b) treeExpanded( )**

c) expanded( ) d) None of the above

98) How to create for Vector elements?

a) Pass vector as parameter for JTree

b) Use method addElements( ) for JTree class.

c) Use method addVector( ) method of JComponent class

d) It is not possible

99) Which of these are the subclasses of Toggle Button class?

a) JRadioButton

b) JCheckBox

c) **Both a & b**

d) None of these

100) Find the Errors

```
import java.awt.*;
import java.swing.*;
/*<applet code="JLabelDemo" width=250 height=150>
</applet>
*/
public class JLabelDemo extends JApplet {
public void init() {
Container contentPane = getContentPane();
ImageIcon ii = new ImageIcon("france.gif");
JLabel jl = new JLabel("France", ii, JLabel.CENTER);
contentPane.add(jl);
}
}
```

a) Its Correct **b)Wrong package**

c)Wrong constructor for creating Icon d)Both b and c

101) Error time :->

```
import java.awt.*;
import javax.swing.*;
/*
<applet code="JTextFieldDemo" width=300 height=50>
</applet>
*/
public class JTextFieldDemo extends JApplet {
JTextField jtf;
public void init() {
Container contentPane = getContentPane();
contentPane.setLayout(new FlowLayout());
jtf = new JTextField(15,”Enter Text Here”);
contentPane.add(jtf);
}
}
```

a) **Program Code is correct**
b) Wrong package imported
c) Wrong constuctor for TextFiled
d) Wrong class extended

102)

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
/*
<applet code="JComboBoxDemo" width=300 height=100>
</applet>
*/
```

```
public class JComboBoxDemo extends JApplet
implements ActionListener {
JLabel jl;
ImageIcon france, germany, italy, japan;
public void init() {
JComboBox jc = new JComboBox();
jc.addItem("France");
jc.addItem("Germany");
jc.addItem("Italy");
jc.addItem("Japan");
jc.addItemListener(this);
contentPane.add(jc);
jl = new JLabel(new ImageIcon("france.gif"));
contentPane.add(jl);
}
public void itemStateChanged(ItemEvent ie) {
String s = (String)ie.getItem();
jl.setIcon(new ImageIcon(s + ".gif"));
}
}
Container contentPane = getContentPane();
contentPane.setLayout(new FlowLayout);
JComboBox jc = new JComboBox();
jc.addItem("France");
jc.addItem("Germany");
jc.addItem("Italy");
jc.addItem("Japan");
jc.addItemListener(this);
contentPane.add(jc);
```

```
jl = new JLabel(new ImageIcon("france.gif"));
contentPane.add(jl);
}
public void itemStateChanged(ItemEvent ie) {
String s = (String)ie.getItem();
jl.setIcon(new ImageIcon(s + ".gif"));
}
}
```

a)Wrong Listener Implemented b)Program code is correct

c)Constructor for setting Layout is wrong d)Both a and c

103) Fill in the empty spaces in the following program so that the program doesn't have any errors:

```
import java.awt.*;
import java.awt._____.*;
import_____.swing.*;
/*
<applet code="JButtonDemo" width=250 height=300>
</applet>
*/
public class JButtonDemo extends JApplet
implements ActionListener {
JTextField jtf;
public void init() {
// Get content pane
Container contentPane = getContentPane();
contentPane.setLayout(new FlowLayout());
// Add buttons to content pane
```

```
ImageIcon france = new ImageIcon("france.gif");
JButton jb = new_____(france);
jb.setActionCommand("France");
jb.addActionListener(this);
contentPane.add(jb);
ImageIcon germany = new ImageIcon("germany.gif");
jb = new JButton(germany);
jb.setActionCommand("Germany");
jb.addActionListener(this);
contentPane.add(jb);
ImageIcon italy = new ImageIcon("italy.gif");
jb = new JButton(italy);
jb.setActionCommand("Italy");Chapter 26:
jb.addActionListener(this);
contentPane.add(jb);
ImageIcon japan = new ImageIcon("japan.gif");
jb = new JButton(japan);
jb.setActionCommand("Japan");
jb.addActionListener(this);
contentPane.add(jb);
// Add text field to content pane
jtf = new JTextField(15);
_____.add(jtf);
}
public void actionPerformed(ActionEvent ae) {
jtf.setText(ae.getActionCommand());
}
}
```

a) **event, javax, JButton, contentPane**

b) event, Java, Button, cpp

c) awt, Javax, JComboBox, contentPane.

d) Swing, event, awt, JTextField

104) What will be the packages required for this program to work?

```
/*
<applet code="JRadioButtonDemo" width=300 height=50>
</applet>
*/
public class JRadioButtonDemo extends JApplet
implements ActionListener {
JTextField tf;
public void init() {
// Get content pane
Container contentPane = getContentPane();
contentPane.setLayout(new FlowLayout());
// Add radio buttons to content pane
JRadioButton b1 = new JRadioButton("A");
b1.addActionListener(this);
contentPane.add(b1);
JRadioButton b2 = new JRadioButton("B");
b2.addActionListener(this);
contentPane.add(b2);
JRadioButton b3 = new JRadioButton("C");
b3.addActionListener(this);
contentPane.add(b3);
// Define a button group
```

```
ButtonGroup bg = new ButtonGroup();
bg.add(b1);
bg.add(b2);
bg.add(b3);
// Create a text field and add it
// to the content pane
tf = new JTextField(5);
contentPane.add(tf);
}
```

a)

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
```

b)

```
import java.Applet.*
import java.awt.*;
```

c)

```
import javax.Swing.Tree.*
import javax.Swing.*
```

d)

```
 import java.awt.*
Import java.JApplet.*;
```

105) Consider the following output. Find the missing statement in the program.

```
 import javax.swing.*;
/* <applet code="JTabbedPaneDemo" width=400 height=100> </applet> */
public class JTabbedPaneDemo extends JApplet
```

```
{
public void init()
{
JTabbedPane jtp = new JTabbedPane();
jtp.addTab("Cities", new CitiesPanel());
jtp.addTab("Colors", new ColorsPanel());
jtp.addTab("Flavors", new FlavorsPanel());
}
}
class CitiesPanel extends JPanel
{
public CitiesPanel()
{
JButton b1 = new JButton("New York");
add(b1);
JButton b2 = new JButton("London");
add(b2);
JButton b3 = new JButton("Hong Kong");
add(b3);
JButton b4 =new JButton("Tokyo"); add(b4);
  }
 }
class ColorsPanel extends JPanel
{
public ColorsPanel()
{
JCheckBox cb1 = new JCheckBox("Red");
add(cb1);
JCheckBox cb2 = new JCheckBox("Green");
```

```
add(cb2);
JCheckBox cb3 = new JCheckBox("Blue");
add(cb3);
 }
 }
```

a) setPane();

b) **getContentPane().add(jtp);**

c) Both a&b

d) None of these

106) What is error in following program?

```
import java.awt.*;
import java.awt.event.*;
 import java.applet.*;
/* <applet code=ScrollDemo.class width=500 height=500> </applet> */
Public class ScrollDemo extends Applet implements AdjustmentListener
{
   Scrollbar s1,s2,s3;
  public void init(
   {
   s1=new Scrollbar(Scrollbar.VERTICAL,0,1,0,255);
    s2=new Scrollbar(Scrollbar.VERTICAL,0,1,0,255);
   s3=new Scrollbar(Scrollbar.VERTICAL,0,1,0,255);
    add(s1);
   add(s2);
   add(s3);
```

```
s1.addAdjustmentListener(this);
s2.addAdjustmentListener(this);
s3.addAdjustmentListener(this);
}
 public void adjustmentChange(AdjustmentEvent ae)
{
setBackground(new Color(s1.getValue(),s2.getValue(),s3.getValue()));
 }
}
```

a) ScrollDemo is not abstract and does not override abstract method

b) adjustmentValueChanged(AdjustmentEvent) in AdjustmentListener public class ScrollDemo extends Applet implements AdjustmentListener

c) Both a&b

d) None of these

107) To get the following output complete the code given below.

```
import java.awt.*;
Import java.awt.event.*;
import javax.swing.*;
class SimpleTableExample extends JFrame
{
Private JPanel topPanel;
 private JTable table;
 private JScrollPane scrollPane;
 public SimpleTableExample()
 {
 setTitle( "Simple Table Application" );
```

```
setSize( 300, 200 );
setBackground( Color.gray );
topPanel = new JPanel();
topPanel.setLayout( new BorderLayout()
);
getContentPane().add( topPanel );
String columnNames[] = { "Column 1", "Column 2","Column 3" };
 String dataValues[][] = { { "12", "234", "67" }, { "-123", "43", "853" },
{ "93","89.2", "109" }, { "279", "9033", "3092" }
};
 topPanel.add( scrollPane, BorderLayout.CENTER );
}
public static void main( String args[] )
{
SimpleTableExample mainFrame = new
SimpleTableExample(); mainFrame.setVisible( true );
 }
 }
```

a)table = new JTable(dataValues,column Names);

scrollPane = new JScrollPane(table);

b)table = new JTable

c)New JScrollPane(t)

d)Both a& b

108) Which statement should be added to display button.

```
import java.awt.*;
import javax.swing.*;
/*
<applet code="JButtonDemo" width=250 height=300>
</applet>
```

```
*/

public class JButtonDemo extends JApplet
 {
public void init()
{
Container contentPane = getContentPane();
contentPane.setLayout(new FlowLayout());

ImageIcon img= new ImageIcon("jpgIcon.jpg");
JButton jb = new JButton(img);
}
}
```

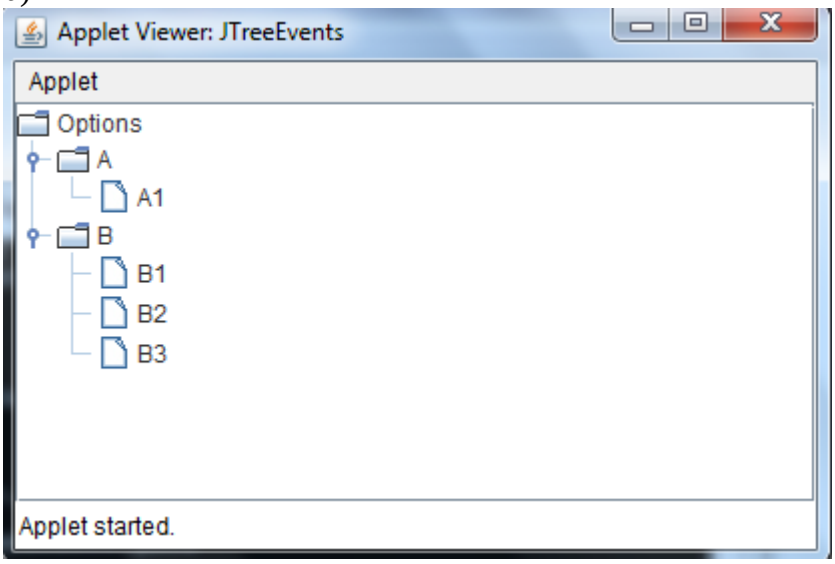
a) ContentPane.add(jb);
b) contentPane.add(jb);
c) contentPane.add();
d) contentPane.addButton(jb);

109) What will be the output of following code:

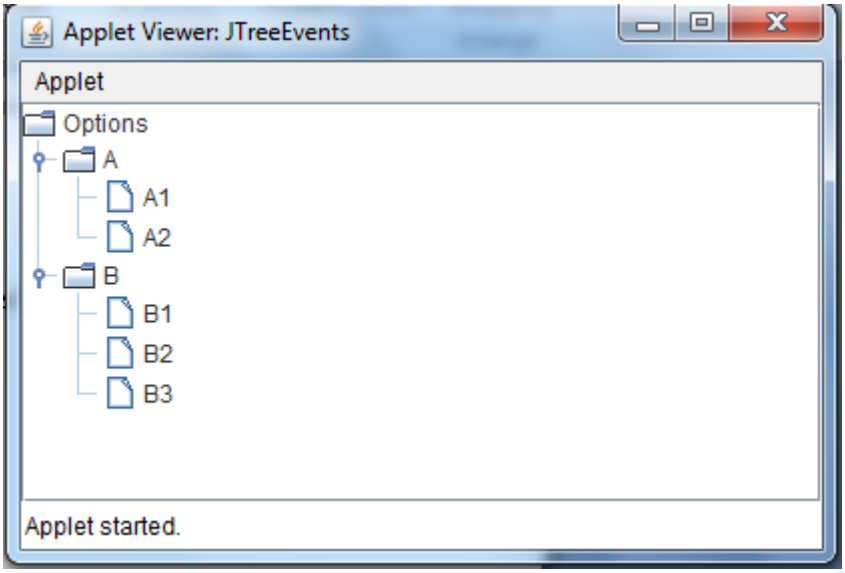
```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import javax.swing.tree.*;

/*<applet code="JTreeEvents" width=400 height=200></applet>*/

public class JTreeEvents extends JApplet
{
JTree tree;

public void init()
{
Container contentPane = getContentPane();
contentPane.setLayout(new BorderLayout());
DefaultMutableTreeNode top = new DefaultMutableTreeNode("Options");
DefaultMutableTreeNode a = new DefaultMutableTreeNode("A");
top.add(a);
DefaultMutableTreeNode a1 = new DefaultMutableTreeNode("A1");
a.add(a1);
DefaultMutableTreeNode a2 = new DefaultMutableTreeNode("A2");
a.add(a2);
DefaultMutableTreeNode b = new DefaultMutableTreeNode("B");
top.add(b);
DefaultMutableTreeNode b1 = new DefaultMutableTreeNode("B1");
b.add(b1);
DefaultMutableTreeNode b2 = new DefaultMutableTreeNode("B2");
b.add(b2);
DefaultMutableTreeNode b3 = new DefaultMutableTreeNode("B3");
```

```
b.add(b3);
tree = new JTree(top);
int v = ScrollPaneConstants.VERTICAL_SCROLLBAR_AS_NEEDED;
int h = ScrollPaneConstants.HORIZONTAL_SCROLLBAR_AS_NEEDED;
JScrollPane jsp = new JScrollPane(tree, v, h);
contentPane.add(jsp, BorderLayout.CENTER);
}
}
```

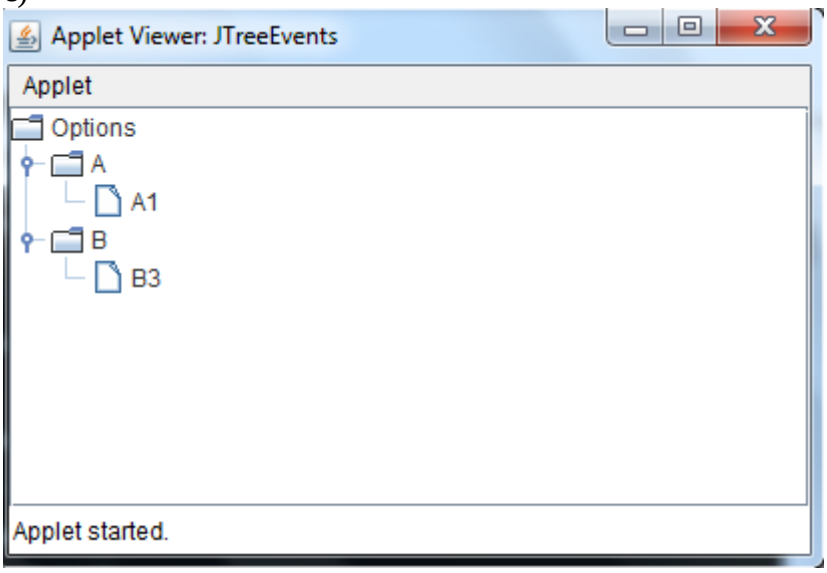
a)



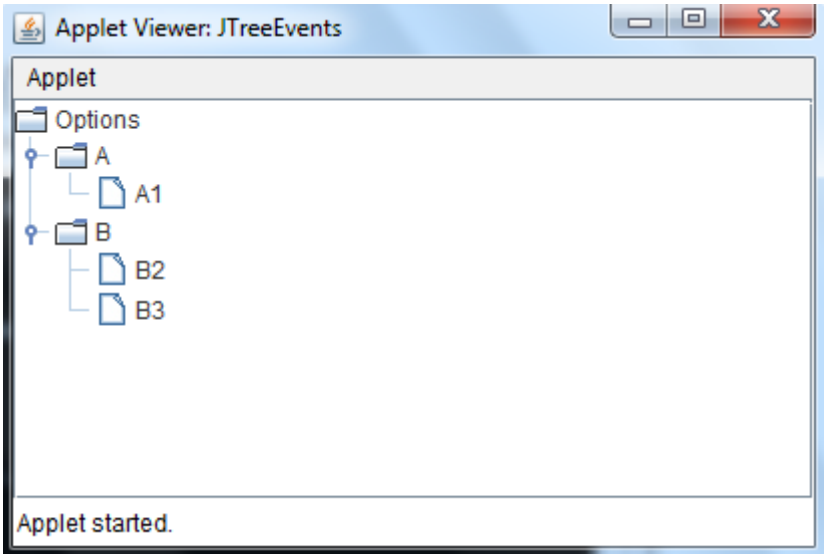
b)



c)


Applet Viewer: JTreeEvents
Applet
Options
A
A1
B
B3
Applet started.

d)


Applet Viewer: JTreeEvents
Applet
Options
A
A1
B
B2
B3
Applet started.

- Events are supported by the **java.awt.event** package.
- The modern approach to handling events is based on the *delegation event model*, which defines standard and consistent mechanisms to generate and process events.
- A *source* generates an event and sends it to one or more *listeners*.
- An *event* is an object that describes a state change in a source.
- Events may also occur that are not directly caused by interactions with a user interface.
- A *source* is an object that generates an event. This occurs when the internal state of that object changes

- Sources may generate more than one type of event.

- A source must register listeners in order for the listeners to receive notifications about a specific type of event. Each type of event has its own registration method.
  the general form:
  public void add*Type*Listener(*Type*Listener *el*)

- When an event occurs, all registered listeners are notified and receive a copy of the event object. This is known as *multicasting* the event.

- In all cases, notifications are sent only to listeners that register to receive them.

- Some sources may allow only one listener to register. The general form of such a method is this:
  public void add*Type*Listener(*Type*Listener *el*) throws java.util.TooManyListenersException

  When such an event occurs, the registered listener is notified. This is known as *unicasting* the event.

- A source must also provide a method that allows a listener to unregister an interest in a specific type of event. The general form of such a method is this:
  public void remove*Type*Listener(*Type*Listener *el*)

- A *listener* is an object that is notified when an event occurs. It has two major requirements.
- First, it must have been registered with one or more sources to receive notifications about specific types of events.
  Second, it must implement methods to receive and process these notifications.

- The methods that receive and process events are defined in a set of interfaces found in **java.awt.event**.

- At the root of the Java event class hierarchy is **EventObject**, which is in **java.util**.
- It is the superclass for all events. Its one constructor is shown here:
  EventObject(Object *src*)

- **EventObject** contains two methods: **getSource( )** and **toString( )**.

- The **getSource( )** method returns the source of the event.
  Object getSource( )

- **toString( )** returns the string equivalent of the event.

- The class **AWTEvent**, defined within the **java.awt** package, is a subclass of **EventObject**. It is the superclass (either directly or indirectly) of all AWT-based events used by the delegation event model.

- **getID( )** method can be used to determine the type of the event.
  int getID( )

- **EventObject** is a superclass of all events.
- **AWTEvent** is a superclass of all AWT events that are handled by the delegation event model

- ActionEvent Generated when a button is pressed, a list item is double-clicked, or a menu item is selected.
- AdjustmentEvent Generated when a scroll bar is manipulated.
- ComponentEvent Generated when a component is hidden, moved, resized, or becomes visible.
- ContainerEvent Generated when a component is added to or removed from a container.
- FocusEvent Generated when a component gains or loses keyboard focus.
- InputEvent Abstract super class for all component input event classes.
- ItemEvent Generated when a check box or list item is clicked; also occurs when a choice selection is made or a checkable menu item is selected or deselected.
- KeyEvent Generated when input is received from the keyboard.
- MouseEvent Generated when the mouse is dragged, moved, clicked, pressed, or released; also generated when the mouse enters or exits a component.
- MouseWheelEvent Generated when the mouse wheel is moved.
- TextEvent Generated when the value of a text area or text field is changed.
- WindowEvent Generated when a window is activated, closed, deactivated, deiconified, iconified, opened, or quit.

- An **ActionEvent** is generated when a button is pressed, a list item is double-clicked, or a menu item is selected.

- The **ActionEvent** class defines four integer constants that can be used to identify any modifiers associated with an action event: **ALT_MASK**, **CTRL_MASK**, **META_MASK**, and **SHIFT_MASK**. In addition, there is an integer constant, **ACTION_PERFORMED**, which can be used to identify action events.

- Constructors:
  ActionEvent(Object *src*, int *type*, String *cmd*)
  ActionEvent(Object *src*, int *type*, String *cmd*, int *modifiers*)
  ActionEvent(Object *src*, int *type*, String *cmd*, long *when*, int *modifiers*)

- You can obtain the command name for the invoking **ActionEvent** object by using the **getActionCommand( )** method,
  String getActionCommand( )

- The **getModifiers( )** method returns a value that indicates which modifier keys (ALT, CTRL, META, and/or SHIFT) were pressed when the event was generated.
  int getModifiers( )

- **getWhen( )** that returns the time at which the event took place. This is called the event's *timestamp*.
  long getWhen( )

- Timestamps were added by **ActionEvent** to help support the improved input focus subsystem
- An **AdjustmentEvent** is generated by a scroll bar.
- The **AdjustmentEvent** class defines integer constants that can be used to identify
- them.

  BLOCK_DECREMENT: The user clicked inside the scroll bar to decrease its value.
  BLOCK_INCREMENT : The user clicked inside the scroll bar to increase its value.
  TRACK :The slider was dragged.
  UNIT_DECREMENT : The button at the end of the scroll bar was clicked to decrease its value.
  UNIT_INCREMENT :The button at the end of the scroll bar was clicked to increase its value.

- In addition, there is an integer constant, **ADJUSTMENT_VALUE_CHANGED**, that indicates that a change has occurred.

- **AdjustmentEvent** constructor:
  AdjustmentEvent(Adjustable *src*, int *id*, int *type*, int *data*)

- The **getAdjustable( )** method returns the object that generated the event.
  Adjustable getAdjustable( )

- The type of the adjustment event may be obtained by the **getAdjustmentType( )** method.
- It returns one of the constants defined by **AdjustmentEvent**.
  int getAdjustmentType( )

- The amount of the adjustment can be obtained from the **getValue( )** method
  int getValue( )

- A **ComponentEvent** is generated when the size, position, or visibility of a component is changed

- There are four types of component events
- The **ComponentEvent** class defines integer constants that can be used to identify them

  COMPONENT_HIDDEN: The component was hidden.
  COMPONENT_MOVED :The component was moved.
  COMPONENT_RESIZED: The component was resized.
  COMPONENT_SHOWN :The component became visible.

- Constructors:
  ComponentEvent(Component *src*, int *type*)

- **ComponentEvent** is the superclass either directly or indirectly of **ContainerEvent**,**FocusEvent**, **KeyEvent**, **MouseEvent**, and **WindowEvent**.

- The **getComponent( )** method returns the component that generated the event.
  Component getComponent( )

- A **ContainerEvent** is generated when a component is added to or removed from a container.

- There are two types of container events
- The **ContainerEvent** class defines **int** constants that can be used to identify them: **COMPONENT_ADDED** and **COMPONENT_REMOVED**.

- **ContainerEvent** is a subclass of **ComponentEvent**

- constructor:
  ContainerEvent(Component *src*, int *type*, Component *comp*)

- You can obtain a reference to the container that generated this event by using the **getContainer( )** method
  Container getContainer( )

- The **getChild( )** method returns a reference to the component that was added to or removed from the container

Component getChild( )

- A **FocusEvent** is generated when a component gains or loses input focus
- These events are identified by the integer constants **FOCUS_GAINED** and **FOCUS_LOST**.

- **FocusEvent** is a subclass of **ComponentEvent**

- constructors:
  FocusEvent(Component *src*, int *type*)
  FocusEvent(Component *src*, int *type*, boolean *temporaryFlag*)
  Focus Event(Component *src*, int *type*, boolean *temporaryFlag*, Component *other*)

- The argument *temporaryFlag* is set to **true** if the focus event is temporary.
- The other component involved in the focus change, called the *opposite component*, is passed in *other*. Therefore, if a **FOCUS_GAINED** event occurred, *other* will refer to the component that lost focus. Conversely, if a **FOCUS_LOST** event occurred, *other* will refer to the component that gains focus.

- You can determine the other component by calling **getOppositeComponent( )**,
  Component getOppositeComponent( )

- The **isTemporary( )** method indicates if this focus change is temporary
  boolean isTemporary( )

- The abstract class **InputEvent** is a subclass of **ComponentEvent** and is the superclass for component input events. Its subclasses are **KeyEvent** and **MouseEvent**.

- **InputEvent** defines several integer constants that represent any modifiers, such as the control key being pressed, that might be associated with the event.

- **InputEvent** class defined the following eight values to represent the modifiers.
  ALT_MASK
  BUTTON2_MASK
  META_MASK
  ALT_GRAPH_MASK
  BUTTON3_MASK
  SHIFT_MASK
  BUTTON1_MASK
  CTRL_MASK

- To test if a modifier was pressed at the time an event is generated, use the **isAltDown( )**, **isAltGraphDown( )**, **isControlDown( )**, **isMetaDown( )**, and **isShiftDown( )** methods.

- **boolean isAltDown( )**
- **boolean isAltGraphDown( )**
- **boolean isControlDown( )**
- **boolean isMetaDown( )**
- **boolean isShiftDown( )**

- You can obtain a value that contains all of the original modifier flags by calling the **getModifiers( )** method.
  int getModifiers( )

- You can obtain the extended modifiers by called **getModifiersEx( )**,
  int getModifiersEx( )

- An **ItemEvent** is generated when a check box or a list item is clicked or when a checkable menu item is selected or deselected.

- There are two types of item events

- DESELECTED :The user deselected an item.
  SELECTED: The user selected an item.

- **ItemEvent** defines one integer constant, **ITEM_STATE_CHANGED**, that signifies a change of state.

- constructor:
  ItemEvent(ItemSelectable *src*, int *type*, Object *entry*, int *state*)

- The **getItem( )** method can be used to obtain a reference to the item that generated an event.
  Object getItem( )

- The **getItemSelectable( )** method can be used to obtain a reference to the **ItemSelectable** object that generated an event.
  ItemSelectable getItemSelectable( )

- Lists and choices are examples of user interface elements that implement the **ItemSelectable** interface.
- The **getStateChange( )** method returns the state change (i.e., **SELECTED** or **DESELECTED**) for the event.

  int getStateChange( )

- A **KeyEvent** is generated when keyboard input occurs.
- There are three types of key events, which are identified by these integer constants: **KEY_PRESSED**,
- **KEY_RELEASED**, and **KEY_TYPED**.
- The first two events are generated when any key is pressed or released. The last event occurs only when a character is generated.

- There are many other integer constants that are defined by **KeyEvent**.

- For example,
- **VK_0** through **VK_9** and **VK_A** through **VK_Z** define the ASCII equivalents of the
- numbers and letters. Here are some others:
- **VK_ENTER**
- **VK_ESCAPE**
- **VK_CANCEL**
- **VK_UP**
- **VK_DOWN**
- **VK_LEFT**
- **VK_RIGHT**
- **VK_PAGE_DOWN**
- **VK_PAGE_UP**
- **VK_SHIFT**
- **VK_ALT**
- **VK_CONTROL**

- The **VK** constants specify *virtual key codes* and are independent of any modifiers, such as control, shift, or alt.

- **KeyEvent** is a subclass of **InputEvent**.

- constructors:
- **KeyEvent(Component *src*, int *type*, long *when*, int *modifiers*, int *code*)**
- **KeyEvent(Component *src*, int *type*, long *when*, int *modifiers*, int *code*, char *ch*)**

- **getKeyChar( )**, which returns the character that was entered, and **getKeyCode( )**, which returns the key code.
- **char getKeyChar( )**
- **int getKeyCode( )**

- If no valid character is available, then **getKeyChar( )** returns **CHAR_UNDEFINED**.
- When a **KEY_TYPED** event occurs, **getKeyCode( )** returns **VK_UNDEFINED**.

- There are eight types of mouse events. The **MouseEvent** class defines the following
- integer constants that can be used to identify them:

- **MOUSE_CLICKED: The user clicked the mouse.**
- **MOUSE_DRAGGED: The user dragged the mouse.**
- **MOUSE_ENTERED: The mouse entered a component.**
- **MOUSE_EXITED :The mouse exited from a component.**
- **MOUSE_MOVED: The mouse moved.**
- **MOUSE_PRESSED: The mouse was pressed.**
- **MOUSE_RELEASED: The mouse was released.**
- **MOUSE_WHEEL :The mouse wheel was moved**

- **MouseEvent** is a subclass of **InputEvent**.
- constructors.
- **MouseEvent(Component *src*, int *type*, long *when*, int *modifiers*, int *x*, int *y*, int *clicks*, boolean *triggersPopup*)**

- The most commonly used methods in this class are **getX( )** and **getY( )**. These return the X and Y coordinates of the mouse when the event occurred.
- **int getX( )**
- **int getY( )**

- **getPoint( )** method to obtain the coordinates of the mouse.
  Point getPoint( )

- The **translatePoint( )** method changes the location of the event.
  void translatePoint(int *x*, int *y*)

- The **getClickCount( )** method obtains the number of mouse clicks for this event.
  int getClickCount( )

- **isPopupTrigger( )** method tests if this event causes a pop-up menu to appear on this platform.
  boolean isPopupTrigger( )

- int getButton( ) : It returns a value that represents the button that caused the event. The return value will be one of these constants defined by **MouseEvent**.
  NOBUTTON BUTTON1 BUTTON2 BUTTON3

- **NOBUTTON** value indicates that no button was pressed or released.

- The **MouseWheelEvent** class encapsulates a mouse wheel event. It is a subclass of **MouseEvent**

- If a mouse has a wheel, it is located between the left and right buttons. Mouse wheels are used for scrolling. **MouseWheelEvent** defines these two integer constants.
- **WHEEL_BLOCK_SCROLL A page-up or page-down scroll event occurred.**
- **WHEEL_UNIT_SCROLL A line-up or line-down scroll event occurred.**

- constructor.
- **MouseWheelEvent(Component *src*, int *type*, long *when*, int *modifiers*, int *x*, int *y*, int *clicks*, boolean *triggersPopup*, int *scrollHow*, int *amount*, int *count*)**

- To obtain the number of rotational units, call **getWheelRotation( )**,
  int getWheelRotation( )

- If the value is positive, the wheel moved counterclockwise. If the value is negative, the wheel moved clockwise.

- To obtain the type of scroll, call **getScrollType( )**
  int getScrollType( )

- If the scroll type is **WHEEL_UNIT_SCROLL**, you can obtain the number of units to scroll by calling **getScrollAmount( )**.
  int getScrollAmount( )

- Text Event are generated by text fields and text areas when characters are entered by a user or program.

- **TextEvent** defines the integer constant **TEXT_VALUE_CHANGED**.

- Constructor:
  TextEvent(Object *src*, int *type*)

- The **TextEvent** object does not include the characters currently in the text component that generated the event.

- There are ten types of window events. The **WindowEvent** class defines integer constants that can be used to identify them.

- **WINDOW_ACTIVATED: The window was activated.**
- **WINDOW_CLOSED :The window has been closed.**
- **WINDOW_CLOSING :The user requested that the window be closed.**
- **WINDOW_DEACTIVATED: The window was deactivated.**
- **WINDOW_DEICONIFIED: The window was deiconified.**
- **WINDOW_GAINED_FOCUS: The window gained input focus.**
- **WINDOW_ICONIFIED :The window was iconified.**
- **WINDOW_LOST_FOCUS: The window lost input focus.**
- **WINDOW_OPENED :The window was opened.**
- **WINDOW_STATE_CHANGED: The state of the window changed.**

- **WindowEvent** is a subclass of **ComponentEvent**

- **Constructors:**
- **WindowEvent(Window *src*, int *type*)**

- **WindowEvent(Window *src*, int *type*, Window *other*)**
- **WindowEvent(Window *src*, int *type*, int *fromState*, int *toState*)**
- **WindowEvent(Window *src*, int *type*, Window *other*, int *fromState*, int *toState*)**

- **getWindow( )**. It returns the **Window** object that generated the event.
- **Window getWindow( )**

- **Window getOppositeWindow()**
- **int getOldState()**
- **int getNewState()**

- **Button :Generates action events when the button is pressed.**
- **Checkbox: Generates item events when the check box is selected or deselected.**
- **Choice :Generates item events when the choice is changed.**
- **List :Generates action events when an item is double-clicked; generates item events when an item is selected or deselected.**
- **Menu: Item Generates action events when a menu item is selected; generates item events when a checkable menu item is selected or deselected.**
- **Scrollbar :Generates adjustment events when the scroll bar is manipulated.**
- **Text components: Generates text events when the user enters a character.**
- **Window :Generates window events when a window is activated, closed, deactivated, deiconified, iconified, opened, or quit.**

- **ActionListener: Defines one method to receive action events.**
- **AdjustmentListener: Defines one method to receive adjustment events.**
- **ComponentListener :Defines four methods to recognize when a component is hidden, moved, resized, or shown.**
- **ContainerListener :Defines two methods to recognize when a component is added to or removed from a container.**
- **FocusListener: Defines two methods to recognize when a component gains or loses keyboard focus.**
- **ItemListener: Defines one method to recognize when the state of an item changes.**
- **KeyListener :Defines three methods to recognize when a key is pressed, released, or typed.**
- **MouseListener: Defines five methods to recognize when the mouse is clicked, enters a component, exits a component, is pressed, or is released.**
- **MouseMotionListener: Defines two methods to recognize when the mouse is dragged or moved.**
- **MouseWheelListener :Defines one method to recognize when the mouse wheel is moved.**
- **TextListener :Defines one method to recognize when a text value changes.**
- **WindowFocusListener: Defines two methods to recognize when a window gains or loses input focus.**
- **WindowListener: Defines seven methods to recognize when a window is activated, closed, deactivated, deiconified, iconified, opened, or quit.**

- ActionListener interface defines the **actionPerformed( )** method that is invoked when an action event occurs.
- **void actionPerformed(ActionEvent *ae*)**

- AdjustmentListener interface defines the **adjustmentValueChanged( )** method that is invoked when an adjustment event occurs.
- **void adjustmentValueChanged(AdjustmentEvent *ae*)**

- ComponentListener interface defines four methods that are invoked when a component is resized, moved, shown, or hidden.
- **void componentResized(ComponentEvent *ce*)**
- **void componentMoved(ComponentEvent *ce*)**
- **void componentShown(ComponentEvent *ce*)**
- **void componentHidden(ComponentEvent *ce*)**

- ContainerListener interface contains two methods.

- When a component is added to a container, **componentAdded( )** is invoked.

- When a component is removed from a container, **componentRemoved( )** is invoked.
- **void componentAdded(ContainerEvent *ce*)**
- **void componentRemoved(ContainerEvent *ce*)**

- FocusListener interface defines two methods. When a component obtains keyboard focus,
- **focusGained( )** is invoked. When a component loses keyboard focus, **focusLost( )** is called.
- **void focusGained(FocusEvent *fe*)**
- **void focusLost(FocusEvent *fe*)**

- ItemListener interface defines the **itemStateChanged( )** method that is invoked when the state of an item changes.
- **void itemStateChanged(ItemEvent *ie*)**

- KeyListener interface defines three methods. The **keyPressed( )** and **keyReleased( )** methods are invoked when a key is pressed and released, respectively. The **keyTyped( )** method is invoked when a character has been entered.

- **void keyPressed(KeyEvent *ke*)**
- **void keyReleased(KeyEvent *ke*)**
- **void keyTyped(KeyEvent *ke*)**

- MouseListener interface defines five methods. If the mouse is pressed and released at the same point, **mouseClicked( )** is invoked. When the mouse enters a component, the **mouseEntered( )** method is called. When it leaves, **mouseExited( )** is called. The **mousePressed( )** and **mouseReleased( )** methods are invoked when the mouse is pressed and released,
- **void mouseClicked(MouseEvent *me*)**
- **void mouseEntered(MouseEvent *me*)**
- **void mouseExited(MouseEvent *me*)**
- **void mousePressed(MouseEvent *me*)**
- **void mouseReleased(MouseEvent *me*)**

- MouseMotionListener interface defines two methods. The **mouseDragged( )** method is called multiple times as the mouse is dragged. The **mouseMoved( )** method is called multiple times as the mouse is moved.

- **void mouseDragged(MouseEvent *me*)**
- **void mouseMoved(MouseEvent *me*)**

- MouseWheelListener interface defines the **mouseWheelMoved( )** method that is invoked when the mouse wheel is moved.
- **void mouseWheelMoved(MouseWheelEvent *mwe*)**

- TextListener interface defines the **textChanged( )** method that is invoked when a change occurs in a text area or text field
- **void textChanged(TextEvent *te*)**

- WindowFocusListener interface defines two methods: **windowGainedFocus( )** and **windowLostFocus( )**. These are called when a window gains or losses input focus.

- **void windowGainedFocus(WindowEvent *we*)**
- **void windowLostFocus(WindowEvent *we*)**

- WindowListener interface defines seven methods. The **windowActivated( )** and **windowDeactivated( )** methods are invoked when a window is activated or deactivated, respectively. If a window is iconified, the **windowIconified( )** method is called. When a window is deiconified, the **windowDeiconified( )** method is called. When a window is opened or closed, the **windowOpened( )** or **windowClosed( )** methods are called, respectively. The **windowClosing( )** method is called when a window is being closed.
- **void windowActivated(WindowEvent *we*)**
- **void windowClosed(WindowEvent *we*)**
- **void windowClosing(WindowEvent *we*)**
- **void windowDeactivated(WindowEvent *we*)**
- **void windowDeiconified(WindowEvent *we*)**
- **void windowIconified(WindowEvent *we*)**
- **void windowOpened(WindowEvent *we*)**

- *adapter class*, that can simplify the creation of event handlers in certain situations. An adapter class provides an empty implementation of all methods in an event listener interface. Adapter classes are useful when you want to receive and process only some of the events that are handled by a particular event listener interface. You can define a new class to act as an event listener by extending one of the adapter classes and implementing only those events in which you are interested.

- Adapter Class: Listener Interface
- ComponentAdapter: ComponentListener
- ContainerAdapter :ContainerListener
- FocusAdapter: FocusListener
- KeyAdapter :KeyListener
- MouseAdapter: MouseListener
- MouseMotionAdapter: MouseMotionListener
- WindowAdapter :WindowListener

- An *anonymous* inner class is one that is not assigned a name

- an *inner class* is a class defined within other class, or even within an expression.

TY DIPLOMA - VIMP QUESTIONS
ADVANCE JAVA PROGRAMMING

Chapter :04

Networking Basics

1) Which is the connectionless protocol?

a) TCP

b) UDP

c) IP

d) HTTP

2) What is the port number of HTTP?

a) 80

b) 23

c) 47

d) 92

3) Which protocol that web browsers and servers use to transfer hypertext pages and images?

a) TCP/IP

b) UDP

c) **HTTP**

d) SMTP

4) Originally, all Internet addresses consisted of ____ bit values.

a) 32

b) 16

c) 24

d) 8

5) What is the full form of DNS?

a) Domain Naming System

b) **Domain Naming Service**

c) Domain Naming Software

d) Domain Naming Security

6) What is URL?

a) Unicast Resource Location

b) Uniform Research Launcher

c) Universal Research Locator

d) **Uniform Resource Locator**

7) Which class is used to encapsulate both the numerical IP address and the domain name for that address?

a) **InetAddress**

b) SocketAddress

c) IPAdress

d) DNSSockets

8) IP address is a _____ protocol that breaks data into small packets and sends them to an area across network.

a. low-level routing

b. high-level routing
c. medium-level routing
d. none of the above

9) _____ is a higher level protocol that manages to robustly string together the packets, sorting and retransmitting them.

a. TCP

b.UDP

c.none of the above

d. both a & b

10) _____ can be used directly to support fast, connectionless and unreliable transport of packets.

a.UDP

b.TCP

c.none of the above

d. both a & b

11) A _____ is anything that has resource that can be shared.

a. server

b. client

c.none of the above

d. both a & b

12) _____ is simply an entity that wants to gain access to a particular server.

a.**Client**

b. server
c. network
d. package

13) A _____ is a numbered socket.

a. port

b. switch
c. URL
d. none of the above

14) A server process must be _____

a. **mutithreaded**

b. single threaded
c. none
d. both a & b

15) TCP/IP reserves lower _____ ports for specific protocols

a. **1024**

b. 1023
c. 1022
d. 1021

16) Port number 21 is for____

a)FTP
b) e-mail
c) Telnet
d) HTTP

17) Port number 23 is for____

a) Telnet
b) FTP
c) e-mail
d) HTTP

18) Port number 25 is for____

a) **e-mail**
e) Telnet

f) FTP
g) HTTP

19) Port number 79 is for____
a) Finger
b) HTTP
c) Telnet
d) FTP

20) Port number 80 is for____
a) HTTP
b) FTP
c) TCP
d) UDP

21) HTTP is a protocol that _____ and _____ use for transfer of hypertext pages and images.
a) web browsers and servers
b) awt
c) swing
d) switch

22) _____ listens to port 80 and _____ connects to port 80.
a.**Server and client**
b. client ,server
c. none of the above
d. all of the above

23) A _____ speaks the client side of a protocol to another server.
a) Proxy
b) proxcy
c) Proksy
d) Proccy

24) _____ is required when clients have restrictions on servers.
a.**Proxy server**

b. proxcy server
c. Proksy server
d. Proccy server

25) URL is an acronym for _____
a)**Uniform Resource Locator.**
b)Uniform Resource Location.

c)Uniform Remote Locator.

d)None of these

26) _____ points to a resource on the World Wide Web
a)**Uniform Resource Locator(URL)**
b)Inet Address

c)ServerSocket

d)TCP/IP

27) A URL specification is based on _____ components.
a)Five
b)Two

c)Three

d)**Four**

28) Which are the four components of URL?
a) Protocol,Server name or IP Address,Port Number
b) **Protocol,Server name or IP Address,Port Number, File Name or directory name**
c)Protocol,Server name or IP Address,Port Number,Host name

d)None of these

29) Java's URL class has several constructors
a)**True**
b)False

30) Java URL class throws which exception

a)MalformedUrlException

b)MalformedURLException

c)malformedURLException

d)MalformedURLExpection

31) Choose the correct two forms of the constructor which allows you to break up the URL into its component parts:

a)URL(String protocolName, String hostName, int port, String path)

b)URL(String protocolName, String hostName, String path)

c)**Both a and b**

d)None of these

32) _____ is a general-purpose class for accessing the attributes of a remote resource.

a)openConnection

b)OpenConnection

c)UrlConnection

d) **URLConnection**

33) The _____ and _____ classes are good enough for simple programs that want to connect to HTTP servers to fetch content.

a) **URL, URLConnection**

b)URL,UrlConnection

c)URLConnection

d)None of the above

34)

Date: Sat Apr 27 12:17:32 CDT 2002 Content-Type: text/html

No expiration information.

Last-Modified: Tue Mar 19 17:52:42 CST 2002

Content-Length: 5299

=== Content ==

The above output is the output for which program?

a) **import java.net.*; import**
java.io.*; import java.util.Date;
class UCDemo
{
public static void main(String args[]) throws Exception { int c;
URL hp = new URL("http://www.internic.net");
URLConnection hpCon = hp.openConnection();
// get date long d =
hpCon.getDate(); if(d==0)
System.out.println("No date information."); else
System.out.println("Date: " + new Date(d));
// get content type
System.out.println("Content-Type: " + hpCon.getContentType());
// get expiration date d =
hpCon.getExpiration();
if(d==0)
System.out.println("No expiration information."); else
System.out.println("Expires: " + new Date(d));
// get last-modified date d =
hpCon.getLastModified();
if(d==0)
System.out.println("No last-modified information."); else
System.out.println("Last-Modified: " + new Date(d));
// get content length int len =
hpCon.getContentLength();
if(len == -1)
System.out.println("Content length unavailable."); else
System.out.println("Content-Length: " + len); if(len != 0) {
System.out.println("=== Content ==="); InputStream input =
hpCon.getInputStream();
int i = len;
while (((c = input.read()) != -1)) { // && (--i > 0)) {
System.out.print((char) c);
} input.close();

```
} else {
System.out.println("No content available.");
}
}
}
```

b) import java.io.*; import
java.util.Date;

class UCDemo

{

public static void main(String args[])

{

int c;

URL hp = new URL("http://www.internic.net"); URLConnection hpCon =
hp.openConnection();

// get date long d =
hpCon.getDate(); if(d==0)

System.out.println("No date information."); else

System.out.println("Date: " + new Date(d));

// get content type

System.out.println("Content-Type: " + hpCon.getContentType());

// get expiration date d =
hpCon.getExpiration();

if(d==0)

System.out.println("No expiration information."); else

System.out.println("Expires: " + new Date(d));

// get last-modified date d =
hpCon.getLastModified();

if(d==0)

System.out.println("No last-modified information."); else

System.out.println("Last-Modified: " + new Date(d));

```
// get content length int len =
hpCon.getContentLength();

if(len == -1)

System.out.println("Content length unavailable."); else

System.out.println("Content-Length: " + len); if(len != 0) {

System.out.println("=== Content ==="); InputStream input =
hpCon.getInputStream(); int i = len;

while (((c = input.read()) != -1)) { // && (--i > 0)) {

System.out.print((char) c);

} input.close();

} else {

System.out.println("No content available.");


}
```

c) import java.net.*; import
java.io.*; import java.util.Date;

```
class UCDemo

{

public static void main(String args[]) throws Exception

{

// get date long d =
hpCon.getDate(); if(d==0)

System.out.println("No date information."); else

System.out.println("Date: " + new Date(d));

// get content type

System.out.println("Content-Type: " + hpCon.getContentType());

// get expiration date d =
hpCon.getExpiration(); if(d==0)
```

```
System.out.println("No expiration information."); else

System.out.println("Expires: " + new Date(d));

// get last-modified date d =
hpCon.getLastModified(); if(d==0)

System.out.println("No last-modified information."); else

System.out.println("Last-Modified: " + new Date(d));

// get content length

int len = hpCon.getContentLength();

if(len == -1)

System.out.println("Content length unavailable."); else

System.out.println("Content-Length: " + len); if(len != 0) {

System.out.println("=== Content ===");

InputStream input = hpCon.getInputStream();

}

}

}
```

d)None of the above

35) Once you make a connection to a remote server, you can use _____ to inspect the properties of the remote object before actually transporting it locally.

a)URL

b)**URLConnection**

c)URL.Connection

d)None of these

36) We can create a URLConnection using the _____ method of a URL object and examine the document's properties and content:

a)openConnection( )

b)**URLConnection()**

c)both a and b

d)None of these

37) The _____ class is used to encapsulate both the numerical IP address and the domain name for that address.

a)IPAddress

b)MACAddress

c) InetAddress

d)None of the above

38) netAddress was specified by which type of IP version?

a) **IPv4**
b) IPv6
c) Both 1 & 2
d) None of the above

39) IPv6 uses a how many bit value to represent an address?

a) 32
b) 64
c) 16
d) **128**

40) IPv4 uses a how many bit value to represent an address?

a) **32**
b) 128
c) 64
d) 16

41) InetAddress can handle _____ IP addresses.

a)IPv4

b)IPv6

c)Both 1 & 2

d)None of the above

42) To create an InetAddress object you have to use one of the available _____
a)Interfaces
b)Classes
c)Factory Methods
d)All of the above

43) Which of the following is a correct factory method to create an object in an InetAddress?
a)getLocalHost()
b)getByName(String hostName)
c)getAllByName(String hostName)
d)All of the above

44) The _____ method returns the InetAddress object that represents the local host.
a)getByName(String hostName)
b)getLocalHost()
c)getAllByName(String hostName)
d)None of the above

45) The _____ method returns an InetAddress for a host name passed to it.
a)getByName(String hostName)
b)getLocalHost()
c)getAllByName(String hostName)
d)None of the above

46) The _____ factory method returns an array of InetAddresses that represent all of the addresses that a particular name resolves to.
a)getByName(String hostName)
b)getLocalHost()
c)getAllByName(String hostName)
d)None of the above

47) The factory method _____ which takes an IP address and returns an InetAddress object.

a)getByName(String hostName)

b)getLocalHost()

c)getAllByName(String hostName)

d)**getByAddress()**

48) Which of the following Instance Method returns true if this object has the same Internet address as other?

a)boolean isMulticastAddress()

b)boolean equals(Object other)

c)String toString()

d)byte[] getAddress()

49) Which of the following Instance Method returns a byte array that represents the object's Internet address in network byte order?

a)byte[] getAddress()

b)String getHostName()

c)String toString()

d)boolean isMulticastAddress()

50) Which of the following Instance Method returns a string that represents the host address associated with the InetAddress object?

a)String toString()

b)boolean equals(Object other)

c)String getHostAddress()

d)None of the above

51) Which of the following Instance Method returns a string that represents the host name associated with the InetAddress object?

a)String getHostAddress()

b)String getByName()

c)String getHostName()
d)String toString()

52) Which of the folowing Instance Method Returns true if this Internet address is a multicast address. Otherwise, it returns false?
a)boolean equals(Object other)

b)boolean isMulticastAddress()
c)boolean isMultiCastAddress()

d)boolean MulticastAddress()

53) Which of the following Instance Method returns a string that lists the host name and the IP address for convenience?
a)String getHostAddress()

b)String getByName()

c)String getHostName()

d)String toString()

54) _____ sockets are used to implement reliable, bidirectional, persistent, point-to- point, stream-based connections between hosts on the Internet.
a)TCP/IP
b)UDP

c)Proxy

d)None of the above

55) _____ can be used to connect Java's I/O system to other programs that may reside either on the local machine or on any other machine on the Internet.
a)Server

b)Client

c)Socket
d)All of the above

56) Which class is designed to be a "listener," which waits for clients to connect before doing anything?

a)Socket

b)Server

C)Both a & b

d)ServerSocket

57) Which are the two constructors used to create client sockets?

a)Socket(String hostName) ,Socket(InetAddress ipAddress, int port)

b)Socket(String hostName, int port) ,Socket(InetAddress ipAddress)

c)Socket(String hostName, int port), Socket(InetAddress ipAddress, int port)

d)None of the above

58) By using following methods a socket can be examined at any time for the address and port information associated with it.

a)InetAddress getInetAddress( )

b)int getPort( )

c)int getLocalPort( )

d)All of the above

59) Which of the following method returns the InetAddress associated with the Socket object?

a)InetAddress getInetAddress( )

b)int getPort( )

c)int getLocalPort( )

d)All of the above

60) Which of the following method returns the remote port to which this Socket object is connected?

a)InetAddress getInetAddress( )

b)int getPort( )

c)int getLocalPort( )

d)All of the above

61) Which of the following method returns the local port to which this Socket object is connected?

a)InetAddress getInetAddress( )

b)int getPort( )

c)int getLocalPort( )

d)All of the above

62) Which exception is thrown by the factory methods getLocalHost() & getByName() when they are unable to resolve the host name?

a)UnknownHostException

b)IOException

c)Both 1 & 2

d)None of the above

63) Which exception is thrown by getAllByName() factory method if it can't resolve the name to at least one address?

a)UnknownHostException

b)IOException

c)Both 1 & 2

d)None of the above

64) Which method returns the InputStream associated with the invoking socket?

a)InputStream getInputStream( )

b)OutputStream getOutputStream( )

c)Both 1 & 2

d)None of the above

65) Which method returns the OutputStream associated with the invoking socket?

a)InputStream getInputStream( )

b)OutputStream getOutputStream( )

c)Both 1 & 2

d)None of the above

66) Fill in the blanks in the below program:

```
import _____ class
InetAddressTest
{
public static void main(String args[]) throws _____ {
InetAddress Address = InetAddress.____();
System.out.println(Address);
Address = InetAddress._____("osborne.com");
System.out.println(Address);
InetAddress SW[] = InetAddress._____("www.nba.com");
for (int i=0; i<SW.length; i++)
System.out.println(SW[i]);
}
}
```

a)java.net.*; , UnknownHostException, getLocalHost(), getByName, getAllByName
b)UnknownHostException, getLocalHost(), getByName, getAllByName

c)java.net.* , UnknownHostException, getLocalHost(), getByName, getAllByName

d)None of the above

67) Which method returns a channel connected to the Socket object?

a)getChannel()
b)getByChannel()

c)getChannelName()

d)Both 1 & 2

68) Which exception is thrown by InetAddress getInetAddress( ), int getPort( ) , int getLocalPort( ) socket methods?

a) UnknownHostException

b) IOException

c) Both 1 & 2

d) None of the above

69) _____ method Creates a socket connecting the local host to the named host and port.

a) Socket(String hostName, int port)

b) Socket(InetAddress ipAddress, int port)

c) Both 1 & 2

d) None of the above

70) _____ method Creates a socket using a preexisting InetAddress object and a port.

a) Socket(String hostName, int port)

b) Socket(InetAddress ipAddress, int port)

c) Both 1 & 2

d) None of the above

71) Fill in the blanks wrt above program

import _____ class
WriteServer

{ public static int serverPort = 998; public static int clientPort = 999; public static int buffer_size = 1024; public static_____ ds; public static byte buffer[] = new byte[buffer_size]; public static void TheClient() throws Exception

{ while(____)

{

DatagramPacket p = new DatagramPacket(buffer, buffer.length); ds.receive(p);

System.out.println(new String(p.getData(), 0, p.getLength()));

} }

public static void main(String args[]) throws Exception

{

if(args.length == 1)

```
{
ds = new DatagramSocket(serverPort);
TheServer();
} else {
public static void TheServer() throws Exception
{ int pos=0; while
(true)
{
int c = System.in.read(); switch
(c) { case -1:
System.out.println("Server Quits.");
return; case '\r':
break; case '\n':
ds.send(new DatagramPacket(buffer,pos,InetAddress.getLocalHost(),clientPort)); pos=0;
break; default: buffer[pos++] = (byte) c;
}
} }
ds = new DatagramSocket(clientPort);
TheClient();
}
}
}
```

a)java.net.* ,DatagramSocket, true
b)java.sql.* ,DatagramSocket,true

c)java.net.* , DatagramPacket,,false

d) java.sql.* ,false, DatagramPacket,,false

72) ds.send(new DatagramPacket(buffer,pos,InetAddress.getLocalHost(),clientPort)); isused for
a)used to send data from client to server.

b)used to send data from server to client.

c)used to send data both ways
d)none of the above

73) _____object is the data container
a)DatagramSocket

b)DataContainer

c)DatagramPacket
d)none of the above

74) DatagramSocket is
a)used to send or receive the DatagramPackets.
b)serialized, predictable, reliable stream of packet data.

c)information passed between machines.

d)none of the above

75) Java implements datagrams by using class/classes
a) DatagramSocket
b)DatagramPackets

c)both a& b
d)none of the above

76) Which is the constructors of Datagram Packet class?
a)DatagramPacket(byte data[ ], int offset, int size)

b)DatagramPacket(byte data[ ], int size, InetAddress ipAddress, int port)

c)DatagramPacket(byte data[ ], int offset, int size, InetAddress ipAddress, int port)

d)All of the above

77) What does byte[ ] getData( ) method do?
a)Returns the byte array of data contained in the datagram
b)Returns the port number.

c)Returns data in form of string

d)returns length of data

78) which is the method of DataPacket class?
a)InetAddress getAddress( )

b)int getPort( )

c)int getLength( )

d)All of the above

79) Which method returns the destination InetAddress, typically used for sending.
a)int getPort( )
b)int getLength( )

c)InetAddress getAddress( )
d)byte[ ] getData( )

80) It provides a serialized, predictable, reliable stream of packet data
a)TCP
b)UDP

c)URL

d)Datagram

81) When the datagram is received, there is no assurance that it hasn't been damaged in trasmissin
a)TRUE
b)FALSE

82) Calling ServerSocket() constructor with port value 'zero' means _____.

a)use a port number that is automatically allocated.

b)use a local port

c)use server port

d)None of above

83) Which of these package contains classes and interfaces for networking?

a)java.util.*

b)java.networking.*

c)java.net.*

d)java.awt.*

84) A ServerSocket can connect to _____ clients.

a) single
b)Multiple

c) **both a and b**

d) none of above

85) A socket identifies _____ in network.

a) **a communication end point**

b) a communication start point

c)Intermediate nodes

d)None of above

86) A_____ is responsible for determining whether code executing in the Java runtime environment has permission to perform a security sensitive operation.

a) Permission object

b)security object

c)**policy object**

d)all of above

87) Connection oriented communication is possible using _____classes of Java.

a)**Socket and ServerSocket**

b)only Socket

c)only ServerSocket

d)DatagramPacket

88) byte[] getData() method of DatagramPacket class returns _____

a)Integer array of data contained in datagram

b)array of data contained in datagram

c)String array of data contained in datagram

d)**Byte array of data contained in datagram**

89) In UDP send() and receive() methods belong to which class?

a)**DatagramSocket**

b)DatagramPacket

c)Socket

d)ServerSocket

90) Java.net package include following classes.

a)URLConnection

b)Socket

c)InetAddress

d)**All of above**

91) Name the class which is used to create a port where the server will listen?

a)DatagramPacket

b)ServerSocket

c)Socket

d)URL

92) Port Number for FTP is

a) **21**

b) 80

c) 20

d) 40

93) Pretty Good Privacy (PGP) is used in security of

a)data

b)**Email**

c) Webpages

d)none of above

94) The constructor for ServerSocket are _____.

a)ServerSocket(int port, int maxQueue)

b)ServerSocket (int port)

c)ServerSocket(int port, int maxQueue, InetAddress localAddress)

d)**All of above**

95) The constructor which is used to create client socket is _____.

a)**Socket(InetAddress IPAddress, int port)**

b)ServerSocket (int port)

c)Socket(InetAddress IPAddress)

d)all of above

96) The factory method which returns an array of InetAddresses that represent all of the addresses that a particular host name resolves to.

a)**getAllByName( )**

b)getByName()

c)getLocalHost( )

d)getHostName( )

97) The java.net.InetAddress class provides methods to get the _____

a)IP of any host name

b)Host name

c)Domain name

d)All of above

98) URL class has several constructors; each can throw a _____.

a)**MalformedURLException**

b)IOException

c)URLException

d)SocketException

99) What is return type of getAddress() method of InetAddress class?

a)byte[]

b)String[]

c)Array[]

d)Double

100) What is Second part of URL address

a)Filename

b)**Hostname**

c)portnumber

d)protocol

101) What is the first part of URL address?

a)Filename

b)Hostname

c)portnumber

d)protocol

102) What is the optional part of URL Address

a)Filename

b)Hostname

c)portnumber

d)protocol

103) What is the return type of the method getAllByName( ) of InetAddress class?

a)byte[]

b)String

c)void

d)InetAddress[ ]

104) Which is the reliable protocol of networking ?

a)TCP

b)UDP

c)HTTP

d)FTP

105) Which method is used to know the full URL of an URL object?

a)**toExternalForm()**

b)ExternalForm()

c)getURL()

d)None of above

106) Which method is used to return the IPAddress of local machine

a)**static InetAddress getLocalHost( )**

b)static InetAddress getHost( )

c)static InetAddress getPort( )

d)All of above

107) Which method of ServerSocket will wait for a client to initiate communications and then communicate with the client

a)initialize()

b)start()

c)socket()

d)accept()

108) Which of the following class defines accept() method?

a)Socket

b)ServerSocket

c)DatagramPacket

d)DatagramSocket

109) Which of these is a bundle of information passed between machines?

a)Datagram

b)Frame

c)Packet

d)socket

110) which of these method of DatagramPacket is used to find the length of byte Array

a)getLength()

b)getlength()

c)getTotalLength()

d)getSize()

111) Which of these method of DatagramPacket is used to find the port number?

a)getPortNumber()

b)getport()

c)getPort()

d)getportNumber()

112) Which of these methods is used to know the type of content used in the URL?

a)**getContentType()**

b)getContentLength( )

c)getcontentType()

d)getcontenttype()

113) _____ class is used for accessing the attribute of a remote resource.

a)**URLConnection**

b)URL

c)url

d)Socket

114) _____ is abstract class for representing access to a system resource.

a)Security

b)**Permission**

c)Policy

d)None of above

115) _____ method of DatagramSocket class is used to receive DatagramPacket.

a)receive(DatagramPacket packet)

b)accept(DatagramPacket packet)

c)Receive(DatagramPacket packet)

d)Accept(DatagramPacket packet)

116) Choose the correct output import java.net.*; class myURL2

```
{
 Public static void main(String args[]) throws MalformedURLException
 {
URL u=new URL("http://www.msbte.com");
System.out.println("Protocol="+u.getProtocol());
System.out.println("Host Name="+u.getHost());
System.out.println("Port Number="+u.getPort());
System.out.println("File Name="+u.getFile());
 }
}
```

a) **Protocol=http Host Name=www.msbte.com Port Number= -1 File Name=**

b)Protocol=http Host Name=www.msbte.com Port Number= -1

c)Protocol=http Host Name=www.msbte.com File Name=

d)Protocol=http Host Name=www.msbte.com Port Number= 1 File Name=

117) Choose the correct output import java.net.*; class networking

```
{
 Public static void main(String args[])throws UnknownHostException  {
InetAddress obj1 =InetAddress.getByName("msbte.com");
InetAddress obj2 =InetAddress.getByName("msbte.com"); boolean x = obj1.equals(obj2);
System.out.print(x);
```

}

}

a)true

b)false

c)1

d)-1

118) If port number is not specified in the URL, getPort() method returns _____

a) 1

b)-1

c)0

d)blank

119) port number of Telnet is_____

a) 20

b) 21

c) **23**

d) 24

120) What is the default length of the queue in following constructor of Serversocket?
ServerSocket(int portno)

a) 80
b) 40

c) 60

d) **50**

121) Which Exception is thrown by DatagramSocket class constructor

a)DatagramSocketException

b)**SocketException**

c)MalformedURLException

d)URLException

122) which is not the method of security class?

a) SocketPermission()

b) FilePermission()

c) **DataPermission()**

d) Non Of Above

123) Which method is used to obtain the Portno of client in Client Program

a)getLocalPort()

b)getLocalHost()

c)getPort()

d)getHost()

124) Which method Returns the output stream of the URL connection ?

a)getInputStream()

b)getoutputStream()

c)getOutputStream()

d)getOutputstream()

125) Which minimum package statements are missing? class InetDemo

```
{
public static void main(String args[])
{ try
{
Socket s=new Socket("127.0.0.1",1234); InputStream ip=s.getInputStream(); OutputStream
op=s.getOutputStream(); Date d=new Date(); String datemsg=String.valueOf(d);
op.write(datemsg.getBytes()); op.close();
} catch(Exception e)
{
 System.out.println(e);
 }
 }
 }
```

a) import java.net.*;

b) import java.util.*;

c) import java.io.*;

d) **All of above**

126) What is the output of the following program?

```
import java.net.*; class
URLDemo
{
public static void main(String args[]) throws
```

```
MalformedURLException{
URL hp = new URL("http://www.Abc.com/downloads");
System.out.println("Protocol: " + hp.getProtocol());
System.out.println("Port: " + hp.getPort());
System.out.println("Host: " + hp.getHost());
System.out.println("File: " + hp.getFile());
System.out.println("Ext:" + hp.toExternalForm());
}
}
```

a)**Protocol: http Port: -1**
Host: www.Abc.com
File: /downloads
Ext:http://www.Abc.com/downloads

b)Protocol: http Port: -1

Host: www.Abc.com

c)Protocol,Port,Host,File,Ext

d)None of these

127) What will be the Output of the following program:

```
import java.net.*; class
InetAddressTest
{
```

```
public static void main(String args[]) throws UnknownHostException {
InetAddress Address = InetAddress.getLocalHost();
System.out.println(Address);
Address = InetAddress.getByName("osborne.com");
System.out.println(Address);
InetAddress SW[] = InetAddress.getAllByName("www.nba.com");
for (int i=0; i<SW.length; i++)
System.out.println(SW[i]);
}
}
```

a)default/206.148.209.138
osbrne.com/198.45.24.162 www.nba.com/64.241.238.153
www.nba.com/64.241.238.142

b)default/206.148.209.138
osborne.com/198.45.24.162
www.nba.com/64.241.238.142

c)default/206.148.209.138 osborne.com/198.45.24.162
www.nba.com/64.241.238.153 www.nba.com/64.241.238.142

d)None of the above

128) Find Error in following program.

```
import java.awt.*;
class URLDemo
{
public static void main(String args[]) throws URLException
{
```

```
URL hp = new URL("http://www.Abc.com/downloads");

System.out.println("Protocol: " + hp.getProtocol());

System.out.println("Port: " + hp.getPort());

System.out.println("Host: " + hp.getHost());

}

}
```

a)Error in package

b)Error in Exception

c)both a and b

d)none of the above

129) What is 1432 in following program.

```
import java.net.*;

import java.io.*;


public class SimpleClient

{

public static void main(String args[])

{ try

{

Socket s1=new Socket("127.0.0.1",1432);

InputStream is=s1.getInputStream();


BufferedReader br=new BufferedReader(new InputStreamReader(is));

System.out.println(br.readLine());

br.close();
s1.close(); }

catch(Exception e)

{}
```

```
}

}
```

a)localhost address

b)port no

c)ip address

d)server address

130) For following code, What will be the correct class to create the object hp?

```
 import java.net.*; class
urld {

public static void main(String args[]) throws MalformedURLException

{

_____  hp = new URL("http://www.google.com/java/javatutorial/");

System.out.println("Protocol: " + hp.getProtocol());

System.out.println("Port: " + hp.getPort());

System.out.println("Host: " + hp.getHost());

System.out.println("File: " + hp.getFile());

System.out.println("Ext:" + hp.toExternalForm());

}

}
```

a) URLConnection
b) **URL**
c) InetAddress
d) Datagram

131) What is the output of this program?

```
 import java.net.*; class
netdemo1

{
```

```
public static void main(String args[]) throws UnknownHostException {
InetAddress Address = InetAddress.getByName("cisco.com");
System.out.println(Address.getHostName());
}
}
```

a) cisco
b) **cisco.com**
c) www.cisco.com
d) None of the mentioned

132) What will be the output of following code?

```
import java.net.*;
class pURL
{
public static void main(String args[]) throws MalformedURLException {
URL hp = new URL("http://www.tutorialspoint.com/awt/awt_event_handling.htm");
System.out.println("Protocol: " + hp.getProtocol());
System.out.println("Port: " + hp.getPort());
System.out.println("Host: " + hp.getHost());
System.out.println("File: " + hp.getFile());
System.out.println("Ext:" + hp.toExternalForm());
} }
```

a)

```
Protocol: http
Port: 2
Host: tutorialspoint.com
File: /awt/awt_event_handling.htm
Ext:http://www.tutorialspoint.com/awt/awt_event_handling.htm
```

b)

Protocol: http
Port: -1
Host: www.tutorialspoint.com
File: /awt/awt_event_handling.htm
Ext:http://www.tutorialspoint.com/awt/awt_event_handling.htm c)
Protocol: http

Port: -1

Host: www.tutorialspoint.com

File: /awt/awt_event_handling.htm
Ext:http:awt_event_handling.htm d)
Protocol: www

Port: -1

Host: www.tutorialspoint

File: /awt/awt_event_handling.htm

Ext:http://www.tutorialspoint.com/awt/awt_event_handling.htm

133) In following code, which method is used for finding the content length?

import java.net.*; import
java.io.*;

import java.util.Date; class
UCDemo11

{

public static void main(String args[]) throws Exception

{ int c;

URL hp = new URL ("http://www.tutorialspoint.com/awt/awt_event_handling.htm");

URLConnection hpCon = hp.openConnection(); int len =
hpCon._____ ; System.out.println("Content-Length:
" + len);

} }

a) findContentType()
b) findContentLength()
c) **getContentLength()**
d) getContentType()

TY DIPLOMA - VIMP QUESTIONS
ADVANCE JAVA PROGRAMMING

Unit : 05
Interacting with database

1) The following classes belong to which package: Connection management, Database Access, Data Types, Database Metadata, Exceptions Warnings:
   a) javax.sql
   b) java.sql
   c) javax.swing
   d) None of the above

2) Which of the following classes/interface allows you to establish a connection to database:
   a) java.sql.Driver
   b) java.sql.DriverManager
   c) java.sql.connection
   d) **All of the above**

3) Which of the following classes/interface allows you to send SQL statement to database for execution and read the result:
   a) java.sql.Statement
   b) java.sql.PreparedStatement
   c) java.sql.ResultSet
   d) **All of the above**

4) What is the description of java.sql.DriverManager?
   a) **This class provides the functionality necessary for managing one or more database drivers.**
   b) This is an interface that abstracts the vendor specific connection protoco
   c) This class is used to discover properties required to obtain the connection
   d) This interface abstracts most of the interaction with the database.

5) What is the description of java.sql.Driver?

   a)This class provides the functionality necessary for managing one or more database drivers.

b) This is an interface that abstracts the vendor specific connection protocol

c)This class is used to discover properties required to obtain the connection
d)This interface abstracts most of the interaction with the database

6) What is the description of java.sql.DriverPropertyInfo?
a) This class provides the functionality necessary for managing one or more database drivers.
b) This is an interface that abstracts the vendor specific connection protocol
c) **This class is used to discover properties required to obtain the connection**
d)This interface abstracts most of the interaction with the database.

7) What is the description of java.sql.Connection?
a) This class provides the functionality necessary for managing one or more database drivers.
b) This is an interface that abstracts the vendor specific connection protocol
c) This class is used to discover properties required to obtain the connection
d) **This interface abstracts most of the interaction with the database.**

8) Which interface lets you execute SQL statements over the underlying connection and access the results?
a) java.sql.Statement
b) java.sql.PreparedStatement
c) java.sql.ResultSet
d)java.sql.CallableStatement

9) This interface lets you execute stored procedures.
a) java.sql.Statement
b) java.sql.PreparedStatement
c) java.sql.ResultSet
d) java.sql.CallableStatement

10) The java.sql package provides several Java _____ class that correspond to some of the SQL types.
a)Connection Management
b)Database Access
c) Data Types

d)Both (a) and (b)

11) Which interface is included in Data Types

a) java.sql.Types
b) java.sql.Date
c) java.sql.Blob
d) All of the above

12) Which interface comes under DatabaseMetadata?

a) java.sql.DatabaseMetadata
b) java.sql.ResultSetMetaData
c) Java.sql.ParameterMetadata
d) All of the above

13) Which of the following comes under SQL Exception and Warnings?

a) java.sql.SQLexception
b) java.sql.SQLWarning
c) java.sql.DataTruncation
d) **Both (b) and (c)**

14) Which of the following is/are charecteristics of JDBC

(1) Supports a wide level of portability.

(2) Provides Java Interfaces that are compatible with Java Applications . These providers are also responsible for providing the driver services.

(3) Provides higher level APIs for application programmers. The JDBC API specification is used as an interface for the applications and DBMS.

(4)The JDBC call to a Java application is made by the SQL Statements. These statements are resposible for entire communication with the database.

a)All of the above

b) None of the above

c) Only 2,3 and 4

d) Only 1,2 and 3

15) Following is/are the packages for JDBC

a) java.sql

b) javax.sql

c) Both a and b

d) None of the above

16) Which of the following is not a function of JDBC

a) Allows Java programs to execute SQL Statements

b) Allows Java programs to retrive the results of the SQL Statements

c) Allows Java programs to manipulate the design of the application

d) Allows Java programs to send user defined requests to the database

17) JDBC-ODBC Bridge can be availed by importing _____ package. a)

java.sql

b) sun.jdbc.odbc

c) java.jdbcodbc

d) None of the above

18) Which of the following is JDBC version was released with Java EE 6

a) JDBC 2.0

b) JDBC 3.0

c) JDBC 4.0

d) JDBC 4.1

19) JDBC API provides _____ and _____ to handle database specific call from users.

a) methods, functions

b) **classes, interfaces**

c) packages, classes

d) interfaces, packages

20) _____ has an important role in JDBC Architechtrue

a) Driver

b) SqlData

c) DriverManager

d) ResultSet

21) _____ refers to the bridge driver (JDBC-ODBC Bridge)

a. Type-1 Driver

b. Type-2 Driver

c. Type-3 Driver

d. Type-4 Driver

22) _____ refers to a partly java and partly native code driver

a. Type-1 Driver

b. Type-2 Driver

c. Type-3 Driver

d. Type-4 Driver

23) _____ refers to pure Java Driver that uses a middleware driver to connect to a database

a. Type-1 Driver

b. Type-2 Driver

c. Type-3 Driver

d. Type-4 Driver

24) _____ refers to pure Java Driver which is directly connected to a database

a. Type-1 Driver

b. Type-2 Driver

c. Type-3 Driver

d. Type-4 Driver

25) OCI stands for _____

a. Order Command Interface

b. Oracle Call Interface

c. Operation Command Interface

d. Open Command Interface

26) Which of the following is not the component of 3-tier architecture of JDBC

a. DataSource Object

b. Java Application

c. Java API

d. Database

27) The javax.sql provides ..... implementations which are used in building server-side applications.

a. JNDI-based lookup to access databases via logical names

b. Connection Pooling

c. Distributed transaction

d. The RowSet

e. all of the above

28) The javax.sql package is used to develop the client/server sided applications and provide server sided extension facilities. State the statement is true or false.

a. True

b. False

29) Using which classes and interfaces of javax.sql package we can establish and manage connection with the data source?

a. Only DataSource

b. DataSource and DataManager

c. DataSource and DriverManager

d. Only DriverManager

30) The ..... package provides a transparent meaning of connection pooling.

a. javax.swing

b. javax.servlet

c. javax.sql

d. java.io.sql

31) ..... implementation are provided by the driver vendor.

a. DataSource

b. DataManager

c. DriverManager.

d. None

32) The ..... Naming service is used to provide a logical name for the DataSource to make a connection.

a. Java Database Connectivity (JDBC)

b. Java Naming anfd Directory Interface (JNDI)

c. Open Database Connectivity (ODBC)

d. Domain Name Server

33) DataSource object can be impeemnted to work with.....

a. three tier infrastructure

b. two tier infrastructure

c. middle tier infrastructure

d. none

34) State true or false

" Connection Pooling means that the connection is reused rather than created each time when it is requested."

a True

b. false

35) ..... Sub class of statement interface.

a. Statement Interface

b. PreparedStatement Interface

c. CallableStatement Interface

d. PreparedStatement Class

36) Which method is invoked on the statement object by p-assing the SQL statements as parameter.

a. getXXX()

b. setXXX()

c. executeXXX()

d. All of the Above

37) ..... can be used to represent a precompiled query , which can be executed multiple times.

a. Statement Interface

b. PreparedStatement Interface

c. CallableStatement Interface

d. PreparedStatement Class

38) The excute method ..... while using the PreparedStatement objects.

a. takes nth number of parameters.

b. takes only one parameters.

c. **do no takes any parametrs.**

d. none of the above.

39) Which method is of the connection object is used to get the object of the PreparedStatement interface?

a. executeStatement()

b. updateStatement()

c. preparedStatement()

d. getStatement()

40) Which method is of the PreparedStatement object are used to set the parametrs of the SQL statements.

a. getXXX()

b. setParameter()

c.setPath()

d. setXXX()

41) We can execute the precompiled SQL statements by using

a. execute()

b. executeUpdate()

c. executeQuery()

d. **all of the above**

42) javax.sql package is also called as _____.

a.**JDBC extension API**

b. API

c.JDBC extension

d. none of the above

43) The javax.sql.DataSource interface represents _____ related to java application

a. **data sources**

b. data packets

c. socket

d. all of the above

44) The javax.sql.CommonDataSource provides the methods that are common between _____ interfaces

a) DataSource,XADataSource,ConnectionPoolDataSource

b) Data.XAData,ConnectionPoolData

c) Source.XASource,ConnectionPoolSource

d) source.xadatasource,connectionpoll

45) Connections made by using _____ objects are implemented on the middle-tier connection pool.

a) DataSource

b) datasopurce

c) datasorce d)none of the above

46) _____ provides a factory for PooledConnection objects.

a) javax.sql.ConnectionPoolDataSource

b) java.sql.ConnectionPoolDataSource

c) javax.sqlConnectionPoolDataSource

d) javax.sql,ConnectionPoolDataSource

47) _____ provides an object to manage connection pools.

a) javax.sql.ConnectionPoolDataSource

b) javax.sql.PooledConnectionInterface

c) javax.sqlConnectionPoolDataSource

d) javax.sql,ConnectionPoolDataSource

48) DriverManager is a …….. class in JDBC API.

a) Abstract

b) **Non-Abstract**

c) main

d) none of the above

49) How many constructor are in the DriverManager Class?

a) One

b) two

c) three

d) four

50) What is the Driver Interface used for?

a) To create Database Object that provide and entry point for database connectivity

b) to contains the results of executing an SQL query.

c) **To create Connection Object that provide an entry point for database connectivity.**

d) none

51) What is connection interface?

a) **It is a standard type that defines an abstraction to access the session established with database server.**

b) It is used to create Connection Object that provides an entry point for database connectivity.

c) both a and b

d) none of the above

52) The Connection interface provides methods to handle

a) ResultSet Object

b) **Connection Object**

c) PreparedStatement Object

d) none of the above.

53) What does the Statement interface return?

a) ResultSet Object

b) **Connection Object**
c) PreparedStatement Object

d) None of the above

54) How many times the query is compiled when used PreparedStatement?

a) **once**

b) twice

c) thrice

d) quadriple

55) How to create an object and execute a query in PreparedStatement?

a)PreparedStatement stmt=new PreparedStatement("insert into Emp values(a,b)");

b) **PreparedStatement stmt=con.prepareStatement("insert into Emp values(a,b)");**
c)PreparedStatement stmt=con.prepareStatement("insert into Emp values(a,b)");

d)none of the above

56) Syntax for ResultSet:

a) ResultSet rs=new ResultSet(“Query-to-be-executed");

b)resultset rs=stmt.executequery(“Query-to-be-executed");

c)ResultSet rs=stmt.executeQuery(“Query-to-be-executed");
d) None of the above

57) What is "jdbc:mysql://localhost:3306/sonoo","root","root"

a)**path to the database**
b)driver location

c)both a and b

d)None of the above

58) Steps for jdbc:

(i) Closing the connection

(ii) Executing SQL statements

(iii) Obtaining a connection

(iv) Creating a JDBC Statement object

a) i, ii, iii, iv

b) ii, iii,iv,i

c) iii, iv , ii , i

d) iii, ii, iv, i

59) Path for Data Sources(ODBC):

a)start->control panel -> system and security -> data sources

b)**start ->control panel->system and security->administrative tools->data sources**

c)start->control panel->data sources

d)start->accessories->data sources

60) Which driver is required for ODBC connectivity

a)MicrosoftODBCForOracle

b) **MicrosoftODBCforOracle**

c) MicrosoftOdbcForOracle

d) MicrosoftOdbcforOracle

61) A Java program cannot directly communicate with an ODBC driver because .....

a) ODBC written in C language
b) ODBC written in C# language
c) ODBC written in C++ language
d) ODBC written in Basic language

62) The JDBC-ODBC Bridge driver translates the JDBC API to the ODBC API and used with _____

a) JDBC drivers
b) ODBC drivers
c) Both A and B
d) None of the above

63) The ..... package contains classes that help in connecting to a database, sending SQL statements to the database, and processing the query results.

a) connection.sql
b) db.sql
c) pkg.sql
d) java.sql

64) The ..... method executes a simple query and returns a singleResult Set object.
a) executeUpdate()
b) executeQuery()
c) execute()
d) noexecute()

65) The ..... method executes an SQL statement that may return multiple results.
a) executeUpdate()
b) executeQuery()
c) execute()
d) noexecute()

66) The ..... object allows you to execute parametrized queries.
a) ResultSet
b) Parametrized
c) PreparedStatement
d) Condition

67) The ..... object provides you with methods to access data from the table.
a) ResultSet
b) Parametrized
c) TableStatement
d) Condition

68) The parameters of the PreparedStatement object are ..... when the user clicks on the Query button.
a) initialized
b) started
c) paused
d) stopped

69) The ..... method sets the query parameters of the PreparedStatement Object. a) putString()
b) insertString()
c) setString()
d) setToString()

70) Connection object can be initialized using the ..... method of the Driver Manager class.
a) putConnection()
b) setConnection()
c) Connection()
d) getConnetion()

71) Which of the following statements is false as different type of statements is concern in JDBC?

a)Regular Statement

b)Prepared Statement

c)Callable Statement

d)Interim Statement

72) JDBC-ODBC bridge supports multiple concurrent open statements per connection?

a)True

b)False

73) Which driver is efficient and always preferable for using JDBC application:

a)Type – 4

b)Type – 1

c)Type – 3

d)Type – 2

74) A Java program cannot directly communicate with an ODBC driver because

a) Statement is wrong

b) ODBC written in C language

c) ODBC written in high level language

d) None of above

75) Abbreviate the term DSN

a) Data Source Name

b) Data Server name

c) Database Server Name

d) Data String Name

76) API stands for

a)Applet Program Interface

b)A**pplication Program Interface**

c)Application procedure Interface

d) None of above

77) API which controls access to the row result of a given Statement and holds data retrieved from a database after you execute an SQL query using Statement objects.

a)java.Sql

b)java.ResultSet

c)Java.sql

d)**java.sql.ResultSet**

78) Application Server used in _____.

a) Three-Tier Mode

b) Two-Tier Mode

c) Milti-Tier Mode

d) Single-Tier Mode

79) Backbone of JDBC Architecture is _____

a) Driver Manager

b) Database Manager

c) Statement Interface

d) Resultset Interface

80) Which method is used to modify stored data

a)execute()

b)executeQuery()

c)executeUpdate()

d)executeResult()

81) How many transaction isolation levels are defined in sql.connection interface?

a)4

b)3

c)5

d)2

82) Which of the following is false as type 4 driver is concern

a)Type 4 driver is “native protocol, pure java” driver

b)Type 4 drivers are 100% Java compatible
c)Type 4 drivers uses Socket class to connect to the database
d)**Type 4 drivers can not be used with Netscape**

83) All raw data types should be read and uploaded to the database as an array of :

a)byte

b)int
c)boolean
d)char

84) Which method is used to perform DML statements in JDBC

a)execute()

b)executeQuery()

c)executeUpdate()
d)None of above

85) JDBC stands for:

a)Java Database Connectivity
b)Java Database Components

c)Java Database Control

d)None of the above is correct.

86) The ..... method executes a simple query and returns a single Result Set object.

a)executeUpdate()

b)**executeQuery()**
c) execute()

d) noexecute()

87) The ..... object allows you to execute parametarized queries.

a)ResultSet

b)Parametrized

c)PreparedStatement

d)Condition

88) Three methods are central to the life cycle of a servlet _____, _____ and _____ .

a)init(), main(), run()

b)init(), service(), destroy()

c)init(), run(), stop()

d)init(), start(), stop()

89) What is data in following program.

```
import java.sql.*;
public class
Selectdbase
{
public static void main(String args[])
{
t
r
y
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection con=DriverManager.getConnection("jdbc:odbc:data");
Statement st=con.createStatement();
ResultSet rs=st.executeQuery("select * from emp");
while(rs.next())
{
System.out.println("id:"+rs.getInt("id"));
System.out.println("salary:"+rs.getString("salary"));
System.out.println("name:"+rs.getString("name1"));
}
con.close();
}
catch(SQLException e)
```

```
{}
catch(Exception e)
{
}
}}
```

a)Driver name
b)database name
c)table name
d)data source name

90) What is output of following program.

```
import java.sql.*;
public class jdbc1
{
public static void main(String args[])
{
t
r
y
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection con=DriverManager.getConnection("jdbc:odbc:data");
Statement st=con.createStatement();
ResultSet rs=st.executeQuery("select * from student");
while(rs.next())
{
System.out.println("id:"+rs.getInt("id"));
System.out.println("name:"+rs.getString("name1"));
}
con.clo
se();
}
catch(SQLException e)
{}
catch(Exception e)
{
}
}}
```

a) "id:"+rs.getInt("id"
b)name:"+rs.getString("name1
c) both a and b
d)values of id and name will be displayed from table.

91) Which is the correct method ********for updating table rows from given option.

```
import java.sql.*;
public class
Updatedbase
{ public static void main(String
args[])
{
t
r
y
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection con=DriverManager.getConnection("jdbc:odbc:data");
Statement st=con.createStatement();
ResultSet rs=st.executeQuery("select * from emp");

St.********("Update emp set name1='jack' where id=2");
System.out.println("\n\nafter change");
rs=st.executeQuery("select * from emp");
while(rs.next())
{
System.out.println("id:"+rs.getInt("id"));
System.out.println("salary:"+rs.getString("salary"));
}
con.clo
se(); }
catch(SQLException e){}
catch(Exception e)
{
}
}}
```

a)executeUpdate()
b)executeQuery()
c)execute()
d)ExecuteUpdate()

92) Find out error from following

```
code. import java.sql.*;
public class Updatedbase
{
public static void main(String args[])
{
t
r
y
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection con=DriverManager.getConnection("jdbc:odbc:data2");
Statement st=con.createStatement();
resultSet rs=st.executeQuery("select * from college");

while(rs.next())
{
System.out.println("id:"+rs.getInt("id"))
System.out.println("name1:"+rs.getString("name1"));
}
con.clo
se(); }
catch(SQLException e)
{}
catch(Exception e)
{
}
}}
```

a)Error in resultSet statement.
b)Error in while loop
c)both a and b
d)Error in catch()

93) Find out error in following program.

```
import java.sql.*;
public class
Selectdbase
{
```

```
public static void main(string args[])
{
t
r
y
{
Class.forname("sun.jdbc.odbc.JdbcOdbcDriver");
Connection con=DriverManager.getConnection("jdbc:odbc:data");
Statement st=con.createStatement();
ResultSet rs=st.executeQuery("select * from emp");
while(rs.next())
{
System.out.println("id:"+rs.getInt("id"));
System.out.println("salary:"+rs.getString("salary"));
System.out.println("name:"+rs.getString("name1"));
}
con.clo
se(); }
catch(SQLException e)
{}
catch(Exception e)
{
}
}}
```

a)Error in main()
b)Error in try block
c)both a and b
d)Error in catch block

94) Find the error in following code:

```
import java.sql.*;
class MysqlCon
{
public static void main(String args[])
```

```
{
t
r
y
{
Class.forName("com.mysql.jdbc.Driver");
Connection con=DriverManager.getConnection (
"jdbc:mysql://localhost:3306/sonoo","root","root");
//here sonoo is database name, root is username and password
Statement stmt=con.createStatement();
ResultSet rs=stmt.executeQuery("select * from emp");
while(rs.next())
System.out.println(rs.getInt(1)+"  "+rs.getString(2)+"  "+rs.getString(3));
  }
catch(Exception e)
{
 System.out.println(e);
}
 }
  }
```

a.missing semicolon

b.**Connection is not terminated.(con.close())**

c.missing brackets

d. both a and c

e. all of the above

95) Find the error in following code:

```
import java.sql.*;
class MysqlCon{
public static void main(String args[]){
try{
Class.forName("com.mysql.jdbc.Driver");
Connection con=DriverManager.getConnection(
"jdbc:mysql://localhost:3306/sonoo","root","root");

//here sonoo is database name, root is username and password
Statement stmt=con.createStatement();
ResultSet rs=executeQuery("select * from
emp");  while(rs.next())
System.out.println(rs.getInt(1)+"  "+rs.getString(2)+"  "+rs.getString(3));
con.close();  } catch(Exception e){  System.out.println(e);}
 } }
```

a)missing semicolon
b)incorrect syntax of Statement
c)incorrect syntax of ResultSet
d)both a and c
e)all of the above

96) Find the error in following code:

```
import java.sql.*;

class MysqlCon{  public static

void main(String args[]){

try{

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection(
```

```
"jdbc:mysql://localhost:3306/sonoo","root","root");
//here sonoo is database name, root is username and password
Statement stmt=createStatement();
ResultSet rs=executeQuery("select * from emp");
while(rs.next())
System.out.println(rs.getInt(1)+"   "+rs.getString(2)+"   "+rs.getString(3));
con.close();   } catch(Exception e){
 System.out.println(e)}
  } }
```

a.missing semicolon

b.incorrect syntax of Statement

c.incorrect syntax of ResultSet

d. both a and c

e. all of the above

97) Find the error in following code:

```
import java.awt.*; class JdbcDemo
{ public  static void main(String a[])
{
try {
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
System.out.println("Driver Loaded");
String url="jdbc:odbc:StudInfo";
```

```
Connection con=DriverManager.getConnection(url);

System.out.println(" Connection to DataBase created");

Statement stmt = con.createStatement();

String query = "select * from StudTable";

ResultSet rs = stmt.executeQuery(query);

while(rs.next()){

System.out.println(" ID : "+ rs.getInt(1));

System.out.println(" Name : "+ rs.getString(2)); System.out.println(" Marks : "+
rs.getInt(3));                                    System.out.println();

}}

catch(ClassNotFoundException e)

{

e.printStackTrac

e();  }

 catch(SQLException e)

{

e.printStackTrace();

}}}
```

a)missing semicolon

b)missing bracket

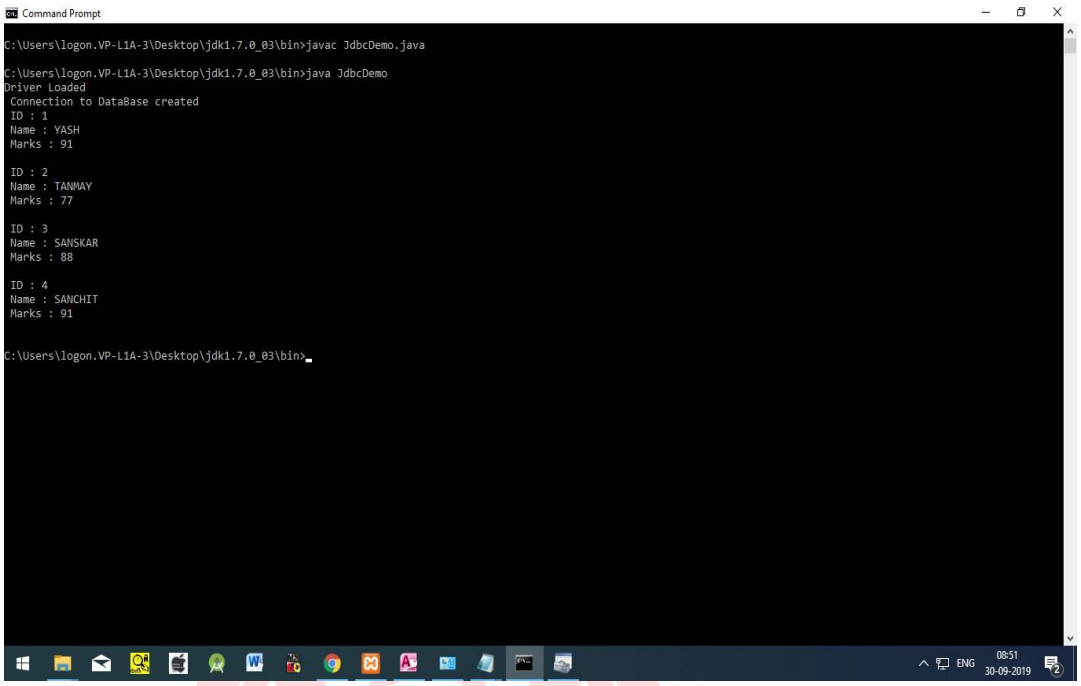
c)incorrect syntax of resultset

d) **None of the above**

98) What will be the output of the code considering the database is created:

```
import java.sql.*; class JdbcDemo  {

public  static void main(String a[])  {

  try {

  Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

  System.out.println("Driver Loaded");

  String url="jdbc:odbc:StudInfo";

  Connection con=DriverManager.getConnection(url);

  System.out.println(" Connection to DataBase created");

  Statement stmt = con.createStatement();

  String query = "select * from StudTable";

   ResultSet rs = stmt.executeQuery(query);


  while(rs.next()){

  System.out.println(" ID : "+ rs.getInt(1));

   System.out.println(" Name : "+ rs.getString(2));

  System.out.println(" Marks : "+ rs.getInt(3));
   System.out.println();

   }}

   catch(ClassNotFoundException e)

   {

   e.printStackTrace();

  }

  catch(SQLException e)

 {

   e.printStackTrace();

   }}}
```

a.



b.it contains error

c.none of the above

99) Find the error in following code:

```
import java.awt.*; class JdbcDemo
{ public  static void main(String a[])
{
  try {
  Class.forName("sun.jdbc.odbc");
  System.out.println("Driver Loaded");
  String url="jdbc:odbc:StudInfo";
  Connection con=DriverManager.getConnection(url);
```

```
System.out.println(" Connection to DataBase created");

Statement stmt = con.createStatement();

String query = "select * from StudTable";

ResultSet rs = stmt.executeQuery(query);


while(rs.next()){

System.out.println(" ID : "+ rs.getInt(1));

System.out.println(" Name : "+ rs.getString(2));

System.out.println(" Marks : "+ rs.getInt(3));
System.out.println();

}}

catch(ClassNotFoundException e)

{

e.printStackTrac

e(); }

catch(SQLException e)

{

e.printStackTrace();

}

}

}
```

a.missing semicolon

b.missing bracket

c.**driver is not suitable**

d. None of the above

100) boolean isLast() method defines _____

a) Determines whether the ResultSet cursor points to the second last row

b) Determines whether the ResultSet cursor points to the last statement

c) Determines whether the ResultSet cursor points to the last Column

d) **Determines whether the ResultSet cursor points to the last row**

101) Callable statement object in JDBC is used to execute a call to

a) stored procedure

b) Statement

c) Prepared Statement

d) Procedure

102) CallableStatement is used to execute_____

a) stored procedure

b) Statement

c) Prepared Statement

d) Procedure

103) Class.forName("_____")

a) sun.jdbc.JdbcOdbcDriver

b) jdbc.odbc.JdbcOdbcDriver

c) sun.jdbc.odbc.JdbcOdbcDriver

d) sun.jdbc.odbc.JdbcOdbc

104) class.forName() method throws

a) NotFoundException

b) **ClassNotFoundException**

c) SQLException

d) Cant Throws any exception

105) Connection object can be initialized using the _____ method of the DriverManager Class.

a) getConnection()

b) getManager()

c) getconnection()

d) Getmanager()

106) createStatement() method without any parameter is used to create a statement with forward only and read only ResultSet Database meta data are retrieved through _____.

a) PreparedStatement object
b) Statement object

c) Connection object

d) CollableStatement object

107) DELETE statement of an SQL is executed by _____.

a) executeUpdate()
b) executeQuery()

c) Execute()

d) executeStatement()

108) DriverManager.getConnection(_____ , _____ , _____) What are the two parameters that are included ?

a) User ID,URL or machine name where server runs, Password

b) URL or machine name where server runs, Password, User ID,

c) URL or machine name where server runs, User ID, Password
d) Database name, User ID, Password

109) Every driver must provide a class that should implement the _____

a) Driver interface

b) Connection Interface

c) Statement Interface

d) Database Interface

110) Following is return type of executeUpdate( ):

a) String

b) Array

c) byte

d) Int

111) execution of delete SQL query in JDBC, ... method must be used. a)

execute()

b) executeQuery()

c) executeUpdate()

d) executeStatement()

112) For execution of INSERT SQL query in JDBC, ... method must be used.

a) executeUpdate()

b) executeStatement()

c) executeQuery()

d) Execute()

113) For execution of SELECT SQL query in JDBC, _____ method must be used.

a) executeQuery()

b) Execute()

c) executeUpdate()

d) executeAll()

114) forName is a_____type method

a) Static
b) Dynamic

115) getConnection() method of which class

a) DriverManager
b) Statement

c) Connection

d) None of above

116) How many JDBC driver types are available by sun Microsystem?

a) 6

b) 4
c) 5

d) 3

117) If a PreparedStatement is a SQL SELECT statement, you execute the statement using _____.

a) PreparedStatement.execute();

b) Statement.executeQuery();

c) PreparedStatement.executeQuery();
d) PreparedStatement.executeUpdate();

118) If you need to use a stored procedure with output parameters, which of the following statement type should be used to call the procedure?

a) CallableStatement
b) PreparedStatement

c) Statement

d) ProcedureStatement

119) In 2-tier architecture , the first tier is generally_____

a) GUI

b) Server

c) Client

d) database

120) In the following JDBC drivers which is known as partly java driver?

a) Pure-Java Driver

b) JDBC-net Pure Java

c) JDBC driver

d) Native-API driver

121) In the three tier model the middle tier of the services acts as a mediator between _____ and _____.

a) **Java application and databases**

b) Client and Server

c) Java application and client

d) Java application and server

122) JDBC is a ---------- interface, which means that it is used to invoke SQL commands directly.

a) high level

b) **low level**

c) middle level

d) top level

123) JDBC Stands for

a) Java Database Connection

b) **Java Database Connectivity**

124) JDBC-ODBC bridge product provide _____ access via _____.

a) JDBC driver, JDBC drivers

b) ODBC drivers, JDBC driver

c) **JDBC driver, ODBC drivers**

d) None

125) Methods of ResultSet() throws _____

a) IOException

b) Exception

c) DatabaseException

d) SQLException

126) Name the type number of driver belongs to JDBC ODBC Bridge driver?

a) Type 1

b) Type 2

c) Type 3

d) Type 4

127) Native API converts _____into the _____ used by DBMS.

a) Native API calls,JDBC API

b) JDBC API, Native API calls

c) JDBC API, pure API calls

d) JDBC API, pure Java

128) Native – protocol pure Java converts ……. in to the …… used by DBMSs directly.

a) JDBC calls, network protocol

b) Native API calls,JDBC API

c) JDBC API, pure API call

d) JDBC API, Native API calls

129) ODBC Drivers are platform _____

a) Dependent

b) Independent

c) Both a and b

d) None

130) ODBC requires configuring _____ which represents the target database.

a) Data String Name

b) Data Source Name

c) Domain Name

d) Database name

131) The PreparedStatement_____symbol is placeholder that is replaced by the input parameter at seen time.

a) ?

b) *

c) &

d) #

132) The ResultSet _____provides methods for retrieving and manipulating the results of executed queries.

a) Statement

b) Package

c) Class

d) Interface

133) The Type 3 architecture is _____

a) **JDBC-Net pure Java**
b) JDBC-ODBC Bridge Driver

c) Native API partly Java driver

d) Native Protocol Pure-Java Driver

134) The_____ is the language for interacting with Database.

a) Structured Query Language
b) Data Manipulation Language

c) Data Definition Language

d) Stored Query Language

135) Type 1 driver is also known as

a) Native API partly Java driver

b) Native Protocol Pure-Java Driver

c) **JDBC-ODBC Bridge Driver**

d) JDBC-Net pure Java

136) Type 3 driver is also known as_____

a) **JDBC-net Pure Java**

b) JDBC-Net pure Java

c) Native API partly Java driver

d) Native Protocol Pure-Java Driver

137) Type II JDBC driver is also known as _____

a) JDBC-net Pure Java

b) JDBC-Net pure Java

c) **Native API partly Java driver**

d) Native Protocol Pure-Java Driver

138) Type IV JDBC driver is also known as _____

a) DBC-net Pure Java

b) JDBC-Net pure Java

c) Native API partly Java driver

d) **Native Protocol Pure-Java Driver**

139) Type4 driver is also known as:

a) 100% Pure Java

b) DBC-net Pure Java

c) JDBC-Net pure Java

d) Native API partly Java driver

140) Where the object of ResultSet maintains a cursor?

a) Second Row

b) **First Row**

c) Last Row

d) Middle Row

141) Which class is used to connect java application to JDBC driver

a) DriverManager

b) Connection

c) Statement

d) ResultSet

142) Which driver is called as a thin driver in JDBC?

a) Type-2 Driver

b) Type-3 Driver

c) Type-1 Driver

d) **Type-4 Driver**

143) Which interface provides methods to execute queries with the database?

a) **Statement interface**

b) Connection

c) Resultset

d) DriverManager

144) Which Interface is used to execute dynamic SQL statements?

a) PreparedStatement

b) Statement

c) CallableStatement

d) Procedurestatement

145) Which is default ResultSet type

a) TYPE_SCROLL_SENSITIVE

b) TYPE_SCROLL_INSENSITIVE

c) TYPE_FORWARD_ONLY

d) TYPE_SCROLL_FIRST

146) Which JDBC driver Type (s) can be used in either applet or servelt code? a)

Type 2

b) Type 3

c) Type 4

d) Both Type 3 and Type 4

147) Which JDBC driver Type(s) can you use in a three-tier architecture and if the Web server and the DBMS are running on the same machine?

a) Type 3

b) Type 2

c) Type 4

d) Both Type 3 and 4

148) Which JDBC driver Types are used for over communications networks? a) Type 2

b) Type 3

c) Type 4

d) **Both Type 3 and Type 4**

149) Which method is used to obtain count of total rows of ResultSet

a) getRow()
b) getrow()

c) GetRow()

d) Row()

150) Which method is used to perform DML statements in JDBC?

a) executeQuery()

b) execute()

c) executeUpdate()
d) executeAll()

151) Which method of class is used to register & dynamically load the driver class?

a) forName()

b) DeviceManager

c) Statement

d) Resultset

152) which of the following function is used to find the column count of the particular ResultSet ?

a) getColumnCount()
b) getRow

c) getcolumnCount()

d) getcolumncount()

153) Which of the following invokes functions in sql ?

a) **Callable statements**
b) Prepared Statement
c) Procedure Statement
d) Statement

154) Which type of driver converts JDBC calls into the network protocol used by the database management system directely

a) Type 4
b) Type 3
c) Type 2
d) Type 1

155) Which type of driver is unqiue in JDBC?

a) Native Protocol Pure-Java Driver
b) **JDBC-Native API**
c) Native API partly Java driver
d) JDBC-ODBC Bridge Driver

156) _____ calls get converted into native C or C++ API calls.

a) Native Protocol Pure-Java Driver
b) **JDBC-Native API**
c) Native API partly Java driver
d) JDBC-ODBC Bridge Driver

157) _____ interface allows storing results of query?

a) ResultSet
b) Connection

c) Statement

d) DriverManager

158) All raw data types including binary documents or images should be read and uploaded to the database as an array of ….

a) Byte
b) Int

c) String

d) Array

159) Analyse the following code and fill the appropriate statement in the blanks

```
import java.sql.*; class DB {  public static void
main(String args[])throws Exception
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection con=DriverManager.getConnection("jdbc:odbc:mydsn");
Statement stmt=con.createStatement();
ResultSet rs=stmt._________________("select * from empwhere id=101");
System.out.println("EmpName\tDesignation\tSalary");
where(rs.next())
{
System.out.println(rs.getString(2)+"\t"+rs.getString(3)+"\t"+rs.getInt(4));
```

```
}
con.clo
se();
 }
 }
```

Ans: executeQuery()

160) Choose missing statements in following code from given options.

```
import java.sql.*;
Class DemoFetch
{ public static void main(String
args[])
{
Connection con;
Statement stmt; ResultSet rs;
String qry, url;  try {
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver"
); url="Jdbc:Odbc:studdsn";
con=DriverManager.getConnection(url,"","");
stmt=con.createStatement();  qry="select *
from stud"; rs=_________________;
System.out.println("Roll
No\tName\tPercentage"); while(rs.next())  {  int
rno=rs.getInt("roll"); String
```

```
nm=rs.getString("sname");  double

per=rs.getDouble("per");

System.out.println(rno+"\t"+nm+"\t"+per);

}

con.clo

se();

 }

catch(Exception e){}

}

}
```

Answer: stmt.executeQuery(qry)

161) Choose missing statements in following code from given options.

```
import java.sql.*;

Class DemoFetch1

{

public static void main(String args[])

{

Connection con;

PreparedStatement pstmt;

ResultSet rs;

String qry,url;

try

{
```

```
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

url="Jdbc:Odbc:studdsn";

con=DriverManager.getConnection(url,"","");

qry="select * from stud";

pstmt=con.prepareStatement(qry);

rs=_____;

System.out.println("Roll No\tName\tPercentage");

while(rs.next())

{ int

rno=rs.getInt("roll"

);

String nm=rs.getString("sname");

double per=rs.getDouble("per");

System.out.println(rno+"\t"+nm+"\t"+per);

}

con.clo

se();

}

catch(Excepti

on e)

{

System.out.println(e.toString());

}

}

}
```

Answer: pstmt.executeQuery(qry)

162)

```
Connection con = DriverManager.getConnection ("jdbc:odbc:wombat", "login", "password");
Statement stmt = con.createStatement();
ResultSet rs = stmt.executeQuery("SELECT a, b, c FROM Table1");
while (rs.next())
{
Integer x = rs.getInt("c");
String s = rs.getString("a");
Float f = rs.getFloat("b");
}
```

What is WRONG with the code above?

Answer: Retrieval of the fields is in the wrong order.

163) Connection object can be initialized using the ..... method of the Driver Manager class.

a) **getConnection()**
b) Getconnection()
c) getStatement()
d) Getstatement()

164) Consider following code and state missing code ?

```
import java.sql.*;
class exp2
```

```
{ public static void main(String args[])throws
Exception
{
 try
{
Class.forName("_____");
Connection con=DriverManager.getConnection("Jdbc:Odbc:demo1dsn");
Statement st=con.createStatement();
ResultSet rs=st.executeQuery("select * from Table1");
System.out.println("After insertion of new record");
while(rs.next())
{
System.out.println(rs.getString(1));
System.out.println(rs.getString(2));
System.out.println(rs.getString(3));
}
st.executeUpdate("insert into Table1 values('jasmine',10,'banglore')");
ResultSet rs1=st.executeQuery("select * from Table1 "); while(rs1.next())
{
System.out.println(rs1.getString(1));
System.out.println(rs1.getString(2));
System.out.println(rs1.getString(3));
}
}
catch(Exception e) {}
}
}
```

Answer: sun.jdbc.odbc.JdbcOdbcDriver

165) Consider the following program. What should be the correction done in the program to get correct output?

```
import java.sql.*;
public class
dbAccess
{
public static void main(String[] args)throws Exception
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection conn =DriverManager.getConnection("jdbc:odbc:ab", "", "");
Statement s = conn.createStatement();
String s1="update Table1 set name1='akash' where rollno='1'";
s.executeQuery(s1); s.close();
conn.close();
}
}
```

Answer: s.executeUpdate(s1)

166) Consider the following program. What should be the method used in following program to get correct output?

```
import java.sql.*;
class Ddemo1
{
public static void main(String args[]) throws Exception
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection c=DriverManager.get**********("jdbc:odbc:ODSN"," "," ");
Statement s=c.createStatement();
ResultSet rs=s.executeQuery("select *from StudTable");
System.out.println("Name" + " \t " + "Roll_No" + " \t " + "Avg");  while(rs.next())
{
System.out.println(rs.getString(1)+" \t "+rs.getInt(2)+" \t \t"+rs.getDouble(3));  s.close();
c.close();
}
}
}
```

Answer: Connection()

167) Consider the following program. Select the statement that should be added to get correct output.

```
import java.sql.*;
class DBEx
{
```

```
public static void main(String args[])
{
Try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection c =DriverManager.getConnection("jdbc:odbc:mydsn","","");
ResultSet rs = s.executeQuery("select * from StudTable");  while(rs.next())
{
System.out.println("Roll No.: "+rs.getInt(1));
System.out.println("Name : "+rs.getString(2));
System.out.println("Branch :"+rs.getString("Branch")+"\n");
}
s.close();
c.close();
}
catch(Excepti
on e)
{
System.out.println("Caught: "+e);
}
}
}
```

Answer: Statement s = c.createStatement();

168) Consider the following program. Which two exceptions are thrown?

```
Package javaapplication21;  import java.sql.*;  public

class db15 { public static void main(String args[])

throws _____, _____

{

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

Connection c=DriverManager.getConnection("jdbc:odbc:MyDSN","","");
System.out.println("Connection Established");

}

}
```

Answer: ClassNotFoundException, SQLException

169) Invoking Class.forName(mydriver) throws _____.

a) ClassNotFoundException

b) IOException

c) SQLException

d) **Both a and c**

170) Permission class is part of_____package

a) **java.security.permission**
b) java.security

c) java.permission

d) None of above

171) PreparedStatement updateemp = con.prepareStatement("insert into emp values(?,?,?)"); How many values are need to insert for prepareStatement paprameter?

a) 3

b) 4

c) 2

d) 1

172) Set XXX() method binds values to the parameters. Where XXX represents

a) Data Type

b) Method

c) Interface

d) Class

173) The JDBC-ODBC bridge is:

a)Three tiered

b)Multithreaded

c)Best for any platform

d)All of the above

TY DIPLOMA - VIMP QUESTIONS
ADVANCE JAVA PROGRAMMING

Chapter 06

1) In the _____ method of servlet connection is created.

a) destroy() b)service() **c) init()** d) connect()

2) A servlet is an instance of _____

a) **HTTPServlet class** b)Cookie c)HttpSessionBindingEvent d)HttpUtils

3) To work with servlet _____ is required

a) **Java Development Kit** b)Java Server Kit c)Java Server Development Kit d)Java Servlet Development

4)Which method is called to process HTTP request?

a) destroy() **b) service()** c) init() d)none of these

5)The server calls _____ method to relinquish any resources, such as files handles that are allocated for servlet.

a)service() b)init() **c)destroy()** d) stop()

6) The life cycle of servlet is managed by _____

a) servlet context b) **servlet container** c) supporting protocol(http or https) d) all of the above

7) The include() method of RequestDispatcher

a) Sends a request to another resource b)**Include resource of file like servlet,jsp or html**

c)Appends the request and response object to the current servlet d)None of the above

8) Servlet creates _____ thread for each

request of client.

a) **single** b)two c)multiple d) None of These

9)Which package contains the classes and interfaces required to build servlet?

a)import javax.servlet.*; b) import javax.servlet.http.*; **c)both a&b** d) None of these

10)The _____ class provides functionality that makes it easy to handle requests and responses.

a)Generic Servlet b)ServletInputStream c)ServletOutputStream d) None of these

11) The call to setContentType() establishes the MIME type of the HTTP response.

a) True b) False

12) servletrunner utility tool listens on _____ port for incoming client requests.

a) 80 b)21 c) 53 **d)8080**

13)Which of the following interface declares the lifecycle method for servlet.

a)Servlet b)Servlet Config c) ServletContext d) ServletResponse

14)_____ interface allows servlet to get initialization parameters.

a)SingleThreadModel b)ServletRequest **c)ServletConfig** d)ServletContext

15) ServletContext interface enables servlets to log events and access information about their environment.

a) True b) False

16) _____ interface is used to read data from a client request.

a)**ServletResponse** b)ServletRequest c) both a&b d) None of these

17) _____ is responsible for managing execution of servlet

a) **Web Container** b) Servlet Context c) JVM d) Server

18) GenericServlet class implements _____ and _____ interfaces.

a)ServletRequest and ServletResponse b) ServletResponse and Servlet

c) Servlet and ServletConfig d)None of these

19) Which of the following class provides an input stream for reading requests from a client

a) ServletException b)GenericServlet c)ServletOutputStream

d) ServletInputStream

20) Which of the following class provides an output stream for writing response to the client

a) ServletException b)GenericServlet c)**ServletOutputStream**

d) ServletInputStream

21) _____ class indicates that a servlet error occurred.

a) ServletException b) Unavailable Exception c) GenericServlet d) None of these

22) Which of the following interface allows a servlet to obtain configuration data when it is loaded

a) ServletConfig b) ServletContext c) ServletRequest d)None of these

23) ServletContextgetServletContext() returns the _____ for this servlet. a) value **b) context** c) enumeration d) None of these

24) Which of the following method returns the value of the initialization parameter named param.

a) ServletContextgetServletContext() **b) String getInitParameter(String** param)

c) Enumeration getInitParameterNames() d)String getServerInfo()

25) Which of the following interface enables the servlet to obtain information about their environment?

a) **ServletContext** b) ServletConfig c)ServletRequest d) ServletResponse

26) Which of the following method returns the port number?

a)String getScheme() b)String getServerName() **c) int getServerPort()** d) String getRemoteHost()

27) _____ method returns the name of the server.

a)String getScheme() **b)String getServerName()** c) int getServerPort() d) String getRemoteHost()

28) _____ method returns the string equivalent of the client hostname.

a) **String getRemoteHost()** b)String getRemoteAddr() c) String getProtocol() d) None of these

29) Which method is used to specify before any lines that uses the PrintWriter?

a)SetPageType() **b) setContentType()** c)setContextType() d)setResponseType()

30) In a web application, running in a web server, servlet is responsible for creating request and response object.

a) True b) False

31) Which of the following method Write s and stack the trace for e to the server log?

a) void log(Exception e, String s) b) void log(String s) c) void log() d) None of these

32) Which of the following method returns an enumeration with the name of servlets in the same namespace in the server?

a)String getInitParameter(String param) b) Enumeration getInitParameterNames()

(c) Enumeration getServletNames() d) None of these

33) _____ returns the real path that corresponds to the virtual path vpath.

a) String getServerInfo() b) String getMimeType(String File)

c) **String getRealPath(String vpath)** d) int getContentLength()

34) The _____ interface is used to indicate that only a single thread should execute the service() method of a servlet.

a**) SingleThreadModel** b) UnithreadModel c) ThreadModel d) None of These

35) If a servlet implements _____ interface , the server creates several instances of it.

a) ServletResponse Interface b) ServlerRequest Interface **c) SingleThreadModel Interface**

d)None of these

36) Which of the following class provides implementations of the basic life cycle methods for a servlet.

a)Servlet InputStream Class b) **GenericServlet Class**

c) ServletException Class d) Servlet OutputStream Class

37) void log(String s) method belongs to which of the following classes.

a) GenericServlet Class b) Servlet OutputStream Class c) Servlet InputStream Class

d) None of These

38)Which of the following interface is an unique object per servlet?

a) ServletConfig b) ServletContext c)ServletRequest d) None of these

39) _____ is an unique object for complete application.

a) ServletConfig **b) ServletContext** c)ServletRequest d) None of these

40) We cannot set attributes in _____ interface that other servlets can use in their implementations.

A)ServletConfig b) ServletContext c)ServletRequest d) both a & b

41) We can set attributes in _____ that other servlets can use in their implementation.

ServletConfig **b) ServletContext** c)ServletRequest d) none of these

42) Which of the following interface is used to forward the request to another resource that can be HTML, JSP or another servlet in same application?

a) Request Dispatcher Interface b)SinglethreadModel Interface

c)ServletResponse Interface d) None of These

43)Which of the following method forwards the request from a servlet to another resource (servlet, JSP file, or HTML file) on the server?

a) void include(ServletRequest request, ServletResponse response)

b) **void forward(ServletRequest request, ServletResponse response)**

c) void include(ServletRequest request)

d) void forward(ServletRequest request)

44) PrintWriter is a character-stream class whereas ServletOutputStream_____ is a class.

a) **byte-stream** b)bit-stream c) string-stream d) io-stream

45)_____ is protocol independent implementation of Servlet

a)ServletInputStream b)ServletOutputStream **c) GenericServlet** d) None of These

46) Which of the following interface guarantees that no two threads will execute concurrently in the servlet's service method ?

a)ServletResponse b) ServletRequest **c) SingleThreadModel** d)ServletConfig

47)Generic servlet is the super class of all servlets.

a) True b) False

48) Servlet runs each request in a _____ ?

a) OS shell b)JVM **c)Separate thread** d)JRE

49)GenericServlet class is encapsulated inside _____ package

a) java.lang **b) javax.servlet** c) java.servlet d) javax.servlet.http

50)Find the error in the following code import java.io.*; import javax.servlet.*;
public class HelloServlet extends GenericServlet{
public void service(ServletRequest request, ServletResponse response) throws ServletException,IOException{
response.setContentType("text/html"); PrintWriter pw = response.getOutputStream(); pw.println("<b> Hello");
pw.close();

a)import javax.servlet.*; b) response.setContentType("text/html")

c) PrintWriter pw = response.getOutputStream(); d)None of these

51)Consider the following program which class should be extended?

```
import java.io.*;
import javax.servlet.*;   public class First
extends ************{  public void
service(ServletRequest req,ServletResponse
res)   throws IOException,ServletException{
res.setContentType("text/html");   PrintWriter
out=res.getWriter();
out.print("<html><body>");
```

```
out.print("<b>hello generic servlet</b>");
out.print("</body></html>");
 }
}
```

a) HttpServlet **b) GenericServlet** c)Servlet d) None of These

52)Which of the following package is missing for the below program? import java.io.*:

```
import javax.servlet.*;
public class PostParameterServlet extends
GenericServlet{ public void service(ServletRequest
request, ServletResponse response) throws
ServletException ,IOException{
Printwriter pw= response.getWriter();
Enumeration e= request.getParameterNames();
While(e.hasMoreElement
s()) { String pname=
(String) e.nextElement();
pw.print(pname + " = ");
Sting pvalue = request.getParameter(pname);
pw.println(pvalue);
}
p
w
.
c
l
o
s
e
```

(
)
;
}
}

a)import.java.util.*; b) import javax.servlet.http.*; c) import java.awt.*; d)None of these

53) Write the missing statement in the below code import javax.servlet.*; public class WelcomeServlet extends GenericServlet{ public void service(ServletRequest request,ServlerResponse response) throws ServletException ,IOException{
response.setContentType("text/htm
l"); PrintWriter pw =
response.getOutputStream();
pw.println(" Hello");
}
}

a)pw.close() b) pw.stop() c) pw.destroy() d) none of these

54) Which statement is missing in the place marked as

********** import java.io.*; import javax.servlet.*;
public class First extends GenericServlet{
public void service(ServletRequest req,ServletResponse res) throws IOException,ServletException{ res.setContentType("text/html");
out.print("<html><body>");
out.print("hello generic servlet");
out.print("</body></html>");
}
}

a) **PrintWriter out=res.getWriter();** b) PrintWriter in = res.getWriter()

c) PrintWriter out=res.putWriter(); d) PrintWriter in = res.putWriter()

55) Which of the following are interfaces in javax.servlet.http package?

a) HttpServletRequest b)HttpServletResponse c)HttpSession **d)All of the above**

56) Which of the following informs an object that it is bound to or unbound from a session?

a)HttpServletRequest b)HttpServlet c)HttpSession **d)HttpSessionBindingListener**

57) ---------- allows session data to be read and written

a)HttpServletRequest b)HttpServlet c)**HttpSession** d)HttpSessionBindingListener

58) ---------- class provides methods to handle HTTP requests and responses

a)HttpServlet b)Cookie c)HttpSessionEvent d)None of the above

59) ---------- class encapsulates session-changed event

a)HttpServlet b)Cookie **c)HttpSessionEvent** d)None of the above

60) Which of the following code can be used to send an error response to the client using the specified status code and error message?

a) request.sendError(statusCode,message) **b) response.sendError(statusCode,message)**

c) header.sendError(statusCode,message) d) None of the above

61) The sendRedirect() method of HttpServletResponse interface can be used to redirect response to another resource, it may be servlet, jsp or html file.

a. True b. False

62) Which methods are used to bind the objects on HttpSession instance and get the objects?

a) setAttribute b)getAttribute **c. Both a & b** d. None of the above

63) getAuthType() method returns authentication scheme

a. True b. False

64) getHeaderNames() returns an enumeration of the header names

a.True b.False

65) ---------- returns part of the URL that identifies the servlet

a)getServletPath() b)getPathInfo() c)getPathTranslated() d)None of the above

66) In getSession(boolean new) method If new is true and no session exists, creates and returns a session for this request. Otherwise, returns the existing session for this request.

a. **True** b.False

67) Which of the following is not a method of HttpServletRequest interface.

a) isRequestedSessionIdFromCookie( ) b)getSession( ) c)getRequestURI( )
d)addCookie(Cookie cookie)

68) SC_NOT_FOUND indicates that the requested resource is not available.

a.True b.False

69) Determines if the session ID must be encoded in the URL identified as url. If so, returns the modified version of url. Otherwise, returns url.

a)encodeRedirectURL(String url) **b)encodeURL(String url)** c)encode(String url

d)None of the above

69) boolean containsHeader(String field) Returns true if the HTTP response header contains a field named field.

a)False **b)True**

70) ---------- Returns true if the server created the session and it has not yet been accessed by the client.

a) invalidate( ) **b)isNew()** c) getLastAccessedTime( ) d)None of the above

71) Invalidate() method is in HttpSessionBindingListener interface

a)True **b)False**

72) Cookies are stored at server side.

a)True **b)False**

73) Some of the information that is saved for each cookie includes the following:

a)The name of the cookie b)The value of the cookie

c)The expiration date of the cookie **d)All of the above**

74) Which method adds cookie to the HTTP response

a)void addCookie() **b)void addCookie(Cookie cookie)**

c)void addCookie(String cookie) d)void addCookie(int i)

75) If an expiration date is not explicitly assigned to a cookie, it is deleted when the current browser session ends. Otherwise, the cookie is saved in a file on the user's machine.

a) **True** b)False

76) Constructor for cookie is

a) Cookie(String name, String value) b) Cookie(String name, int value)

c) Cookie(Char name, String value) d)None Of These

77) ---------- method Returns true if the cookie must be sent using only a secure protocol. Otherwise, returns false.

a) getSecure( ) b) getName( ) c)clone() d)None of these

78) The HttpServlet class extends GenericServlet.

a)False **b)True**

79) Which of the following is not a method of HttpServlet class?

a)doDelete() b)doGet() c)doHead() **d)getValue()**

80) Which method is Called by the server when an HTTP request arrives for this servlet.

a)getLastModifies() **b)service()** c)doPut() d)None of the above

81) HttpSessionEvent encapsulates EventObject

a) **True** b)False

82) --------- obtains the session to which the listener is being bound or unbound

a)getName() **b)getSession()** c)getSessionName() d)None Of the above

83) A servlet developer overrides which of the following methods?

a)doDelete() b)doGet() c)doHead() **d)All of the above**

84) ---------- method returns true if the cookie contains session id.Otherwise returns false.

a)Boolean isRequestedSessionIdFromCookie()

b) Boolean isRequestedSessionId()

c) Boolean isSessionIdFromCookie()

d)None of the above

85) The servlet is invoked when a form on a Web page is submitted.

a)True b)False

86) Which method returns copy of this object?

a)getclone() **b)clone()** c)setclone() d)None of these

87) Which class allows state information to be stored on a client machine?

a) **Cookie** b)HttpServlet c)HttpSession d)None of these

88) Which method returns the URL?

a)getURL() b)URL() **c)getRequestURL()** d)None of the above

89) Method isRequestedSessionFromCookie() returns true if a cookie contains session id otherwise false

a)True b)False

90) Which method redirects the client to the URL?

a) **sendRedirect(String url)** b)Redirect(String url)

b) sendError(String url) d)None of the above

91) Which method adds field to the header with date value equal to msec?

a)void setDateHeader(String field,long msec) b) void setDateHeader(String field,int msec)

c) void setDateHeader(long msec) d) void setDate(String field,long msec)

92) Which method sets status code for this response to code

a)void setStatus(int code) b) void setStatus()

c) void Status(int code) d)None of the above

93) Which method returns the time when the client last made a request for this session

a)void getLastAccessedDate() **b)long getLastAccessedTime()**

c)getAccessedTime() d)None of the above

94) Which method performs an HTTP DELETE?

a) void doDelete(HttpServletRequest req,HttpServletResponse res) throws IOException, ServletException

b) void Delete(HttpServletRequest req,HttpServletResponse res) throws IOException, ServletException

c) void doDelete(HttpRequest req,HttpServletResponse res) throws IOException, ServletException

d) void doDelete(HttpRequest req,HttpResponse res) throws IOException, ServletException

95) Which method performs an HTTP GET?

a) void doGet(HttpServletRequest req,HttpServletResponse res) throws ServletException

b) **void doGet(HttpServletRequest req,HttpServletResponse res) throws IOException, ServletException**

c) void Get(HttpServletRequest req,HttpServletResponse res) throws IOException,

ServletException d)None of the above

96) Which method invalidates this session and removes it from the context?

a) **void invalidate()** b)void validate() c)void verify() d)void removeAttribute()

97) Which of the following is not a method of HttpSession interface?

a) getAttribute() b)setAttribute() **c)setHeader()** d)isNew()

98) Which of the following is a method of HttpServletResponse interface?

a) getAttribute() b)setAttribute() **c)setHeader()** d)isNew()

99) SC_OK indicates that HTTP request succeeded.

a) **True** b)False

100) SC_NOT_OK indicates that requested resource is not available.

a) True **b)False**

101) Which of the following is true about cookies?

a)Cookies are stored on client b)Cookies contain state information

c)Cookies track user activities **d)All of the above**

102) Which of the following is not a method of cookie class?

a) Clone() b)getMaxAge() **c)doGet()** d)getName()

103) Which one is not a constructor for cookie?

a) Cookie() b)Cookie(String name) **c)a& b** d)Cookie(String name,String value)

104) Which of the following is not a method of HttpServlet Class?

a) **setComment()** b)doDelete() c)doGet() d)doOptions()

105) HttpServlet class methods throw which exceptions?

a) IOException b)ServletException c)IllegalstateException **d)a & b**

106) Which of the following is constructor for HttpSessionEvent Class?

a)HttpSessionEvent() **b) HttpSessionEvent(Httpsession session)**

c) HttpSessionEvent(String value) d)None of the above

107) Action parameter of tag specifies ----------

a)Location **b)URL** c)Path d)None of the above

108) ---------- method returns any path information that is located after the servlet path and before a query string of the URL

a)String getPathInfo() b)String getPath()

c)String getMethod() d)None of the above

109) ---------- returns an array of the cookies in this request

a)Cookie[] getCookies() b)Cookie[] getMaxCookies()

c)Cookie[] getMinCookies() d)None of the above

110) Which method returns int equivalent of the header field named field?

a) int getHeader() b)int getIntHeader() **c)int getIntHeader(String field)** d)None of these

111) ---------- method returns name of the user who issues this request.

a) **String getRemoteUser()** b)String getUser() c)String getRemote() d)None of these

112) A servlet can write a cookie to a user's machine via the _____ method of the HttpServletResponse interface.

a) **addCookie( )** b)Cookie( ) c) CookieDo( ) d)updateCookie()

113) A servlet can write a cookie to a user's machine via the addCookie( ) method of the _____ interface

a) ServletRequest b)HttpServletRequest c) **HttpServletResponse** d)ServletResponse

114) Correct signature of Cookies is:

a)Cookie(String value, String name) b)Cookie(int value, String name)

c)**Cookie(String name, String value)** d)Cookie(int value, int name)

115) Information that is saved for each cookie includes the following:

a)The name of the cookie, the value of the cookie b)The expiration date of the cookie

c)The domain and path of the cookie d)**All of above**

116) Following is the correct syntax for creating Cookies Object c:

a)Cookie c = new Cookie(MyCookie, data); b)**Cookie c = new Cookie("MyCookie", data);**

c)cookie c = new cookie("MyCookie", data); d)Cookie c = new Cookie(data,"MyCookie");

117) Following method is used for Sets the maximum age of the cookie in seconds.

a)public void setmaxage(int expiry) b)public void Setmaxage(String expiry)

c)public void setMaxAge(String expiry) d)**public void setMaxAge(int expiry)**

118) Following is the method of HttpServletRequest interface is used to return all the cookies from the browser.

a)Cookie[] getcookies() b)private Cookie[] getCookies() c) **public Cookie[] getCookies()** d)public Cookie[] setCookies()

119) Which one is following method indicates if secure protocol to be used while sending this cookie?

a)**public void setSecure(boolean secure)** b) public void setsecure(int secure)

c) private void setSecure(boolean secure) d)public boolean setSecure(boolean secure)

120) Following method is used for setting up comments in the cookie.

a)public void setComment(int purpose) b)**public void setComment(String purpose)**

c)private void setpath(String path) d)public void setPath(String path)

121) Following method is used to Specifies a path for the cookie to which the client should return the cookie.

a)**public void setPath(String path)** b)public void setpath(String value)

c)public void setPath(int path) d)private void setpath(String path)

122) Following method is returns the cookie protocol version.

a) **int getVersion( )** b)int setversion( ) c)int GetVersion( ) d)String getVersion( )

123) Following method is used to Sets the domain in which this cookie is visible.

a)**public void setDomain(String pattern)** b)public void setDomain(int pattern)

c)public void getDomain(String pattern) d)private void setdomain(String pattern)

124) _____ is a way to maintain state (data) of an user.

a)**Session Tracking** b)Cookie tracking c)HttpServletState d)Session

125) _____encapsulates session events.

a) httpsessionevent b)**HttpSessionEvent** c)HttpSessionTrackingEvent
d)HttpSessionEventObject

126) Following method returns the session in which the event occurred.

a)HttpSession setSession( ) b)httpsession getsession( )

c)**HttpSession getSession( )** d)HttpSession setSessionEvent( )

127) HTTP is a _____ protocol.

a) **Stateless** b)state oriented c)stateful d)datagram

128) A session can be created via the getSession( ) method of _____.

a)HttpServletResponse b)**HttpServletRequest** c)HttpRequest d)HttpResponse 129) In some applications, it is necessary to _____ so that information can be collected from several interactions between a browser and a server.

a) save date and time information b)save creation of session

b) **save state information** d) save objects

130) A session can be created via the _____ method of HttpServletRequest

a) getSessionCreate( ) b)SetSession( ) c)setsession( ) d) **getSession( )**

131) The _____ method is overridden to process any HTTP POST requests that are sent to servlet.

a) doGet() b)**doPost( )** c)DoPost() d)DOGET()

132) _____ interface enables a servlet to obtain information about a client request.

a)**HttpServletRequest** b)HttpServletResponse c)httpservletrequest d)Http Request

133) Identify the correct sequence of creating cookies. **a)**

1) Create a Cookie object.

2) Set the maximum Age.

3) Place the Cookie in HTTP response header. b)

1) Set the maximum Age.

2) Place the Cookie in HTTP response header.

3) Create a Cookie object.

c)

1) Place the Cookie in HTTP response header.

2) Set the maximum Age. 3) Create a Cookie object.

d)

1) Set the maximum Age.

2) Create a Cookie object.

3)Place the Cookie in HTTP response header.

134) Following are the methods in connection with HttpSession interface.

a) String getId( ) b)void invalidate( ) c)long getLastAccessedTime( ) d)**All of above**

135) The HttpSession interface is implemented by the _____.

a) Session b) Cookies c)client d)**server**

136) A cookie is stored on a _____ and contains state information.

a) Session b) Cookies c)**client** d)server

137) Which class encapsulates a session-changed event?

a) Cookie b)HttpServlet c)**HttpSessionEvent** d)HttpSessionBindingEvent

138) Following method returns the time (in milliseconds since midnight, January 1, 1970, GMT) when this session was created.

a) int getcreationtime( ) b)long CreationTime( ) c)**long getCreationTime( )** d)long getCreation( )

139) Methods of HttpSession interface throw an _____ if the session has already been invalidated.

a) IllegalState b)IllegalException c)LegalStateException d)**IllegalStateException**

140) Following method returns true if the requested session ID is valid in the current session context.

a)boolean RequestedSessionIdValid( ) b)boolean isRequestedSessionId( )

c)boolean isRequestedValid( ) d)**boolean isRequestedSessionIdValid( )**

141) The Java _____ specification defines an application programming interface for communication between the Web server and the application program. a) **Servlet**

b) Server

c) Program

d) Randomize

142) The doGet() method in the example extracts values of the parameter's type and number by using _____

a) **request.getParameter()**
b) request.setParameter()
c) responce.getParameter()
d) responce.getAttribute()

143) Dynamic interception of request and responses to transform the information is done by

a)Servlet Container b)Servlet config c)Servlet context d)**Servlet filter**

144) State true or false.

i) init() is called after start()
in applet ii) applets are used
for networking
iii) inheritance is a part of Java Foundation
Classes iv) final does not prevent inheritance

a) i-true, ii-true, iii-false, iv-true
b) **i-false, ii-false, iii-false, iv-false**
c) i-true, ii-true, iii-true, iv-true
d) i-true, ii-false, iii-false, iv-false

145) Identify Error in the following servlet code import java.io.*;

import javax.servlet.*; public class ColorGetServlet extends

HttpServlet {

public void doGet(HttpServletRequest request,HttpServletResponse response) throws ServletException, IOException {

```
String color = request.getParameter("color"); response.setContentType("text/html");
PrintWriter pw = response.getWriter(); pw.println("<B>The selected color is: ");
pw.println(color); pw.close();
}
}
```

Answer: Package javax.servlet.http.* not imported

146) Find error in the following code import java.io.*; import javax.servlet.*; import javax.servlet.http.*;

```
public class ColorPostServlet extends HttpServlet
{
public void doPost(HttpServletRequest request,HttpServletResponse response) throws
ServletException, IOException
{
String color = request.getParameter("color");
PrintWriter pw = response.getWriter();
pw.println("<B>The selected color is: ");
pw.println(color); pw.close();
}
}
```

Answer:

Error: response.setContentType("text/html"); missing

145) Which is the missing statement in following code to get proper output:

```
import
java.io.*;
import
java.util.*;
```

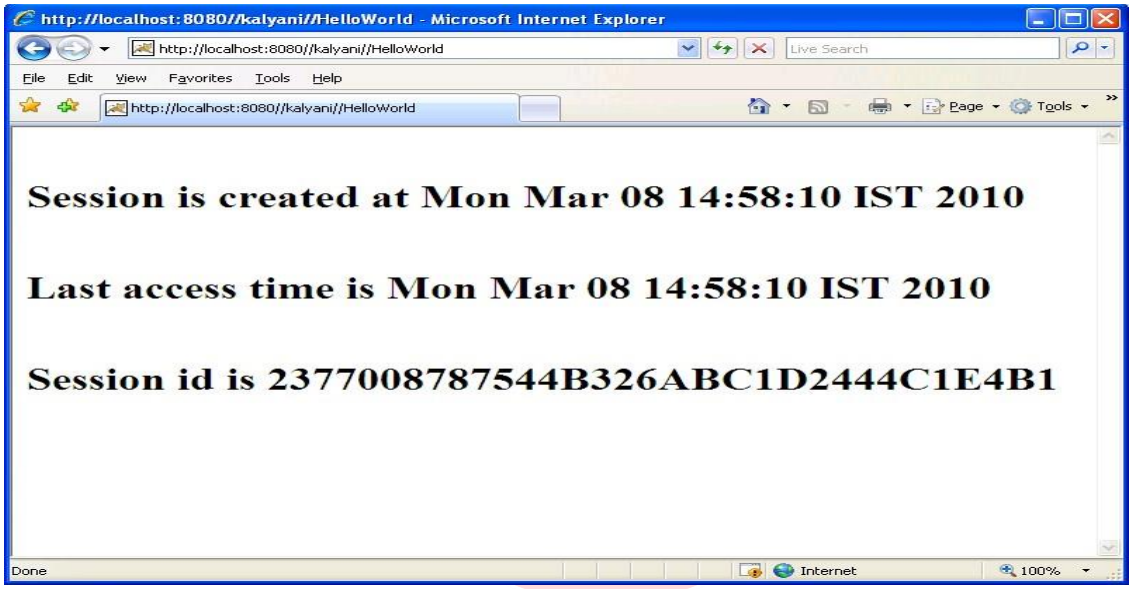
```
public class HelloWorld extends HttpServlet
{
 public void doGet(HttpServletRequest request, HttpServletResponse response)
 throws IOException, ServletException
 {
 response.setContentType("te
 xt/html"); PrintWriter out =
 response.getWriter();
 HttpSession hp=request.getSession(true);
 Date create=new
 Date(hp.getCreationTime()); Date
 access=new
 Date(hp.getLastAccessedTime());
 out.println("<h1><br>Session is created at
 "+create); out.println("<h1><br>Last
 access time is "+access);
 out.println("<h1><br>Session id is
 "+hp.getId()); }
}
```

a)Missing ;
b)Missing()
c)Missing Package Statement
d)Missing {

146) What is the missing statement in the following program to get the proper output

```
import
java.io.*;
import
java.util.*;
public class HelloWorld extends HttpServlet

{
public void doGet(HttpServletRequest request, HttpServletResponse response)
throws IOException, ServletException
{
 response.setContentType("t
 ext/html"); PrintWriter out =
 response.getWriter();
 HttpSession hp=request.getSession(true);
 Date create=new
 Date(hp.getCreationTime()); Date
```

```
access=new
Date(hp.getLastAccessedTime());
out.println("<h1><br>Session is created at
"+create); out.println("<h1><br>Last
access time is "+access);
out.println("<h1><br>Session id is
"+hp.getId());
}
}

/*OUTPUT*/
```



a)Missing semicolon

b)missing {}

c)missing Package statement

d)none of the above

147) What is at the place of _____ to get output

```
import
java.io.*;
import
javax.servlet.*;
import
javax.servlet.ht
tp.*; public
class example
```

```
extends
_____
 {
 public void doGet(HttpServletRequest request, HttpServletResponse response)throws
 IOException, ServletException
 {
 response.setContentType("te
 xt/html"); PrintWriter out =
 response.getWriter();
 // print out cookies
 String name = request.getParameter("t1");
 String value =
 request.getParameter("t2"); Cookie c =
 new Cookie(name, value);
 response.addCookie(c); String n1 =
 c.getName(); String v1 = c.getValue();
 out.println("<h3> The Name  Of Cookie
 is:</h3>"+n1); out.println("<h3> The Value  Of
 Cookie is:</h3>"+v1);
 }
 }
```

a) HttpServlet
b)HttpResponse
c)HttpRequest
d)All of the above

148) What we should write at the place of_____ to get the proper output.

```
import
java.io.*;
import
javax.servlet.*;
import
javax.servlet.ht
tp.*;
public class password extends HttpServlet
{ public void doGet(HttpServletRequest request, HttpServletResponse
 response)throws IOException, ServletException
 {
 response.setContentType("text/html");
 PrintWriter out = _____
```

```
out.println("<html>");
out.println("<head>");
out.println("<title>Hello
World!</title>");
out.println("</head>");
out.println("<body>");
String a=
request.getParameter("t1");
String b=
request.getParameter("t2");
int  a1
=a.len
gth();
int
b1=
b.leng
th();
if(b.le
ngth()
<=6)
{
out.println("<h3> Welcome  To  Home Page<h3>");
}
e
l
s
e
{
out.println("<h3>Password Should not more than 6 Character<h3>");
}
out.println("
</body>");
out.println("
</html>");
}
}
```

a)request.getWriter();

b)response.getWriter();

c)response.printWriter();

d)none of the above

149) What should we write at the place of _____to get the proper output.

```
import
java.io.*;
import
javax.servle
t.*;
public class GenericServletDemo extends GenericServlet
{ public void service(ServletRequest request ,_____
 response) throws
 ServletException,IOException
 {
 response.setContentType("text/html");
 PrintWriter out=response.getWriter();
 out.println("<HTML>");
 out.println("<HEAD><TITLE>hello page</TITLE></HEAD>"); out.println("<BODY>");
 String name=(String) request.getParameter("user1");
 String password=(String)
 request.getParameter("password1"); String phone=(String)
 request.getParameter("phone1");
 out.println("Name="+name+"<br>");
 out.println("Passsword="+password+"<br>");
 out.println("Phone="+phone+"<br>");
 out.println("</BODY></HTML>");
 }
 }
```

a)HttpReuest
b) ServletResponse
c)HttpResponse
d)none of the above

150) Choose correct package for following code from given options. import java.io.*; import javax.servlet.*;

```
public class HelloServlet extends *******
{ public void service(ServletRequest
request, ServletResponse response)
throws ServletException, IOException
{
response.setContentType("te
xt/html"); PrintWriter pw =
```

```
response.getWriter();
pw.println("<B>Hello!");
pw.close();
}
}
```

a) GenericServlet
b) genericServlet
c) HttpServlet
d) httpServle

151) In following Java program fill statement showing ******.Select any one option from given options

```
import
java.io.*;
import
java.util.*;
import javax.servlet.*;

public class PostParametersServlet
extends GenericServlet { public
 void service(ServletRequest
 request, ServletResponse
 response) throws
 ServletException, IOException {
 // Get print writer.
PrintWriter pw = response.getWriter();


            // Get enumeration of parameter names
Enumeration e = request.getParameterNames();
;
// Display parameter names and values.
while(e.hasMoreElements()) {
String pname =
(String)e.nextElement();
pw.print(pname + " = ");
String pvalue = *******;
pw.println(pvalue);
}
pw.close();
```

```
}
}
```

a) request.getParameterName(pname);
b) request.Getparametername(pname);
c) Request.Getparametername(pname);
d) request.getParametername(pname);

152) Consider the following program and identify the missing method attribute ***** if you want to implement HTTP POST.

```
<html>
<body>
<center>
<form name="Form1"
method=*****
action="http://localhost:8080/examples/servlet/ColorPostServlet">
<B>Color:</B>
<select name="color" size="1">
<option value="Red">Red</option>
<option value="Green">Green</option>

<option value="Blue">Blue</option>
</select>
<br><br>
<input type=submit value="Submit">
</form>
</body>
</html>
```

a)HTTP POST
b)HTTP GET
c)post
d)get

153) Consider the following program. Identify the missing word ****.

```
<html>
```

```
<body>
<center>
<form name="Form1"
action="http://****/examples/servlet/ColorGetServlet">
<B>Color:</B>
<select name="color" size="1">
<option value="Red">Red</option>
<option value="Green">Green</option>
<option value="Blue">Blue</option>
</select>
<br><br>
<input type=submit value="Submit">
</form>
</body>
</html>
```

a)localhost:8080

b) localhost:0808

c) locallhost:8080

d) localhost:80800

154) Consider the following program and identify the missing method name ****** if you want to implement HTTP POST.

```
import java.io.*; import javax.servlet.*;
import javax.servlet.http.*; public class
ColorGetServlet extends HttpServlet {
public void ******(HttpServletRequest
request, HttpServletResponse response)
throws ServletException, IOException {
String color =
request.getParameter("color");
response.setContentType("text/html");
PrintWriter pw = response.getWriter();
pw.println("<B>The selected color is: ");
pw.println(color);
pw.close();
}
}
```

a)doGet()

b)doPost()

c)HTTP GET()

d)HTTP POST()