



V2V EDTECH LLP

Online Coaching at an Affordable Price.

OUR SERVICES:

- Diploma in All Branches, All Subjects
- Degree in All Branches, All Subjects
- BSCIT / CS
- Professional Courses



+91 93260 50669



v2vedtech.com



V2V EdTech LLP



v2vedtech



WINTER– 18 EXAMINATION

Subject Name: Database Management System

Model Answer

Subject Code:

22319

Important Instructions to examiners:

- 1) The answers should be examined by key words and not as word-to-word as given in the model answer scheme.
- 2) The model answer and the answer written by candidate may vary but the examiner may try to assess the understanding level of the candidate.
- 3) The language errors such as grammatical, spelling errors should not be given more Importance (Not applicable for subject English and Communication Skills).
- 4) While assessing figures, examiner may give credit for principal components indicated in the figure. The figures drawn by candidate and model answer may vary. The examiner may give credit for any equivalent figure drawn.
- 5) Credits may be given step wise for numerical problems. In some cases, the assumed constant values may vary and there may be some difference in the candidate's answers and model answer.
- 6) In case of some questions credit may be given by judgement on part of examiner of relevant answer based on candidate's understanding.
- 7) For programming language papers, credit may be given to any other program based on equivalent concept.

Q. No.	Sub Q. N.	Answer	Marking Scheme
1		Attempt any FIVE of the following :	10 M
	a	List disadvantages of typical file processing system.	2 M
	Ans	Disadvantages of file processing system 1. Data redundancy and inconsistency 2. Difficulty in accessing data 3. Data isolation 4. Integrity problems 5. Atomicity problems	any 4 disadvantages – 1/2 Mark each disadvantage
	b	Define i)Data Abstraction ii)Data Redundancy	2 M
	Ans	1. Data Abstraction : Many end users are not computer trained so it is needed to hide complex data structures from them. Hiding complexity of data structures from end user through different levels is known as data abstraction. It has 3 levels :	1 mark Data Abstraction description 1 mark Data Redundancy description



	<p>a. Physical level</p> <p>b. logical level</p> <p>c. view level</p> <p>2. Data redundancy :</p> <p>The repetition of information is known as redundancy .This redundancy leads to higher storage and access cost.</p> <p>It may lead to data inconsistency, that is different copies of the same data may have different values.</p>	
c	<p>Define the term:</p> <p>i) Candidate key</p> <p>ii) Primary key</p>	2 M
Ans	<p>Candidate key: In a relation, there may be a primary key or may not, but there may be a key or combination of keys which uniquely identify the record. Such a key is called as Candidate key.</p> <p>OR</p> <p>A candidate key is a column, or set of columns, in a table that can uniquely identify any database record without referring to any other data.</p> <p>The candidate key can be simple (having only one attribute) or composite as well.</p> <p>For Example, {STUD_NO, COURSE_NO} is a composite candidate key for relation STUDENT_COURSE.</p> <p>Primary key: A key which is selected by the designer to uniquely identify the entity is called as Primary key. A primary key cannot contain duplicate values and it can never contain null values inside it.</p> <p>Example, RollNo attribute is a primary key for Relation Student.</p>	1 mark Candidate key 1 mark Primary Key
d	<p>List Four DDL commands with syntax.</p>	2 M
Ans	<p>DDL commands</p> <p>1. Create</p> <p>Syntax : create table <table_name>(Column_name1 datatype1, column_name2 Datatype2,...Column_nameN DatatypeN);</p> <p>2. Drop</p>	1/2 mark for each command and 1/2 mark for syntax



	<p>Syntax: drop table <table_name>;</p> <p>3. Desc</p> <p>Syntax: describe <table_name>;</p> <p>OR</p> <p>Desc <table_name></p> <p>4. Truncate</p> <p>Syntax: truncate table <table_name>;</p> <p>5. Alter</p> <p>Syntax: Alter table <table_name> add Column_name Datatype (size);</p>	
e	Define Normalization, list its types.	2 M
Ans	<p>Normalization:</p> <p>Normalization can be defined as process of decomposition/division of database tables to avoid the data redundancy.</p> <p>Types of Normalization:</p> <ol style="list-style-type: none">1. 1NF2. 2NF3. 3NF4. BCNF	1 mark for Normalization definition and 1 mark for types
f	Enlist four aggregate functions.	2 M
Ans	SUM() AVG() MAX() MIN() COUNT()	any 4 functions
g	Define Cursor. List the two types of cursor.	2 M



	Ans	<p>Cursor: The Oracle Engine uses a work area for its internal processing in order to execute an SQL statement. This work area is private to SQL's operations and is called a Cursor.</p> <p>OR</p> <p>A cursor is a temporary work area created in the system memory when a SQL statement is executed.</p> <p>Types of cursor are:</p> <ol style="list-style-type: none"> 1) Implicit cursor 2) Explicit cursor 	1 mark Cursor definition and 1 mark for types of cursor																					
2		Attempt any THREE of the following :	12 M																					
	a	Distinguish between network model and hierarchical model.	4 M																					
	Ans	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 5%;">Sr. No.</th> <th style="width: 45%;">Hierarchical model</th> <th style="width: 45%;">Network model</th> </tr> </thead> <tbody> <tr> <td>1.</td> <td>Hierarchical model is not more popular than network model</td> <td>Network model is more popular than the hierarchical and relational model.</td> </tr> <tr> <td>2.</td> <td>It does not uses client server architecture</td> <td>It uses client –server architecture</td> </tr> <tr> <td>3.</td> <td>One to many relationship is maintained.</td> <td>One to many and many to many relationship is maintained.</td> </tr> <tr> <td>4.</td> <td>Hierarchical model is based on tree like structure with one root.</td> <td>Network model is based on tree like structure with many roots.</td> </tr> <tr> <td>5.</td> <td>One child or many children have only one parent</td> <td>Many children have many parent</td> </tr> <tr> <td>6.</td> <td>Main application of hierarchical model is in the mainframe database system.</td> <td>It is upgraded version of hierarchical model so used in network</td> </tr> </tbody> </table>	Sr. No.	Hierarchical model	Network model	1.	Hierarchical model is not more popular than network model	Network model is more popular than the hierarchical and relational model.	2.	It does not uses client server architecture	It uses client –server architecture	3.	One to many relationship is maintained.	One to many and many to many relationship is maintained.	4.	Hierarchical model is based on tree like structure with one root.	Network model is based on tree like structure with many roots.	5.	One child or many children have only one parent	Many children have many parent	6.	Main application of hierarchical model is in the mainframe database system.	It is upgraded version of hierarchical model so used in network	any 4 points
Sr. No.	Hierarchical model	Network model																						
1.	Hierarchical model is not more popular than network model	Network model is more popular than the hierarchical and relational model.																						
2.	It does not uses client server architecture	It uses client –server architecture																						
3.	One to many relationship is maintained.	One to many and many to many relationship is maintained.																						
4.	Hierarchical model is based on tree like structure with one root.	Network model is based on tree like structure with many roots.																						
5.	One child or many children have only one parent	Many children have many parent																						
6.	Main application of hierarchical model is in the mainframe database system.	It is upgraded version of hierarchical model so used in network																						
	b	Explain set Operators with example.	4 M																					
	Ans	<p>Set operators combine the results of two component queries into a single result. Queries containing set operators are called as compound queries. Set operators in SQL are represented with following special keywords as: Union, Union all, intersection & minus. Consider data from two tables emp and employee as</p>	1 mark for explanation and 1 mark for example each																					



Emp

Employee

Ename
a
b
c
d

Ename
c
e

1) Union: The Union of two or more sets contains all elements, which are present in either or both. Union works as or.

E.g. select ename from emp union select ename from employee;

The output considering above data is :

Output

Ename
a
b
c
d
e

2) Union all: The Union of 2 or more sets contains all elements, which are present in both, including duplicates.

E.g. select ename from emp union all select ename from employee;

The output considering above data is:

Output



Ename
a
b
c
c
d
e

3) Intersection: The intersection of two sets includes elements which are present in both. E.g. select ename from emp intersect select ename from employee;

The output considering above data is:

Output

Ename
c

4) Minus: The minus of two sets includes elements from set1 minus elements of set2.

E.g. select ename from emp minus select ename from employee;

The output considering above data is:

Ename
a
b
d

c Explain any four String functions with example.

4 M

Ans i) Lower(char)-

Returns the input string with all letters in lower case.

Example: SQL>Select lower ('RAJESH') from dual;

1 mark for explanation and 1 mark for example each



Output: rajesh

ii) Upper(char)-

Returns the input string with all letters in upper case.

Example: SQL>Select upper ('rajesh') from dual;

Output: RAJESH

iii) Ltrim(char,set)-

It removes or trims from left of character string

. Example: SQL>Select Ltrim('university','univ') from dual;

Output: ersity

iv) Rtrim(char,set)-

It removes or trims from right of character string.

Example: SQL>Select Rtrim('university','sity') from dual;

Output: univer

v) Length(char)-

It returns length of character string.

Example: SQL> Select length('University') from dual; Output:10

vi) Concat(str1,str2,...)-

Returns the string that result from concatenating the arguments.

Example: Select Concat('employee', 'name') from dual;

Output: employeename

vii) Lpad(str, len, padstr)-

Returns the string str, left-padded with the string padstr to a length of len characters.

Example: Select lpad(ename,10,'*') from emp where empno=7782;

viii) Rpad(str,len,padstr)-

Returns the string str, right-padded with the string padstr to a length of len characters.

Example: Select rpad(ename,10,'*') from emp where empno=7782;



	viii) Substr(Char,m,n)- It returns a portion of char, beginning at a character m, n character long. Example: Select substr('College',3,4) from dual; Output: lleg	
d	Describe exception handling in brief.	4 M
Ans	<p>Exception Handling: Exception is nothing but an error. Exception can be raise when DBMS encounters errors or it can be raised explicitly.</p> <p>When the system throws a warning or has an error it can lead to an exception. Such exception needs to be handled and can be defined internally or user defined.</p> <p>Exception handling is nothing but a code block in memory that will attempt to resolve current error condition.</p> <p>Syntax:</p> <p>DECLARE ;</p> <p>Declaration section</p> <p>...executable statement;</p> <p>EXCEPTION</p> <p>WHEN ex_name1 THEN ;</p> <p>Error handling statements/user defined action to be carried out;</p> <p>END;</p> <p>Types of Exception:</p> <p>1) Predefined Exception/system defined exception/named exception: Are always automatically raised whenever related error occurs. The most common errors that can occur during the execution of PL/SQL. Not declared explicitly i.e. cursor already open, invalid cursor, no data found, zero divide and too many rows etc. Programs are handled by system defined Exceptions.</p> <p>2) User defined exception: It must be declare by the user in the declaration part of the block where the exception is used. It is raised explicitly in sequence of statements using:</p> <p>Raise_application_error(Exception_Number, Error_Message);</p>	any relevant 4 points 1 mark each
3	Attempt any THREE of the following :	12 M
a	Describe commit and rollback with syntax and example.	4 M



	Ans	<p>Commit:</p> <p>The COMMIT command saves all transactions to the database since the last COMMIT or ROLLBACK command</p> <p>The syntax: SQL> COMMIT;</p> <p>Or</p> <p>COMMIT WORK;</p> <p>Example :</p> <p>SQL>Commit;</p> <p>Rollback:</p> <p>The ROLLBACK command is used to undo transactions that have not already been saved to the database.</p> <p>The ROLLBACK command can only be used to undo transactions since the last COMMIT or ROLLBACK command was issued.</p> <p>The syntax for ROLLBACK is:</p> <p>ROLLBACK TO SAVEPOINT_NAME;</p> <p>OR</p> <p>ROLLBACK;</p> <p>OR</p> <p>ROLLBACK WORK;</p> <p>Example:</p> <p>SQL>ROLLBACK;</p>	Description and syntax – 1 Mark example 1 Mark for each
	b	Explain joins in SQL with examples.	4 M
	Ans	<p>JOIN:</p> <p>A SQL join is an instruction to combine data from two sets of data (i.e. two tables). A JOIN clause is used to combine rows from two or more tables, based on a related column between them. SQL Join types are as follows:</p> <p>1) INNER JOIN or EQUI JOIN:</p> <p>A join which is based on equalities is called equi join. In equi join comparison operator “=” is used to perform a Join.</p>	Definition: 2 marks, Any 2 types with description: 1 mark each



Syntax:

```
SELECT tablename.column1_name,tablename.column1_name  
  
FROM table_name1,table_name2  
  
where table_name1.column_name=table_name2.column_name;
```

Example:

```
Select stud_info.stud_name, stud_info.branch_code, branch_details.location
```

```
From stud_info, branch_details
```

```
Where Stud_info.branch_code=branch_details.branch_code;
```

2) SELF JOIN:

The SQL SELF JOIN is used to join a table to itself, as if the table were two tables, temporarily renaming at least one table in the SQL statement.

Syntax:

```
SELECT a.column_name, b.column_name  
  
FROM table1 a, table1 b  
  
WHERE a.common_field = b.common_field;
```

Example:

```
Select x.stud_name, y.stud_name
```

```
from stud_info x, stud_info y
```

```
Where x.leader= y.stud_id;
```

3) LEFT OUTER JOIN:

A left outer join retains all of the rows of the “left” table, regardless of whether there is a row that matches on the “right” table.

Syntax:

```
Select column1name,column2name  
  
from table1name any_alias1 ,table2name any_alias2  
  
on any_alias1.columnname(+) = any_alias2.columnname;
```

OR



```
Select column1name,column2name  
from table1name left outer join table2name  
on table1name.columnname= table2name.columnname;
```

Example:

```
select last_name, department_name  
from employees e, departments d  
on e.department_id(+) = d.department_id;
```

OR

```
select last_name, department_name  
from employees left outer join departments  
on employees.department_id = departments.department_id;
```

4) RIGHT OUTER JOIN:

A right outer join retains all of the rows of the “right” table, regardless of whether there is a row that matches on the “left” table.

Syntax:

```
Select column1name, column2name  
from table1name any_alias1, table2name any_alias2  
on any_alias1.columnname =any_alias2.columnname (+);
```

OR

```
Select column1name, column2name  
from table1name any_alias1 right outer join table2 name any_alias2  
on any_alias1.columnname =any_alias2.columnname;
```

Example:

```
Select last_name,department_name from employees e, departments d on  
e.department_id = d.department_id(+);
```

OR



	<p>Select last_name, department_name from employees e right outer join departments d on e.department_id = d.department_id;</p> <p>5) NON EQUI JOIN:</p> <p>Non equi joins is used to return result from two or more tables where exact join is not possible.</p> <p>Syntax:</p> <p>Select aliasname.column1name, aliasname.column2name from tablename alias where <condition using range>;</p> <p>For example:</p> <p>In emp table and salgrade table. The salgrade table contains grade and their low salary and high salary. Suppose you want to find the grade of employees based on their salaries then you can use NON EQUI join.</p> <p>Select e.empno, e.ename, e.sal, s.grade from emp e, salgrade s where e.sal between s.lowsal and s.hisal;</p>	
c	Explain function in PL/SQL with example.	4 M
Ans	<p>Function:</p> <p>Function is a logically grouped set of SQL and PL/SQL statements that perform a specific task. A function is same as a procedure except that it returns a value. A function is created using the CREATE FUNCTION statement.</p> <p>Syntax:</p> <pre>CREATE [OR REPLACE] FUNCTION function_name [(parameter_name [IN OUT IN OUT] type [, ...])] RETURN return_datatype {IS AS} BEGIN < function_body > END [function_name];</pre>	Description: 2 marks, Example : 2 marks



Where,

- *function-name* specifies the name of the function.
- [OR REPLACE] option allows the modification of an existing function.
- The optional parameter list contains name, mode and types of the parameters. IN represents the value that will be passed from outside and OUT represents the parameter that will be used to return a value outside of the procedure.
- The function must contain a **return** statement.
- The *RETURN* clause specifies the data type you are going to return from the function.
- *function-body* contains the executable part.
- The AS keyword is used instead of the IS keyword for creating a standalone function.

Example:

```
CREATE OR REPLACE FUNCTION Success_cnt
```

```
RETURN number
```

```
IS cnt number(7) := 0;
```

```
BEGIN
```

```
SELECT count(*) into cnt
```

```
FROM candidate where result='Pass';
```

```
RETURN cnt;
```

```
END;
```

```
/
```

d Explain database security with its requirements in detail.

4 M

Ans

Database security

Database security refers to the collective measures used to protect and secure a database or database management software from illegal use and malicious threats and attacks.

Requirements of Database Security:

1. For prevention of data theft such as bank account numbers, credit card information, passwords, work related documents or sheets, etc.
2. To make data remain safe and confidential.
3. To provide confidentiality which ensures that only those individuals should ever

Definition 1
Mark, 3 mark
for detail
description



be able to view data they are not entitled to.

4. To provide integrity which ensures that only authorized individuals should ever be able change or modify information.
5. To provide availability which ensure that the data or system itself is available for use when authorized user wants it.
6. To provide authentication which deals with the desire to ensure that an authorized individual.
7. To provide non-repudiation which deals with the ability to verify that message has been sent and received by an authorized user.

OR

1. Confidentiality: The principle of confidentiality specifies that only sender and intended recipients should be able to access the contents of a message. Confidentiality gets compromised if an unauthorized person is able to access the contents of a message

2. Integrity: when the contents of the message are changed after the sender sends it, but before it reaches the intended recipient, we say that the integrity of the message is lost.

3. Authentication: Authentication helps to establish proof of identities. The Authentication process ensures that the origin of a message is correctly identified.

4. **Availability:** The goal of availability s to ensure that the data, or the system itself, is available for use when the authorized user wants it.

4 Attempt any **THREE** of the following :

12 M

a Explain the four roles of database administrator.

4 M

Ans **1. Schema Definition** The Database Administrator creates the database schema by executing DDL statements. Schema includes the logical structure of database table (Relation) like data types of attributes, length of attributes, integrity constraints etc.

2. Storage structure and access method definition The DBA creates appropriate storage structures and access methods by writing a set of definitions which is translated by data storage and DDL compiler.

3. Schema and physical organization modification DBA writes set of definitions to modify the database schema or description of physical storage organization.

1 Mark for each role



4. Granting authorization for data access The DBA provides different access rights to the users according to their level. Ordinary users might have highly restricted access to data, while you go up in the hierarchy to the administrator, you will get more access rights. Integrity constraints specifications: Integrity constraints are written by DBA and they are stored in a special file which is accessed by database manager while updating data.

5. Routine Maintenance some of the routine maintenance activities of a DBA is given below.

- (i) Taking backup of database periodically
- (ii) Ensuring enough disk space is available all the time.
- (iii) Monitoring jobs running on the database.
- (iv) Ensure that performance is not degraded by some expensive task submitted by some users.

6. Integrity- constraint specification: Integrity constraints are written by DBA and they are stored in a special file, which is accessed by database manager, while updating the data.

b State and Explain 1 NF and 2 NF with example.

4 M

Ans First Normal Form (1NF):

A relation R is said to be in first normal form (1NF) if the domain of all attributes of R are atomic.

OR

A table is in the first normal form if it contains no repeating elements groups. Example:
Supplier(sno,sname,location,pno,qty)

SNO	SNAME	LOCATION	PNO	QTY
S1	Abc	Mumbai	P1	200
S2	Pqr	Pune	P2	300
S3	Lmn	Delhi	P1	400

The above relation is in 1NF as all the domains are having atomic value. But it is not in 2NF.

Second Normal Form (2NF):

A relation is said to be in the second normal form if it is in first normal form and all the non key attributes are fully functionally dependent on the primary key.

For 1NF - 2 Marks, For 2NF -2 Marks



Example:

In the above relation NAME, LOCATION depends on SNO and QTY on (SNO, PNO) so the table can be split up into two tables as Supplier(SNO,SNAME,LOCATION) and SP(SNO,PNO,QTY) and now both the tables are in second normal form.

Supplier

SNO	SNAME	LOCATION
S1	Abc	Mumbai
S2	Pqr	Pune
S3	Lmn	Delhi

Supplier_Product

SNO	PNO	QTY
S1	P1	200
S2	P2	300
S3	P1	400

c Draw the block structure of PL/SQL. List advantages of PL/SQL.

4 M

Ans

Declare (Optional)

--Use for declaring variables

Begin (Mandatory)

--Use for writing executable code;

Exception (Optional)

--Use to write exceptions to be catch during run time.

End; (Mandatory)

--To terminate PL-SQL block/ code.

For block structure - 2 Marks, For advantages -2 Marks



	<p>Advantages of PL/SQL:</p> <ol style="list-style-type: none">1. PL/SQL is portable and high transaction processing language.2. PL/SQL is in fact procedural language but it also supports object oriented programming.3. It allows user to write as well as access the functions and procedures from outside the programs.4. It has got built in libraries of packages.	
d	Write step by step syntax to create, open and close cursor in PL/SQL.	4 M
Ans	<p>A cursor holds the rows (one or more) returned by a SQL statement.</p> <p>Declaring: This term is used to declare a cursor so that memory initialization will take place.</p> <p>A cursor is declared by defining the SQL statement that returns a result set.</p> <p>Example:</p> <p>Declare CURSOR Winter_18 IS SELECT roll_no, std_name, percentage FROM student;</p> <p>Opening: A Cursor is opened and populates data by executing the SQL statement defined by the cursor.</p> <p>Example:</p> <p>Open Winter_18;</p> <p>Closing a Cursor: This forces cursor for releasing the allocated memory assigned/occupied by cursor.</p> <p>Example:</p> <p>CLOSE Winter_18;</p>	2 marks, Opening: 1 mark, Closing cursor: 1 mark
e	Explain Transaction ACID properties.	4 M
Ans	<p>ACID properties of transaction</p> <ol style="list-style-type: none">1. Atomicity: When one transaction takes place, many operations occur under one transaction. Atomicity means either all operations will take place property and reflect in the database or none of them will be reflected.2. Consistency: Consistency keeps the database consistent. Execution of a transaction	For each property - 1 Mark



needs to take place in isolation. It helps in reducing complications of executing multiple transactions at a time and preserves the consistency of the database.

3. Isolation: It is necessary to maintain isolation for the transactions. This means one transaction should not be aware of another transaction getting executed. Also their intermediate result should be kept hidden.

4. Durability: When a transaction gets completed successfully, it is important that the changes made by the transaction should be preserved in database in spite of system failures.

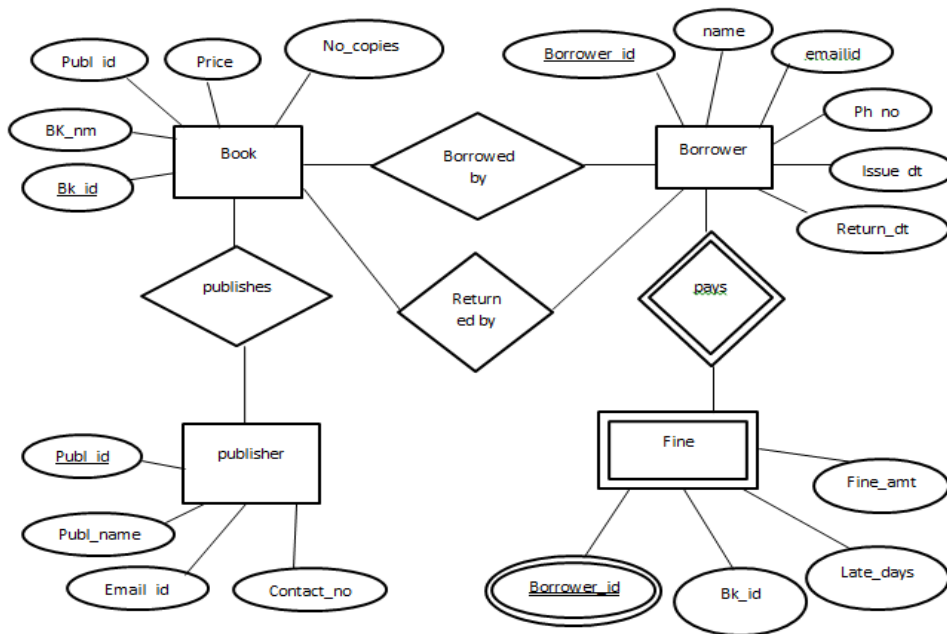
5 Attempt any TWO of the following :

12 M

a Draw an E-R diagram of library management system considering issue and return, fine calculation facility, also show primary key, weak entity and strong entity.

6 M

Ans



Correct entities: 2M,
correct symbols: 2M,
Correct relationships: 2M

**b Consider the following database
Employee(emp_id,emp_name,emp_city,emp_addr,emp_dept,join_date)**

6 M

- i) Display the emp_id of employee who live in city 'Pune' or 'Nagpur'.**
- ii) Change the employee name 'Ayush' to 'Ayan'.**
- iii) Display the total number of employee whose dept is 50.**

Ans i) Display the emp_id of employee who live in city 'Pune' or 'Nagpur'

select emp_id
from Employee

Each query :
2M



		<p>where emp_city='Pune' or emp_city='Nagpur'</p> <p>ii) Change the employee name 'Ayush' to 'Ayan'</p> <p>update Employee</p> <p>set emp_name='Ayan'</p> <p>where emp_name='Ayush'</p> <p>iii) Display the total number of employee whose dept is 50</p> <p>Select count(*)</p> <p>from Employee</p> <p>where emp_dept=50;</p>	
	c	Consider the following schema Depositor (ACC_no, Name, PAN, Balance). Create a view on Depositor having attributes(ACC_No,PAN) where balance is greater than 100000	6 M
	Ans	<p>create view v1</p> <p>as</p> <p>select ACC_No,PAN</p> <p>from Depositor</p> <p>where balance > 100000;</p>	Correct logic 3M, Correct syntax :3M
6		Attempt any TWO of the following :	12 M
	a	<p>Create a sequence</p> <p>i) Sequence name is Seq_1, Start with 1, increment by 1, minimum value 1, maximum value 20.</p> <p>ii) Use a seq_1 to insert the values into table Student(ID Number(10), Name char (20));</p> <p>iii) Change the Seq_1 max value 20 to 50.</p> <p>iv) Drop the sequence.</p>	6 M
	Ans	<p>i) create sequence Seq_1 start with 1 increment by 1 minvalue 1 maxvalue 20;</p> <p>ii) insert into student values(Seq_1.nextval,'ABC');</p>	Query 1: 2M, Query 2: 2M, Query 3 : 1M, Query 4 : 1M



		iii) Alter sequence Seq_1 maxvalue 50; iv) Drop sequence Seq_1;	
b		Write a PL/SQL program which accepts the customer_ID from the user. If the enters an invalid ID then the exception invalid_id is raised using exception handling.	6 M
Ans		DECLARE c_id numeric(10); invalid_id_Exception Exception; BEGIN c_id:=&c_id; if(c_id<0) then raise invalid_id_Exception; end if; EXCEPTION WHEN invalid_id_Exception THEN dbms_output.put_line('Invalid customer id'); END;	Correct logic : 3M, Correct syntax : 3M
c		i) create user 'Rahul' ii) grant create, select,insert,update, delete, drop privilege to 'Rahul' iii) Remove the select privilege from user 'Rahul'	6 M
Ans		(i) create user Rahul identified by rahul1234; (ii) 1) assuming table Employee for granting permissions to user 'Rahul' for select, insert, update and delete privilege) Grant select, insert,update,delete on employee to Rahul; 2) for create and drop privilege which are system privileges not specific to any object such as table Grant connect, resource, DBA to Rahul; iii) (assuming table Employee for revoking permissions to user 'Rahul') Revoke select on Employee from Rahul;	each query : 2M



V2V EDTECH LLP

Online Coaching at an Affordable Price.

OUR SERVICES:

- Diploma in All Branches, All Subjects
- Degree in All Branches, All Subjects
- BSCIT / CS
- Professional Courses



+91 93260 50669



v2vedtech.com



V2V EdTech LLP



v2vedtech



SUMMER – 19 EXAMINATION

Subject Name: Database Management System Model Answer

Subject Code: 22319

Important Instructions to examiners:

- 1) The answers should be examined by key words and not as word-to-word as given in the model answer scheme.
- 2) The model answer and the answer written by candidate may vary but the examiner may try to assess the understanding level of the candidate.
- 3) The language errors such as grammatical, spelling errors should not be given more Importance (Not applicable for subject English and Communication Skills).
- 4) While assessing figures, examiner may give credit for principal components indicated in the figure. The figures drawn by candidate and model answer may vary. The examiner may give credit for any equivalent figure drawn.
- 5) Credits may be given step wise for numerical problems. In some cases, the assumed constant values may vary and there may be some difference in the candidate's answers and model answer.
- 6) In case of some questions credit may be given by judgement on part of examiner of relevant answer based on candidate's understanding.
- 7) For programming language papers, credit may be given to any other program based on equivalent concept.

Q. No.	Sub Q. N.	Answer	Marking Scheme
1.		Attempt any FIVE of the following:	10 M
	a	Define : (i) Instance (ii) Schema	2 M
	Ans	(i) Instance: The data stored in database at a particular moment of time is called instance of database. (ii) Schema: Design of a database is called the schema. Schema is of three types: Physical schema, logical schema and view schema.	1 M for each Definition
	b	List any four advantages of DBMS.	2 M
	Ans	<ul style="list-style-type: none">• Controlling Redundancy• Maintaining Integrity• Inconsistency can be avoided• Data can be shared• Restricting unauthorized access• Providing Backup and Recovery• Concurrency Control• Better security.	(½ M for any advantage)
	c	State any two E.F. Codd's rule for RDBMS.	2 M
	Ans	1. The Information rule: All information in an RDBMS is represented logically in just one way - by values in tables.	½ M for each rule , ½ M each



	<p>2. The Guaranteed Access rule: Each item of data in an RDBMS is guaranteed to be logically accessible by resorting to a combination of table name, primary key value, and column name.</p> <p>3. The Systematic Treatment of Null Values rule: Null values (distinct from an empty character string or a string of blank characters and distinct from zero or any other number) are supported in a fully relational DBMS for representing missing</p> <p>4. The Dynamic Online Catalog Based on the Relational Model rule: The database description is represented at the logical level in the same way as ordinary data, so that authorized users can apply the same relational database.</p> <p>5. The Comprehensive Data Sublanguage rule: A relational system may support several languages and various modes of terminal for data definition, view definition, data manipulation etc.</p> <p>6. The View Updating rule: All views of the data which are theoretically updatable must be updatable in practice by the DBMS.</p> <p>7. The High-level Insert, Update, and Delete rule: The capability of handling a base relation or a derived relation as a single database to perform all DML operations.</p> <p>8. The Physical Data Independence rule: Application programs and terminal activities remain logically unchanged whenever any changes are made in either storage representations or access methods.</p> <p>9. The Logical Data Independence rule: Application programs and terminal activities remain logically unchanged when information preserving changes of any kind are made to the base tables.</p> <p>10. The Integrity Independence rule: Integrity constraints must be definable in the RDBMS sub-language and stored in the system catalogue and not within individual application programs.</p> <p>11. The Distribution Independence rule: An RDBMS has distribution independence. Distribution independence implies that users should not have to be aware of whether a database is distributed.</p> <p>12. The No subversion rule: If the database has any means of handling a single record at a time that low-level language must not be able avoid the integrity rules which are expressed in a higher-level language that handles multiple records at a time.</p>	proper statement
d	List DCL commands.	2 M
Ans	DCL is Data Control Language: 1. GRANT 2. REVOKE	1 M for each command
e	Define Normalization and list its types.	2 M
Ans	Normalization is a process of organizing the data in database to avoid data redundancy, insertion anomaly, update anomaly & deletion anomaly.	1 M for definition, 1 M for the types



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2013 Certified)

		Types of normalization are : <ul style="list-style-type: none">• First normal form(1NF)• Second normal form(2NF)• Third normal form(3NF)• Boyce & Codd normal form (BCNF)• Fourth normal form(4NF)	
	f	Write syntax for creating synonyms with example	2 M
	Ans	Syntax to create synonym: CREATE SYNONYM SYNONYM_name FOR Table_name; Example to create synonym: CREATE SYNONYM offices FOR locations;	1 M for correct syntax, 1 M for correct example
	g	State any four PL/SQL data types.	2 M
	Ans	1. NUMBER or NUMBER(P,S) 2. PLS_INTEGER 3. CHAR 4. RAW 5. ROWID 6. VARCHAR2 7. DATE	½ M for each data type
2		Attempt any THREE of the following:	12 M
	a	Explain overall structure of DBMS with the help of diagram.	4 M
	Ans	Components of DBMS structure are classified in 3 categories as: 1. Query processor : Embedded DML pre compiler: It converts DML statements embedded in application. Program to normal procedural calls in host language. DML Compiler: It translates DML statements of high level language into low level instruction that a query evaluation engine understands. DDL interpreter: It interprets DDL statements and records them in a set of tables containing metadata. Query evaluation Engine: It executes low level instructions generated by DML compiler and issued by query processor to select efficient ways to execute query. DDL interpreter. It has following components,	2 M for correct diagram, 2 M for correct explanation



2. Storage Manager Components :

Transaction manager: It ensures that the database remains in consistent state despite of the system failure and that concurrent transaction execution proceeds without conflicting.

File Manager: It manages the allocation of space on disk storage and data structures used to represent information stored on disk

Buffer Manager: It is responsible for fetching data from disk storage into main memory and deciding what data to cache memory.

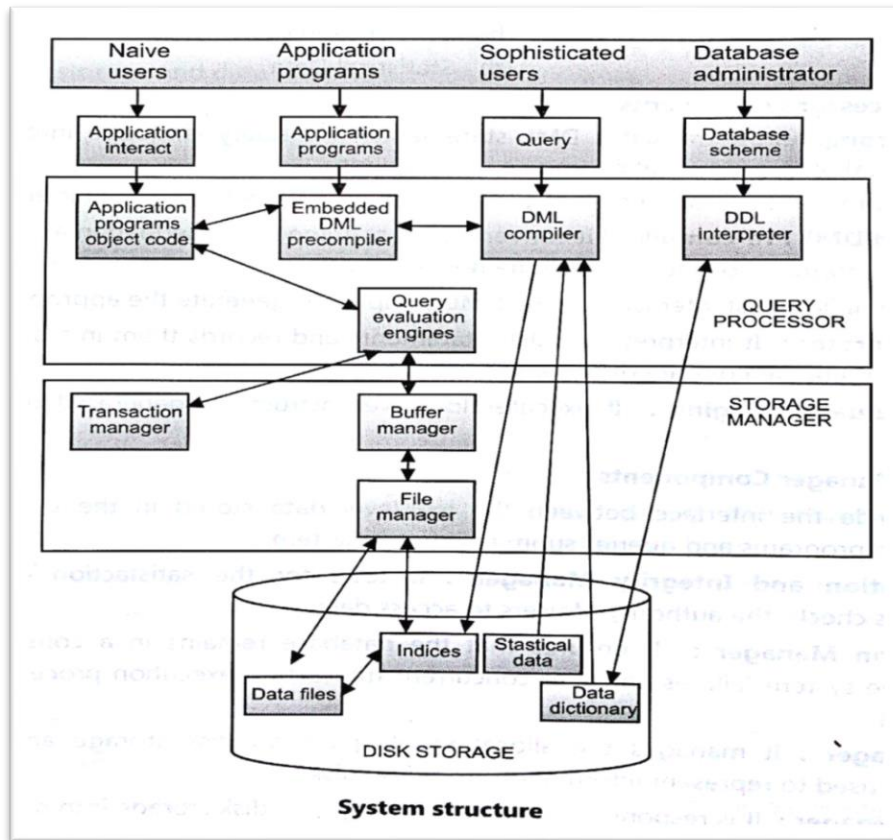
3. Disk storage :

Data files: It stores the database.

Data Dictionary: It stores metadata that hold particular values.

Indices: Provide fast access to data items that hold particular values.

Statistical data: It stores statistical information about the data in the database.



b	Explain difference between delete and truncate command with example.	4 M
----------	---	------------

Ans	DELETE Command : <ul style="list-style-type: none"> • It is DML (Data Manipulation Language) command. 	
------------	---	--



		<ul style="list-style-type: none">• It is used to remove all or specific records of table.• WHERE clause can be used to remove specific records.• Syntax: DELETE FROM Table_name; <p>OR</p> <p>DELETE FROM Table_name WHERE Condition;</p> <ul style="list-style-type: none">• Example: DELETE FROM Employees WHERE Emp_id=100;• ROLLBACK command can be used to get deleted record. <p>TRUNCATE Command :</p> <ul style="list-style-type: none">• It is a DDL(Data Definition Language) command• It is used to remove all records permanently.• WHERE clause can be used as it removes all records.• Syntax: TRUNCATE TABLE Table_name;• Example: TRUNCATE TABLE Employees;• ROLLBACK command cannot be used to get records.• New records can be added into a table as structure remains intact. <p>OR</p>	<p>(2 M for proper explanation of each command) or (any 4 differences)</p>
--	--	--	--



		<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 50%;">DELETE</th> <th style="width: 50%;">TRUNCATE</th> </tr> </thead> <tbody> <tr> <td>It is DML(Data Manipulation Language) command</td> <td>It is a DDL(Data Definition Language) command</td> </tr> <tr> <td>It is used to remove all or specific records of table.</td> <td>It is used to remove all records permanently.</td> </tr> <tr> <td>WHERE clause can be used to remove specific records.</td> <td>WHERE clause can be used as it removes all records.</td> </tr> <tr> <td>Syntax: DELETE FROM Table_name; OR DELETE FROM Table_name WHERE Condition;</td> <td>Syntax: TRUNCATE TABLE Table_name;</td> </tr> <tr> <td>Example: DELETE FROM Employees WHERE Emp_id=100;</td> <td>Example: TRUNCATE TABLE Employees;</td> </tr> <tr> <td>ROLLBACK command can be used to get deleted record.</td> <td>ROLLBACK command cannot be used to get records. New records can be added into a table as structure remains intact.</td> </tr> </tbody> </table>	DELETE	TRUNCATE	It is DML(Data Manipulation Language) command	It is a DDL(Data Definition Language) command	It is used to remove all or specific records of table.	It is used to remove all records permanently.	WHERE clause can be used to remove specific records.	WHERE clause can be used as it removes all records.	Syntax: DELETE FROM Table_name; OR DELETE FROM Table_name WHERE Condition;	Syntax: TRUNCATE TABLE Table_name;	Example: DELETE FROM Employees WHERE Emp_id=100;	Example: TRUNCATE TABLE Employees;	ROLLBACK command can be used to get deleted record.	ROLLBACK command cannot be used to get records. New records can be added into a table as structure remains intact.	
DELETE	TRUNCATE																
It is DML(Data Manipulation Language) command	It is a DDL(Data Definition Language) command																
It is used to remove all or specific records of table.	It is used to remove all records permanently.																
WHERE clause can be used to remove specific records.	WHERE clause can be used as it removes all records.																
Syntax: DELETE FROM Table_name; OR DELETE FROM Table_name WHERE Condition;	Syntax: TRUNCATE TABLE Table_name;																
Example: DELETE FROM Employees WHERE Emp_id=100;	Example: TRUNCATE TABLE Employees;																
ROLLBACK command can be used to get deleted record.	ROLLBACK command cannot be used to get records. New records can be added into a table as structure remains intact.																
c		Write and explain syntax for creating view with example.	4 M														
Ans	<p>A view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database.</p> <p>View has two types:</p> <ol style="list-style-type: none"> 1. Simple view: The fields in a view are fields from one table in the database. 2. Complex view: The fields in a view are fields from more than one table in the database. You can add SQL functions, WHERE, and JOIN statements to a view and present the data as if the data were coming from different table. <p>CREATE VIEW Syntax Create view view_name As</p>		<p>2 M for correct syntax, 1 M for explanation, 1 M for correct example</p>														



	<p>Select column1, column2...</p> <p>From table_name</p> <p>Where condition ;</p> <p>Example</p> <p>Create view mumbai_customers AS</p> <p>Select customer_name,contact_name</p> <p>From customers</p> <p>Where city='Mumbai';</p>	
d	Explain PL/SQL block structure with the help of diagram.	4 M
Ans.	<p>PL/SQL Block Structure :</p> <p style="text-align: center;"><i>Declare</i></p> <p style="text-align: center;"><i>Declaration of memory variables</i></p> <p style="text-align: center;"><i>BEGIN (Mandatory)</i></p> <p style="text-align: center;"><i>SQL executable statements</i></p> <p style="text-align: center;"><i>Exception</i></p> <p style="text-align: center;"><i>Handling errors</i></p> <p style="text-align: center;"><i>END; (Mandatory)</i></p> <p>Explanation of PL/SQL Block Structure:</p> <p>Declaration section</p> <p>A block begins with declarative section where variables, cursors are declared. It is an Optional block.</p> <p>Execution section</p> <p>Executable SQL or PL/SQL Statements are needed to write here</p>	<p>PL/SQL block structure 2M, Explanation 2M</p>



		<p style="text-align: center;">for the execution. It is mandatory block.</p> <p>Exception section It is used to handles the exceptions. It is an Optional block.</p> <p>End statement It is used to indicate termination of PL/SQL block. It is mandatory.</p>																			
3		Attempt any THREE of the following:	12 M																		
	a	State and explain 2NF with example.	4 M																		
	Ans	<p>A table is said to be in 2NF if both the following conditions hold:</p> <ul style="list-style-type: none"> • Table is in 1NF (First normal form) • No non-prime attribute is dependent on the proper subset of any candidate key of table. • San attribute that is not part of any candidate key is known as non-prime attribute. • Example: Suppose a school wants to store the data of teachers and the subjects they teach. They create a table that looks like this: Since a teacher can teach more than one subjects, the table can have multiple rows for a same teacher. <table border="1" style="margin-left: auto; margin-right: auto; border-collapse: collapse; text-align: center;"> <thead> <tr> <th>teacher_id</th> <th>Subject</th> <th>teacher_age</th> </tr> </thead> <tbody> <tr> <td>111</td> <td>Math's</td> <td>38</td> </tr> <tr> <td>111</td> <td>Physics</td> <td>38</td> </tr> <tr> <td>222</td> <td>Biology</td> <td>38</td> </tr> <tr> <td>333</td> <td>Physics</td> <td>40</td> </tr> <tr> <td>333</td> <td>Chemistry</td> <td>40</td> </tr> </tbody> </table> <p>CandidateKeys: {teacher_id,subject}</p> <p>Non-prime attribute: teacher_age The table is in 1 NF because each attribute has atomic values. However, it is not in 2NF because non-prime attribute teacher_age is dependent on teacher_id alone which is a proper subset of candidate key. This violates the rule for 2NF as the rule says “no non-prime attribute is dependent on the proper subset of any candidate key of the table “To make the table complies with 2NF we can break it in two tables like this: teacher details tab</p>	teacher_id	Subject	teacher_age	111	Math's	38	111	Physics	38	222	Biology	38	333	Physics	40	333	Chemistry	40	<p>State : 1M Explanation with example: 3M</p>
teacher_id	Subject	teacher_age																			
111	Math's	38																			
111	Physics	38																			
222	Biology	38																			
333	Physics	40																			
333	Chemistry	40																			



		<table border="1"><thead><tr><th>teacher_id</th><th>teacher_age</th></tr></thead><tbody><tr><td>111</td><td>38</td></tr><tr><td>222</td><td>40</td></tr><tr><td>333</td><td>40</td></tr></tbody></table> <p>teacher_subject Table:</p> <table border="1"><thead><tr><th>Teacher_id</th><th>Subject</th></tr></thead><tbody><tr><td>111</td><td>Math's</td></tr><tr><td>111</td><td>Physics</td></tr><tr><td>222</td><td>Biology</td></tr><tr><td>333</td><td>Physics</td></tr><tr><td>333</td><td>Chemistry</td></tr></tbody></table>	teacher_id	teacher_age	111	38	222	40	333	40	Teacher_id	Subject	111	Math's	111	Physics	222	Biology	333	Physics	333	Chemistry	
teacher_id	teacher_age																						
111	38																						
222	40																						
333	40																						
Teacher_id	Subject																						
111	Math's																						
111	Physics																						
222	Biology																						
333	Physics																						
333	Chemistry																						
b	Explain any four aggregate functions with example.	4 M																					
Ans	<p>An aggregate function is a function where the values of multiple rows are grouped together as input on certain criteria to form a single value of more significant meaning.</p> <p>Aggregate functions are :</p> <ol style="list-style-type: none">1) Count()2) Sum()3) Avg()4) Min()5) Max() <p>1. Count () - 1) It returns number of rows from the given table if no attribute is mentioned.</p> <p>2) If some attribute is mentioned, it gives total number of not null values</p>	<p>Any 4 aggregate functions with example : 1M each</p>																					



	<p>for that attribute.</p> <p>Eg :Select count(*) from emp;</p> <p>Returns total number of records from emp table.</p> <p>1) Select count(telephone) from emp;</p> <p>Returns total number of employees having telephone numbers.</p> <p>2. Sum() - It give total of all values from a numeric attribute of the given table,</p> <p>Eg :Select sum(salary) from emp;</p> <p>Returns total salary drawn of all employees from the emp table.</p> <p>3. Avg () - It gives average of all the numeric values of the given attribute from the table.</p> <p>Eg :Select Avg(salary) from emp;</p> <p>Returns average salary of employees from emp table.</p> <p>4. Min () - It gives minimum of all the values of the numeric given attribute from the table.</p> <p>Eg :Select Min(salary) from emp;</p> <p>Returns minimum salary value from emp table,</p> <p>5. Max () - It gives maximum of all the values of the numeric given attribute from the table.</p> <p>Eg :Select Max(salary) from emp;</p> <p>retunes maximum salary value from emp table,</p>	
c	Explain exception handling in PL/SQL with example.	4 M
Ans	<p>Exception handling in PL/SQL:</p> <p>An exception is an error condition during a program execution. PL/SQL supports programmers to catch such conditions using EXCEPTION block in the program and an appropriate action is taken against the error condition.</p> <p>There are two types of exceptions –</p> <ul style="list-style-type: none">• System-defined (built in) exceptions	<p>Explanation : 2M , example :2M</p>



- User-defined exceptions

The general syntax for exception handling is as follows :

```
DECLARE
<declarations section>
BEGIN
<executable command(s)>
EXCEPTION
<exception handling goes here >
WHEN exception1 THEN
exception1-handling-statements
WHEN exception2 THEN
exception2-handling-statements
.....
.....
END;
```

Raising Exceptions

Exceptions are raised by the database server automatically whenever there is any internal database error, but exceptions can be raised explicitly by the programmer by using the command **RAISE**. Following is the simple syntax for raising an exception

```
DECLARE
exception_name EXCEPTION; BEGIN
IF condition THEN
RAISE exception_name;
END IF;
EXCEPTION
WHEN exception_name THEN
statement;
END;
```

You can use the above syntax in raising the Oracle standard exception or any user-defined exception.

Example :

DECLARE

```
A number:=20;
B number:=0;
C number;
```

BEGIN

```
dbms_output.put_line('First Num : '||A);
dbms_output.put_line('Second Num : '||B);
```



		<pre> C:= A / B; --Raise built in Exception if B is 0 dbms_output.put_line(' Result ' C);-- and then Result will not be displayed EXCEPTION WHEN ZERO_DIVIDE THEN dbms_output.put_line(' Trying to Divide by zero :: Error '); END; </pre>							
	d	Explain states of transaction with the help of diagram.	4 M						
	Ans	<div style="text-align: center;"> <pre> graph LR Active((Active)) --> PartiallyCommitted((Partially Committed)) Active --> Failed((Failed)) PartiallyCommitted --> Committed((Committed)) PartiallyCommitted --> Failed Failed --> Aborted((Aborted)) </pre> </div> <p>Active –the initial state; the transaction stays in this state while it is executing</p> <p>Partially committed –after the final statement has been executed.</p> <p>Failed - after the discovery that normal execution can no longer proceed.</p> <p>Aborted – after the transaction has been rolled back and the database restored to its state prior to the start of the transaction. Two options after it has been aborted: restart the transaction - can be done only if no internal logical error kill the transaction Committed –after successful completion.</p>	diagram : 1M, explanation : 3M						
	4	Attempt any THREE of the following:	12 M						
	a	State difference between relational and hierarchical model.	4 M						
	Ans	<table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 50%;">Relational model</th> <th style="width: 50%;">Hierarchical model.</th> </tr> </thead> <tbody> <tr> <td>A database model to manage data as tuples grouped into relations(tables)</td> <td>A structure of data organized in a tree like model using parent child relationships.</td> </tr> <tr> <td>Arranges data in tables</td> <td>Arranges data in tree like structure</td> </tr> </tbody> </table>	Relational model	Hierarchical model.	A database model to manage data as tuples grouped into relations(tables)	A structure of data organized in a tree like model using parent child relationships.	Arranges data in tables	Arranges data in tree like structure	Any 4 differences : 1M each
Relational model	Hierarchical model.								
A database model to manage data as tuples grouped into relations(tables)	A structure of data organized in a tree like model using parent child relationships.								
Arranges data in tables	Arranges data in tree like structure								



		<p>Represents both “one to many” and”many to many” relationships.</p> <p>Easier to access data</p> <p>Flexible</p> <p>Example</p> <table border="1" style="margin-bottom: 5px;"> <thead> <tr><th>Student ID</th><th>First name</th><th>Last name</th></tr> </thead> <tbody> <tr><td>52-743965</td><td>Charles</td><td>Peters</td></tr> <tr><td>48-209689</td><td>Anthony</td><td>Scndrup</td></tr> <tr><td>14-204968</td><td>Rebecca</td><td>Phillips</td></tr> </tbody> </table> <table border="1" style="margin-bottom: 5px;"> <thead> <tr><th>ProviderID</th><th>Provider name</th></tr> </thead> <tbody> <tr><td>156-983</td><td>UnitedHealth</td></tr> <tr><td>146-823</td><td>Blue Shield</td></tr> <tr><td>447-784</td><td>Carefirst Inc.</td></tr> </tbody> </table> <table border="1"> <thead> <tr><th>Student ID</th><th>ProviderID</th><th>Type of plan</th><th>Start date</th></tr> </thead> <tbody> <tr><td>52-743965</td><td>156-983</td><td>HSA</td><td>04/01/2016</td></tr> <tr><td>48-209689</td><td>146-823</td><td>HMO</td><td>12/01/2015</td></tr> <tr><td>14-204968</td><td>447-784</td><td>HSA</td><td>03/14/2016</td></tr> </tbody> </table>	Student ID	First name	Last name	52-743965	Charles	Peters	48-209689	Anthony	Scndrup	14-204968	Rebecca	Phillips	ProviderID	Provider name	156-983	UnitedHealth	146-823	Blue Shield	447-784	Carefirst Inc.	Student ID	ProviderID	Type of plan	Start date	52-743965	156-983	HSA	04/01/2016	48-209689	146-823	HMO	12/01/2015	14-204968	447-784	HSA	03/14/2016	<p>Represents “one to many” relationship</p> <p>Difficult to access data</p> <p>Less flexible</p> <p>Example :</p>	
Student ID	First name	Last name																																						
52-743965	Charles	Peters																																						
48-209689	Anthony	Scndrup																																						
14-204968	Rebecca	Phillips																																						
ProviderID	Provider name																																							
156-983	UnitedHealth																																							
146-823	Blue Shield																																							
447-784	Carefirst Inc.																																							
Student ID	ProviderID	Type of plan	Start date																																					
52-743965	156-983	HSA	04/01/2016																																					
48-209689	146-823	HMO	12/01/2015																																					
14-204968	447-784	HSA	03/14/2016																																					
	b	List the SQL operations and explain range searching operations between and pattern matching operator ‘like’ with example.	4 M																																					
	Ans	<p>Types of SQL operators :</p> <ol style="list-style-type: none"> 1) SQL Arithmetic Operators 2) SQL Comparison Operators 3) SQL Logical Operators <p>Arithmetic operators are used to perform arithmetic operations on numbers. They are +,-,*, / and %.</p> <p>Comparison operators are used in between two variables to compare their values. They are <,>,<=,>=,=,!< or <>,!< and !>.'</p> <p>Logical operators are used for the Boolean results in sql queries for comparison of values from the attributes of the tables. Eg: Any, Exists, All, Like, Between, In etc.</p> <p>Between operator: The BETWEEN operator is used to search for values that are within a set of values, given the minimum value and the maximum value inclusive of both the limits.</p> <p>Eg: select * from emp where salary between 40000 and 50000;</p> <p>This will results in rows from emp table where salary falls in the range of 40000 to 50000.</p>	<p>List of operators : 2M, between operator : 1M, Like operator : 1M</p>																																					



	<p>Like operator :</p> <p>The LIKE operator is used to compare a value to similar values using wildcard operators. It uses two wild characters as ‘%’ and ‘_’ where ‘%’ represents all characters of the pattern and ‘_’ represents one single character from pattern.</p> <p>Eg :</p> <p>Select ename from emp where ename like ‘S%’;</p> <p>This will return all employee names starting with ‘S’.</p> <p>Select ename from emp where ename like ‘_a%’;</p> <p>This will return all employee names whose second character is ‘a’.</p>	
c	Explain cursor with example.	4 M
Ans	<p>A cursor is a temporary work area created in system memory when a SQL statement is executed. A cursor is a set of rows together with a pointer that identifies a current row. It is a database object to retrieve data from a result set one row at a time. It is useful when we want to manipulate the record of a table in a singleton method, in other words one row at a time. In other words, a cursor can hold more than one row, but can process only one row at a time. The set of rows the cursor holds is called the active set.</p> <p>Each cursor contains the followings 4 steps,</p> <ol style="list-style-type: none">1. Declare Cursor: In this part we declare variables and return a set of values.2. Open: This is the entering part of the cursor.3. Fetch: Used to retrieve the data row by row from a cursor.4. Close: This is an exit part of the cursor and used to close a cursor.5. Eg: <p>Declare</p> <pre>enumemp.eno%type; enemp.ename%type;</pre> <p>Cursor cur is select eno, ename from emp where jobname = “mgr”;</p> <p>Begin</p>	<p>Explanation : 2M, example : 2M</p>



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2013 Certified)

	<pre>Open cur; Loop Fetch cur into enum,en; Exit when cur%NOTFOUND; Dbms_output.put_line(„emp num “ enum “ emp name „ en); End loop; Close cur; End; /</pre> <p>The example shows fetching multiple records using cursor. A cursor is a temporary work area created in system memory when a SQL statement is executed. A cursor is a set of rows together with a pointer that identifies a current row.</p> <p>In the example, the cursor is defined to hold the rows as defined by the select query. Once the cursor is defined, the next step is to open the cursor. When the cursor is opened, it is ready to retrieve the rows. This is done using the fetch statement. Since there are many rows, a loop is used to display the values of all the rows. Once the rows are fetched, the cursor should be closed.</p>	
d	State the use of database trigger and also list types of trigger.	4 M
Ans	Use of trigger <p>Trigger: A trigger is a stored procedure in database which automatically invokes whenever a special event in the database occurs. A trigger can be invoked when a row is inserted into a specified table or when certain table columns are being updated.</p> <p>Triggers are written to be executed in response to any of the following events –</p> <p>A database manipulation (DML) statement (DELETE, INSERT, or UPDATE)</p> <p>Database definition (DDL) statements (CREATE, ALTER, or DROP).</p> <p>A database operation (SERVERERROR, LOGON, LOGOFF, STARTUP, or SHUTDOWN).</p> <p>Triggers can be defined on the table, view, schema, or database with which the event is associated.</p>	Use : 3M List of types : 1M



	<p>Triggers can be written for the following purposes –</p> <ul style="list-style-type: none">● Generating some derived column values automatically● Enforcing referential integrity● Event logging and storing information on table access● Auditing● Synchronous replication of tables● Imposing security authorizations● Preventing invalid transactions <p>Types of trigger</p> <ul style="list-style-type: none">• DML Triggers• DDL Triggers• Logon Triggers	
e	Explain recovery techniques with example.	4 M
Ans	<p>When recovering the database, it is must redo the effects of the previous transactions. This is called Rolling Forward or simple Forward Recovery. Not all but some active transaction that didn't complete successfully needs to rollback, when the disk drive crashed. Such kind of rollback is called Backward Recovery.</p> <p>The Redo Log and Rolling Forward (REDO operation)</p> <p>The redo log is a set of operating system files that record all changes made to any database buffer, including data, index, and rollback segments, whether the changes are committed or uncommitted. The redo log protects changes made to database buffers in memory that have not been written to the data files.</p> <p>The first step of recovery from an instance or disk failure is to roll forward, or reapply all of the changes recorded in the redo log to the data files. Because rollback data is also recorded in the redo log, rolling forward also regenerates the corresponding rollback segments.</p> <p>Rolling forward proceeds through as many redo log files as necessary to bring the database forward in time. Rolling forward usually includes online redo log files and may include archived redo log files.</p> <p>After roll forward, the data blocks contain all committed changes as</p>	<p>Explanation : 3M, Example 1M</p>



	<p>well as any uncommitted changes that were recorded in the redo log.</p> <p>Rollback Segments and Rolling Back (UNDO operation) Rollback segments record database actions that should be undone during certain database operations. In database recovery, rollback segments undo the effects of uncommitted transactions previously applied by the rolling forward phase. After the roll forward, any changes that were not committed must be undone. After redo log files have reapplied all changes made to the database, then the corresponding rollback segments are used. Rollback segments are used to identify and undo transactions that were never committed, yet were recorded in the redo log and applied to the database during roll forward. This process is called rolling back.</p> <div data-bbox="532 762 1105 1182" data-label="Diagram"><p>The diagram, titled "Transaction Recovery", shows a timeline of transactions T₁ through T₅. A vertical line marks a "Checkpoint" where T₁, T₂, and T₃ are active. After the checkpoint, T₄ and T₅ are active. A "Failure" occurs after T₅. Below the timeline, it lists: UNDO: T₂, T₃; REDO: (blank); Last Checkpoint; Active transactions: T₂, T₃. The source "Transactions and Recovery" is noted at the bottom.</p></div> <p>(Descriptive example can be considered)</p>	
5	Attempt any TWO of the following:	12 M
a	Draw an E-R diagram of library management system considering issue and return, Fine calculation facility. Consider appropriate entities.	6M



Ans		<p>Correct entities: 2M,</p> <p>correct symbols: 2M,</p> <p>Correct relationships: 2M</p>
b	<p>Consider the table Student (name, marks, dept, age, place, phone, birthdate). Write SQL query for following.</p> <p>i) To list students having place as ‘Pune’ or ‘Jalgaon’</p> <p>ii) To list students having same department(dept) as that of ‘Rachana’</p> <p>iii) To change marks of ‘Rahul’ from 81 to 96.</p> <p>iv) To list student name and marks from ‘Computer’ dept.</p> <p>v) To list student name who have marks less than 40.</p> <p>vi) To list students who are not from ‘Mumbai’;</p>	6M
Ans	<pre> select name from Student where place= 'Pune' or place='Jalgaon'; (OR) select name from Students where place in('Pune', 'Jalgaon'); ii)select name from Student where dept=(select dept from student where name='Rachana'); iii)update Student set marks=96 where name= 'Rahul'; v)select name,marks from Student where dept='Computer'; iv)select name from Student where marks<40; v)select * from Student where place != 'Mumbai'; </pre>	<p>Each Correct Query : 1M</p>
c	<p>Create simple and composite index. Write command to drop above index.</p>	6M
Ans	<p><u>Create simple index</u> Syntax: Create index index_name on <tablename><column name>; (OR)</p>	<p>Simple index 2M,</p> <p>Composite</p>



		<p>E.g.: Create index idx_empno on employee (empno); <u>Create composite index:</u> Syntax: Create index index_name on <tablename><Column_name1, Column_name2>; (OR) E.g.: Create index idx_ename_eno on employee (ename, empno);</p> <p><u>Drop Index:</u> Syntax: Drop index <index_name>; (OR) E.g. (Assuming idx_empno created on employee table) Drop index idx_empno;</p>	<p>index: 2M Drop index 2M (Note: Either syntax or example can be considered. Any other example allowed.)</p>
6		Attempt any TWO of the following:	12 M
	a	<p>i) Write a command to create table student(RNO,name marks, dept) with proper datatypes and RNo as primary key ii) Write a command to create and drop sequence.</p>	6M
	Ans	<p>i) create table student (RNO number(5) constraint student_RNO_pk primary key, name varchar2(20), marks number(4), dept varchar2(20)); (OR) create table student (RNO number(5) , name varchar2(20), marks number(4), dept varchar2(20), constraint student_RNO_pk primary key(RNO),);</p> <p>ii) Create Sequence: Create sequence <seq_name> Start with [initial value] Increment by [value] Minvalue [minimum value] Maxvalue [maximum value] [cycle/no cycle] [cache value / No cache] [order / No order];</p>	<p>Correct query: 3M Create sequence : 2M Drop sequence :1M (Note: For (ii) Either syntax or example can be considered. Any other example allowed)</p>



	<p>(OR)</p> <p>(Creating sequence for Employee number of emp table.)</p> <p>Create sequence emp_eno_seq start with 1 increment by 1 maxvalue 100 no cycle no cache;</p> <p>Drop sequence:</p> <p>Drop sequence<Sequence Name>;</p> <p>(OR)</p> <p>Drop sequence emp_eno_seq;</p>	
b	Write a PL/SQL program to calculate factorial of a given number.	6M
Ans	<pre>DECLARE num number:=&num; fact number:=1; BEGIN while num!=0 loop fact:=fact*num; num:=num-1 end loop; dbms_output.put_line('Factorial =' fact); END; /</pre> <p>(OR)</p> <pre>DECLARE num number:=&num; fact number:=1; i number; BEGIN for i in 1..num loop fact:=fact*i; end loop;</pre>	<p>Correct Syntax: 3M,Correct logic : 3M</p> <p>(Note: Any other logic can be considered)</p>



		dbms_output.put_line('Factorial=' fact); END; /	
c		Write SQL command for following i) Create user ii) Grant privileges to user. iii) Remove privileges from user.	6M
Ans		i) Create user CREATE USER <username> IDENTIFIED BY <password>; (OR) CREATE USER RAJ IDENTIFIED BY RAJ123; ii) Grant privileges to user. GRANT <privilege list> ON <relation name or view name> TO <user list>; (OR) (assuming table Employee for granting permissions to user 'RAJ' for select, insert, update and delete privilege) GRANT SELECT, INSERT, UPDATE, DELETE ON EMPLOYEE TO RAJ; iii) Remove privileges from user. REVOKE <privilege list> ON <relation name or view name > FROM <user list>; (OR) (assuming table Employee for revoking permissions to user 'RAJ') REVOKE SELECT, INSERT, UPDATE, DELETE ON EMPLOYEE FROM RAJ;	Each correct command: 2M (Note: Either syntax or example can be considered. Any other example allowed)



V2V EDTECH LLP

Online Coaching at an Affordable Price.

OUR SERVICES:

- Diploma in All Branches, All Subjects
- Degree in All Branches, All Subjects
- BSCIT / CS
- Professional Courses



+91 93260 50669



v2vedtech.com



V2V EdTech LLP



v2vedtech



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

Important Instructions to examiners:

- 1) The answers should be examined by key words and not as word-to-word as given in the model answer scheme.
- 2) The model answer and the answer written by candidate may vary but the examiner may try to assess the understanding level of the candidate.
- 3) The language errors such as grammatical, spelling errors should not be given more Importance (Not applicable for subject English and Communication Skills).
- 4) While assessing figures, examiner may give credit for principal components indicated in the figure. The figures drawn by candidate and model answer may vary. The examiner may give credit for any equivalent figure drawn.
- 5) Credits may be given step wise for numerical problems. In some cases, the assumed constant values may vary and there may be some difference in the candidate's answers and model answer.
- 6) In case of some questions credit may be given by judgement on part of examiner of relevant answer based on candidate's understanding.
- 7) For programming language papers, credit may be given to any other program based on equivalent concept.

Q. No	Sub Q.N.	Answer	Marking Scheme
1.	(a) Ans.	Attempt any FIVE of the following: State any two advantages of DBMS over file processing system. Advantages of DBMS over file processing system: <ul style="list-style-type: none">• Reduction in Data redundancy• Data consistency and integrity• Data security• Privacy• Easy access of data• Easy recovery• Flexibility	10 2M <i>Any two advantages 1M each</i>
	(b) Ans.	Draw three level architecture of DBMS.	2M



WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

		<p>The diagram illustrates the three-level architecture of a database system. At the top is the External level, which contains User 1, User 2, User 3, and User n. Below each user is a corresponding View (View 1, View 2, View 3, and View n). Arrows point from each view to its respective user. Below the external level is the Conceptual level, which is connected to all views. Below the conceptual level is the Internal level, which is connected to the conceptual level. At the bottom is the Database, represented by a cylinder icon, which is connected to the internal level.</p>	<p><i>Correct diagram</i> 2M</p>
	<p>(c) Ans.</p>	<p>Define table and field. Table: A table is a collection of related data held in table format. It is a set of data elements using a model of vertical columns and horizontal rows. Field: Each table contains field which is a data structure, used to hold the data. It can also be termed as attribute.</p>	<p>2M <i>Each definition 1M</i></p>
	<p>(d) Ans.</p>	<p>Enlist DML commands.</p> <ul style="list-style-type: none"> • Insert - used to insert new row into table • Delete- used to delete a row from the table • Update – used to modify data in the table. • Select – used to view data from a table. 	<p>2M <i>Any 2 commands 1M each</i></p>
	<p>(e) Ans.</p>	<p>Define primary key and foreign key. Primary key is an attribute or set of attributes used to identify an entity from an entity set. All the values of a primary key should be unique and null values are not allowed. Foreign key is an attribute of an entity which is the primary key of another entity. It is used to show relation between entities. The table containing foreign key is called the child table.</p>	<p>2M <i>Each definition 1M</i></p>
	<p>(f) Ans.</p>	<p>List any four string functions in SQL. Initcap(String) – converts first character of string to upper case Upper(String) – converts the string to upper case Lower(String) – converts string to lower case Length(String) – returns the number of characters in the string Instr(String, sub) – returns the location of the substring Lpad(String,char,number) – returns the string left padded with the character specified to a total of length specified. Rpad(String,char,number) – returns the string right padded with the</p>	<p>2M <i>Any four string functions 1/2M each</i></p>



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

		<p>character specified to a total of length specified. Ltrim(String) -removes white space or other specified characters from the left end of the string Rtrim(String)--removes white space or other specified characters from the right end of the string Replace(String, char,char) – replace all occurrence of a substring by another substring Substring(String,number) – extracts substring from the string Translate(String,char,char) – replace all occurrence of characters by other characters</p>																						
	<p>(g) Ans.</p>	<p>State any two advantages of functions in PL/SQL. Advantages of functions in PL/SQL:</p> <ul style="list-style-type: none"> • Work can be divided into smaller modules so that it can be manageable and also enhances the readability of the code. • It promotes reusability. • It is secure, as the code is in the database and hides the internal database details from the user. • It improves performance against running SQL queries multiple times. 	<p>2M</p> <p><i>Any two advantages 1M each</i></p>																					
2.	<p>(a) Ans.</p>	<p>Attempt any THREE of the following: Distinguish between Network and Hierarchical model. (Any four points)</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">Sr. No.</th> <th style="width: 45%;">Network Model</th> <th style="width: 45%;">Hierarchical model</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">1</td> <td>Represents tree like structure with many roots</td> <td>Represents tree like structure with one root</td> </tr> <tr> <td style="text-align: center;">2</td> <td>Reflects M:N(many to many) relations</td> <td>Reflects 1:N (one-to-many)relations</td> </tr> <tr> <td style="text-align: center;">3</td> <td>Allows a child to have more than one parent</td> <td>There can be only one parent node</td> </tr> <tr> <td style="text-align: center;">4</td> <td>Relationship is represented as pointers or links</td> <td>Relationships between records is of parent-child type</td> </tr> <tr> <td style="text-align: center;">5</td> <td>This model is free from such inconsistency as there is only a single occurrence of a record set.</td> <td>There are multiple occurrence of child records and therefore inconsistency</td> </tr> <tr> <td style="text-align: center;">6</td> <td>Searching a record is easy as there are multiple paths</td> <td>Searching a record is difficult as a child can be</td> </tr> </tbody> </table>	Sr. No.	Network Model	Hierarchical model	1	Represents tree like structure with many roots	Represents tree like structure with one root	2	Reflects M:N(many to many) relations	Reflects 1:N (one-to-many)relations	3	Allows a child to have more than one parent	There can be only one parent node	4	Relationship is represented as pointers or links	Relationships between records is of parent-child type	5	This model is free from such inconsistency as there is only a single occurrence of a record set.	There are multiple occurrence of child records and therefore inconsistency	6	Searching a record is easy as there are multiple paths	Searching a record is difficult as a child can be	<p>12 4M</p> <p><i>Any four points 1M each</i></p>
Sr. No.	Network Model	Hierarchical model																						
1	Represents tree like structure with many roots	Represents tree like structure with one root																						
2	Reflects M:N(many to many) relations	Reflects 1:N (one-to-many)relations																						
3	Allows a child to have more than one parent	There can be only one parent node																						
4	Relationship is represented as pointers or links	Relationships between records is of parent-child type																						
5	This model is free from such inconsistency as there is only a single occurrence of a record set.	There are multiple occurrence of child records and therefore inconsistency																						
6	Searching a record is easy as there are multiple paths	Searching a record is difficult as a child can be																						



WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

		to a data element.	reached only through a parent
(b) Ans.	<p>Explain any four set operators in SQL with example. Set operators combine the results of two component queries into a single result. Queries containing set operators are called as compound queries. Set operators in SQL are represented with following special keywords as: Union, Union all, intersection & minus.</p> <p>1) Union: The Union of two or more sets contains all elements, which are present in either or both. Union works as or. The duplicates of both the tables will appear only once. <i>E.g.</i> select ename from emp1 union select ename from emp2;</p> <p>2) Union all: The Union of 2 or more sets contains all elements, which are present in both, including duplicates. <i>E.g.</i> select ename from emp1 union all select ename from emp2;</p> <p>3) Intersection: The intersection of two sets includes elements which are present in both. <i>E.g.</i> select ename from emp1 intersect select ename from emp2;</p> <p>4) Minus: The minus of two sets includes elements from set1 minus elements of set2. <i>E.g.</i> select ename from emp1 minus select ename from emp2;</p>		<p>4M</p> <p><i>1M each for explanation of operators with example</i></p>
(c) Ans.	<p>Describe Views and write a command to create view. A view is a virtual table based on the result set of the SQL statement. The fields in a view are fields from one or more than one table in the database. SQL functions, where, join statements can be added to a view and the data in it can be presented as if it were from one table. The database engine recreates the data, using the view's SQL statement, every time a user queries a view. A view can be updated using the create or replace view command. For deleting a view, drop query can be used.</p> <p>General syntax to create a view: create view viewname as select query. <i>Eg:</i> create view vw_student as select stud_id, stud_name,ssc_per from student;</p>		<p>4M</p> <p><i>Explanation 3M</i></p> <p><i>General syntax/example 1M</i></p>
(d) Ans.	<p>Explain implicit and explicit cursors. A cursor is a temporary work area created in system memory when an SQL statement is executed. A cursor is a set of rows together with a pointer that identifies a current row. It is a database object to retrieve</p>		<p>4M</p>



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

		<p>data from result set on row at a time.</p> <p>Implicit cursor: these types of cursors are generated and used by the system during the manipulation of a DML query. An implicit cursor is also generated by the system when a single row is selected by a SELECT command.</p> <p>Programmers cannot control the implicit cursors.</p> <p>Explicit cursor: this type of cursor is created by the user when the select command returns more than one row, and only one row is to be processed at a time. An explicit cursor can move from one row to another in a result set. An explicit cursor uses a pointer that holds the record of a row.</p> <p>To create an explicit cursor the following steps are used.</p> <ol style="list-style-type: none">1. Declare cursor: this is done in the declaration section of PL/SQL program.2. Open: this step is done before the cursor is used to fetch the records.3. Fetch: used to retrieve data row by row from the cursor.4. Close: once the processing of the data is done, the cursor can be closed.	<p><i>Each explanation 2M</i></p>
3.	(a) Ans.	<p>Attempt any THREE of the following: State and explain 3NF with example.</p> <p>3NF:</p> <p>An entity is said to be in the third normal form when,</p> <ol style="list-style-type: none">1) It satisfies the criteria to be in the second normal form.2) There exists no transitive functional dependency. (Transitive functional dependency can be explained with the relationship link between three tables. If table A is functionally dependent on B, and B is functionally dependent on C then C is transitively dependent on A). <p>Let us consider the Schema given: (Supplier_no,SupplierName,Supplier_city,Order_no,Order_quantity, Order_amount,Product_code,Product name,rate)</p> <p>Step 1.To convert it into 2NF, We have to decompose the given table into two tables with fully functional dependencies and establishing a referential integrity constraint relationship among the two tables.</p> <p>Table2: Supplier Details (Supplier_no,Supplier_name,Supplier_city)</p> <p>Table 3:Order Details (Order_no,Order_quantity,Order_amount,Supplier_no Product_code,</p>	<p>12 4M</p> <p><i>Explanation 2M</i></p> <p><i>Any example 2M</i></p>



WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

		<p>product_name,rate) Now the above two tables are in 2NF Step 2: To convert the above tables in 3NF, we have to decompose them in three tables satisfying the transitive dependencies property. Table 4: Supplier Details (Supplier_no,Supplier_name,Supplier_city) Table 5: Product Details: (Product_code, product_name,rate) Table 6: Order Details (or Transaction Details) ((Order_no,Supplier_no,Product_code,Order_quantity,Order_amount)) Hence the above three tables are satisfying Transitive dependencies. Thus they are in 3NF.</p>	
	<p>(b) Ans.</p>	<p>Define index. Explain it's types. An Index is a schema object. It is used by the oracle server to improve the speed of retrieval of the rows from a table .Indexes are of two types based on number of columns included in the index. The types of index are: 1) Simple index: An index created on a single column of table is called as simple index Syntax: SQL>Create Index index_name on tablename(attribute); <i>Example:</i>Create index emp_index on emp(empno); 2) Composite Index: An index created on more than one column is called composite index. Syntax: SQL>Create Index index_name on tablename(attribute1,attribute2); <i>Example:</i> Create index emp_index on emp(empno,ename);</p>	<p style="text-align: center;">4M</p> <p style="text-align: center;"><i>Definitio n 1M</i></p> <p style="text-align: center;"><i>Each type 1½M</i></p>
	<p>(c) Ans.</p>	<p>Explain Exception handling with it's types. An exception is an error condition during a program execution. PL/SQL supports programmers to catch such conditions using EXCEPTION block in the program and an appropriate action is taken against the error condition. There are two types of exceptions – 1) System-defined exceptions/Predefined exceptions/Built-in exceptions</p>	<p style="text-align: center;">4M</p> <p style="text-align: center;"><i>Explana tion 2M</i></p>



WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

	<p>2) User-defined exception</p> <p>Predefined exceptions- PL/SQL provides predefined Exception, which are executed when any database rule is violated by a program. Example: NO_DATA_FOUND, ZERO_DIVIDE.</p> <p>Syntax for Predefined Exception Handling:</p> <p>The general syntax for exception handling is as follows.</p> <pre>DECLARE <declarations section> BEGIN <executable command(s)> EXCEPTION <exception handling goes here > WHEN exception1 THEN exception1-handling-statements WHEN exception2 THEN exception2-handling-statements WHEN exception3 THEN exception3-handling-statements WHEN others THEN exception3-handling-statements END;</pre> <p>User defined Exceptions:</p> <p>PL/SQL allow us to define our own exception according to the need of our program. A user defined exception must be declared and then raised explicitly.</p> <p>Syntax for User defined Exception:<pre>DECLARE exception_name EXCEPTION; BEGIN IF condition THEN RAISE exception_name; END IF; EXCEPTION WHEN exception_name THEN statement; END;</pre></p>	<p><i>Types</i> <i>2M</i></p>
--	---	-----------------------------------



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

	<p>(d) Ans.</p>	<p>Explain ACID properties of traction. A transaction can be defined as a group of tasks. A single task is the minimum processing unit which cannot be divided further.</p> <p>ACID Properties A transaction is a very small unit of a program and it may contain several lowlevel tasks. A transaction in a database system must maintain Atomicity, Consistency, Isolation, and Durability – commonly known as ACID properties – in order to ensure accuracy, completeness, and data integrity.</p> <ul style="list-style-type: none">• Atomicity: This property states that a transaction must be treated as an atomic unit, that is, either all of its operations are executed or none. There must be no state in a database where a transaction is left partially completed. States should be defined either before the execution of the transaction or after the execution/abortion/failure of the transaction.• Consistency: The database must remain in a consistent state after any transaction. No transaction should have any adverse effect on the data residing in the database. If the database was in a consistent state before the execution of a transaction, it must remain consistent after the execution of the transaction as well.• Isolation: In a database system where more than one transaction are being executed simultaneously and in parallel, the property of isolation states that all the transactions will be carried out and executed as if it is the only transaction in the system. No transaction will affect the existence of any other transaction.• Durability: The database should be durable enough to hold all its latest updates even if the system fails or restarts. If a transaction updates a chunk of data in a database and commits, then the database will hold the modified data. If a transaction commits but the system fails before the data could be written on to the disk, then that data will be updated once the system springs back into action.	<p>4M</p> <p><i>Explanation of each property 1M</i></p>
4.	<p>(a) Ans.</p>	<p>Attempt any THREE of the following: Explain strong and weak entity set. Strong entity set: An entity set that has sufficient attributes to form a primary key is</p>	<p>12 4M</p>



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

	<p>called as Strong entity set. <i>Example:</i> Employee is a Strong entity with attributes as empid, name, address, salary, birthdate among which empid can be considered as primary key.</p> <p>Weak entity set: The entity set which does not have sufficient attributes to form a primary key is called as Weak entity set. A weak entity is an entity that cannot be uniquely identified by its attributes alone; therefore, it must use a foreign key in conjunction with its attributes to create a primary key. The foreign key is typically a primary key of an entity it is related to. <i>Example:</i> Employee has "dependents" with name, birthdate, and relationship to employee and it can be related to employee with the help of empid, so "dependents" is a weak entity which depends on strong entity "Employee".</p>	<p><i>Each entity set 2M</i></p>
<p>(b) Ans.</p>	<p>Describe create & alter command with syntax & example.</p> <p>1) The SQL CREATE TABLE statement is used to create a new table.</p> <p>Syntax</p> <p>The basic syntax of the CREATE TABLE statement is as follows –</p> <p><i>CREATE TABLE table_name</i></p> <p>(</p> <p> <i>column1 datatype (size),</i></p> <p> <i>column2 datatype(size),</i></p> <p> <i>column3 datatype(size),</i></p> <p> </p> <p>);</p> <p>Example:</p> <p>CREATE TABLE Persons</p> <p>(</p> <p> PersonIDnumber(10),</p> <p> LastNamevarchar2(20),</p> <p> FirstNamevarchar2(20),</p> <p> Address varchar2(20),</p> <p> City varchar2(20)</p>	<p>4M</p> <p><i>Each command 2M</i></p>



WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

	<p>);</p> <p>2)The ALTER TABLE statement is used to add, delete, or modify columns in an existing table.</p> <p>The ALTER TABLE statement is also used to add and drop various constraints on an existing table.</p> <p>i) To add Columns in a table</p> <p><i>Syntax:</i>ALTER TABLE <i>table_name</i> ADD <i>column_name datatype</i>;</p> <p><i>Example</i> ALTER TABLE Customers ADD Email varchar2(20);</p> <p>ii) To delete a column in a table</p> <p>ALTER TABLE <i>table_name</i> DROP COLUMN <i>column_name</i>;</p> <p><i>Example</i> ALTER TABLE Customers DROP COLUMN Email;</p> <p>iii) To modify a column in a table</p> <p><i>Syntax:</i> ALTER TABLE <i>table_name</i> MODIFY COLUMN <i>column_name datatype</i>;</p> <p><i>Example</i> ALTER TABLE Customers MODIFY COLUMN <i>customeridnumeric(10)</i>;</p> <p>iv) To add Constraints in A table</p> <p><i>Syntax:</i> ALTER TABLE <i>table_name</i> ADD constraint <i>constraintname (column_name)</i>;</p> <p><i>Example:</i> ALTER TABLE Customers ADD constraint primary key(CustomerID);</p>	
--	---	--



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

	<p>(c) Ans.</p>	<p>Define database trigger. How to create and delete trigger? Triggers are stored programs, which are automatically executed or fired when some events occur. Triggers are, in fact, written to be executed in response to any of the following events –</p> <ul style="list-style-type: none">• A database manipulation (DML) statement (DELETE, INSERT, or UPDATE)• A database definition (DDL) statement (CREATE, ALTER, or DROP). <p>Triggers can be defined on the table, view, schema, or database with which the event is associated.</p> <p>Creating Triggers</p> <p>The syntax for creating a trigger is – CREATE [OR REPLACE] TRIGGER trigger_name { BEFORE AFTER INSTEAD OF } { INSERT [OR] UPDATE [OR] DELETE } [OF col_name] ON table_name [REFERENCING OLD AS o NEW AS n] [FOR EACH ROW] WHEN (condition) DECLARE Declaration-statements BEGIN Executable-statements EXCEPTION Exception-handling-statements END;</p> <p>To delete a trigger: <i>Syntax:</i> DROP TRIGGER trigger_name.</p>	<p>4M</p> <p><i>Definition 1M</i></p> <p><i>Create 2M</i></p> <p><i>Delete 1M</i></p>
	<p>(d) Ans.</p>	<p>Explain any one control structure in PL/SQL with example. PL/SQL has three categories of control statements: conditional selection statements, loop statements and sequential control statements. PL/SQL categories of control statements are:</p> <ul style="list-style-type: none">• Conditional selection statements, which run different statements for different data values.	<p>4M</p>



WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

		<p>The conditional selection statements are IF and CASE.</p> <ul style="list-style-type: none">• Loop statements, which run the same statements with a series of different data values. <p>The loop statements are the basic LOOP, FOR LOOP, and WHILE LOOP.</p> <p>The EXIT statement transfers control to the end of a loop. The CONTINUE statement exits the current iteration of a loop and transfers control to the next iteration. Both EXIT and CONTINUE have an optional WHEN clause, where you can specify a condition.</p> <p>Sequential control statements, which are not crucial to PL/SQL programming.</p> <p>The sequential control statements are GOTO, which goes to a specified statement, and NULL, which does nothing.</p> <p>1)Conditional Control: IF and CASE Statements: The IF statement lets us execute a sequence of statements conditionally. That is, whether the sequence is executed or not depends on the value of a condition. There are three forms of IF statements: IF-THEN, IF-THEN-ELSE, and IF-THEN-ELSIF. The CASE statement is a compact way to evaluate a single condition and choose between many alternative actions.</p> <p>IF-THEN Statement:The simplest form of IF statement associates a condition with a sequence of statements enclosed by the keywords THEN and END IF (not ENDIF), as follows: IF condition THEN sequence_of_statements END IF;</p> <p>IF-THEN-ELSE Statement: The second form of IF statement adds the keyword ELSE followed by an alternative sequence of statements, as follows: IF condition THEN sequence_of_statements1</p>	<p><i>Explanation of Any one control structure 4M</i></p>
--	--	--	---



WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

	<pre>ELSE sequence_of_statements2 END IF;</pre> <p>IF-THEN-ELSIF Statement: The third form of IF statement uses the keyword ELSIF (not ELSEIF) to introduce additional conditions, as follows:</p> <pre>IF condition1 THEN sequence_of_statements1 ELSIF condition2 THEN sequence_of_statements2 ELSE sequence_of_statements3 END IF;</pre> <p>CASE Statement: Like the IF statement, the CASE statement selects one sequence of statements to execute.</p> <pre>IF grade = 'A' THEN dbms_output.put_line('Excellent'); ELSIF grade = 'B' THEN dbms_output.put_line('Very Good'); ELSIF grade = 'C' THEN dbms_output.put_line('Good'); ELSIF grade = 'D' THEN dbms_output.put_line('Fair'); ELSIF grade = 'F' THEN dbms_output.put_line('Poor'); ELSE dbms_output.put_line('No such grade'); END IF;</pre> <p>END CASE;</p> <p>2) Iterative Control: LOOP and EXIT Statements: LOOP statements let us execute a sequence of statements multiple times. There are three forms of LOOP statements: LOOP, WHILE-LOOP, and FOR-LOOP.</p> <p>LOOP: The simplest form of LOOP statement is the basic (or</p>	
--	--	--



WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

	<p>infinite) loop, which encloses a sequence of statements between the keywords LOOP and END LOOP, as follows:</p> <pre>LOOP sequence_of_statements END LOOP;</pre> <p>WHILE-LOOP:The WHILE-LOOP statement associates a condition with a sequence of statements enclosed by the keywords LOOP and END LOOP, as follows:</p> <pre>WHILE condition LOOP sequence_of_statements END LOOP;</pre> <p>Before each iteration of the loop, the condition is evaluated. If the condition is true, the sequence of statements is executed, then control resumes at the top of the loop. If the condition is false or null, the loop is bypassed and control passes to the next statement.</p> <p>FOR-LOOP:Whereas the number of iterations through a WHILE loop is unknown until the loop completes, the number of iterations through a FOR loop is known before the loop is entered. FOR loops iterate over a specified range of integers. The range is part of an <i>iteration scheme</i>, which is enclosed by the keywords FOR and LOOP. A double dot (..) serves as the range operator. The syntax follows:</p> <pre>FOR counter IN [REVERSE] lower_bound..higher_bound LOOP sequence_of_statements END LOOP</pre> <p>3)Sequential Control: GOTO and NULL Statements: Unlike the IF and LOOP statements, the GOTO and NULL statements are not crucial to PL/SQL programming.</p> <p>GOTO Statement</p> <p>The GOTO statement branches to a label unconditionally.</p> <pre>BEGIN ... GOTO insert_row;</pre>	
--	---	--



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

		... <<insert_row>> INSERT INTO empVALUES ... END;	
(e) Ans.	<p>Describe database backups with it's types.</p> <p>Regular backups are required to protect database and ensure <i>its</i> restoration in case of failure. Various <i>backup types</i> provide different protection to our database. Backing up and restoring data is one of the most important responsibilities of IT professionals</p> <p>Three common types of database backups can be run on a desired system: normal (full), incremental and differential.</p> <p>i) Normal or Full Backups:</p> <p>When a normal or full backup runs on a selected drive, all the files on that drive are backed up. This, of course, includes system files, application files, user data — everything. Those files are then copied to the selected destination (backup tapes, a secondary drive or the cloud), and all the archive bits are then cleared.</p> <p>Normal backups are the fastest source to restore lost data because all the data on a drive is saved in one location.</p> <p>ii) Incremental Backups:</p> <p>A common way to deal with the long running times required for full backups is to run them only on weekends. Many businesses then run incremental backups throughout the week since they take far less time. An incremental backup will grab only the files that have been updated since the last normal backup. Once the incremental backup has run, that file will not be backed up again unless it changes or during the next full backup.</p> <p>iii) Differential Backups:</p> <p>An alternative to incremental database backups that has a less complicated restore process is a differential backup. Differential backups and recovery are similar to incremental in that these backups grab only files that have been updated since the last normal backup. However, differential backups do not clear the archive bit. So a file</p>	4M <i>Description</i> 2M <i>Types</i> 2M	



WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

		<p>that is updated after a normal backup will be archived every time a differential backup is run until the next normal backup runs and clears the archive bit.</p>	
5.	<p>(a)</p> <p>Ans.</p>	<p>Attempt any TWO of the following: Draw an ER diagram for library management system. (Use Books, Publisher & Member entities). <i>(Note: Consider any relevant diagram)</i></p> <div style="text-align: center;"> </div>	<p>12 6M</p> <p><i>Correct entities</i> 2M</p> <p><i>Correct symbols</i> 2M</p> <p><i>Correct relations</i> hips 2M</p>
(b)	Ans.	<p>Write a command to crate table student (rollno, Stud_name, branch, class, DOB, City, Contact_no) and write down queries for following:</p> <p>(i) Insert one row into the table (ii) Save the data (iii) Insert second row into the table (iv) Undo the insertion of second row (v) Create save point S₁. (vi) Insert one row into the table.</p>	6M



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

	<p>SQL>Create table student(Rollno number(5), Stud_name char(10), branch varchar(10), class varchar(10), DOB date, city varchar(15), Contact_no number(12));</p> <p>(i) Insert one row into the table: SQL>Insert into student values(1,'Ram','CO','FirstYear','12-jun-2001','Pune',98576867)</p> <p>(ii) Save the data: SQL> commit; (OR) SQL> commit work;</p> <p>(iii)Insert second row into the table: SQL>Insert into student values(2,'Raj','CO','FirstYear','22-Sep-2002','Mumbai',98896863)</p> <p>(iv)Undo the insertion of second row: SQL> rollback; (OR) SQL> rollback work;</p> <p>(v)Create savepoint s1: SQL>Savepoint s1;</p> <p>(vi) insert one row into the table: SQL>Insert into student values(3,'Beena','CO','FirstYear','30-Dec-2002','Mumbai',97846455)</p>	<p><i>Create table 3M</i></p> <p><i>Each correct Query ½M each</i></p>
(c)	<p>Consider following schema: EMP (empno, deptno, ename, salary, designation, join_date, DOB, dept_location). Write down SQL queries for following: (i) Display employees name & number in decreasing order of salary.</p>	6M



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

	Ans. (ii) Display employee name & employee number whose designation is Manager. (iii) Display age of employees with ename. (iv) Display total salary of all employees. (v) Display employee names having deptno as 20 and dept_location is Mumbai (vi) Display name of employee who earned lowest salary. (i) Display employees name & number in descending order of salary: SQL> select ename,empno from EMP order by salary desc; (ii) Display employee name & employee number whose designation is Manager. SQL> select ename,empno from EMP where designation='Manager'; (iii) Display age of employees with ename SQL>select round ((sysdate - DOB) /365, 0) as "age",ename from EMP; OR select months_between(TRUNC(sysdate),DOB)/12 as "age",ename from EMP ; (**Note consider any other logic also) (iv) Display total salary of all employees. SQL> select sum(salary) from EMP; (v) Display employee names having deptno as 20 and dept_location is Mumbai. SQL> select enamefrom EMP where deptno=20 and dept_location='Mumbai'; (vi) Display name of employee who earned lowest salary SQL> select ename from EMP where salary=(select min(salary) from EMP);	<i>Each correct Query 1M</i>
6.	(a) Attempt any TWO of the following: Consider the structure for book table as Book-Master (bookid, bookname, author, no_of copies, price) Write down SQL queries for following:	12 6M



WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

	<p>Ans.</p>	<p>(i) Write a command to create Book_master table. (ii) Get authorwise list of all books. (iii) Display all books whose price is between ₹ 500 & ₹ 800. (iv) Display all books with details whose name start with 'D'. (v) Display all books whose price is above ₹ 700. (vi) Display all books whose number of copies are less than 10.</p> <p>(i)Write a command to create Book_Master table table. SQL>Create table Book-Master(bookid number(5), bookname char(10), authorvarchar(20), no_of_copiesnumber(10), price number(10,2));</p> <p>(ii)Get authorwise list of all books. SQL>Select sum(no_of copies) from Book_Master group by author;</p> <p>(iii)Display all books whose price is between Rs.500 & Rs. 800 SQL> Select * from Book_Master where price between 500 and 800;</p> <p style="text-align: center;">OR</p> <p>SQL> Select * from Book_Master where price >=500 and price<=800;</p> <p>(iv) Display all books with details whose name start with 'D' SQL> Select bookname from Book_Master where bookname like 'D%';</p> <p>(v)Display all books whose price is above Rs. 700 SQL>Select * from Book_Master where price >700;</p> <p>(vi) Display all books whose number of copies are less than 10 SQL>Select * from Book_Master where no_of_copies<10;</p>	<p>Each correct Query 1M</p>
	<p>(b)</p>	<p>Write a PL/SQL program to print n even numbers using For Loop. (Note: Any other logic can be allowed)</p>	<p>6M</p>



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

	Ans.	<pre> declare num number; n number:=&n; begin for num in 1..n loop if(mod(num,2)=0) then dbms_output.put_line('Even no are :' num); end if; end loop; end;</pre>	<p><i>Correct logic 3M</i></p> <p><i>Correct syntax 3M</i></p>
	(c) Ans.	<p>Describe database privileges. Write down the procedure for granting & revoking privileges in database objects to the users.</p> <p>Database privileges: When multiple users can access database objects, authorization can be controlled to these objects with privileges. Every object has an owner. Privileges control if a user can modify an object owned by another user. Privileges are granted or revoked either by the instance administrator, a user with the ADMIN privilege or, for privileges to a certain object, by the owner of the object.</p> <p>1) System Privileges: System privileges are privileges given to users to allow them to perform certain functions that deal with managing the database and the server e.g Create user, Create table, Drop table etc.</p> <p>2) Object Privileges: Object privileges are privileges given to users as rights and restrictions to change contents of database object – where database objects are things like tables, stored procedures, indexes, etc. Ex. Select, insert, delete, update, execute, references etc</p> <p><u>Procedure for granting privileges</u> Grant: This command is used to give permission to user to do operations on the other user's object. Syntax: Grant<object privileges>on<object name>to<username>[with grant option] ; Example: Grant select, update on emp to user1;</p> <p><u>Procedure for revoking privileges</u> Revoke: This command is used to withdraw the privileges that has</p>	<p>6M</p> <p><i>Databases Privileges 2M</i></p> <p><i>Procedure for granting privileges 2M</i></p>



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION
(Autonomous)
(ISO/IEC - 27001 - 2005 Certified)

WINTER – 2019 EXAMINATION
MODEL ANSWER

Subject: Database Management System

Subject Code: 22319

	<p>been granted to a user. Syntax: Revoke <object privileges>on<object name>from <username> ; Example: Revoke select, update on emp from user1;</p>	<p><i>Procedures for revoking privileges 2M</i></p>
--	---	---



V2V EDTECH LLP

Online Coaching at an Affordable Price.

OUR SERVICES:

- Diploma in All Branches, All Subjects
- Degree in All Branches, All Subjects
- BSCIT / CS
- Professional Courses



+91 93260 50669



v2vedtech.com



V2V EdTech LLP



v2vedtech



WINTER – 2022 EXAMINATION

Subject Name: Database Management System

Model Answer

Subject Code:

22319

Important Instructions to examiners:

- 1) The answers should be examined by key words and not as word-to-word as given in the model answer scheme.
- 2) The model answer and the answer written by candidate may vary but the examiner may try to assess the understanding level of the candidate.
- 3) The language errors such as grammatical, spelling errors should not be given more Importance (Not applicable for subject English and Communication Skills).
- 4) While assessing figures, examiner may give credit for principal components indicated in the figure. The figures drawn by candidate and model answer may vary. The examiner may give credit for any equivalent figure drawn.
- 5) Credits may be given step wise for numerical problems. In some cases, the assumed constant values may vary and there may be some difference in the candidate's answers and model answer.
- 6) In case of some questions credit may be given by judgement on part of examiner of relevant answer based on candidate's understanding.
- 7) For programming language papers, credit may be given to any other program based on equivalent concept.
- 8) As per the policy decision of Maharashtra State Government, teaching in English/Marathi and Bilingual (English + Marathi) medium is introduced at first year of AICTE diploma Programme from academic year 2021-2022. Hence if the students in first year (first and second semesters) write answers in Marathi or bilingual language (English +Marathi), the Examiner shall consider the same and assess the answer based on matching of concepts with model answer.

Q. No.	Sub Q. N.	Answer	Marking Scheme
1		Attempt any FIVE of the following:	10 M
	a)	Define i) Data Abstraction ii) Instance.	2 M
	Ans	i) Data Abstraction: Hiding complexity of data structures from end user through different levels is known as data abstraction. Many end users are not computer trained so it is needed to hide complex data structures from them. It has 3 levels : a. Physical level b. logical level c. view level ii) Instance: The data stored in database at a particular moment of time is called instance of database. Example: Let's say a table teacher in our database whose name is School, suppose the table has 50 records so the instance of the database has 50 records for now and tomorrow we are going to add another fifty records so tomorrow the instance have total 100 records. This is called an instance.	Each correct definition: 1M



	b)	State any two advantages of DBMS.	2 M
	Ans	<ul style="list-style-type: none">• Reduction in Data redundancy• Data consistency and integrity• Data security• Privacy• Easy access of data• Easy recovery• Flexibility	Any two correct advantages each 1M
	c)	Define Normalization. Enlist its types.	2 M
	Ans	<p>Normalization is a process of organizing the data in database to avoid data redundancy, insertion anomaly, update anomaly & deletion anomaly.</p> <p>Types of normalization are:</p> <ul style="list-style-type: none">• First normal form(1NF)• Second normal form(2NF)• Third normal form(3NF)• Boyce & Codd normal form (BCNF)• Fourth normal form(4NF)	1 M for correct definition, 1 M for correct types
	d)	Write syntax for creating and Renaming a table.	2 M
	Ans	<p><u>Syntax of the CREATE TABLE statement is as follows:</u></p> <pre>CREATE TABLE table_name (column1 datatype (size), column2 datatype(size), column3 datatype(size),);</pre> <p><u>Syntax of RENAME TABLE statement is as follows:</u></p> <pre>RENAME old_table _name To new_table_name ;</pre>	Correct Syntax of Create Table: 1M, Correct Syntax of Rename Table: 1M
	e)	Enlist arithmetic and logical SQL operators.	2 M
	Ans	<p><u>SQL Arithmetic Operators:</u></p> <p>Addition Operator (+)</p>	SQL arithmetic operators: 1M, SQL logical



		<p>Subtraction Operator (-)</p> <p>Multiplication Operator (+)</p> <p>Division Operator (-)</p> <p>Modulus Operator (+)</p> <p><u>SQL Logical Operators:</u></p> <p>ALL operator</p> <p>AND operator</p> <p>OR operator</p> <p>BETWEEN operator</p> <p>IN operator</p> <p>NOT operator</p> <p>ANY operator</p> <p>LIKE operator</p>	operators: 1M
	f)	Write syntax for creating and dropping views.	2 M
	Ans	<p>CREATE VIEW Syntax:</p> <p>create view view_name As</p> <p>select column1, column2...</p> <p>from table_name</p> <p>where condition ;</p> <p>DROP VIEW Syntax:</p> <p>DROP VIEW view_name;</p>	<p>Correct syntax for create view: 1M,</p> <p>Correct syntax for drop view: 1M</p>
	g)	State two advantages of PL/SQL	2 M
	Ans	<p>Advantages of PL/SQL are:</p> <ul style="list-style-type: none">• Work can be divided into smaller modules so that it can be manageable and also enhances the readability of the code.• It promotes reusability.• It is secure, as the code is in the database and hides the internal database details from the user.• It improves performance against running SQL queries multiple times.	<p>Any two correct advantages each 1M</p>

2.	Attempt any <u>THREE</u> of the following:	12 M
	a) Explain three level architecture of Database system.	4 M
Ans	<div style="text-align: center;"> <p>External level</p> <p style="text-align: center;">Fig: Three Level Architecture of DBMS</p> </div> <p>This architecture has three levels:</p> <ol style="list-style-type: none"> 1. External level 2. Conceptual level 3. Internal level <p>1. External level</p> <p>It is also called view level because several users can view their desired data from this level which is internally fetched from database with the help of conceptual and internal level mapping.</p> <p>The user doesn't need to know the database schema details such as data structure; table definition etc. user is only concerned about data which is what returned back to the view level after it has been fetched from database which is present at the internal level. External level is the top level of the three level DBMS architecture.</p> <p>2. Conceptual level</p> <p>It is also called logical level. The whole design of the database such as relationship among data, schema of data etc. are described in this level.</p> <p>Database constraints and security are also implemented in this level of architecture.</p>	<p>Explanation=3M</p> <p>Diagram=1M</p>



	<p>This level is maintained by DBA (database administrator).</p> <p>3. Internal level</p> <p>This level is also known as physical level. This level describes how the data is stored in the storage devices.</p> <p>This level is also responsible for allocating space to the data. This is the lowest level of the architecture.</p>	
b)	<p>Write SQL queries for following:</p> <p>i) Create table student with following attributes using suitable data types. Roll no., as primary key, name, marks as not null and city.</p> <p>ii) Add column Date of Birth in above student table.</p> <p>iii) Increase the size of attribute name by 10 in above student table.</p> <p>iv) Change name of Student table to stud.</p>	4 M
Ans	<p>i) CREATE TABLE Student (Rollno int PRIMARY KEY, name varchar(30) NOT NULL, marks int NOT NULL, city varchar(20));</p> <p>ii) ALTER TABLE student ADD DateofBirth varchar(20);</p> <p>iii) ALTER TABLE student Modify name varchar(40);</p> <p>iv) RENAME Student to Stud;</p>	Correct Query 1M Each
c)	<p>Write and Explain the syntax for creating and dropping indexes with an example.</p>	4 M
Ans	<p>CREATE INDEX</p> <p>The CREATE INDEX command is used to create indexes in tables. It allows duplicate values. Indexes are used to retrieve data from the database very fast. The users cannot see the indexes; they are just used to speed up searches/queries.</p> <p>Syntax:</p> <p>CREATE INDEX index_name</p>	Correct explanation of CREATE INDEX Syntax with example: 2M,



	<p>ON table_name (column1, column2, ...);</p> <p>Example:</p> <p>The following SQL creates an index named id_firstname on the FirstName column in the Student table:</p> <pre>CREATE INDEX id_firstname ON Student (FirstName);</pre> <p>DROP INDEX</p> <p>The DROP INDEX statement is used to delete an index in a table.</p> <p>Syntax:</p> <pre>DROP INDEX index_name ON table_name;</pre> <p>Example:</p> <pre>DROP INDEX id_firstname ON Student;</pre>	<p>Correct explanation of DROP INDEX Syntax with example: 2M</p>
	d) Write a PL/SQL code to print reverse of a number.	4 M
Ans	<p>PL/SQL code to print reverse of a number:</p> <pre>declare n number; i number; rev number:=0; r number; begin n:=&n; while n>0 loop r:=mod(n,10); rev:=(rev*10)+r; n:=trunc(n/10); end loop; dbms_output.put_line('reverse is ' rev); end;</pre>	<p>Correct Logic: 2M, Correct Code: 2M,</p> <p>Any other correct logic shall be considered</p>
3.	Attempt any <u>THREE</u> of the following:	12 M
	a) Write down any four Dr. E.F Codd's rules.	4 M
Ans	<p>Dr. E. F. Codd's Rules:</p> <p>Rule 1: The information Rule: all data viewed to users.</p>	Any 4: 1 M each



	<p>Rule 2: Guaranteed Access Rule: all user get access to database Rule 3: Systematic treatment of null values: null value should be empty Rule 4: Dynamic online Catalog: record all transactions in database Rule 5: Data Sub language rule: use only one language Rule 6: View updating rule: table and view updated simultaneously Rule 7: High level insert, delete and update: multiple insert delete update Rule 8: Physical data independence: hardware change Rule 9: Logical data independence: structure change Rule 10: Integrity independence: store correct data Rule 11: Distribution independence: distributed database Rule 12: No subversion rule: no version of language used.</p>	
b)	State the use of group by and order by clauses.	4 M
Ans	<p>Group by Clause: Group by clause is used to collect the data as multiple records and group them to produce the result. Syntax: SELECT column_name, function(column_name) FROM table_name WHERE condition GROUP BY column_name;</p> <p>Ex: select avg(sal) from emp_details group by deptno;</p> <p>Order by Clause: To view the data in sorted order, the order by clause is used. By default, the data is sorted in ascending order. Syntax: SELECT expressions FROM tables [WHERE conditions] ORDER BY expression [ASC DESC];</p> <p>Ex: select deptno from emp_details order by deptno; Ex: select deptno from emp_details order by deptno desc; (for descending order)</p>	<p>2 M=group by</p> <p>2 M=order by</p>
c)	Explain the steps of cursor implementation with syntax and example.	4 M
Ans	<p>Cursor Implementation: Steps to create Cursor:</p> <p>a. Declaring cursor: Cursor is declared in the declaration section. Syntax: cursor <cursor_name>is<select query>; Ex: cursor a is select ename from emp_details where empno=3;</p> <p>b. Opening cursor: After declaring the cursor, the cursor needs to open. Syntax: open <cursor_name>;</p>	<p>2 M=cursor implementation</p> <p>1M=syntax</p> <p>1M=example</p>



	<p>Ex: open a;</p> <p>c. Fetching a record from cursor: Once the cursor is declared and opened, we need to get records or rows from the cursor. These records are accessed using the FETCH statement. Syntax: fetch <cursor_name> into <variable_list>; Ex: fetch a into name;</p> <p>d. Closing cursor: Once the cursor is opened and processing is over, we need to close it. Syntax: close <cursor_name>; Ex: close a;</p> <p>Example:</p> <pre>declare name emp_details.ename%type; cursor a is select ename from emp_details where empno=3;<i>//cursor declaration</i> begin open a;<i>//opening the cursor</i> loop fetch a into name;<i>//fetching the rows from cursor</i> update emp_details set comm=3000 where empno=3; exit when a%notfound; dbms_output.put_line('record updated'); end loop; close a;<i>//closing the cursor</i> end;</pre>	2 M
d)	Explain ACID properties of transaction.	4 M
Ans	<p>ACID Properties of Transaction:</p> <ol style="list-style-type: none">1. Atomicity2. Consistency3. Isolation4. Durability <p>1. Atomicity: Atomicity means all the operations included in the single transaction gets executed at a time or none.</p> <p>2. Consistency: Consistency means update or edits the same data stored at different locations.</p> <p>3. Isolation: Isolation means all the transactions gets executed independent of each other.</p> <p>4. Durability: Durability means data can be saved in database permanently until user change it.</p>	1M for each properties



4.		Attempt any <u>THREE</u> of the following:	12 M
	a)	Describe any four responsibilities of Database Administrator.	4 M
	Ans	<p>Responsibilities of Database Administrator (DBA):</p> <ol style="list-style-type: none"> 1. Schema Definition: Database or schema can be designed or defined by DBA. 2. Creating storage structure: DBA allocate or decide the space to store the database. 3. Create grant access methods: Different access methods to access the database can be granted by DBA to the users. 4. Schema modification: The database or schema which is already defined can be modified by DBA as per the requirements. 5. Granting authorization: To access the different databases, DBA can grant the authorization to authorized users only. 6. Performance tuning: The problems/errors arise in database accessing; can be resolved by DBA to increase the performance. 7. Regular maintenance: DBA can monitor the transactions in database and maintain the database error free by doing the regular maintenance. 	Any 4: 1 M each
	b)	Explain Primary and Unique key constraint with syntax.	4 M
	Ans	<p><u>Primary key constraint:</u> Primary key constraint applied to any column can't accept the duplicate and null values. This constraint can be applied at the time of table creation.</p> <p><u>Syntax for Primary Key:</u> CREATE TABLE <Table_Name> (Column1 datatype, Column2 datatype, CONSTRAINT <Name> PRIMARY KEY (Column name));</p> <p><u>Example:</u> CREATE TABLE CUSTOMERS (ID INT NOT NULL, AGE INT NOT NULL, ADDRESS CHAR (25), PRIMARY KEY (ID));</p> <p style="text-align: center;">OR</p>	<p>1 M=primary key use</p> <p>1M=syntax for primary key</p> <p>1 M=unique key use</p> <p>1M=syntax for unique key</p>



	<p>CREATE TABLE CUSTOMERS (ID INT NOT NULL, NAME VARCHAR (20) NOT NULL, AGE INT NOT NULL, ADDRESS CHAR (25) , SALARY DECIMAL (18, 2), PRIMARY KEY (ID, NAME));</p> <p><u>Unique key constraint:</u> Unique key constraint applied to any column can't accept the duplicate values (only accepts unique values). This constraint can be applied at the time of table creation. <u>Syntax for Unique key:</u></p> <p>CREATE TABLE <Table_Name> (Column1 datatype, Column2 datatype,CONSTRAINT <Name> UNIQUE KEY (Column name));</p> <p><u>Example:</u></p> <p>CREATE TABLE students (S_Id int NOT NULL, LastName varchar (255) NOT NULL, FirstName varchar (255), City varchar (255), UNIQUE (S_Id))</p> <p>OR</p> <p>CREATE TABLE students (S_Id int NOT NULL, LastName varchar (255) NOT NULL, FirstName varchar (255), City varchar (255), CONSTRAINT uc_studentId UNIQUE (S_Id, LastName))</p>	
c)	Write and Explain the syntax for creating database trigger.	4 M
Ans	Database trigger: Triggers can be referred as stored procedures that are fired or executed when an	Explanation of syntax:2M Example of



	<p>INSERT, UPDATE or DELETE statement is given against the associated table.</p> <p><u>Syntax:</u></p> <pre>create trigger [trigger_name] [before after] {insert update delete} on [table_name] [for each row] [trigger_body]</pre> <p><u>Explanation of syntax:</u></p> <ol style="list-style-type: none">1. create trigger [trigger_name]: Creates or replaces an existing trigger with the trigger_name.2. [before after]: This specifies when the trigger will be executed.3. {insert update delete}: This specifies the DML operation.4. on [table_name]: This specifies the name of the table associated with the trigger.5. [for each row]: This specifies a row-level trigger, i.e., the trigger will be executed for each row being affected.6. [trigger_body]: This provides the operation to be performed as trigger is fired <p><u>Example:</u> Given Student Report Database, in which student marks assessment is recorded. In such schema, create a trigger so that the total and percentage of specified marks is automatically inserted whenever a record is insert.</p> <p>Here, as trigger will invoke before record is inserted so, BEFORE Tag can be used.</p> <pre>create trigger stud_marks before INSERT on Student for each row set Student.total = Student.subj1 + Student.subj2 + Student.subj3, Student.per = Student.total * 60 / 100;</pre>	Trigger=2M
d)	Write a trigger which invokes on deletion of record on emp table.	4 M
Ans	Delete Trigger: create or replace trigger trg1 before delete on emp_details declare begin	4 M Or any other correct logic for



		raise_application_error(-20000,'cannot delete the record'); end;	delete trigger
e)		Explain Database Recovery techniques in detail.	4 M
Ans		Database Recovery Techniques: Database recovery techniques are used to restore the original data in system from backup. Backward and forward recovery is two types of database recovery. Recovery Techniques: 1. Log based recovery. 2. Shadow paging recovery 3. Checkpoints 1. Log based recovery: It records sequence of log records, which includes all activities done by database users. It records the activities when user changes the database. In case of database failure, by referring the log records users can easily recover the data. 2. Shadow paging recovery: This technique is the alternative for log based recovery. In this technique, database is divided into pages that can be stored on the disk. The page table is used to maintain the record of location of pages. In case of database failure, page table is used to recover the parts of database. 3. Checkpoints: Checkpoint records all committed transactions into logs. When system fails, it check log to determine recovery action.	Any 2 techniques: 2 M each
5.		Attempt any <u>TWO</u> of the following:	12 M
a)		Draw the overall architecture of DBMS. Explain storage manager and query processor components.	6 M
Ans:		Storage manager components: 1. Buffer Manager The Buffer Manager allocates the space to the buffer to store data in it. 2. File Manager The task of the file manager is to manage the space allocation in disk for storing the information and also the data structures used for representing that information.	2M=Correct Diagram with naming 2M= storage manager components

3. Authorization and Integrity Manager

Allows only authorized users to access data and should be hidden from the public users. The Authorization and Integrity Manager verifies the authority of the user trying to access the data and it also checks the integrity constraints when the database is modified.

4. Transaction Manager

A transaction in DBMS is nothing but a very small unit of the program. The Transaction Manager manages all the transaction (program) execution.

2M=query processor

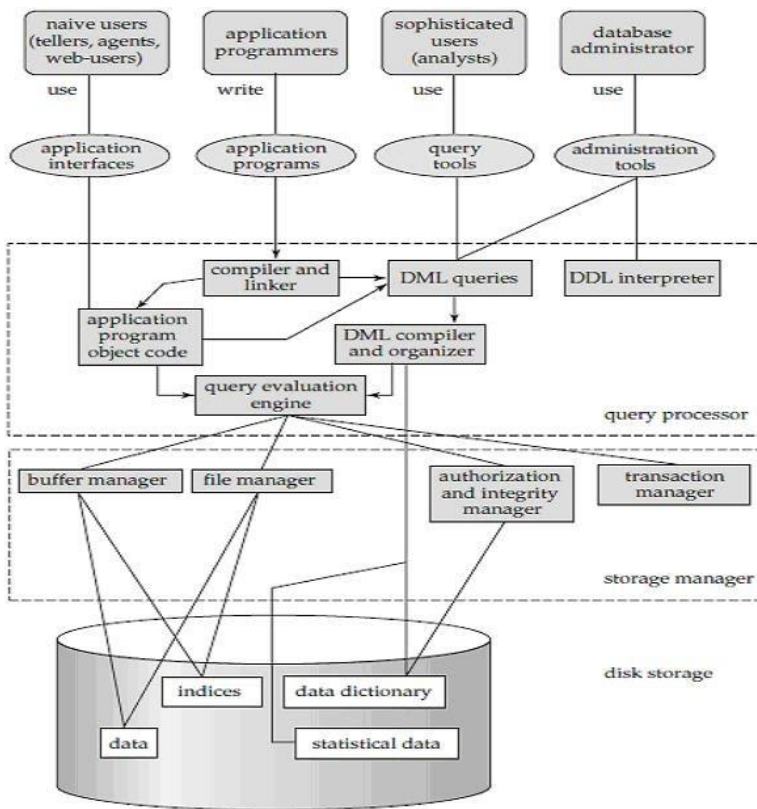


Fig: Overall architecture of DBBMS

Query Processor Components:

Query processor handles the query processing. It processes the query given by the user.

1. DDL Interpreter:

DDL expands to Data Definition Language. DDL Interpreter interprets the DDL statements such as schema definition statements like create, delete, etc.

2. DML Compiler:

DML expands to Data Manipulation Language. DML Compiler compiles (or translates) the DML statements such as select, update and delete statements into low-



	<p>level instructions which is nothing but the machine-readable object code to make it executable.</p> <p>The DML compiler is also responsible for query optimization.</p> <p>3. Query Evaluation Engine:</p> <p>The Query Evaluation Engine is also referred as "Query execution engine".</p> <p>Query Evaluation Engine first interprets the SQL commands to access the data from the database and then it returns the answer to that query.</p>	
b)	<p>Write the SQL queries for following EMP table. Emp (empno, deptno, ename, salary, designation, city.)</p> <p>i) Display average salary of all employees.</p> <p>ii) Display names of employees who stay in Mumbai or Pune.</p> <p>iii) Set the salary of employee 'Ramesh' to 50000.</p> <p>iv) Display names of employees whose salaries are less than 50000.</p> <p>v) Remove the Record of employees whose deptno is 10.</p> <p>vi) Remove the column deptno from EMP table.</p>	6 M
Ans	<p>i. select avg(salary) from emp;</p> <p>ii. select ename from emp where city='Mumbai' or city='Pune';</p> <p>iii. update emp set salary=50000 where ename='Ramesh';</p> <p>iv. select ename from emp where salary<50000;</p> <p>v. delete from emp where deptno=10;</p> <p>vi. alter table emp drop column deptno;</p>	1 M each
c)	<p>Write and Explain the syntax for creating, Altering and dropping the sequence.</p>	6 M
Ans	<p><u>Syntax for creating sequence:</u></p> <p>CREATE SEQUENCE sequence_name</p> <p>START WITH initial_value</p> <p>INCREMENT BY increment_value</p> <p>MINVALUE minimum value</p> <p>MAXVALUE maximum value</p> <p>CYCLE NOCYCLE ;</p> <p>.where as:</p>	2 M



	<p>sequence_name: Name of the sequence.</p> <p>initial_value: starting value from where the sequence starts. Initial_value should be greater than or equal to minimum value and less than equal to maximum value.</p> <p>increment_value: Value by which sequence will increment itself. Increment_value can be positive or negative.</p> <p>minimum_value: Minimum value of the sequence. maximum_value: Maximum value of the sequence.</p> <p>cycle: When sequence reaches its set_limit it starts from beginning.</p> <p>nocycle: An exception will be thrown if sequence exceeds its max_value.</p> <p><u>Example:</u> CREATE SEQUENCE sequence_1 start with 1 increment by 1 minvalue 0 maxvalue 100 cycle;</p> <p><u>Alter sequence:</u> Syntax: alter sequence <sequence_name> maxvalue <number>; Alter sequence can change the maxvalue in the sequence created.</p> <p><u>Dropping sequence:</u> Syntax: drop sequence <sequence_name>; To drop the sequence the DROP command is used.</p>	<p>2 M</p> <p>2 M</p>
6.	Attempt any <u>TWO</u> of the following:	12 M
a)	<p>Write SQL queries for following. Consider table stud (roll no, name, subl, sub2, sub3)</p> <p>i) Display names of student who got minimum mark in subl.</p> <p>ii) Display names of students who got above 40 marks in sub2.</p> <p>iii) Display count of Students failed in sub2.</p> <p>iv) Display average marks of subl of all students.</p> <p>v) Display names of students whose name start with 'A' by arranging them in ascending order of subl marks.</p> <p>vi) Display student name whose name ends with h' and subject 2 marks are</p>	6 M



	between 60 to 75.	
Ans	i. select name from stud where sub1= (select min(sub1) from stud); ii. select name from stud where sub2>40; iii. select count(*) from stud where sub2<40; iv. select avg(sub1) from stud; v. select name from stud where name like 'A%' order by sub1; vi. select name from stud where name like '%h' and sub2 between 60 and 75;	1 M each
b)	Write a PL/SQL code to check whether specified employee is present in Emp table or not. Accept empno from user. If employee does not exist display message using exception handling.	6 M
Ans	<u>PL/SQL Program:</u> declare no emp.empno%type; begin no:=&no; select empno into no from emp where empno=no; dbms_output.put_line('Empno is present: ' no); exception when no_data_found then dbms_output.put_line('Empno not present'); end;	2M =declaration 2M=select statement 2M=proper exception message
c)	Write SQL queries for following. 1) Create user named 'user1' having Password '1234 ii) Assign 'insert' and update' Privilege to 'user1'. ii) Remove update Privilege assigned to the user1. iv) Assign the resource Permission to user1.	6 M
Ans	i. create user user1 identified by 1234; ii. grant insert, update on system.emp to user1; iii. revoke update on system.emp from user1; iv. grant create session, unlimited tablespace, create table to user1;	1 ½ M each