Avinashilingam Institute for Home Science and Higher Education for Women

(Deemed to be University under Category 'A' by MHRD, Estd. u/s 3 of UGC Act 1956)
Re-accredited with 'A+' Grade by NAAC. Recognised by UGC Under Section 12B
Coimbatore - 641 043, Tamil Nadu, India

Bachelor's Degree Examination –November 2019

III Semester

Class: II UG **Time: 3 Hours**
Major: Computer Applications **Max.Marks : 100**

18BCAC10– Relational Database Management Systems

10 x 1 = 10

Part A
Choose the correct answer

Choose the correct answer.

1. The relational model is concerned with____
   a) Data Integrity b) Data Manipulation c) Both [a] & [b] d) None of these

2. A _____ is a file or a set of files that contains a database's metadata.
   a) Entity b) Data dictionary c) Attribute d) Entity set

3. In a relational database a referential integrity constraint can be specified with the help of _____
   a) Primary key b) Foreign key c) Objects d) Entity integrity

4. The First Normal form is based on the concept of _____
   a) Atomicity b) Functional dependency c) Durability d) Security

5. _____ is used to cancel previously granted or denied permissions.
   a) Revoke b) Grant c) Connect d) Reconnect.

6. Which one of the following is a DDL command?
   a) Drop b) Delete c) Update d) Select

7. _____ allows sending an entire block of statements to the database at one time.
   a) Query b) PL/SQL c) Command d) Constraint

8. _____ holds the one or more rows returned by a SQL statement.
   a) Functions b) Operators c) Cursor d) Trigger

9. A problem that arises during the execution of a program is _____
   a) trigger b) state c) exception d) event

10. ____ is a special type of stored procedure that automatically executes when an Event occurs in the database server.
    a) template b) exception c) trigger d) Inheritance

Part B 5 x 6 = 30

Answer ALL questions

Each answer should not exceed 400 words or two pages

11. a) Expound briefly on Relational Database elements.

OR

11. b) Summarize Three-Scheme architectura with neat sketch.

12. a) Illuminate Functional dependencies in detail.

OR

12. b) Elucidate concurrency control techniques with example.

13. a) Elucidate Date functions with example.

OR

13. b) Write a note on Aggregate functions with example.

14. a) What is a cursor? Give its syntaoc.

OR

14. b) Describe basic structure of PL/SQL with example.

15. a) Distinguish between procedures and functions.

OR

15.b) How is an exception handled in PL/SQL? Explain with suitable example.

Part C 5 x 12 = 60

Answer ALL questions

Each answer should not exceed 800 words or four pages

16. a) Elucidate Data Independence in detail.

OR

16. b) Dramatize various notations in E-R diagram with example.

17. a) Explain the normal forms based on primary keys.

OR

17. b)Summarize second and third normal forms with example.

18. a) Elucidate the types of SQL command in detail.

OR

18. b) Describe character functions with example.

19. a) Elucidate Looping statements in PL/SQL with example.

OR

19. b) Clarify PL/SQL character set and data types in detail

20. a) Explicate Triggers with example.

OR

20. b) Describe User defined procedures with example.

****

SHEME OF EVALUATION

SECTION A

1. c) Both [a] & [b]
2. b) Data dictionary
3. b) Foreign key
4. a) Atomicity
5. a) revoke.
6. a) drop
7. b) PL/SQL
8. c) cursor
9. c) exception
10. c) trigger

SECTION B

11. a)

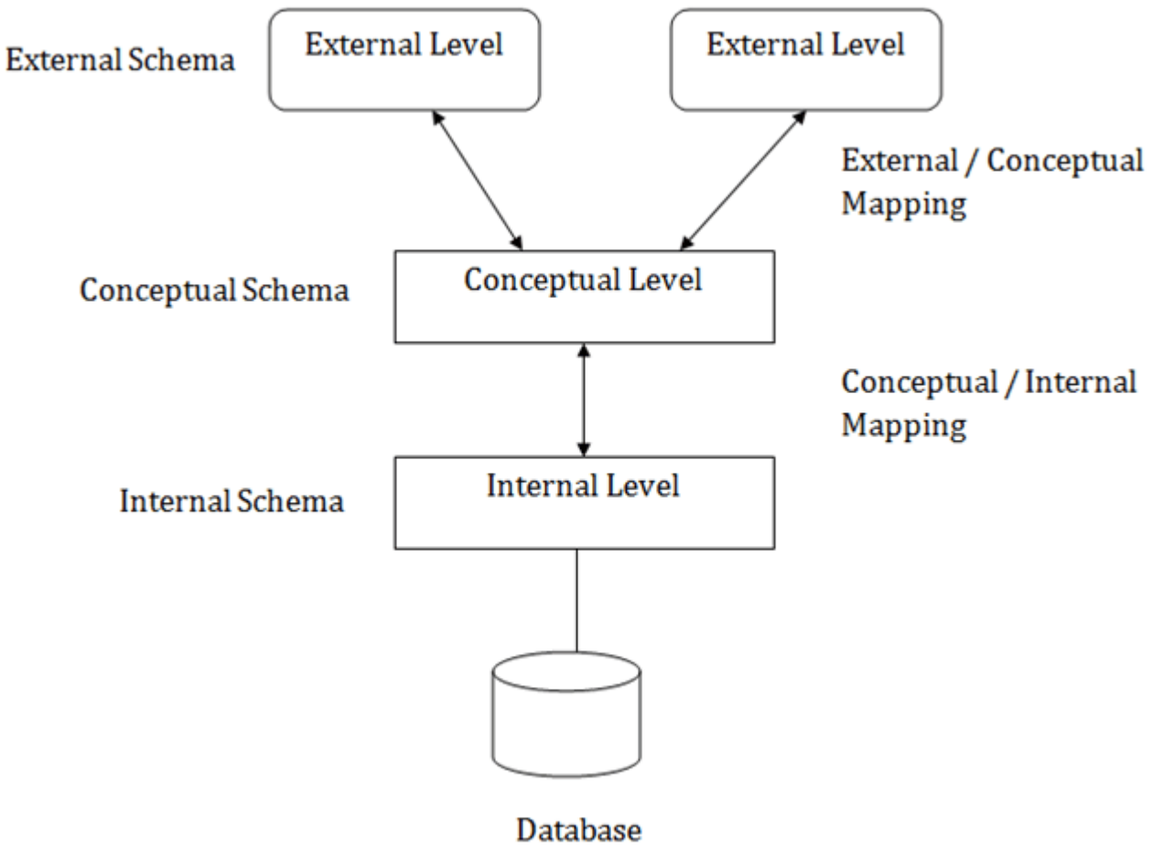
Relational databases are based on the relational model. The relational model is a group of rules set forth by E. F. Codd based on mathematical principles (relational algebra), and it defines how database management systems should function. The basic structures of a relational database (as defined by the relational model) are tables, columns (or fields), rows (or records), and keys.

OR

11. b) Three schema Architecture

- The three schema architecture is also called ANSI/SPARC architecture or three-level architecture.
- This framework is used to describe the structure of a specific database system.
- The three schema architecture is also used to separate the user applications and physical database.
- The three schema architecture contains three-levels. It breaks the database down into three different categories.

The three-schema architecture is as follows:



12. a)

A *functional dependency* (FD) is a relationship between two attributes, typically between the PK and other non-key attributes within a table. For any relation R, attribute Y is functionally dependent on attribute X (usually the PK), if for every valid instance of X, that value of X uniquely determines the value of Y. This relationship is indicated by the representation below :

X ————> Y

The left side of the above FD diagram is called the *determinant*, and the right side is the *dependent*.

OR

12. b) Concurrency control is the procedure in DBMS for managing simultaneous operations without conflicting with each another. Concurrent access is quite easy if all users are just reading data. There is no way they can interfere with one another. Though for any practical database, would have a mix of reading and WRITE operations and hence the concurrency is a challenge.

Concurrency control is used to address such conflicts which mostly occur with a multi-user system. It helps you to make sure that database transactions are performed concurrently without violating the data integrity of respective databases.

13. a) In SQL, dates are complicated for newbies, since while working with database, the format of the date in table must be matched with the input date in order to insert. In various scenarios instead of date, datetime (time is also involved with date) is used.
Default date functions are:

- **NOW():** Returns the current date and time. Example:
- SELECT NOW();

  Output:

  2017-01-13 08:03:52

- **CURDATE()**: Returns the current date. Example:
- SELECT CURDATE();

  Output:

  2017-01-13

OR

b) **The following are the most commonly used SQL aggregate functions:**

- AVG – calculates the average of a set of values.
- COUNT – counts rows in a specified table or view.
- MIN – gets the minimum value in a set of values.
- MAX – gets the maximum value in a set of values.
- SUM – calculates the sum of values.

14. a)

PL/SQL data type	**SQL data type**	**Description**
INT	**INT**	Signed four-byte integer numeric data
INTEGER	INTEGER	Signed four-byte integer numeric data
LONG	**CLOB** (32760)	Character **large object** data
LONG RAW	**BLOB** (32760)	Binary **large object** data

.

OR

14. b)

In PL/SQL, as in most other procedural languages, the smallest meaningful grouping of code is known as a *block*. A block is a unit of code that provides execution and scoping boundaries for variable declarations and exception handling. PL/SQL allows you to create *anonymous blocks* (blocks of code that have no name) and *named blocks*, which may be packages, procedures, functions, triggers, or object types.

A PL/SQL block has up to four different sections, only one of which is mandatory:

Header

Used only for named blocks. The header determines the way the named block or program must be called. Optional.

Declaration section

Identifies variables, cursors, and subblocks that are referenced in the execution and exception sections. Optional.

Execution section

Statements the PL/SQL runtime engine will execute at runtime. Mandatory.

15.a) **Function** must return a value but in Stored **Procedure** it is optional. Even a **procedure** can return zero or n values. **Functions** can have only input parameters for it whereas **Procedures** can have input or output parameters . **Functions** can be called from **Procedure** whereas **Procedures** cannot be called from a **Function**.

OR

15. b)

An **exception** is a **PL**/**SQL** error that is raised during program execution, either implicitly by Times or explicitly by your program. **Handle** an **exception** by trapping it with a **handler** or propagating it to the calling environment.

SECTION-C

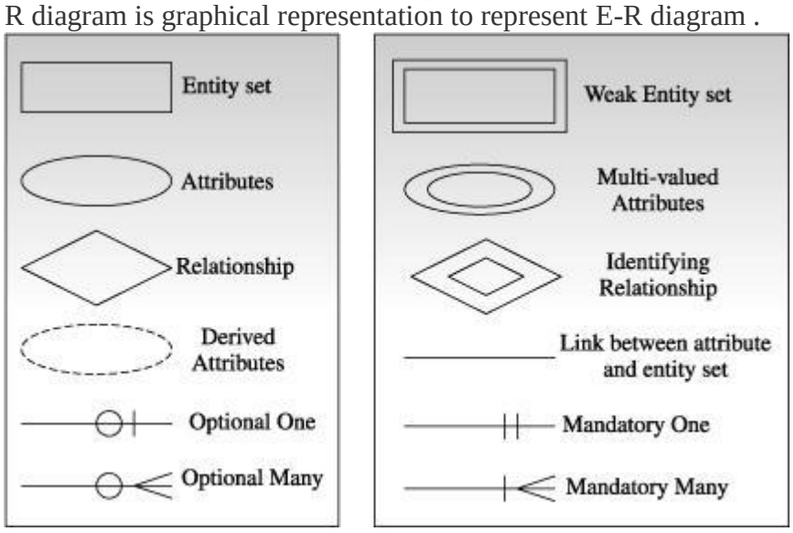
16. a) **Data independence** is the type of data transparency that matters for a centralized DBMS. It refers to the immunity of user applications to changes made in the definition and organization of data.there are two types of data independence physical and logical data independence .

Physical data independence deals with hiding the details of the storage structure from user applications. The application should not be involved with these issues, since there is no difference in the operation carried out against the data.

The data independence and operation independence together gives the feature of data abstraction. There are two levels of data independence.

OR

16. b) An E-R model is normally expressed as an entity-relationship diagram (called ER diagram). E-R diagram is graphical representation to represent E-R diagram .



17. a) **Definition:** The **normal form** of a relation refers to the highest normal form condition that it meets, and hence indicates the degree to which it has been normalized.

Normal forms, when considered *in isolation* from other factors, do not guarantee a good database design. It is generally not sufficient to check separately that each relation schema in the database is, say, in BCNF or 3NF. Rather, the process of normalization through decomposition must also confirm the existence of additional properties that the relational schemas, taken together, should possess. These would include two properties:

The **nonadditive join or lossless join property**, which guarantees that the spurious tuple generation problem discussed in Section 15.1.4 does not occur with respect to the relation schemas created after decomposition.

The **dependency preservation property**, which ensures that each functional dependency is represented in some individual relation resulting after decomposition.

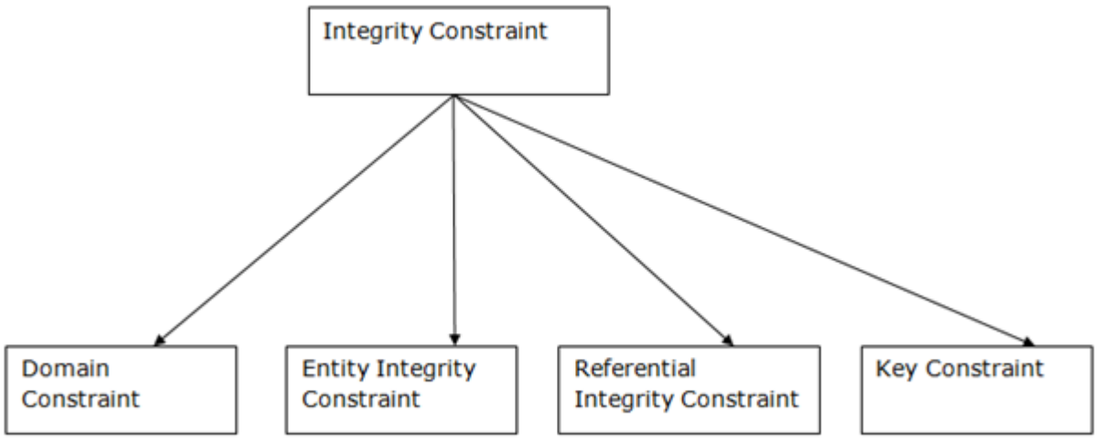
OR

17. b)

Integrity Constraints

- Integrity constraints are a set of rules. It is used to maintain the quality of information.
- Integrity constraints ensure that the data insertion, updating, and other processes have to be performed in such a way that data integrity is not affected.
- Thus, integrity constraint is used to guard against accidental damage to the database.

Types of Integrity Constraint



1. Domain constraints

- Domain constraints can be defined as the definition of a valid set of values for an attribute.
- The data type of domain includes string, character, integer, time, date, currency, etc. The value of the attribute must be available in the corresponding domain.

Example:

ID	NAME	SEMENSTER	AGE
1000	Tom	1st	17
1001	Johnson	2nd	24
1002	Leonardo	5th	21
1003	Kate	3rd	19
1004	Morgan	8th	A

Not allowed. Because AGE is an integer attribute

2. Entity integrity constraints

- The entity integrity constraint states that primary key value can't be null.
- This is because the primary key value is used to identify individual rows in relation and if the primary key has a null value, then we can't identify those rows.
- A table can contain a null value other than the primary key field.

Example:

EMPLOYEE

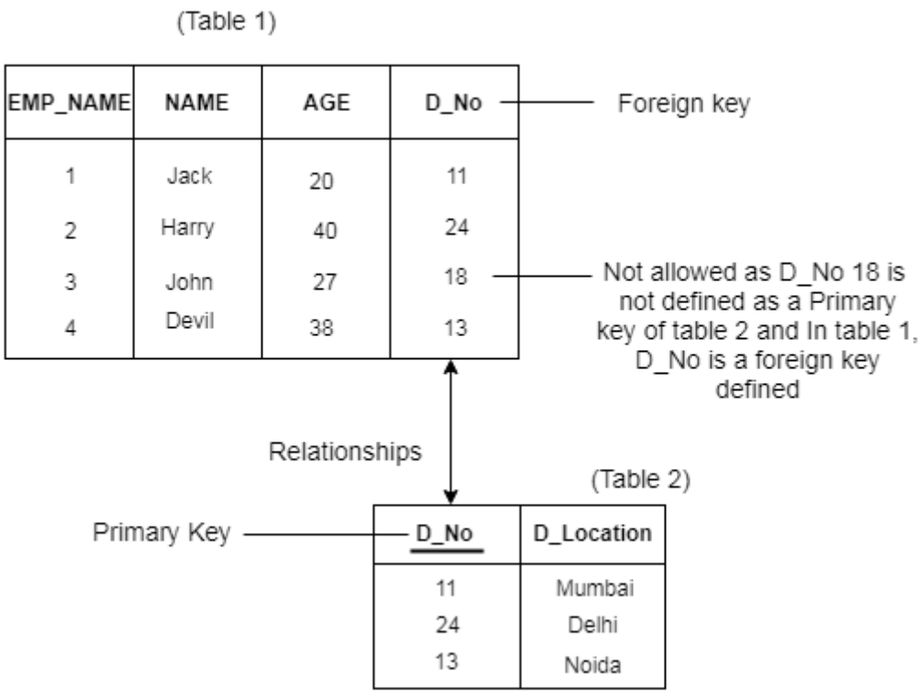
EMP_ID	EMP_NAME	SALARY
123	Jack	30000
142	Harry	60000
164	John	20000
	Jackson	27000

Not allowed as primary key can't contain a NULL value

3. Referential Integrity Constraints

- A referential integrity constraint is specified between two tables.
- In the Referential integrity constraints, if a foreign key in Table 1 refers to the Primary Key of Table 2, then every value of the Foreign Key in Table 1 must be null or be available in Table 2.

Example:



4. Key constraints

- Keys are the entity set that is used to identify an entity within its entity set uniquely.
- An entity set can have multiple keys, but out of which one key will be the primary key. A primary key can contain a unique and null value in the relational table.

Example:

ID	NAME	SEMENSTER	AGE
1000	Tom	1st	17
1001	Johnson	2nd	24
1002	Leonardo	5th	21
1003	Kate	3rd	19
1002	Morgan	8th	22

Not allowed. Because all row must be unique

18. a) Types of SQL Commands

The following sections discuss the basic categories of commands used in SQL to perform various functions. These functions include building database objects, manipulating objects, populating database tables with data, updating existing data in tables, deleting data, performing database queries, controlling database access, and overall database administration.

The main categories are

- DDL (Data Definition Language)
- DML (Data Manipulation Language)
- DQL (Data Query Language)
- DCL (Data Control Language)

- Data administration commands
- Transactional control commands

(OR)

18. b) A character function is a function that takes one or more character values as parameters and returns either a character value or a number value. The Oracle Server and PL/SQL provide a number of different character datatypes, including CHAR, VARCHAR, VARCHAR2, LONG, RAW, and LONG RAW. In PL/SQL, the three different datatype families for character data are:

VARCHAR2

A variable-length character datatype whose data is converted by the RDBMS

CHAR

The fixed-length datatype

RAW

A variable-length datatype whose data is not converted by the RDBMS, but instead is left in "raw" form

When a character function returns a character value, that value is always of type VARCHAR2 (variable length), with the following two exceptions: UPPER and LOWER. These functions convert to upper- and lowercase, respectively, and return CHAR values (fixed length) if the strings they are called on to convert are fixed-length CHAR arguments.

PL/SQL provides a rich set of character functions that allow you to get information about strings and modify the contents of those strings in very high-level, powerful ways.

Name	**Description**
ASCII	Returns the ASCII code of a character.
CHR	Returns the character associated with the specified collating code.
CONCAT	Concatenates two strings into one.
INITCAP	Sets the first letter of each word to uppercase. All other letters are set to lowercase.
INSTR	Returns the location in a string of the specified substring.
LENGTH	Returns the length of a string.
LOWER	Converts all letters to lowercase.
LPAD	Pads a string on the left with the specified characters.
LTRIM	Trims the left side of a string of all specified characters.
REPLACE	Replaces a character sequence in a string with a different set of characters.
RPAD	Pads a string on the right with the specified characters.
RTRIM	Trims the right side of a string of all specified characters.
SOUNDEX	Returns the "soundex" of a string.
SUBSTR	Returns the specified portion of a string.
TRANSLATE	Translates single characters in a string to different characters.
UPPER	Converts all letters in the string to uppercase.

(OR)

19. a)

S.No	**Loop Type & Description**
1	PL/SQL Basic LOOP

In this loop structure, sequence of statements is enclosed between the LOOP and the END LOOP statements. At each iteration, the sequence of statements is executed and then control resumes at the top of the loop.

PL/SQL WHILE LOOP

2 Repeats a statement or group of statements while a given condition is true. It tests the condition before executing the loop body.

PL/SQL FOR LOOP

3 Execute a sequence of statements multiple times and abbreviates the code that manages the loop variable.

Nested loops in PL/SQL

4 You can use one or more loop inside any another basic loop, while, or for loop.

OR

19. b)

- CHARACTER Data Type
- NUMBER Data Type
- BOOLEAN Data Type
- DATE Data Type
- LOB Data Type

User defined function is a database object which encapsulates one or more **sql** statements for reuse , which can accept zero or more parameters and return either a value or table. **User defined** fuctions contain useful logic for using another queries or other database objects like stored **procedures**.