MySQL MCQ Test 1

1. Which type of database management system is MySQL?

a) Object-oriented

b) Hierarchical

**c) Relational**

d) Network

2. What is data in a MySQL database organized into?

a) Objects

**b) Tables**

c) Networks

d) File systems

3. What represents an ‘attribute’ in a relational database?

a) Table

b) Row

**c) Column**

d) Object

4. What represents a ‘tuple’ in a relational database?

a) Table

**b) Row**

c) Column

d) Object

d) Server daemon program

5. The MySQL server used in its client/server architecture is \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

a) mysqla

b) mysqlb

c) mysqlc

**d) mysqld**

6. What is the degree of a table with 1000 rows and 10 column?

1. **10**
2. 100
3. 1000
4. None of the above.

7. Which of the following command is used to get all the columns in a table?

1. #
2. **\***
3. @
4. $

8. Total no of attributes/columns present in a relation/table is called?

1. **Degree d(R)**
2. Cardinality |R|
3. Domain
4. All of the above

9. Total no if tuples present in a relation or Rows present in a table, is called?

1. Degree d(R)
2. **Cardinality |R|**
3. Domain
4. All of the above

10. Which statement is used to select a default database?

**a) USE**

b) CREATE

c) DROP

d) SCHEMA

11. Which keyword is the synonym for DATABASE?

a) TABLE

b) OBJECT

c) DB

**d) SCHEMA**

12. Which statement is used to drop an existing database?

1. **DROP DATABASE**
2. DELETE DATABASE
3. DEL DATABASE
4. None of the above

13. To create a database only if it doesn’t already exist, which clause is used?

a) IF EXISTS

**b) IF NOT EXISTS**

c) CREATE EXISTS

d) EXISTS IF

14. Which statement is used to see the list of for an existing database?

a) SHOW CREATE DATABASE

**b) SHOW SCHEMAS**

c) SHOW CREATE

d) SHOW CREATE DATABASE TABLE

15. Which statement is used to see the list of for an existing database?

a) SHOW CREATE DATABASE

**b) SHOW DATABASES**

c) SHOW CREATE

d) SHOW CREATE DATABASE TABLE

16. If the default database is dropped, the default database is unset (the DATABASE() function returns \_\_\_\_\_).

1. **NULL**
2. Empty
3. Undefined
4. Not of the above

17. What does db1 represent in the following code snippet?

CREATE DATABASE If NOT EXISTS db1

a) A table

b) A row

**c) A database**

d) None of the above.

18. A\_\_\_\_\_ is a query that retrieves rows from more than one table or view:

a) Start

b) End

**c) Join**

d) All of the mentioned

19. A condition in JOINS is referred to as \_\_\_\_\_\_\_\_\_\_

a) Join in SQL

**b) Join condition**

c) Join in SQL & Condition

d) None of the mentioned

20. SELECT \* FROM EMP INNER JOIN DEPT ON EMP.DEPTNO = DEPT.DEPTNO;

In the above statement, which of the following word is optional.

1. **INNER**
2. JOIN
3. ON
4. None of the above

21. SELECT \* FROM EMP INNER JOIN DEPT ON EMP.DEPTNO = DEPT.DEPTNO; will display

1. **All matching rows from table EMP and DEPT**
2. All rows from EMP table
3. All rows from DEPT table
4. None of the above

22. Which are the join types in join condition:

a) Cross join

b) Natural join

c) Join with USING clause

**d) All of the mentioned**

23. Which product is returned in a join query have no join condition:

a) Equijoins

**b) Cartesian**

c) Both Equijoins and Cartesian

d) None of the mentioned

24. Which is a join condition contains an equality operator:

**a) Equijoins**

b) Cartesian

c) Both Equijoins and Cartesian

d) None of the mentioned

25. Which join refers to join records from the write table that have no matching key in the left table are include in the result set:

a) Left outer join

**b) Right outer join**

c) Full outer join

d) Half outer join

26. SELECT \* FROM EMP INNER JOIN DEPT D ON EMP.DEPTNO = DEPT.DEPTNO; will display

1. All matching rows from table EMP and DEPT
2. All rows from EMP table
3. All rows from DEPT table
4. **Will give an error**

27. The CARTESIAN JOIN is also known as

1. JOIN
2. **CROSS JOIN**
3. SIMPLE JOIN
4. NATURAN JOIN

28. In the absence of a WHERE condition the INNER JOIN will behave like a.

1. CARTESIAN JOIN
2. PRODUCT JOIN
3. CROSS JOIN
4. **All of the above**

29. In the presence of WHERE condition this JOIN will function like a.

1. INNER JOIN
2. EQUI JOIN
3. **Both A and B**
4. None of the above.

30. RIGHT OUTER JOIN, return all records from the right table, and the matched records from the left table.

1. **true**
2. false

MySQL MCQ Test 2

1. LEFT OUTER JOIN, return all records from the left table, and the matched records from the right table.

1. **true**
2. false

2. INNER JOIN, returns records that have matching values in both tables.

1. **true**
2. false

3. To selects all orders with customer and shipper information. What statement will you issue to get the result?

1. SELECT \* FROM Orders INNER JOIN Customers ON Orders.CustomerID = Customers.CustomerID) INNER JOIN Shippers ON Orders.ShipperID = Shippers.ShipperID);
2. SELECT \* FROM ((Orders INNER JOIN Customers ON Orders.CustomerID = Customers.CustomerID) INNER JOIN Shippers ON Orders.ShipperID = Shippers.ShipperID);
3. **Both A and B**
4. None of the above

4. INNER JOIN, in the absence of a join condition produce a \_\_\_\_\_\_\_\_\_\_\_ between the specified tables.

1. **Cartesian JOIN**
2. LEFT JOIN
3. RIGHT JOIN
4. None of the above.

5. (comma sign), in the absence of a join condition produce a \_\_\_\_\_\_\_\_\_\_\_ between the specified tables.

1. **Product JOIN**
2. LEFT JOIN
3. RIGHT JOIN
4. None of the above.

6. Redundant columns of a NATURAL join \_\_\_\_\_\_\_\_\_\_\_

1. **do not appear.**
2. will display at the last.
3. will appear for multiple times.
4. None of the above.

7. SELECT select\_list FROM table\_list WHERE row\_constraint GROUP BY grouping\_columns; Which of these is not optional?

**a) select\_list**

b) table\_list

c) row\_constraint

d) grouping\_columns

8. In inner join, result is produced by matching rows in one table with rows in another table.

**a) True**

b) False

9. The join where all possible row combinations are produced is called \_\_\_\_\_\_\_\_\_

a) INNER JOIN

b) OUTER

c) NATURAL

**d) CARTESIAN**

10. The clause that filters JOIN results is called \_\_\_\_\_\_\_\_\_

**a) WHERE**

b) SORT

c) GROUP

d) GROUP BY

11. CROSS JOIN and JOIN are similar to \_\_\_\_\_\_\_\_\_\_

**a) INNER JOIN**

b) NATURAL JOIN

c) OUTER JOIN

d) CARTESIAN JOIN

12. The comma operator can also be used to join tables.

**a) True**

b) False

13. The left and right joins are also known as \_\_\_\_\_\_\_\_\_\_

a) INNER JOIN

b) NATURAL JOIN

**c) OUTER JOIN**

d) CARTESIAN JOIN

14. What is joining a table to itself called?

a) COMPLETE

**b) SELF**

c) OBSOLETE

d) CROSS

15. In which join all the rows from the left table appear in the output irrespective of the content of the other table?

a) RIGHT JOIN

**b) LEFT JOIN**

c) INNER JOIN

d) OUTER JOIN

16. The join in which all the rows from the right table appear in the output irrespective of the content of the other table is \_\_\_\_\_\_\_\_\_\_\_

a) CARTESIAN JOIN

b) CROSS JOIN

c) INNER JOIN

**d) RIGHT JOIN**

17. Write a SQL statement to display a list with salesman name, customer name and their cities for the salesmen and customer who belongs to the same city.

1. SELECT salesman.name AS "Salesman", customer.cust\_name, customer.city FROM salesman, customer WHERE salesman.city = city;
2. SELECT salesman.name AS "Salesman", customer.cust\_name, customer.city FROM salesman, customer WHERE city = customer.city;
3. **SELECT salesman.name AS "Salesman", customer.cust\_name, customer.city FROM salesman, customer WHERE salesman.city = customer.city;**
4. All of the above.

18. If you join multiple tables that have the same column name, you have to use qualifier to refer to that column in the SELECT statement to avoid the ambiguous column error.

1. **true**
2. false

19. USING (customerNumber) is equivalent to

1. **ON c.customerNumber = o.customerNumber**
2. FROM c.customerNumber = o.customerNumber
3. IN c.customerNumber = o.customerNumber
4. None of the above.

20. To avoid a Cartesian product, always include a valid join condition in a WHERE clause.

1. **True**
2. False

21. Select the correct query/queries for cross join:

1. **Select \* FROM Table1 T1 CROSS JOIN Table1 T2;**
2. Select \* FROM Table1 T1 ALL CROSS JOIN Table1 T2;
3. Select \* FROM Table1 T1, Table1 T1;
4. Select \* FROM Table1 T1 CROSS Table1 T2;

22. Select the correct query/queries for cross join:

1. Select \* FROM Table1 T1 CROSS JOIN Table1 T1;
2. Select \* FROM Table1 T1 ALL CROSS JOIN Table1 T2;
3. **Select \* FROM Table1 T1, Table1 T2;**
4. Select \* FROM Table1 T1 CROSS Table1 T2;

23. LEFT JOIN and LEFT OUTER JOIN are equivalent.

1. **True**
2. False

24. You can drop OUTER keyword and just say LEFT JOIN or RIGHT JOIN or FULL JOIN.

1. **True**
2. False

25. To specify a normal join, using keyword inner is \_\_\_\_\_\_\_\_\_\_\_.

1. Mandatory
2. **Optional**
3. Independent
4. Free

26. ON predicate is written like a

1. From clause
2. Select clause
3. **Where clause**
4. Define clause

27. USING predicate can be used as an alternative of \_\_\_\_\_\_\_\_\_\_\_.

1. From clause
2. Select clause
3. **Joining condition**
4. Define clause

28. Inner join returns rows when there is at least one match in both tables.

1. **True**
2. False

29. In an outer join, rows are returned even when there are no matches through the JOIN criteria on the second table.

1. **True**
2. False

30. In natural joins, rows are returned from two tables based on common column names.

1. **True**
2. False

MySQL MCQ Test 3

1. Full join is used to return records from both left and right outer join.

1. **True**
2. False

2. The JOIN which does Cartesian product is called?

a) Left Join

b) Left Outer Join

c) Right Outer Join

**d) Cross Join**

3. What is the other name of INNER JOIN?

**a) Equi Join**

b) In Join

c) Out Join

d) All of the above

4. Which join is used for joining the table to itself?

a) In

b) Natural

c) Cross

**d) Self**

5. The \_\_\_\_\_\_\_ returns a number of rows equal to the product of all rows in all the tables being joined.

1. Outer join
2. Inner join
3. **Cartesian product**
4. Self Join

6. \_\_\_\_\_\_\_\_\_return rows only when there is at least one row from both tables that matches the join condition.

1. **Inner joins**
2. Outer joins
3. Self Join
4. Left outer join

7. The join in which all the rows from the right table appear in the output irrespective of the content of the other table is \_\_\_\_\_\_\_\_\_\_\_\_\_\_

a) CARTESIAN JOIN

b) CROSS JOIN

c) INNER JOIN

**d) RIGHT JOIN**

8. In SQL the statement

SELECT \* FROM R, S; is equivalent to

1. SELECT \* FROM R NATURAL JOIN S.
2. **SELECT \* FROM R CROSS JOIN S.**
3. SELECT \* FROM R UNION JOIN S.
4. SELECT \* FROM R INNER JOIN S.

9. In SQL the word ‘natural’ can be used with

1. inner join
2. full outer join
3. **right outer join**
4. all of the above

10. In SQL the word ‘natural’ can be used with

1. cross join
2. full outer join
3. **left outer join**
4. all of the above

11. If two relations R and S are joined, then the non-matching tuples of both R and S are ignored in

1. left outer join
2. right outer join
3. full outer join
4. **inner join**

12. If two relation R and S are joined where in both the relation there are no common attributes, then what happens if natural join is issued?

1. It works like left outer join
2. It works like right outer join
3. It works like inner join
4. **It works like cross join**

13. The common column is eliminated in

1. Theta join
2. Outer join
3. **Natural join**
4. Composed join

14. What type of join is needed when you wish to include rows that do not have matching values?

1. Equi-join
2. Natural join
3. **Outer join**
4. All of the above.

15. What type of join is needed when you wish to return rows that do have matching values?

1. Equi-join
2. Natural join
3. Simple join
4. **All of the above.**

16. Which of the following is one of the basic approaches for joining tables?

1. Subqueries
2. Union Join
3. Natural join
4. **All of the above**

17. The following SQL is which type of join:

SELECT CUSTOMER\_T. CUSTOMER\_ID, ORDER\_T. CUSTOMER\_ID, NAME, ORDER\_ID FROM CUSTOMER\_T, ORDER\_T WHERE CUSTOMER\_T. CUSTOMER\_ID = ORDER\_T. CUSTOMER\_ID

1. **Equi-join**
2. Natural join
3. Outer join
4. Cartesian join

18. How many tables may be included with a join?

1. One
2. Two
3. Three
4. **All of the above.**

19. The following SQL is which type of join:

SELECT CUSTOMER\_T. CUSTOMER\_ID, ORDER\_T. CUSTOMER\_ID, NAME, ORDER\_ID FROM CUSTOMER\_T, ORDER\_T;

1. Equi-join
2. Natural join
3. Outer join
4. **Cartesian join**

20. The facility that allows nesting one select statement into another is called \_\_\_\_\_\_\_\_\_\_

a) Nesting

b) Binding

**c) Subquerying**

d) Encapsulating

21. Which subquery returns a single value?

**a) Scalar**

b) Column

c) Row

d) Table

22. Usage of aggregates in WHERE clause is allowed.

a) True

**b) False**

23. Which operators are used when a subquery returns multiple rows to be evaluated in comparison to the outer query?

**a) IN and NOT IN**

b) EXISTS and NOT EXISTS

c) OUTER JOIN and INNER JOIN

d) LEFT JOIN and RIGHT JOIN

24. The 'ALL' subquery performs which operation?

a) Row

**b) Column**

c) Table

d) Database

25. Which of these operators perform similar operations like 'ALL ' and 'ANY '?

**a) SOME**

b) MANY

c) SELECT

d) GROUP

26. Which operators test whether a subquery returns any rows?

a) IN and NOT IN

**b) EXISTS and NOT EXISTS**

c) PRESENT

d) ABSENT

27. An uncorrelated subquery does not contain any reference to the values from the outer query.

**a) True**

b) False

29. Analyse the following statement:

SELECT \* from EMP where true = (SELECT if (null is null, true, false));

1. **Will return all records from EMP table.**
2. Will return all records whose comm is not null.
3. Error, because Subquery returns more than 1 row
4. None of the above

30. Which of these operators does not perform relative-value comparisons?

a) =

**b) ==**

c) <=

d) >=

118. The EXISTS keyword will be true if:

1. **Any row in the subquery meets the condition only**
2. All rows in the subquery fail the condition only
3. Both of these two conditions are met
4. Neither of these two conditions is met

119. How can you find rows that do not match some specified condition?

1. EXISTS
2. **NOT EXISTS**
3. NOT EXISTENCES
4. None of the mentioned

120. Table Employee has 10 records. It has a non-NULL SALARY column which is also UNIQUE.

The SQL statement

SELECT COUNT(\*) FROM Employee WHERE SALARY > ANY (SELECT SALARY FROM EMPLOYEE);

Prints

1. 10
2. **9**
3. 5
4. 0

121. The facility that allows nesting one select statement into another is called \_\_\_\_\_\_\_\_\_\_

a) Nesting

b) Binding

**c) Subquerying**

d) Encapsulating

122. Which subquery returns a single value?

**a) Scalar**

b) Column

c) Row

d) Table

123. Usage of aggregates in WHERE clause is allowed.

a) True

**b) False**

124. Which operators are used when a subquery returns multiple rows to be evaluated in comparison to the outer query?

**a) IN and NOT IN**

b) EXISTS and NOT EXISTS

c) OUTER JOIN and INNER JOIN

d) LEFT JOIN and RIGHT JOIN

125. Analyse the following statement:

SELECT \* from EMP where true = (SELECT if (comm is null, true, false) from EMP);

1. Will return all records whose comm is null.
2. Will return all records whose comm is not null.
3. **Error, because Subquery returns more than 1 row**
4. None of the above

126. Analyse the following statement:

SELECT \* from (SELECT \* from EMP);

1. Will print all records from EMP table.
2. Will give an error because of invalid tbl\_name;
3. **Will give an error because every derived table must have its own alias**
4. Brackets are to be removed to get the output.

127. Analyse the following statement:

SELECT \* from (SELECT \* from EMP) E;

1. **Will print all records from EMP table.**
2. Will give an error because of invalid tbl\_name;
3. Will give an error because every derived table must have its own alias
4. Brackets are to be removed to get the output.

128. Analyse the following statement:

SELECT (SELECT 1, 2) from EMP

1. Will return all records from EMP table.
2. Will return all records whose comm is not null.
3. **Error, because Subquery returns more than 1 column value.**
4. None of the above

129. Analyse the following statement:

SELECT \* from EMP where sal > (select sal from EMP where ename='SMITH');

1. **Will return all records from EMP table whose sal is more than SIMTI's salary.**
2. Will return all records whose salary is as same as SMITH's salary.
3. Error, because Subquery returns more than 1 row
4. None of the above

130. A subquery in an SQL SELECT statement is enclosed in:

1. braces -- {...}.
2. CAPITAL LETTERS.
3. **Parenthesis -- (...).**
4. brackets -- [...].

131. Scalar subquery should return.

1. **1 row, 1 column**
2. 1 row, multiple columns
3. Multiple rows, 1 column
4. None of the above.

132. Which of the following are not valid subquery type:

1. Single row subquery
2. Correlated subqueries
3. Nested subqueries
4. **All are valid**

133. Select ID, GPA from student grades order by GPA \_\_\_\_\_\_\_\_\_\_\_\_

In order to give only 10 rank on the whole we should use:

**a) Limit 10**

b) Upto 10

c) Only 10

d) Max 10

134. Select \_\_\_\_\_\_\_\_ dept\_name from instructor;

Here which of the following displays the unique values of the column?

a) All

b) From

**c) Distinct**

d) Name

135. Select ID, name, dept name, salary \* 1.1 where instructor;

The query given below will not give an error. Which one of the following has to be replaced to get the desired output?

a) Salary\*1.1

b) ID

**c) Where**

d) Instructor

136. Select \* from student join takes using (ID);

The above query is equivalent to :

**a) Select \* from student inner join takes using (ID);**

b) Select \* from student outer join takes using (ID);

c) Select \* from student left outer join takes using (ID);

d) All of the mentioned

137. The \_\_\_\_\_\_ clause allows us to select only those rows in the result relation of the \_\_\_\_ clause that satisfy a specified predicate.

**a) Where, from**

b) From, select

c) Select, from

d) From, where

138. Parenthesis is not mandatory for sub-queries.

1. True
2. **False**

139. Which of the following are DML commands in MySQL Database?

1. HAVING
2. GROUP BY
3. INTERSECT
4. **INSERT**

140. Which of following commands is a DDL (Data Definition Language) command?

1. DELETE
2. INSERT
3. **TRUNCATE**
4. None of the above

141. Which of the following commands is used to populate table rows with data?

1. DELETE
2. INSERT
3. **SELECT**
4. UPDATE

142. Which of the following can be used to insert rows in tables?

1. SELECT
2. **INSERT**
3. Sub-queries
4. All of the above

143. Which of the following commands is used to change the rows that already exist in a table?

1. INSERT
2. UNION
3. **UPDATE**
4. CHANGE

144. What is true about the UPDATE command?

1. It can update only one row at a time
2. It can update only 100 rows at a time
3. **It can update unlimited rows at a time in bulk**
4. None of the above

145. Which of the following clauses decides how many rows are to be updated?

1. SELECT
2. **WHERE**
3. FROM
4. All of the above

146. Which of the following commands can be used to remove existing records from a table?

1. UPDATE
2. INSERT
3. MINUS
4. **DELETE**

147. What among the following is true about the DELETE statement?

1. The DELETE statement has to be accompanied by the WHERE clause
2. **It is not mandatory to write a WHERE clause with the DELETE statement**
3. DELETE can remove data from multiple tables at a time
4. None of the above

148. What among the following is a TRUNCATE statement equivalent to?

1. To a DELETE statement
2. To an UPDATE statement
3. **A DELETE statement without a WHERE clause**
4. None of the above

149. Which of the following situations indicate that a DML operation has taken place?

1. **When new rows are added to a table**
2. When two queries are combined
3. When a table is truncated
4. None of the above

150. VALUES is mandatory to be used if we use the keyword INSERT

1. **True**
2. False

151. VALUES can add multiple rows at a time during the INSERT

1. **True**
2. False

152. Which keyword is mandatory to be used if we use the keyword INSERT?

1. Value
2. Values
3. Val
4. **Either A or B**

153. Consider a database table COLORS name whose attributes are ID (primary key), Color.

ID = {1, 2, 3, 4, 5, 6}

Color = {Red, Blue, Red, Blue, Green, Yellow}

INSERT into COLORS VALUES (7,'Black')

How many rows the above INSERT command will insert?

**a) 1**

b) 2

c) NULL

d) None of the mentioned

154. Consider a database table COLORS name whose attributes are ID (primary key), Color.

ID = {1, 2, 3, 4, 5, 6}

Color = {Red, Blue, Red, Blue, Green, Yellow}

INSERT into COLORS VALUE (7,'Black'), (8,'Cyan')

How many rows the above INSERT command will insert?

1. 1
2. **2**
3. NULL
4. None of the mentioned

155. Which of the following commands allows undoing the changed data?

1. **ROLLBACK**
2. COMMIT
3. INSERT
4. UPDATE

156. What is true about an INSERT statement which tries to insert values into a virtual column?

1. **It cannot insert values in the Virtual column**
2. It can insert values
3. It throws an error
4. All of the above

157. Which of the following commands allows the user to insert multiple rows with a single statement?

1. **INSERT ... values (...), (...)**
2. INSERT ... values (...), values (...)
3. INSERT ALL ... values (...)
4. None of the above

158. DDL statements are used for which of the following database objects?

1. **Tables**
2. Sub-queries
3. Rows
4. Columns

159. What is the basic unit of storage in Database that contains data?

1. View
2. Column
3. Query
4. **Table**

160. Which of the following database objects generate numeric values?

1. Table
2. View
3. Index
4. **AUTO\_INCREMENT**

161. Which of the following characters can be used to name a table?

1. A to Z
2. a to z
3. 0 to 9
4. **All of the above**

162. Consider following relations with values.

T1 = {10}

T2 = {-5, 0, +5, 10}

What will be output of the given statement?

SELECT \* FROM T2 WHERE C1 >ALL (SELECT C1 FROM T1);

1. 0
2. 10
3. Null
4. **Empty set**

163. Consider following relations with values.

T1 = {10}

T2 = {-5, 0, +5, 10}

What will be output of the given statement?

SELECT \* FROM T2 WHERE C1 >=ALL (SELECT C1 FROM T1);

1. 0
2. **10**
3. Null
4. Empty set

164. Consider following relations with values.

T1 = {10}

T2 = {12, 6, NULL, -100}

What will be output of the given statement?

SELECT \* FROM T2 WHERE C1 >ALL (SELECT C1 FROM T1);

1. 0
2. -100
3. **12**
4. Empty set

165. Consider following relations with values.

T1 = {10}

T2 = {12, 6, NULL, -100}

What will be output of the given statement?

SELECT \* FROM T2 WHERE C1 <ALL (SELECT C1 FROM T1);

1. 6
2. -100
3. **Both A and B**
4. Empty set

166. Consider following relations with values.

T1 = {10, 3}

T2 = {12, 6, NULL, -100}

What will be output of the given statement?

SELECT \* FROM T2 WHERE C1 >ALL (SELECT C1 FROM T1);

1. 6
2. **12**
3. Both A and B
4. Empty set

167. Consider following relations with values.

T1 = {10, 3}

T2 = {12, 6, NULL, -100}

What will be output of the given statement?

SELECT C1 FROM T1 UNION SELECT C1 FROM T2;

1. {10, 3, 12, 6, -100}
2. {10, 3, 12, 6, NULL}
3. **{10, 3, 12, 6, NULL, -100}**
4. None of the above

168. The TRUNCATE TABLE statement removes all the data of a table and resets the auto-increment value to zero.

1. **True**
2. False

169. The facility that allows nesting one select statement into another is \_\_\_\_\_\_\_\_\_\_\_\_\_\_

a) nesting

b) binding

**c) subquerying**

d) encapsulating

170. The union operation is represented by

a) ∩

**b) U**

c) –

d) \*

171. The intersection operator is used to get the \_\_\_\_\_ tuples.

a) Different

**b) Common**

c) All

d) Repeating

172. The union operation automatically \_\_\_\_\_\_\_\_\_\_, unlike the select clause.

a) Adds tuples

b) Eliminates unique tuples

c) Adds common tuples

**d) Eliminates duplicate**

173. If we want to retain all duplicates, we must write \_\_\_\_\_\_\_\_ in place of union.

**a) Union all**

b) Union some

c) Intersect all

d) Intersect some

174. (SELECT course id FROM SECTION WHERE semester = ’Fall’ AND YEAR= 2009)

EXCEPT (SELECT course id FROM SECTION WHERE semester = ’Spring’ AND YEAR= 2010);

This query displays

a) Only tuples from second part

b) Only tuples from the first part which has the tuples from second part

c) Tuples from both the parts

**d) Tuples from first part which do not have second part**

175. For like predicate which of the following is true.

i) % matches zero OF more characters.

ii) \_ matches exactly one CHARACTER.

**a) i-only**

b) ii-only

c) Both of the mentioned

d) None of the mentioned

176. The number of attributes in relation is called as its

a) Cardinality

**b) Degree**

c) Tuples

d) Entity

177. \_\_\_\_\_ clause is an additional filter that is applied to the result.

a) Select

b) Group-by

**c) Having**

d) Order by

178. The \_\_\_\_\_\_\_\_\_\_\_\_\_ is essentially used to search for patterns in target string.

**a) Like Predicate**

b) Null Predicate

c) In Predicate

d) Out Predicate

179. What does UNION operator do in a SQL statement?

a) Bring common data from the listed tables.

b) Bring data which is not common from the listed tables.

c) Bring all data from the listed tables.

**d) Bring all distinct from the listed tables.**

180. Which one is correct syntax for applying UNION operator?

a) SELECT column\_name(s) FROM table\_name1 UNION table\_name2

b) SELECT column\_name(s) FROM table\_name1

**UNION**

**SELECT column\_name(s) FROM table\_name2**

c) UNION SELECT column\_name(s) FROM table\_name1

SELECT column\_name(s) FROM table\_name2

d) SELECT FROM table\_name1 AND table\_name2

181. How can we get all records (redundant as well as non-redundant) from union operator?

**a) Using ‘ALL’ operator with UNION.**

b) Using ‘Distinct’ operator with UNION.

c) We get all records (redundant as well as non-redundant) with UNION operator by default.

d) None of the above.

182. The column names in the result of a UNION (of tables) are always equal to the column names in the 1st SELECT statement with the UNION operator, true or false?

**a) True**

b) False

183. Is UNION or UNION ALL operator valid for LONG data type column?

a) True

**b) False**

184. Can we use UNION operator in SELECT statement which contains TABLE collection expressions?

a) True

**b) False**

185. UNION operator requires an extra overhead of removing redundant rows, is it true?

**a) True**

b) False

186. What is true about order by with Union operator?

a) Order By can be issued in each result set.

**b) It can be issued for the overall result set.**

c) Both A & B.

d) None of the above

187. The result set will have Column names from the first query, correct?

**a) True**

b) False

188. If we know the records returned by our query are unique then which operator will be not used UNION or UNION ALL?

1. Union
2. **Union ALL**
3. INTERSECT
4. MINUS

189. \_\_\_\_\_\_ operator merges the result sets of two component queries with duplicate rows:

1. UNION
2. **UNION ALL**
3. INTERSECT
4. MINUS

190. \_\_\_\_\_\_\_\_\_\_ is used to combine the result from multiple SELECT statements into a single result set.

1. **UNION**
2. INTERSECT
3. MINUS
4. All of the above

191. The column names from the \_\_\_\_\_\_\_\_\_\_ SELECT statement are used as the column names for the results returned.

1. **first**
2. second
3. Both A and B
4. None of the above

192. What will be the output of the following statement?

mysql> SELECT REPEAT('a',1) UNION SELECT REPEAT('b',10);

1. **a**

**bbbbbbbbbb**

1. 1

10

1. a

b

1. None of the above

193. What will be the output of the following statement?

mysql> SELECT 1 UNION SELECT 'b';

1. **1**

**b**

1. 1

1

1. a

b

1. None of the above

194. What will be the output of the following statement?

mysql> SELECT null UNION SELECT null;

1. 0
2. 1
3. **null**
4. undefined

195. The default behaviour for UNION is that duplicate rows are removed from the result.

1. **true**
2. false

196. Select all deptno from EMP union select deptno from dept;

What will be the output of the given statement?

1. **Will display unique deptno**
2. Will display deptno including duplicates.
3. Will raise an error
4. None of the above.

197. Select deptno from EMP union all select deptno from dept;

What will be the output of the given statement?

1. Will display unique deptno
2. **Will display deptno including duplicates.**
3. Will raise an error
4. None of the above.

198. To apply ORDER BY to an individual SELECT, place the clause inside the \_\_\_\_\_\_ that enclose the SELECT

1. **()**
2. {}
3. []
4. None

199. To apply LIMIT to an individual SELECT, place the clause inside the \_\_\_\_\_\_that enclose the SELECT

1. **()**
2. {}
3. []
4. None

200. If we know the records returned by our query are unique then which operator will be used UNION or UNION ALL?

1. **Union**
2. Union ALL
3. INTERSECT
4. MINUS

201. \_\_\_\_\_\_ operator merges the result sets of two component queries with unique rows:

1. **UNION**
2. UNION ALL
3. INTERSECT
4. MINUS

202. An UNION operation to combine multiple result sets into one.

1. **True**
2. False

203. Union operator is a:

1. Unary Operator
2. Ternary Operator
3. **Binary Operator**
4. Not an operator

204. Which of the following returns all distinct rows selected by either query?

1. INTERSECT
2. MINUS
3. **UNION**
4. UNION ALL

206. Drop table is?

1. DML Statement
2. **DDL Statement**
3. Query Statement
4. None of the above

207. Which among the following is the correct syntax for modifying the definition of an existing table?

**a) ALTER TABLE person MODIFY person\_id INT AUTO\_INCREMENT;**

b) ALTER TABLE person person\_id INT AUTO\_INCREMENT;

c) ALTER TABLE person MODIFY person\_id;

d) ALTER TABLE person

208. In order to add a new column to an existing table in SQL, we can use the command

1. MODIFY TABLE
2. EDIT TABLE
3. **ALTER TABLE**
4. ALTER COLUMNS

209. Which keyword is used to specify the foreign key after the table is created?

a) SETUP

b) SET

**c) ALTER TABLE**

d) SPECIFY

210. Which of these commands will delete a table called ABC if you have appropriate authority:

1. **DROP TABLE ABC**
2. DROP ABC WHERE confirm = "YES"
3. DELETE ABC WHERE confirm = "YES"
4. DROP ABC

211. The command to eliminate a table from a database is:

1. REMOVE TABLE CUSTOMER;
2. **DROP TABLE CUSTOMER;**
3. DELETE TABLE CUSTOMER;
4. UPDATE TABLE CUSTOMER;

212. The DROP TABLE statement:

1. Deletes the table structure only.
2. **Deletes the table structure along with the table data.**
3. Works whether or not referential integrity constraints would be violated.
4. Is not an SQL statement.

213. What SQL command can be used to delete columns from a table?

1. MODIFY TABLE TableName DROP ColumnName
2. MODIFY TABLE TableName DROP COLUMN ColumnName
3. ALTER TABLE TableName DELETE ColumnName
4. **ALTER TABLE TableName DROP COLUMN ColumnName**

214. What SQL command can be used to delete columns from a table?

1. MODIFY TABLE TableName DROP ColumnName
2. MODIFY TABLE TableName DROP COLUMN ColumnName
3. **ALTER TABLE TableName DROP ColumnName**
4. ALTER TABLE TableName DELETE COLUMN ColumnName

215. What SQL command can be used to add columns to a table?

1. ALTER TABLE TableName APPEND ColumnName
2. **ALTER TABLE TableName ADD COLUMN ColumnName**
3. MODIFY TABLE TableName ADD ColumnName
4. MODIFY TABLE TableName ADD COLUMN ColumnName

216. What SQL command can be used to add columns to a table?

1. **ALTER TABLE TableName ADD ColumnName**
2. ALTER TABLE TableName APPEND COLUMN ColumnName
3. MODIFY TABLE TableName ADD ColumnName
4. MODIFY TABLE TableName ADD COLUMN ColumnName

217. The command to eliminate a table from a database is:

1. **DROP TABLE CUSTOMER;**
2. DELETE TABLE CUSTOMER;
3. REMOVE TABLE CUSTOMER;
4. UPDATE TABLE CUSTOMER;

218. ALTER TABLE can be used to update or modify values of a column.

1. **True**
2. False

219. ALTER table command, can be used to add constraints to a table.

1. **True**
2. False

220. ALTER table command, can be used to add foreign key constraints to a table.

1. **True**
2. False

221. ALTER table command, allows renaming an existing table

1. **True**
2. False

223. Change Keywords of Alter table allows you to:

1. Drop the table.
2. **Change Name of Column**
3. Drop the column constraint
4. None of the above.

224. Change Keywords of Alter table allows you to:

1. Drop the table.
2. **Change Column Data Type**
3. Drop the column constraint
4. None of the above.

225. What command you will issue to add a new column in the STUDENT table?

1. **Alter table STUDENT ADD COLUMN lastName varchar(25)**
2. Alter table STUDENT ADDING COLUMN lastName varchar(25)
3. Alter table STUDENT ADDED COLUMN lastName varchar(25)
4. None of the above.

226. What command you will issue to add a new column in the STUDENT table?

1. **Alter table STUDENT ADD lastName varchar(25)**
2. Alter table STUDENT ADDING COLUMN lastName varchar(25)
3. Alter table STUDENT ADDED COLUMN lastName varchar(25)
4. None of the above.

227. The command you will use to DROP the CUSTOMER relation:

1. **DROP TABLE CUSTOMER;**
2. DELETE TABLE CUSTOMER;
3. REMOVE TABLE CUSTOMER;
4. UPDATE TABLE CUSTOMER;

228. DROP TABLE CUSTOMER will eliminate the:

1. **Table**
2. Column
3. Rows
4. Constraints

229. Which command you will issue to provide a primary key after the table is created?

a) SETUP

b) SET

**c) ALTER TABLE**

d) SPECIFY

230. Which command you will issue to provide a NOT NULL to a column after the table is created?

a) Change Table

b) Modify Table

**c) ALTER TABLE**

d) None of the above

231. Which of the following is DDL?

1. Drop
2. Alter
3. Delete
4. **Both A and B**

232. DROP is a \_\_\_\_\_\_\_\_\_\_\_\_\_\_ statement in SQL.

1. Query
2. Embedded SQL
3. **DDL**
4. DCL

233. The clause alter table in SQL can be used to

1. add an attribute
2. delete an attribute
3. alter the default values of an attribute
4. **all of the above**

234. Which of the following example creates users table whose PRIMARY KEY is user\_id column:

1. CREATE TABLE users (USER\_ID INT AUTO\_INCREMENT PRIMARY KEY, USERNAME VARCHAR (40));
2. CREATE TABLE users(USER\_ID INT AUTO\_INCREMENT, USERNAME VARCHAR (40),

PRIMARY KEY (USER\_ID));

1. CREATE TABLE users(USER\_ID INT AUTO\_INCREMENT, USERNAME VARCHAR (40), CONSTRAINT PK\_USER\_ID PRIMARY KEY (USER\_ID));
2. **All of the above.**

235. Can I define multiple unique constraints on a table?

1. **Yes**
2. No

236. Which of the following is NOT a type of SQL constraint?

1. PRIMARY KEY
2. FOREIGN KEY
3. **ALTERNATE KEY**
4. UNIQUE

237. The value of Primary key

1. can be duplicated
2. can be null
3. **cannot be null**
4. none of these

238. In an RDBMS relationship between tables are created by using

1. Alternate Key and Foreign Key
2. **Primary key and Foreign Key**
3. Candidate Key and Foreign Key
4. Composite Key and Foreign Key

239. In the following code, InnoDB is \_\_\_\_\_\_\_\_\_\_

CREATE TABLE student (Name CHAR (30), Student\_id INT, PRIMARY KEY (student\_id)) ENGINE = InnoDB;

a) Database name

b) Table name

c) Reference engine

**d) Storage engine**

240. Columns that are part of a PRIMARY KEY are made NOT NULL even if not declared that way

**a) True**

b) False

241. What is the role of “CONSTRAINS” in defining a table in Mysql?

a) Declaring primary key

b) Declaring Foreign Key

c) Restrictions on columns

**d) All of the mentioned**

242. Find out the logical error in the following query?

CREATE TABLE person (Person\_id VARCHAR (20), Name VARCHAR (20), Address VARCHAR (20), Mobile\_no SMALLINT);

a) Lesser number of columns

b) Incorrect definition

**c) Primary key is missing**

d) None of the mentioned

243. What is meaning of “REFERENCES” in table definition?

a) Primary key

b) NULL

c) Foreign Key

**d) A ”foreign Key” belong to this particular table**

244. What is the role of “CONSTRAINS” in defining a table in Mysql?

a) Declaring primary key

b) Declaring Foreign Key

c) Restrictions on columns

**d) All of the mentioned**

245. In the following query, what does “person\_id” stands for?

CREATE TABLE person (Person\_ id SMALLINT, Fname VARCHAR (20), Lname VARCHAR (20), CONSTRAINT pk\_person PRIMARY KEY (person\_id));

a) Normal attribute of the table

b) Supreme key

c) Composite key

**d) Primary key**

246. In the following query “person\_id” can be

SELECT person\_id, fname, l name, Birth\_date FROM person

WHERE person\_id=1;

a) Only Primary Key

**b) Primary Key or any other Attribute**

c) Only attribute but not a primary Key

d) None of the mentioned

247. Indexes are created in conjunction with \_\_\_\_\_\_\_\_constraints.

1. **Primary key**
2. Check constraint
3. Not null
4. None of the above

248. Indexes are created in conjunction with \_\_\_\_\_\_\_\_ constraints.

1. Check constraint
2. **Unique**
3. Not null
4. None of the above

249. Which of the following objects are dropped automatically when a table is dropped.

1. Procedure
2. **Constraints**
3. Views
4. Synonyms

250. A table can have more than one primary key

1. True
2. **False**

251. The keys that can have NULL values are

1. Primary Key
2. Unique Key
3. Foreign Key
4. **Both b and c**

252. The UNIQUE and FOREIGN keys cannot have NULL values

1. true
2. **false**

253. Which of the following constraint does not enforce uniqueness?

a) UNIQUE  
b) Primary key  
**c) Foreign key**  
d) None of the mentioned

254. Term that is known to commit current transaction, is

1. Rollback work
2. **Commit work**
3. Trace work
4. Transit work

255. In SQL, which command(s) is (are) used to change a table's storage characteristics?

1. **ALTER TABLE**
2. MODIFY TABLE
3. CHANGE TABLE
4. All of the above

256. To include integrity constraint in an existing relation use:

a) Create table

b) Modify table

**c) Alter table**

d) Drop table

257. Which of the following is not an integrity constraint?

a) Not null

**b) Positive**

c) Unique

d) Check ‘predicate’

258. CREATE TABLE Employee (Emp\_id NUMERIC NOT NULL, Name VARCHAR (20), dept\_name VARCHAR (20), Salary NUMERIC UNIQUE (Emp\_id, Name));

INSERT INTO Employee VALUES (1002, Ross, CSE, 10000)

INSERT INTO Employee VALUES (1006, Ted, Finance,);

INSERT INTO Employee VALUES (1002, Rita, Sales, 20000);

What will be the result of the query?

a) All statements executed

b) Error in create statement

**c) Error in insert into Employee values (1006, Ted, Finance, );**

d) Error in insert into Employee values (1008, Ross, Sales, 20000);

259. CREATE TABLE Manager (ID NUMERIC, Name VARCHAR (20), budget NUMERIC, Details VARCHAR (30));

In order to ensure that the value of budget is non-negative which of the following should be used?

**a) Check (budget>0)**

b) Check (budget<0)

c) Alter (budget>0)

d) Alter (budget<0)

260. Foreign key is the one in which the \_\_\_\_\_\_\_\_ of one relation is referenced in another relation.

a) Foreign key

**b) Primary key**

c) References

d) Check constraint

261. CREATE TABLE course (. . . FOREIGN KEY (dept name) REFERENCES department . . . );

Which of the following is used to delete the entries in the referenced table when the tuple is deleted in course table?

a) Delete

**b) Delete cascade**

c) Set null

d) All of the mentioned

262. Data integrity constraints are used to:

a) Control who is allowed access to the data

b) Ensure that duplicate records are not entered into the table

**c) Improve the quality of data entered for a specific property (i.e., table column)**

d) Prevent users from changing the values stored in the table

263. How can a SQL developer add a key on a table?

a) While creating a table

b) With Alter table command

**c) All of the Mentioned**

d) None of the above

264. What is true about Unique and primary key?

1. **Unique can have multiple NULL values but Primary can’t have.**
2. Unique can have single NULL value but Primary can’t have even single.
3. Both can have duplicate values
4. None of the Mentioned.

265. Use of UNIQUE while defining an attribute of a table in SQL means that the attribute values are

1. **distinct values**
2. cannot have NULL
3. both (A) & (B)
4. same as primary key

266. Use of UNIQUE and NOT NULL while defining an attribute of a table in SQL means that the attribute values are

1. distinct values
2. cannot have NULL
3. **both (A) & (B)**
4. same as foreign key

267. Primary key in a relation R is always associated with an INDEX object

1. **True**
2. False

268. What is a view?

1. A view is a special stored procedure executed when certain event occurs.
2. **A view is a virtual table which results of executing a pre-compiled query. A view is not part of the physical database schema, while the regular tables are.**
3. A view is a database diagram.
4. None of these

269. What is an SQL virtual table that is constructed from other tables?

1. **View**
2. A relation
3. Just another table
4. Query results

270. Which of the following is not a limitation of view?

a) ORDER BY Does Not Work

**b) Index Created on View Used Often**

c) Cross Database Queries Not Allowed in Indexed View

d) Adding Column is Expensive by Joining Table outside View

271. Which of the following statement is true?

a) Views could be looked as an additional layer on the table which enables us to protect intricate or sensitive data based upon our needs

b) Views are virtual tables that are compiled at run time

c) Creating views can improve query response time

**d) All of the Mentioned**

272. SQL Server has mainly how many types of views?

a) one

**b) two**

c) three

d) four

273. Dynamic Management View is a type of

**a) System Defined Views**

b) User Defined View

c) Simple View

d) Complex View

274. Syntax for creating views is

**a) CREATE VIEW AS SELECT**

b) CREATE VIEW AS UPDATE

c) DROP VIEW AS SELECT

d) CREATE VIEW AS UPDATE

275. You can delete a view with \_\_\_\_\_\_\_\_\_\_\_ command.

**a) DROP VIEW**

b) DELETE VIEW

c) REMOVE VIEW

d) TRUNCATE VIEW

276. \_\_\_\_\_\_\_\_\_\_\_ is stored only in the Master database.

a) Database-scoped Dynamic Management View

b) Complex View

c) Catalos View

**d) None of the mentioned**

277. Which of the following creates a virtual relation for storing the query?

a) Function

**b) View**

c) Procedure

d) None of the mentioned

278. Which of the following is the syntax for views where v is view name?

a) Create view v as “query name”;

b) Create “query expression” as view;

**c) Create view v as “query expression”;**

d) Create view “query expression”;

279. SELECT course\_id FROM physics\_fall\_2009 WHERE building = ’Watson’;

Here the tuples are selected from the view. Which one denotes the view?

a) Course\_id

b) Watson

c) Building

**d) physics\_fall\_2009**

280. Updating the value of the view

**a) Will affect the relation from which it is defined**

b) Will not change the view definition

c) Will not affect the relation from which it is defined

d) Cannot determine

281. SQL view is said to be updatable (that is, inserts, updates or deletes can be applied on the view) if which of the following conditions are satisfied by the query defining the view?

a) The from clause has only one database relation

b) The query does not have a group by or having clause

c) The select clause contains only attribute names of the relation, and does not have any expressions, aggregates, or distinct specification

**d) All of the mentioned**

281. Which of the following is used at the end of the view to reject the tuples which do not satisfy the condition in where clause?

a) With

b) Check

**c) With check**

d) All of the mentioned

283. Consider the two relations instructor and department

ID Name Dept\_name Salary

1001 Ted Finance 10000

1002 Bob Music 20000

1003 Ron Physics 50000

Department:

Dept\_name Building Budget

Biology Watson 40000

Chemistry Painter 30000

Music Taylor 50000

Which of the following is used to create view for these relations together?

**a) CREATE VIEW instructor\_info AS SELECT ID, name, building FROM instructor, department WHERE instructor.dept name= department.dept name;**

b) CREATE VIEW instructor\_info SELECT ID, name, building FROM instructor, department;

c) CREATE VIEW instructor\_info AS SELECT ID, name, building FROM instructor;

d) CREATE VIEW instructor\_info AS SELECT ID, name, building FROM department;

284. For the view CREATE VIEW INSTRUCTOR\_INFO AS SELECT ID, NAME, BUILDING FROM INSTRUCTOR, DEPARTMENT WHERE INSTRUCTOR.DEPT NAME = DEPARTMENT.DEPT NAME;

If we insert tuple into the view as insert into instructor info values (’69987’, ’White’, ’Taylor’);

What will be the values of the other attributes in instructor and department relations?

a) Default value

**b) Null**

c) Error statement

d) 0

285. CREATE VIEW faculty AS SELECT ID, name, dept name FROM instructor;

Find the error in this query.

a) Instructor

b) Select

c) View …as

**d) None of the mentioned**

286. Which one is not applicable while querying on a view?

a) GROUP BY

b) SELECT

c) ORDER BY

**d) All can be given**

287. Refer below query which leads to create a view named vwEmployee.

CREATE vwEmployee VIEW AS SELECT nothing FROM dbo.Employee WHERE ID < 100

Now, tell the problem in query?

a) Above query is correct.

**b) View name must be after keyword view and ‘nothing’ is not a keyword, so should be replaced with \*.**

c) Replace nothing with view name.

d) Replace nothing with column names.

288. How can you drop more than one View in single command?

a) Drop viewname1 + viewname2 + viewname (n);

b) Drop viewname1; Drop viewname2; Drop viewname (n);

c) Drop viewname1; viewname2; viewname (n);

**d) Drop viewname1, viewname2, viewname (n);**

289. Views are also called as:

a) Complex tables

b) Simple tables

**c) Virtual tables**

d) Actual Tables

290. Are views stored in Databases?

1. **Yes**
2. No

291. Can we show computed values in views from different columns of a table?

1. **Yes**
2. No

292. The clause in SQL that specifies that the query result should be sorted in ascending or descending order based on the values of one or more columns is

1. **View**
2. Order by
3. Group by
4. Having

293. SQL views can be used to hide:

1. Columns and rows only.
2. Complicated SQL syntax only.
3. Both of the above can be hidden by an SQL view.
4. None of the above is correct.

294. The SQL statement to create a view is:

1. **CREATE VIEW.**
2. MAKE VIEW.
3. SELECT VIEW.
4. INSERT VIEW.

295. For what purposes are views used?

1. To hide columns only
2. To hide rows only
3. To hide complicated SQL statements only
4. **All of the above are uses for SQL views.**

296. What is an SQL virtual table that is constructed from other tables?

1. Just another table
2. **A view**
3. A relation
4. Query results

297. A reason for using an SQL view to hide columns is:

1. To simplify a result only.
2. To prevent the display of sensitive data only.
3. **To accomplish both of the above.**
4. None of the above are reasons for using an SQL view.

298. What does 'firstName' represent in the following code snippet?

CREATE TABLE STUDENT (firstName VARCHAR (30), Roll\_num INT, Address CHAR (30), Phone CHAR (12));

a) A table

b) A row

**c) An attribute**

d) An object

299. CREATE SCHEMA is a synonym for \_\_\_\_\_\_\_\_\_\_\_ as of mysql 5.0.2.

1. **CREATE DATABASE**
2. CREATE TABLE
3. CREATE VIEW
4. None of the above

300. In the following code, InnoDB is \_\_\_\_\_\_\_\_\_\_

CREATE TABLE STUDENT (Name CHAR (30), Student\_id INT, PRIMARY KEY (student\_id)) ENGINE = InnoDB;

a) Database name

b) Table name

c) Reference engine

**d) Storage engine**

301. How many attributes are there in employee relation?

CREATE TABLE employee (Emp\_name CHAR (30), Emp\_id INT);

a) 30

b) 1

**c) 2**

d) 3

302. What does ‘abc’ & ‘xyz’ specify?

CREATE TABLE ABC (xyz);

**a) Table name and column specs**

b) Column specs and table name

c) Table name and number of columns

d) Table name and number of rows

303. In the CREATE TABLE statement, the engine name specified is case sensitive.

a) True

**b) False**

304. What can be used to check for views that have been invalidated by DROP or ALTER operations?

a) CREATE TABLE

b) VERIFY TABLE

c) DETAILS TABLE

**d) CHECK TABLE**

305. Which among the following is the correct syntax for creating table?

**a) CREATE TABLE name;**

b) CREATE name;

c) CREATE TABLE

d) All of the mentioned

306. SQL allows comparison operations on data type’s i.e.

1. time
2. date
3. int
4. **All datatypes**

307. The SQL ALTER statement can be used to:

1. Change the table data.
2. **Change the table structure.**
3. Delete rows from the table.
4. Add rows to the table.

308. You can use \_\_\_\_\_\_\_\_\_ any time, if you forget the names of the columns in your table or what types they have.

1. DESC <tname>
2. DESCRIBE <tname>
3. EXPLAIN <tname>
4. **All of the above.**

309. The MySQL CREATE TABLE command is used to create a new table into the database. A table creation command requires?

1. Name of the table
2. Names of fields
3. Datatypes for each field
4. **All of the above**

310. The table name can be specified as db\_name.tbl\_name to create the table in a specific database.

1. **True**
2. False

311. To create one table from another, add a SELECT statement at the end of the CREATE TABLE statement.

1. **True**
2. False

312. You can use the \_\_\_\_\_\_\_\_\_\_ keyword when creating a temporary table.

1. TEMP
2. **TEMPORARY**
3. Not Permanent
4. None of the above.

313. A TEMPORARY table is visible only within the current session

1. **True**
2. False

314. A TEMPORARY is dropped automatically when the session is closed.

1. **True**
2. False

315. To create one table from another, add a \_\_\_\_\_\_\_ statement at the end of the CREATE TABLE statement.

1. Update
2. **Select**
3. Fetch
4. None of the above.

316. Use \_\_\_\_\_\_\_\_\_ to create an empty table based on the definition of another table, including any column attributes and indexes defined in the original table.

1. **CREATE TABLE ... LIKE**
2. CREATE TABLE ... FROM
3. CREATE TABLE ... INTO
4. None of the above.

318. If neither NULL nor NOT NULL is specified, the column is treated as though \_\_\_\_\_\_ had been specified.

1. **NULL**
2. NOT NULL
3. DEFAULT
4. UNIQUE

319. What is the meaning of “Temporary Tables” in Mysql?

1. Rows returned by sub query
2. Permanent tables
3. **Virtual tables**
4. All of the mentioned

320. What is the true about the following statement?

CREATE TABLE TEMP (1C INT);

1. **Table will be created**
2. Invalid table name
3. Invalid column name
4. None of the above

321. What is the true about the following statement?

CREATE TABLE TEMP (\_C1 INT);

1. **Table will be created**
2. Invalid table name
3. Invalid column name
4. None of the above

322. What is the true about the following statement?

CREATE TABLE TEMP ($C1 INT);

1. **Table will be created**
2. Invalid table name
3. Invalid column name
4. None of the above

323. Which of these take more space?

Variable length columns, fixed length columns

a) Variable length

b) Same

**c) Fixed length**

d) Indeterminate

324. Which of these is not a valid name for a column

1. To
2. **From**
3. Far
4. Near

325. SQL permits attribute names to be repeated in the same relation.

1. **FALSE**
2. TRUE

326. The command used to delete a particular column in a relation is \_\_\_\_\_\_\_\_\_\_\_\_

1. UPDATE TABLE
2. TRUNCATE COLUMN
3. **ALTER, DROP**
4. DELETE COLUMN

327. The command used to add multiple column in a relation is \_\_\_\_\_\_\_\_\_\_\_\_

1. Alter table temp add column(c3, c4);
2. Alter table temp add column(c3 int), add column (c4 int);
3. **Alter table temp append column(c3 int), add column (c4 int);**
4. All of the above

328. The command used to change the datatype of a particular column in a relation is \_\_\_\_\_\_\_\_\_\_\_\_

1. UPDATE TABLE
2. TRUNCATE COLUMN
3. **ALTER, MODIFY**
4. DELETE COLUMN

329. The command used to rename a particular column in a relation is \_\_\_\_\_\_\_\_\_\_\_\_

1. UPDATE TABLE
2. TRUNCATE COLUMN
3. **ALTER, CHANGE**
4. DELETE COLUMN

330. The command used to delete multiple column in a relation is \_\_\_\_\_\_\_\_\_\_\_\_

1. UPDATE TABLE
2. ALTER MULTIPLE DROP
3. **ALTER, DROP**
4. DELETE COLUMN

331. The command used to delete multiple column in a relation is \_\_\_\_\_\_\_\_\_\_\_\_

1. ALTER TABLE T DROP COLUMN (C3 AND C10);
2. ALTER TABLE T DROP COLUMN (C3) AND (C10);
3. **ALTER TABLE T DROP COLUMN C3, DROP COLUMN C10;**
4. All of the above

332. The SQL ALTER statement can be used to:

1. **Change the table structure.**
2. Change the table data.
3. Add rows to the table.
4. Delete rows from the table.

333. How will you create a table by the name "New Student"?

1. Create table "New Student" (...)
2. Create table 'New Student' (...)
3. Create table [New Student] (...)
4. **Create table `New Student` (...)**

334. Identify the table name in the following statement.

INSERT INTO STUDENT VALUES ('Kyle', 'M', NULL);

**a) Student**

b) Values

c) Kyle

d) M

335. Identify the column name where the NULL data is passed.

INSERT INTO STUDENT (ID, Gender, City) VALUES ('Kyle', 'M', NULL);

a) ID

b) Gender

**c) City**

d) None of the above

336. Which Statement is used to insert the values in the table?

**a) INSERT INTO**

b) Insert

c) INSERT

d) None of the mentioned

337. In the following query, what does “person” stands for?

INSERT INTO person (person\_id, fname, lname) VALUES (1, 'S', 'P');

a) Composite attributes

b) Multivalued attributes

**c) Table name**

d) None of the mentioned

338. Will this query produce any error?

INSERT INTO person (person\_id, fname, lname) VALUES (1, 'S', 'U'), (1, 'T', 'U');

/\* where person\_id is a primary key \*/

**a) Error**

b) No Error

c) Depends

d) None of the mentioned

339. Will this query produce any error?

Note: person relation has no records.

INSERT INTO person (person\_id, fname, lname) VALUES (1, 'S', 'U'), (2, 'T', 'U');

/\* where person\_id is a primary key \*/

a) Error

**b) No Error**

c) Depends

d) None of the mentioned

340. “INSERT” is same as “UPDATE”?

**a) NO**

b) YES

c) May be

d) None of the mentioned

341. Which SQL statement is used to insert new data in a database?

1. INSERT NEW
2. **INSERT INTO**
3. ADD NEW
4. ADD ROW

342. Which of the following can add a row to a table?

1. Add
2. **Insert**
3. Update
4. Alter

343. Which SQL statement is used to insert a new data in a Student database?

1. **INSERT INTO student….**
2. INSERT IN student….
3. ADD student….
4. INSERT ALL student….

344. Insert into instructor values (10211, 'Smith', 'Biology', 66000); what type of statement is this?

1. Query
2. **DML**
3. Relational
4. DDL

345. Which is right statement to insert multiple records?

1. **INSERT INTO job\_question VALUES ('col1', 'col2'), ('col1', 'col2'), ...**
2. INSERT INTO job\_question VALUES ('col1', 'col2'), VALUES ('col1', 'col2'), ...
3. A and B Both
4. None of the above

346. Weather the following insert statement is valid or invalid?

DEPT (ID, NAME, LOC)

INSERT INTO DEPT VALUES ((SELECT 1), 'A ', 'B ');

1. **valid**
2. invalid
3. no idea
4. None of the above.

347. Is it necessary to insert the value in each column of the table?

a) Yes

**b) No**

c) Depends on the server

d) Depends on the usage of the table

348. Which SQL keyword is used to sort the result-set?

1. SORT BY
2. ORDER
3. **ORDER BY**
4. SORT

349. UPDATE is a \_\_\_\_\_\_\_\_\_\_ statement that modifies rows in a table.

1. **DML**
2. DDL
3. DCL
4. None of the above.

350. In given table TEMP (C1 INT, C2 INT)

C1 C2

----- -----

1 1

What will be the value of C1 and C2 column after giving the following update statement?

UPDATE T SET C1 = C1 + 1, C2 = C1;

1. 1 1
2. **2 2**
3. 1 2
4. None

351. In given table TEMP (C1 INT, C2 INT)

C1 C2

----- -----

1 1

What will be the value of C1 and C2 column after giving the following update statement?

UPDATE T SET C2 = C1, C1 = C1 + 1;

1. 1 1
2. **2 1**
3. 1 2
4. None

352. In given table TEMP (C1 INT, C2 INT)

C1 C2

----- -----

1 1

What will be the value of C1 and C2 column after giving the following update statement?

UPDATE T SET C2 = C1, C1 = C1 + 1, C2 = C1;

1. 1 1
2. 2 1
3. 1 2
4. **2 2**

353. Is “UPDATE TEMP SET C1 = 10, SET C2 = 10” statement a valid statement?

1. Yes
2. **No**
3. Can’t say
4. None

354. In given table TEMP (C1 INT PRIMARY KEY, C2 INT)

C1 C2

----- -----

2 1

1 1

3 1

What will be the value of C1 column after giving the following update statement?

UPDATE TEMP SET C1 = C1 + 1;

1. [3, 2, 4]
2. [2, 1, 4]
3. **ERROR 1062 (23000): Duplicate entry '3' for key 'PRIMARY'**
4. None of the above

355. In given table TEMP (C1 INT PRIMARY KEY, C2 INT)

C1 C2

----- -----

2 1

1 1

3 1

What will be the value of C1 column after giving the following update statement?

UPDATE TEMP SET C1 = C1 + 1 ORDER BY C1 DESC;

1. **[3, 2, 4]**
2. [2, 1, 4]
3. ERROR 1062 (23000): Duplicate entry '3' for key 'PRIMARY'
4. None of the above

356. In given table TEMP (C1 INT PRIMARY KEY, C2 INT, C3 INT)

C1 C2 C3

---- ---- -------

2 1 NULL

1 1 NULL

3 1 NULL

What will be the value of C3 column after giving the following update statement?

UPDATE TEMP SET C3 = IF (C1 = 1,100, IF (C1=2, 200,'300'));

1. [100, 200, 300]
2. **[200, 100, 300]**
3. ERROR 1062 (23000): Duplicate entry '3' for key 'PRIMARY'
4. None of the above

357. In given table TEMP (C1 INT PRIMARY KEY, C2 INT, C3 INT)

C1 C2 C3

---- ---- -------

2 1 200

1 1 100

3 1 300

What command will you issue, if you what to update column C3 to NULL.

1. **UPDATE TEMP SET C3 = NULL**
2. UPDATE TEMP SET C3 IS NULL
3. UPDATE TEMP SET C3 IS ''
4. None of the above.

358. What command you will issue to update first 5 records?

1. UPDATE TEMP SET C3 = 1 where C1 <=5;
2. UPDATE TEMP SET C3 = 1 limit C1 <=5;
3. **UPDATE TEMP SET C3 = 1 limit 5;**
4. None of the above.

359. If you omit the \_\_\_\_\_\_\_\_\_\_\_ clause, all records in the table will be updated!

1. **WHERE**
2. HAVING
3. Either A or B
4. None of the above.

360. Qualifying the name of column with the table name is not necessary in single-table updates.

**a) True**

b) False

361. What is the value of val2?

UPDATE t SET val1 = val1 + 2, val2 = val1;

a) Previous val1

**b) Updated val1**

c) Unchanged

d) val1 + 1

362. UPDATE statement is a DML statement. What does DML stand for?

**a) Data Manipulation Language**

b) Data Manipulation Level

c) Data Markup Language

d) Data Markup Level

363. Which keyword in the UPDATE statement is used to assign values to columns?

a) ASSIGN

**b) SET**

c) MARK

d) GET

364. In given table TEMP (C1 INT PRIMARY KEY, C2 INT, C3 INT)

C1 C2 C3

---- ---- -----

2 1 0

1 1 0

3 1 0

What will be the value of C3 column after giving the following update statement?

UPDATE TEMP SET C3= 1.56 WHERE C1 = 1;

1. 1
2. 1.56
3. **2**
4. None of the above.

365. Which SQL statement is used to update data in a database?

1. UPDATE from
2. **UPDATE**
3. SAVE AS
4. MODIFY

366. DELETE FROM tbl\_name does not regenerate the table but instead deletes all rows, one by one.

1. **True**
2. False

367. If you do not specify a list of column names for INSERT ... VALUES, values for every column in the table must be provided.

1. **True**
2. False

368. Which keyword is used to delete all the rows from the table?

**a) TRUNCATE**

b) REMOVE

c) DELETE ALL

d) CLEAR

369. You want to delete a record from parent table and if child records are present then?

1. **The statement will raise an error.**
2. Parent record will be deleted.
3. Parent and Child record will be deleted.
4. None of the above.

370. Deletion of an employee from table also deletes that employee from another table. This kind of delete is called \_\_\_\_\_\_\_\_\_\_\_\_

a) Transparent

b) Concrete

c) Elaborate

**d) Cascaded**

371. Which statement is used to delete an existing row from the table?

**a) DELETE**

b) WHERE

c) MODIFY

d) None of the mentioned

372. In the following query how many rows will be deleted?

DELETE PERSON WHERE PERSON\_ID=1;

/\*person\_id is a primary key \*/

**a) 1**

b) 0

c) No row

d) None

373. In the following query how many rows will be deleted?

DELETE PERSON WHERE PERSON\_ID<10;

/\*person\_id is a primary key \*/

**a) 0-9**

b) 1-7

c) No row

d) None

374. Which statement is used to delete an existing row from the table?

**a) DELETE**

b) WHERE

c) MODIFY

d) None of the mentioned

375. In the following query how many rows will be deleted?

DELETE person WHERE person\_id=1 or person\_id=2;

/\*person\_id is a primary key \*/

a) 1

**b) 2**

c) No row

d) None

376. In the following query how many rows will be deleted?

DELETE person WHERE person\_id=1 and person\_id=2;

/\*person\_id is a primary key \*/

a) 1

b) 2

**c) No row**

d) None

377. In the following query how many rows will be deleted?

DELETE person WHERE person\_id=1 and person\_id=1;

/\*person\_id is a primary key \*/

**a) 1**

b) 2

c) No row

d) None

378. In the following query how many rows will be deleted?

DELETE person WHERE person\_id<10;

/\*person\_id is a primary key \*/

**a) 0-9**

b) 1-7

c) No row

d) None

379. The number of rows in the table is 10. Suppose all rows are deleted using truncate. The new row starts with sequence number \_\_\_\_\_\_\_\_\_\_\_\_\_

a) 11

**b) 1**

c) 100

d) 101

380. In a MyISAM table, if the maximum value of an AUTO\_INCREMENT increment column is 12 and that row is deleted, the next value generated is \_\_\_\_\_\_\_\_\_\_\_\_\_

a) 12

**b) 13**

c) 1

d) 14

381. In InnoDB tables, when a table is emptied with the TRUNCATE TABLE, the counter begins at \_\_\_\_\_\_\_\_\_\_\_\_\_

a) 0

**b) 1**

c) -1

d) Arbitrary

382. Which SQL statement is used to delete data from a database table?

1. COLLAPSE
2. **DELETE**
3. MODIFY
4. REMOVE

383. Which SQL statement is used to delete data FROM a database?

1. DELETE INTO …
2. **DELETE FROM …**
3. DELETE AS …
4. DELETE IN …

384. The command to remove rows from a table 'CUSTOMER' is:

1. DROP FROM CUSTOMER ...
2. UPDATE FROM CUSTOMER ...
3. REMOVE FROM CUSTOMER ...
4. **DELETE FROM CUSTOMER WHERE ...**

385. Aliases are case sensitive.

1. **True**
2. False

386. Examine the following statement.

SELECT fName AS 'First Name' FROM EMPLOYEE.

What is the meaning of 'First Name'?

1. Column Name
2. Table Name
3. **Alias Name**
4. None of the above

387. You can rename …………… temporarily by giving another name known as Alias.

1. a table
2. a column
3. a view
4. **All of the above**

388. Relation R(col1, col2)

R = {1, 2}

What is the value of col2?

UPDATE R SET COL2 = COL1, COL1 = COL1 + 5;

a) {1, 2}

b) {1, 6}

**c) {6, 1}**

d) None of the above

389. Relation R(col1, col2)

R = {1, 2}

What is the value of col2?

UPDATE R SET COL1 = COL1 + 5, COL2 = COL1;

a) {1, 2}

b) {1, 6}

**c) {6, 6}**

d) None of the above

390. Character data can be stored as

a) Fixed length string

b) Variable length string

**c) Either Fixed or Variable length string**

d) None of the mentioned

391. Which declaration represents that “character data will consume the same number of bytes as declared and is right padded”?

**a) Char**

b) Varchar

c) Both Char and Varchar

d) None of the mentioned

392. Which declaration doesn’t use the same number of bytes and consumption of bytes depends on the input data?

**a) Varchar**

b) Char

c) Both Varchar and Char

d) None of the mentioned

393. The maximum length of the char columns is

**a) 255 bytes**

b) 65, 535 bytes

c) 256 bytes

d) None of the mentioned

394. The maximum length of the varchar columns is

**a) Upto 65, 535 bytes**

b) Upto 256 bytes

c) Upto 65, 567 bytes

d) None of the mentioned

395. In MySQL database variable length column is declared by

**a) Varchar**

b) Varchar 3

c) Varchar2

d) None of the mentioned

396. An integer or floating-point column can have the additional attribute AUTO\_INCREMENT.

1. **True**
2. False

397. An AUTO\_INCREMENT attribute on a column can be given only on \_\_\_\_\_\_ datatype.

1. Integer
2. Floating-point
3. **Either A or B**
4. None of the above.

398. Which “text type” has the maximum number of bytes?

a) Tiny text

b) Text

c) Medium text

**d) Long text**

399. Which among the following have the maximum bytes?

a) Varchar

b) Char

**c) Text type**

d) Both Varchar and Char

400. What will happen if the data being loaded into a text column exceeds the maximum size of that type?

a) Extra memory will be allocated

b) Process terminate

**c) Data will be truncated**

d) Depend on the system

401. Which data type is more suitable for storing “small notes” in Mysql?

a) Char

b) Varchar

**c) Mediumtext**

d) Varchar2

402. Which data type is more suitable for storing “documents” in Mysql?

a) Varchar

**b) Longtext**

c) Varchar2

d) Either Varchar or Longtext

403. “Numeric Data” is used to store

a) Whole numbers

b) Natural numbers

c) Rational numbers

**d) Both Whole and Natural numbers**

404. Which Numeric Data type has the largest range?

a) Mediumint

b) Smallint

**c) Int**

d) Tinyint

405. What will be the storage pattern for “float(4, 2)” in Mysql?

**a) Total of four digits, two to the left of decimal and two to the right of decimal**

b) Total of six digits

c) Total of four digits, not distributed uniformly

d) None of the mentioned

406. Which among the following are the correct representation of “float(4, 2)”?

a) 24.33

b) 124.4

c) 12.123

**d) Both 24.33 and 12.123**

407. Which among the following is the correct representation of “float(5,0)”?

a) 12345.123

b) 12345.1

**c) 12345**

d) 123.123

408. Which among the following is the correct representation of “float(1, 1)”?

**a) Error**

b) Total of 1 digit

c) Total of 2 digit, one digit at right of the decimal, one digit at left of the decimal

d) None of the mentioned

409. There can be only \_\_\_ AUTO\_INCREMENT column per table

1. **1**
2. 2
3. 3
4. 4

410. AUTO\_INCREMENT column must be indexed

1. **True**
2. False

411. AUTO\_INCREMENT column cannot have a DEFAULT value.

1. **True**
2. False

412. An AUTO\_INCREMENT column works properly only if it contains only positive values.

1. **True**
2. False

413. Which of the following is a valid SQL type?

1. CHAR
2. NUMERIC
3. FLOAT
4. **All of the above**

414. Triggers enable to enforce data integrity constraints.

**a) True**

b) False

415. Which statement is used to create a trigger?

**a) CREATE TRIGGER**

b) CREATE TRIGGERS

c) PRODUCE TRIGGER

d) PRODUCE TRIGGERS

416. For which of the following are triggers not supported?

a) Delete

b) Update

c) Insert

**d) Views**

417. Which statement is used to remove a trigger?

a) REMOVE

b) DELETE

**c) DROP**

d) CLEAR

418. Triggers are invoked automatically by the server.

**a) True**

b) False

419. ITERATE means

1. **"Start the loop again".**
2. "Stop the loop again".

420. What is abc in the following statement?

CREATE TRIGGER abc (...) (...) ON def FOR EACH ROW ghi;

**a) Trigger name**

b) Table name

c) Trigger statement

d) Update statement

421. What is def in the following statement?

CREATE TRIGGER abc (...) (...) ON def FOR EACH ROW ghi;

a) Trigger name

**b) Table name**

c) Trigger statement

d) Update statement

422. What is ghi in the following statement?

CREATE TRIGGER abc (...) (...) ON def FOR EACH ROW ghi;

a) Trigger name

b) Table name

**c) Trigger statement**

d) Update statement

423. What is def in the following statement?

DECLARE abc HANDLER FOR def ghi;

a) Action

**b) Condition value**

c) Statement

d) Null

424. If you drop a table, any triggers for the table are also dropped.

1. **True**
2. False

425. Which statement use to delete triggers?

1. **DROP TRIGGER table\_name.trigger\_name**
2. TRIGGER table\_name.trigger\_name
3. DELETE TRIGGER table\_name.trigger\_name
4. None of the above.

426. What will be the output of the following code?

DROP PROCEDURE IF EXISTS P1;

DELIMITER $$

CREATE PROCEDURE P1()

BEGIN

DECLARE X VARCHAR (5);

SET X = NULL;

SELECT IFNULL(X, 'ABCDEFGHI');

END $$

DELIMITER ;

1. NULL
2. ABCDE
3. **ABCDEFGHI**
4. None of the above

427. How many values can be returned from a stored procedure?

1. **0**
2. 1
3. 2
4. 3

428. Which procedure parameter enables the caller to pass in a value and get back a value?

a) IN

b) OUT

**c) INOUT**

d) GETINOUT

429. The IN, OUT and INOUT keywords do not apply to stored functions.

**a) True**

b) False

430. A stored procedure is invoked using the statement \_\_\_\_\_\_\_\_\_\_

a) INVOKE

b) SEE

**c) CALL**

d) RETURN

431. A stored procedure is invoked using the statement \_\_\_\_\_\_\_\_\_\_

a) INVOKE

b) SEE

**c) CALL()**

d) RETURN

432. A stored procedure is invoked using the statement \_\_\_\_\_\_\_\_\_\_

a) INVOKE

b) CALL

c) CALL()

**d) Either B or C**

433. Which of the below statement is correct:

1. DROP PROCEDURE dbo.My\_Proc;

2. DROP PROCEDURE dbo.My\_Proc\_1, dbo.My\_Proc\_2, dbo.My\_Proc\_3;

3. DROP PROCEDURE IF EXISTS dbo.My\_Proc\_1;

1. Only 1 is correct
2. Both 1 & 2 are correct
3. **All three are correct**
4. None of above is correct

434. Examine the following code of MySQL.

drop procedure if exists pl1;

delimiter $$

create procedure pl1()

begin

DECLARE X VARCHAR (5);

SET X = 'ABCDEFGHI';

SELECT x;

end $$

delimiter ;

What will be the output?

1. ABCDE
2. **Error**
3. NULL
4. No Error.

435. Examine the following code of MySQL.

drop procedure if exists pl1;

delimiter $$

create procedure pl1()

begin

DECLARE X VARCHAR (50);

SET X = 'ABCDEFGHI';

SELECT left(x, 4);

end $$

delimiter ;

What will be the output?

1. ABCDE
2. **ABCD**
3. NULL
4. Error.

436. By default we use semicolon \_\_\_\_\_\_ as a delimiter.

1. **(;)**
2. ($)
3. ($$)
4. None of the above

437. By default, a stored procedure is associated with the default.......

1. Table
2. **Database**
3. View
4. All of the above

438. To associate the stored procedure explicitly with a given database, specify........

1. **db\_name.sp\_name**
2. db\_name\_sp\_name
3. db\_name$sp\_name
4. All of the above.

439. Stored procedures that take no arguments can be invoked using

1. CALL
2. CALL()
3. Execute
4. **Either A or B**

440. To get back a value from a procedure using........

1. **OUT**
2. OUTER
3. OUTSIDE
4. None of the above

441. To get back a value from a procedure using........

1. **INOUT**
2. OUTER
3. OUTSIDE
4. None of the above

442. Can we call stored procedure from within another stored procedure or function.

1. **True**
2. False

443. DELIMITER //

CREATE PROCEDURE GetAllProducts()

BEGIN

SELECT \* FROM products;

END //

DELIMITER ;

1. Invalid delimiter (//) character
2. **The procedure will print all product details**
3. Invalid command select in stored procedure
4. None of the above

444. To declare a variable inside a stored procedure, you use the \_\_\_\_\_\_\_ statement.

1. **DECLARE**
2. DEFINE
3. DEF
4. All of the above

445. You use the DECLARE statement as follows in stored procedure:

1. **DECLARE variable\_name datatype(size) DEFAULT default\_value;**
2. DECLARE datatype(size) variable\_name DEFAULT default\_value;
3. DECLARE variable\_name DEFAULT default\_value datatype(size);
4. All of the above

446. By default, all parameters are of \_\_\_\_\_ parameters in stored procedure.

1. **IN**
2. OUT
3. INOUT
4. Return

447. You cannot specify IN, OUT or INOUT modifiers to the parameters in stored procedure.

1. True
2. **False**

448. You can specify \_\_\_\_\_\_\_ modifiers to the parameters in stored procedure.

1. IN
2. OUT
3. OUTPUT
4. **Either A or B**

449. You can specify \_\_\_\_\_\_\_ modifiers to the parameters in stored procedure.

1. OUT
2. INOUT
3. OUTPUT
4. **Either A or B**

450. You can specify \_\_\_\_\_\_\_ modifiers to the parameters in stored procedure.

1. IN
2. INOUT
3. OUTPUT
4. **Either A or B**

451. \_\_\_\_\_\_\_\_\_\_\_\_\_\_ block is used to write compound statements.

1. **BEGIN ... END**
2. BEGIN ... ENDING
3. BEGIN ... BEGIN END
4. All of the above

452. To provide a default value for a variable, include a \_\_\_\_\_\_\_\_\_\_ clause.

1. DEFAUT
2. **DEFAULT**
3. DEFAILLIR
4. DECEIVE

453. What will be the output of the following code?

Drop procedure if exists pl1;

DELIMITER $$

CREATE PROCEDURE pl1()

BEGIN

DECLARE a INT DEFAULT 10;

DECLARE b, c INT;

SET a = a + 100;

SET b = 2;

SET c = a + b;

BEGIN

DECLARE c INT;

SET c = 5;

SELECT a, b, c;

END;

# SELECT a, b, c;

END$$

DELIMITER ;

1. **{110, 2, 5}**
2. {110, 2, 12}
3. {null, null, 5}
4. {0, 0, 5}

454. What will be the output of the following code?

Drop procedure if exists pl1;

DELIMITER $$

CREATE PROCEDURE pl1()

BEGIN

DECLARE a INT DEFAULT 10;

DECLARE b, c INT;

SET a = a + 100;

SET b = 2;

SET c = a + b;

BEGIN

DECLARE c INT;

SET c = 5;

# SELECT a, b, c;

END;

SELECT a, b, c;

END$$

DELIMITER ;

1. {110, 2, 5}
2. **{110, 2, 12}**
3. {null, null, 5}
4. {0, 0, 5}

455. In MySQL stored procedures, user variables are referenced with an \_\_\_ prefixed to the user variable name.

1. **@**
2. #
3. $
4. &

456. What will be the output of the following code?

Drop procedure if exists pl1;

DELIMITER $$

CREATE PROCEDURE pl1()

BEGIN

SET @x = 15;

SET @y = 10;

SELECT @x, @y, @x-@y;

END$$

delimiter ;

1. **{15, 10, 5}**
2. {15, 10, 0}
3. {15, 10, -5}
4. None of the above

457. What will be the output of the following code?

Drop procedure if exists pro1;

delimiter $

CREATE PROCEDURE pro1(IN var1 INT)

BEGIN

SELECT \* FROM EMP LIMIT var1;

END$

delimiter ;

mysql> call pro1(7)

1. Will display first record
2. **Will display first seven record**
3. Will display seventh record
4. None of the above.

458. What is PRO1 in the following statement?

delimiter $

CREATE PROCEDURE PRO1(IN var1 INT)

BEGIN

SELECT \* FROM EMP LIMIT var1;

END$

delimiter ;

a) Trigger name

b) Table name

**c) Procedure name**

d) Update statement

459. \_\_\_\_\_\_\_\_\_\_\_ statement can be used to change the characteristics of a stored procedure.

1. **Alter procedure**
2. Modify procedure
3. Change procedure
4. None of the above.

460. Which of the following statement is proper to create the stored procedure.

1. CREATE PROCEDURE PRO1...
2. **CREATE PROCEDURE PRO1()...**
3. CREATE PRO1 PROCEDURE...
4. None of the above.

461. Specifying a parameter as IN, OUT, or INOUT is valid only for a PROCEDURE.

1. **True**
2. False

462. Consider the following table having records.

TEMP = {1, 2, NULL, NULL, NULL, 3, 4}

What will be the output of the following code?

Drop procedure if exists PRO1;

delimiter $

CREATE PROCEDURE PRO1(out var1 INT)

BEGIN

SELECT count(\*) into var1 from BLANKTABLEA;

END$

delimiter ;

1. 0
2. 3
3. 4
4. **7**

463. Consider the following table having records.

TEMP = {1, 2, NULL, NULL, NULL, 3, 4}

What will be the output of the following code?

Drop procedure if exists PRO1;

delimiter $

CREATE PROCEDURE PRO1(out var1 INT)

BEGIN

SELECT count(\*) into var1 from BLANKTABLEA WHERE ID IS NULL;

END$

delimiter ;

1. 0
2. **3**
3. 4
4. 7

463. A stored procedure in SQL is a\_\_\_\_\_\_\_\_\_\_\_

1. Block of functions
2. **Group of SQL statements.**
3. None
4. All of the above

464. Which statement(S) is/are incorrect

1. Stored procedure can be shared by multiple programs
2. Stored procedures are in compiled form.
3. Stored procedure is a group of SQL statements
4. **All are correct.**

465. Which of the following is true concerning a procedure?

1. You do not create them with SQL.
2. They do not need to have a unique name.
3. **They include procedural and SQL statements.**
4. They are the same thing as a function.

466. Which of the following is used to input the entry and give the result in a variable in a procedure ?

a) Put and get

b) Get and put

c) Out and In

**d) In and out**

467. The format for compound statement is

**a) Begin ……. end**

b) Begin ……. endwith

c) Begin ……. repeat

d) Both Begin ……. end and Begin ……. Endwith

468. DECLARE is permitted only inside a BEGIN ... END compound statement and must be at its start, before any other statements.

1. **True**
2. False

469. To declare local variables, use the DECLARE statement.

1. **True**
2. False

470. Variables can be set directly with the SET statement.

1. **True**
2. False

471. It is not permitted to assign the value DEFAULT to stored procedure.

1. **True**
2. False

472. Stored procedures support execution of which SQL commands?

1. Only the SELECT command
2. SELECT & INSERT commands
3. Only the UPDATE command
4. **SELECT, INSERT, DELETE, and UPDATE commands**

473. Local variables declared within a stored routine using what statement?

1. DEFINE
2. ASSIGN
3. INVOKE
4. **DECLARE**

474. When declaring variable it must take place within which block?

1. START/FINISH block
2. **BEGIN/END block**
3. DELIMITER/DELIMITER block
4. OPEN/CLOSE block

475. Which statement is used when someone want to immediately exit a loop of a BEGIN … END block pending the value of a variable or outcome of a particular task?

1. **LEAVE**
2. END
3. FINISH
4. DELIMITER

476. Examine the following code.

Drop procedure if exists PRO1;

delimiter $

CREATE PROCEDURE PRO1(x int)

B:BEGIN

SELECT x;

END B$

delimiter ;

What will be the output after the procedure is called?

mysql> call PRO1(123);

1. 0
2. Null
3. **123**
4. None of the above.

477. Examine the following code.

Drop procedure if exists PRO1;

delimiter $

CREATE PROCEDURE PRO1(x int)

Begin:BEGIN

SELECT x;

END Begin$

delimiter ;

What will be the output after the procedure is called?

mysql> call PRO1(123);

1. 0
2. **Error**
3. 123
4. None of the above.

478. A BEGIN ... END block can be labelled.

1. **True**
2. False

479. The statement\_list itself is optional, so the empty compound statement (BEGIN...END) is

1. **legal.**
2. [illegal](https://www.google.co.in/search?rlz=1C1CHBF_enIN788IN788&q=illegal+affairs&spell=1&sa=X&ved=0ahUKEwiiua3LvvXaAhXJKo8KHZucA-QQBQgjKAA)
3. invalid
4. None of the above.

480. BEGIN ... END blocks can be nested.

1. **True**
2. False

481. To produce a stored function, which statement is used?

a) PRODUCE FUNCTION

**b) CREATE FUNCTION**

c) PRODUCE PROCEDURE

d) CREATE PROCEDURE

482. How many values can be returned from a given stored function?

a) 0

**b) 1**

c) 2

d) 3

483. To associate the stored function explicitly with a given database, specify........

1. **db\_name.function\_name**
2. function\_name.db\_name
3. db\_name$function\_name
4. All of the above.

484. Suppose a stored function named PI() is written in the database ‘sampdb’. How would it be called?

a) PI()

**b) sampdb.PI()**

c) MySQL.PI()

d) db.PI()

485. By default, a stored procedure is associated with the default.......

1. Table
2. **Database**
3. View
4. All of the above

486. What is the standard delimiter?

1. **;**
2. :
3. $
4. /

487. Do we have to put the parentheses after the name of the stored function?

1. **True**
2. False

488. The IN, OUT and INOUT keywords do not apply to stored functions.

**a) True**

b) False

489. Keywords and function names are not case sensitive.

1. **True**
2. False

490. A stored function is a special kind stored program that returns a

1. **Single value.**
2. Multiple value.
3. Simple value
4. Complex value

491. You use the DECLARE statement as follows in stored function:

1. **DECLARE variable\_name datatype(size) DEFAULT default\_value;**
2. DECLARE datatype(size) variable\_name DEFAULT default\_value;
3. DECLARE variable\_name DEFAULT default\_value datatype(size);
4. All of the above

492. By default, all parameters are of \_\_\_\_\_ parameters in stored function.

1. **IN**
2. OUT
3. INOUT
4. Return

493. You cannot specify IN, OUT or INOUT modifiers to the parameters in stored function.

1. **True**
2. False

494. You cannot specify \_\_\_\_\_\_\_ modifiers to the parameters in stored function.

1. IN
2. OUT
3. OUTPUT
4. **Either A or B**

495. You cannot specify \_\_\_\_\_\_\_ modifiers to the parameters in stored function.

1. OUT
2. INOUT
3. OUTPUT
4. **Either A or B**

496. You cannot specify \_\_\_\_\_\_\_ modifiers to the parameters in stored function.

1. IN
2. INOUT
3. OUTPUT
4. **Either A or B**

493. You must specify the data type of the return value in the RETURNS statement.

1. **True**
2. False

494. You must specify the data type of the return value in the \_\_\_\_\_\_\_\_ statement.

1. RETURN
2. **RETURNS**
3. OUT
4. None of the above.

495. \_\_\_\_\_\_statement is used to exit the loop construct.

1. **LEAVE**
2. LEAVING
3. LEFT
4. None of the above.

496. The \_\_\_\_\_\_statement terminates execution of a stored function and returns the value expr to the function caller.

1. **RETURN**
2. ACTION
3. RESEND
4. All of the above

497. There must be at least \_\_\_\_\_ RETURN statement in a stored function.

1. Zero
2. **One**
3. Two
4. Multiple

498. \_\_\_\_\_\_\_\_\_\_\_ statement can be used to change the characteristics of a stored function.

1. **Alter function**
2. Modify function
3. Change function
4. None of the above.

499. If there are no parameters in the function, an empty parameter list of () should be used.

1. **True**
2. False

450. For a FUNCTION, parameters are always regarded as IN parameters.

1. **True**
2. False

451. The RETURNS clause may be specified only for a FUNCTION, for which it is mandatory.

1. **True**
2. False

452. User defined variables are session specific.

**a) True**

b) False

453. The synonym for last\_insert\_id session variable is \_\_\_\_\_\_\_\_\_\_\_\_\_

a) insert\_id

**b) identity**

c) sql\_auto\_is\_null

d) sql\_big\_selects

454. Which of the following has a return type in its specification and must return a value specified in that type?

1. **Function**
2. Procedure
3. Package
4. None of the above

455. Create function dept count(dept\_name varchar (20))

begin

declare d count integer;

select count(\*) into d count from instructor where instructor. Dept\_name= dept\_name

return d count;

end $

Find the error in the above statement.

**a) Return type missing**

b) Dept\_name is mismatched

c) Reference relation is not mentioned

d) All of the mentioned

456. It is not permitted to assign the value DEFAULT to function parameters.

1. **True**
2. False

457. What statement/s is/are used to set the value of a declared stored routine variable?

1. **SET statement**
2. SELECT….. INTO statement
3. SELECT statement
4. SET….. INTO statement

458. The RETURN statement terminates execution of a stored function and returns the value expr to the function caller.

1. **True**
2. False

459. There must be at least one RETURN statement in a stored function.

1. **True**
2. False

460. What is the purpose of index in MySQL server?

1. To enhance the query performance
2. To provide an index to a record
3. To perform fast searches
4. **All of the mentioned**

461. A SQL query will not work if there are no indexes on the relations - Is it true?

1. **NO**
2. YES

462. The maximum number of indexes on MyISAM table is \_\_\_\_\_\_\_\_\_\_

1. 0
2. 1
3. 2
4. **more than 1**

463. Using indexes optimizes query performance.

1. **True**
2. False

464. Which of the following is valid SQL for an Index?

1. **CREATE INDEX ID**
2. CHANGE INDEX ID
3. ADD INDEX ID
4. REMOVE INDEX ID

465. Index is used to find the rows matching a WHERE clause quickly.

1. **True**
2. False

466. The \_\_\_\_\_\_\_\_\_\_\_\_\_\_ is used to display the defined indexes on a table.

1. **SHOW INDEX FROM table\_name**
2. SHOW INDEXES FROM table\_name
3. SHOW IND FROM table\_name
4. None of the above

467. CREATE INDEX cannot be used to create a PRIMARY KEY.

1. **True**
2. False

468. Consider the following relation R.

R(ID, NAME)

ID = {1, 2, 1}

NAME = {'Saleel', 'Vrushali', 'Sharmin'}

What will happen if this statement is issued?

CREATE UNIQUE INDEX IND1 ON R(ID);

1. Index will be applied on ID column
2. **Index will not be applied on ID column**
3. Error invalid create index syntax.
4. None of the above

469. Consider the following relation R.

R(ID, NAME)

ID = {1, 2, 1}

NAME = {'Saleel', 'Vrushali', 'Sharmin'}

What will happen if this statement is issued?

CREATE INDEX IND1 ON R(ID);

1. **Index will be applied on ID column**
2. Index will not be applied on ID column
3. Error invalid create index syntax.
4. None of the above

470. CREATE INDEX enables you to add indexes to existing tables.

1. **True**
2. False

471. Which clause is used to determine “which column to include in the query sets”?

**a) SELECT**

b) FROM

c) WHERE

d) ORDER BY

472. Which clause is used to “Identifies table from which to draw table and how the table should be joined”?

**a) FROM**

b) SELECT

c) ORDER By

d) WHERE

473. Which clause is used to “Filters out unwanted data”?

a) FROM

**b) WHERE**

c) SELECT

d) ORDER BY

474. Which clause is used to “group rows together by common columns values”?

a) SELECT

**b) GROUP BY**

c) FROM

d) WHERE

475. Which clause is used to “filter out unwanted Groups”?

**a) HAVING**

b) FROM

c) WHERE

d) SELECT

476. Which clause is used to “sort the rows of the final result set by one or more columns”?

a) HAVING

**b) ORDER BY**

c) WHERE

d) FROM

477. Which clause is used to “Modify the existing field of the table”?

**a) ALTER**

b) FROM

c) SELECT

d) MODIFY

478. Which among the following is not a “query clause”?

**a) WHERE**

b) MODIFY

c) ALTER

d) FROM

479. “MODIFY” is used with which “Query clause”?

**a) ALTER**

b) FROM

c) WHERE

d) ORDER BY

480. If emp\_id contain the following set {-1, -2, 2, 3, -3, 1}, what will be the output on execution of the given query?

SELECT emp\_id FROM person ORDER BY emp\_id;

1. **{-3, -2, -1, 1, 2, 3}**
2. {-1, 1, -2, 2, -3, 3}
3. {1, 2, 3, -1, -2, -3}
4. None of the mentioned

481. If emp\_id contain the following set {9, 7, 6, 4, 3, 1, 2}, what will be the output on execution of the given query?

SELECT emp\_id FROM person ORDER BY emp\_id DESC;

1. **{9, 7, 6, 4, 3, 1, 2}**
2. {1, 2, 3, 4, 6, 7, 9}
3. {2, 1, 3, 4, 6, 7, 9}
4. None of the mentioned

482. If emp\_id contain the following set {9, 7, 6, 4, 3, 1, 2}, what will be the output on execution of the given query?

SELECT emp\_id FROM person ORDER BY emp\_id;

1. {9, 7, 6, 4, 3, 1, 2}
2. **{1, 2, 3, 4, 6, 7, 9}**
3. {2, 1, 3, 4, 6, 7, 9}
4. None of the mentioned

483. Is there any error in the following query?

SELECT emp\_id, title, start\_date, fname, fed\_id FROM person ORDER BY RIGHT (fed\_id, 3);

1. Yes
2. **No error**
3. Depends
4. None of the mentioned

484. What will be the output of a query given below?

SELECT person\_id, Fname, lname FROM person;

1. **Show only columns (person\_id, Fname, lname) and rows related to these columns**
2. Show only columns (person\_id, Fname, lname)
3. Show all rows
4. Show all columns except (person\_id, Fname, lname)

485. [DROP TABLE](https://dev.mysql.com/doc/refman/8.0/en/drop-table.html) removes one or more tables.

1. **True**
2. False

486. A TEMPORARY table is visible only with the session that created it.

1. **True**
2. False

487. Which of the following are the types of sub-queries?

1. Ordered sub-queries
2. Grouped sub-queries
3. **Single row sub-queries**
4. None of the above

488. Which of the following is true about sub-queries?

1. They execute after the main query executes
2. They execute in parallel to the main query
3. The user can execute the main query and then, if wanted, execute the sub-query
4. **They execute before the main query executes.**

489. Which of the following clause is mandatorily used in a sub-query?

1. **SELECT**
2. WHERE
3. ORDER BY
4. GROUP BY

490. In which of the following clauses can a sub-query be used?

1. HAVING
2. WHERE
3. FROM
4. **All of the above**

491. Which of the following single-row operators can be used for writing a sub-query?

1. >=
2. <
3. =
4. **All of the above**

492. Which of the following multi-row operators can be used with a sub-query?

1. IN
2. ANY
3. ALL
4. **All of the above**

493. A type of query that is placed within a WHERE or HAVING clause of another query is called?

a) Master query

**b) Sub query**

c) Super query

d) Multi-query

494. To obtain the structure of a MySQL table, the command to use is:

a) STRUCTURE [TableName].

b) DESCRIBE [TableName].

c) DESCRIBE STRUCTURE [TableName].

**d) DESC TABLE [TableName].**

495. DML is provided for

1. Description of logical structure of database
2. Addition of new structure in the database system.
3. **Manipulation & processing of database**
4. Definition of physical structure of database system

496. 'AS' clause is used in SQL for

1. Selection operation
2. **Rename Operation**
3. Join operation
4. Projection Operation

497. Count function in SQL returns the number of

1. **values**
2. distinct values
3. groups
4. columns

498. The statement in SQL which allows to change the definition of a table is

1. **Alter**
2. Update
3. Create
4. Select

499. Which of the following operation is used if we are interested in only certain columns of a table?

1. **PROJECTION**
2. SELECTION
3. UNION
4. JOIN

500. Which command is used to select distinct subject (SUB) from the table (BOOK)?

1. SELECT ALL FROM BOOK
2. **SELECT DISTINCT SUB FROM BOOK**
3. SELECT SUB FROM BOOK
4. All of the above

501. A SELECT command without a WHERE clause returns?

a) All the records from a table that match the previous WHERE clause

**b) All the records from a table, or information about all the records**

c) SELECT is invalid without a WHERE clause

d) Nothing

502. In a SELECT with a GROUP BY clause, a WHERE clause, and a HAVING clause, the WHERE conditions are applied before the HAVING conditions

**a) True**

b) False

c) Either True or False

d) None of the above

503. Which of these is a valid call to a function?

1. concat('a', 123)
2. concat('a', '123')
3. concat(a, '123')
4. **Both A and B**

504. Which of the following is NOT available in MySQL:

1. REVOKE
2. **GET**
3. LIKE
4. JOIN

505. Which of the following is available in MySQL:

1. CREATE VIEW
2. CREATE SCHEMA
3. CREATE TRIGGER
4. **All of the above**

506. If a column contains 7 values and 6 distinct values, the cardinality is \_\_\_\_\_\_\_\_\_\_

a) 7

**b) 6**

c) 13

d) 1

507. Which function is used to divides one numeric expression by another and get the remainder?

1. POWER
2. **MOD**
3. ROUND
4. REMAINDER

508. The virtual table that it’s created by data from the result of an SQL 'Select' statement is called \_\_\_\_\_\_\_\_\_

1. **View**
2. Synonym
3. Sequence
4. Transaction

509. Which statement in SQL allows us to change the definition of a table is?

1. **ALTER**
2. UPDATE
3. CREATE
4. SELECT

510. Which of the following do you need to consider when you make a table in SQL?

1. Data types
2. Primary keys
3. Default values
4. **All of the above.**

511. The SQL statement that queries or reads data from a table is \_\_\_\_\_\_\_\_ .

1. **SELECT**
2. READ
3. QUERY
4. None of the above is correct.

512. In SQL "all attributes" can be displayed by using symbol?

1. ?
2. ^
3. &
4. **\***

513. What SQL structure is used to limit column values of a table?

1. The LIMIT constraint
2. **The CHECK constraint**
3. The VALUE constraint
4. None of the above is correct.

514. Which of these return a result to the client?

**a) Stored functions**

b) Stored procedures

c) Triggers

d) Events

515. Which of these is defined to execute when the table row is modified?

a) Stored functions

b) Stored procedures

**c) Triggers**

d) Events

516. What executes on a time activated basis according to a schedule?

a) Stored program

**b) Events**

c) Triggers

d) Stored procedures

517. Stored programs improve database security.

**a) True**

b) False

518. Which character does the mysql client program recognize as a statement delimiter?

a) :

b) .

**c) ;**

d) ,

519. Which command is used to redefined the mysql delimiter?

a) redefine\_delim

b) delim\_redefine

**c) delimiter**

d) redefine

520. Which of the following characters cannot be used as a delimiter?

a) ,

b) .

c) ;

**d) \**

521. Stored routines refers to stored functions and procedures.

**a) True**

b) False

522. Stored programs refer to stored objects of how many of the following types?

functions, procedures, triggers, events

a) 0

b) 1

c) 3

**d) 4**

523. Which of the following statements does not modify the table?

a) INSERT

b) UPDATE

c) DELETE

**d) SELECT**

524. Which statement upgrades the database directory name encoding?

**a) ALTER DATABASE**

b) ALTER SERVER

c) ALTER EVENT

d) ALTER FUNCTION

525. The statement that alters an existing event to have the given definition is \_\_\_\_\_\_\_\_\_\_\_\_\_

**a) ALTER EVENT**

b) ALTER DATABASE

c) ALTER FUNCTION

d) ALTER DATABASE

526. Which statement can produce the same output as ‘SHOW COLUMNS’?

a) DESCRIBE

**b) DISPLAY**

c) SEE

d) GET

527. The clause that is used to display information that matches a given pattern is \_\_\_\_\_\_\_\_\_\_\_\_\_

a) WHERE

b) IS

c) SAME

**d) LIKE**

528. The MySQL INSERT statement allows you to insert one or more rows into a table.

1. **True**
2. False

529. We often use the BETWEEN operator in the WHERE clause of the \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ statements.

1. SELECT
2. UPDATE
3. DELETE
4. **All of the above**

530. A NULL value is treated as a blank.

a) True

**b) False**

531. A table may be joined to itself.

**a) True**

b) False

c) None of the above

532. What SQL clause is used to restrict the rows returned by a query?

a) AND

**b) WHERE**

c) HAVING

d) FROM

533. The USE command?

a) Is used to load code from another file

b) Has been deprecated and should be avoided for security reasons

c) Is a pseudonym for the SELECT command

**d) Should be used to choose the database you want to use once you’ve connected to MySQL**

534. Which of these is not a comment specifying construct?

a) #

b) /\* \*/

c) —

**d) !#**

535.

536.

537.

538.

539.

540.

541. If there are no matching rows, [COUNT()](https://dev.mysql.com/doc/refman/8.0/en/group-by-functions.html#function_count) returns?

1. **0**
2. 1
3. empty
4. null

542. Count(), returns a count of the \_\_\_\_\_\_\_\_\_ values of expr in the rows retrieved by a SELECT statement.

1. 0
2. NULL
3. **non-NULL**
4. Both A and B

543. COUNT(\*) will returns a count of the number of \_\_\_\_\_\_\_\_\_ retrieved, whether or not they contain NULL values.

1. **rows**
2. values
3. columns
4. Will generate an error.

544. If there are no matching rows, COUNT(DISTINCT) returns?

1. **0**
2. 1
3. null
4. empty

545. What statement will you issue to find out the highest salary from EMP table statement?

1. **SELECT MAX (SAL) FROM EMP**
2. SELECT MAXIMUN (SAL) FROM EMP
3. SELECT GREATEST (SAL) FROM EMP
4. None of the above

546. What is the meaning of “GROUP BY” clause in Mysql?

**a) Group data by column values**

b) Group data by row values

c) Both a and b

d) None of the mentioned

547. Which clause is similar to “HAVING” clause in Mysql?

a) SELECT

**b) WHERE**

c) FROM

d) None of the mentioned

548. What is the meaning of “HAVING” clause in Mysql?

**a) To filter out the row values**

b) To filter out the column values

c) Both a and b

d) None of the mentioned

549. “COUNT” keyword belongs to which categories in Mysql?

**a) Aggregate functions**

b) Operators

c) Clauses

d) All of the mentioned

550. Which among the following belongs to an “aggregate function”?

**a) COUNT**

b) UPPER

c) LOWER

d) All of the mentioned

551. Which of the following belongs to an “aggregate function”?

a) COUNT

b) SUM/AVG

c) MIN/MAX

**d) All of the mentioned**

552. Which clause is used with an “aggregate functions”?

**a) GROUP BY**

b) SELECT

c) WHERE

d) Both a and c

553. What is the significance of the statement “GROUP BY d.name” in the given query?

SELECT d.name, COUNT (emp\_id) emp\_no FROM department d INNER JOIN Employee e ON d.dept\_id = e.emp\_id GROUP BY d.name

a) Aggregation of the field “name” of both table

**b) Aggregation of the field “name” of table “department”**

c) Sorting of the field “name”

d) None of the mentioned

554. What is the significance of the statement “HAVING COUNT (emp\_id)>2” in the given query?

SELECT d.name, COUNT (emp\_id) emp\_no FROM department d INNER JOIN Employee e ON d.dept\_id=e.emp\_id GROUP BY d.name HAVING COUNT (emp\_id)>2

a) Filter out all rows whose total emp\_id below 2

b) Selecting those rows whose total emp\_id>2

**c) Both a and b**

d) None of the mentioned

555. Aggregate functions are functions that take a \_\_\_\_\_\_\_\_\_\_\_ as input and return a single value.

1. **Collection of values**
2. Single value
3. Aggregate value
4. Both Collection of values & Single value

556. SELECT \_\_\_\_\_\_\_\_\_\_ FROM instructor WHERE dept name = 'Comp. Sci.';

Which of the following should be used to find the mean of the salary?

1. Mean(salary)
2. **Avg(salary)**
3. Sum(salary)
4. Count(salary)

557. SELECT COUNT (\_\_\_\_ ID) FROM teaches WHERE semester = ‘Spring’ AND YEAR = 2010;

If we do want to eliminate duplicates, we use the keyword \_\_\_\_\_\_in the aggregate expression.

1. **Distinct**
2. Count
3. Avg
4. Primary key

558. All aggregate functions except \_\_\_\_\_ ignore null values in their input collection.

1. Count(attribute)
2. **Count(\*)**
3. Avg
4. Sum

559. A Boolean data type that can take values true, false, and\_\_\_\_\_\_\_\_

a) 1

b) 0

c) Null

**d) Unknown**

560. The \_\_\_\_ connective tests for set membership, where the set is a collection of values produced by a select clause. The \_\_\_\_ connective tests for the absence of set membership.

a) Or, in

b) Not in, in

**c) in, not in**

d) In, or

561. Which of the following should be used to find all the courses taught in the Fall 2009 semester but not in the Spring 2010 semester.

**a) SELECT DISTINCT course id FROM SECTION WHERE semester = 'Fall' AND YEAR = 2009 AND course id NOT IN (SELECT course id FROM SECTION WHERE semester = 'Spring' AND YEAR = 2010);**

b) SELECT DISTINCT course\_id FROM instructor WHERE name NOT IN ('Fall', 'Spring');

c) (SELECT course id FROM SECTION WHERE semester = 'Spring' AND YEAR = 2010)

d) SELECT COUNT (DISTINCT ID) FROM takes WHERE (course id, sec id, semester, YEAR) IN (SELECT course id, sec id, semester, YEAR FROM teaches WHERE teaches.ID = 10101);

562. The phrase “greater than at least one” is represented in SQL by \_\_\_\_\_

a) < all

b) < some

c) > all

**d) > some**

563. Which of the following is used to find all courses taught in both the Fall 2009 semester and in the Spring 2010 semester.

**a) SELECT course id FROM SECTION AS S WHERE semester = ‘Fall’ AND YEAR = 2009 AND EXISTS (SELECT \* FROM SECTION AS T WHERE semester = ‘Spring’ AND YEAR = 2010 AND S.course id = T.course id);**

b) SELECT name FROM instructor WHERE salary > SOME (SELECT salary FROM instructor WHERE dept name = ‘Biology’);

c) SELECT COUNT (DISTINCT ID) FROM takes WHERE (course id, sec id, semester, YEAR) IN (SELECT course id, sec id, semester, YEAR FROM teaches WHERE teaches.ID = 10101);

d) (SELECT course id FROM SECTION WHERE semester = ‘Spring’ AND YEAR = 2010)

564. We can test for the nonexistence of tuples in a subquery by using the \_\_\_\_\_ construct.

a) Not exist

**b) Not exists**

c) Exists

d) Exist

565. SELECT dept\_name, ID, avg (salary) FROM instructor GROUP BY dept\_name;

This statement IS erroneous because

a) Avg(salary) should not be selected

**b) ID should not be used in group by clause**

c) Misplaced group by clause

d) Group by clause is not valid in this query

566. SQL applies predicates in the \_\_\_\_\_\_\_ clause after groups have been formed, so aggregate functions may be used.

a) Group by

**b) With**

c) Where

d) Having

567. Aggregate functions can be used in the select list or the \_\_\_\_\_\_\_ clause of a select statement or subquery. They cannot be used in a \_\_\_\_\_\_ clause.

a) Where, having

**b) Having, where**

c) Group by, having

d) Group by, where

568. The \_\_\_\_\_\_\_\_ keyword is used to access attributes of preceding tables or subqueries in the from clause.

a) In

**b) Lateral**

c) Having

d) With

569. Which of the following creates temporary relation for the query on which it is defined ?

**a) With**

b) From

c) Where

d) Select

570. TEMP table has total 11 rows and following data.

C1 C2

------ ------

1 1

2 2

NULL NULL

NULL NULL

NULL NULL

3 3

4 4

NULL NULL

5 5

NULL NULL

NULL 6

What will be output after giving following statement?

mysql> SELECT COUNT(\*) FROM TEMP;

1. 0
2. 1
3. **11**
4. NULL

571. TEMP table has total 11 rows and following data.

C1 C2

------ ------

1 1

2 2

NULL NULL

NULL NULL

NULL NULL

3 3

4 4

NULL NULL

5 5

NULL NULL

NULL 6

What will be output after giving following statement?

mysql> SELECT COUNT(\*) FROM TEMP GROUP BY C1;

1. 0
2. **6**
3. 11
4. NULL

572. Which of the following is not an aggregate function?

a) Avg

b) Sum

**c) With**

d) Min

573. Which of the following is not a valid aggregate function?

a) COUNT

b) MIN

c) MAX

**d) COMPUTE**

574. Which statement is used to count number of rows in table?

1. **SELECT COUNT(\*) FROM placement\_question;**
2. SELECT COUNT ALL(\*) FROM placement\_question;
3. SELECT ROWS(\*) FROM placement\_question;
4. All the above

575. Which SQL keyword is used to retrieve a maximum value?

1. TOP
2. MOST
3. UPPER
4. **MAX**

576. Which SQL keyword is used to retrieve a minimum value?

1. TOP
2. MOST
3. UPPER
4. **MIN**

577. Which of the following is a SQL aggregate function?

1. LEFT
2. **AVG**
3. JOIN
4. LEN

578. Which of the following group functions ignore NULL values?

1. MAX
2. **COUNT**
3. SUM
4. All of the above

579. "COUNT" keyword belongs to which categories in Mysql?

1. **Aggregate functions**
2. Operators
3. Clauses
4. All of the mentioned

580. "MIN" keyword belongs to which categories in Mysql?

1. **Aggregate functions**
2. Operators
3. Clauses
4. All of the mentioned

581. "MAX" keyword belongs to which categories in Mysql?

1. **Aggregate functions**
2. Operators
3. Clauses
4. All of the mentioned

582. "SUM" keyword belongs to which categories in Mysql?

1. **Aggregate functions**
2. Operators
3. Clauses
4. All of the mentioned

583. "AVG" keyword belongs to which categories in Mysql?

1. **Aggregate functions**
2. Operators
3. Clauses
4. All of the mentioned

584. Having clause is processed after the GROUP BY clause.

1. **True**
2. False

585. The sequence of the columns in a GROUP BY clause has no effect in the ordering of the output.

1. True
2. **False**

586. You want all dates when any employee was hired. Multiple employees were hired on the same date and you want to see the date only once.

Query - 1

Select distinct hiredate from hr.employee Order by hiredate;

Query - 2

Select hiredate from hr.employees Group by hiredate Order by hiredate;

Which of the above query is valid?

1. Query - 1
2. Query - 2
3. **Both**
4. None of the above

587. All aggregate functions ignore NULLs except for............

1. Distinct
2. **Count (\*)**
3. Average()
4. None of above

588. What is true about the following statement?

SELECT DEPT, AVG(SALARY) FROM EMPLOYEE GROUP BY DEPT;

1. AVG is the aggregate function name.
2. (SALARY) is the column on which the average is computed.
3. DEPT is the column by which the rows will be grouped.
4. **All of the above**

589. Aggregate function SUM() cannot be given in……..

1. Order by clause
2. Group by clause
3. Where clause
4. Both B and C

590. Aggregate function MAX() be given in……..

1. Having by clause
2. Group by clause
3. Where clause
4. All of the above.

591. The HAVING clause is used to restrict the results returned by the GROUP BY clause.

1. **True**
2. False

592. The GROUP BY Clause is used to group rows with same values.

1. **True**
2. False

593. The GROUP BY clause must appear after the FROM and \_\_\_\_\_\_\_ clauses.

1. **WHERE**
2. HAVING
3. ORDER BY
4. LIMIT

594. MySQL allows you to sort the groups in ascending or descending orders.

1. **True**
2. False

595. The default order is........... in Group By clause.

1. **Ascending**
2. Descending
3. Invalid
4. None of the above

596. If the GROUP BY clause is omitted, the HAVING clause behaves like the \_\_\_\_\_ clause.

1. **WHERE**
2. ORDER BY
3. LIMIT
4. None of the above

597. The HAVING clause applies a filter condition to each \_\_\_\_\_\_\_\_\_\_.

1. First 10 rows
2. **group rows**
3. individual row
4. None of the above

598. The WHERE clause applies the filter condition to each \_\_\_\_\_\_\_\_\_\_.

1. First 10 rows
2. group rows
3. **individual row**
4. None of the above

599. What are the results of the following queries if col is an integer column?

1. SELECT \* FROM mytbl WHERE num\_col = '4';

2. SELECT \* FROM mytbl WHERE num\_col = 4;

**a) Same**

b) Different

c) 1 is an error

d) 2 is an error

600. What is the meaning of "EMPTY SET"?

**a) No values**

b) Error

c) Access denied

d) None of the mentioned

601. Select odd one out?

a) Equality Conditions

b) Inequality Conditions

c) Range condition

**d) Between**

602. The following query belongs to which condition types?

SELECT fname FROM person WHERE title = ’TELLER’;

**a) Equality condition**

b) Inequality condition

c) Range condition

d) All of the mentioned

603. The following query belongs to which condition types?

SELECT fname FROM person WHERE fed\_id=’111-11-111’;

**a) Equality condition**

b) Inequality condition

c) Range condition

d) All of the mentioned

604. The following query belongs to which condition types?

SELECT firstName FROM person WHERE Dept\_id = (SELECT Dept\_id FROM department WHERE locationID = 1001);

/\* where locationID is a unique key \*/

**a) Equality condition**

b) Inequality condition

c) Range condition

d) All of the mentioned

605. Is the following query belongs to the “Equality condition”?

SELECT product\_type.name, product.name FROM product\_type INNER JOIN Product ON product\_type.dept = Product.dept WHERE product\_type.name = ’customers\_accounts’;

**a) Yes**

b) No

c) Depends

d) None of the mentioned

606. What is the meaning of “Equality Conditions”?

**a) Equal to**

b) Not equal to

c) Both Equal to and Not equal to

d) None of the mentioned

607. What is the meaning of “Inequality Condition”?

**a) Not-equal-to**

b) Equal-to

c) Both Not-equal-to and Equal-to

d) None of the mentioned

608. Does the following query belong to the “Inequality condition”?

SELECT product\_type.name, product.name FROM product\_type INNER JOIN Product ON product\_type.dept=Product.dept WHERE product\_type.name <> ’customers\_accounts’;

**a) Yes**

b) No

c) Depends

d) None of the mentioned

609. Does the following query belong to the “Inequality condition”?

SELECT product\_type.name, product.name FROM product\_type INNER JOIN Product ON product\_type.dept=Product.dept WHERE product\_type.name ! = ’customers\_accounts’;

**a) Yes**

b) No

c) Depends

d) None of the mentioned

610. What is the meaning of “Range Conditions”?

a) Expression is equal to Expression

b) Expression is not equal to Expression

**c) Expression fall under certain range**

d) None of the mentioned

611. Which among the following data types can be used with “Range Condition”?

a) Numeric data type

b) Temporal data type

c) String data type

**d) Both Numeric and Temporal data type**

612. The following query belongs to which “Condition Types”?

SELECT emp\_id, fname, lname FROM account WHERE start\_date < ’2007-10-08’;

a) Equality conditions

b) Inequality condition

**c) Range condition**

d) None of the mentioned

613. What will be the output of the following query?

SELECT \* FROM employee WHERE start\_date BETWEEN ‘2007-01-01’ AND ‘2008-01-01’;

**a) All employees details between 2007 and 2008**

b) All employees details before 2008

c) All employees details from 2007 to 2008

d) None of the mentioned

614. Fill the blanks with suitable options?

BETWEEN \_\_\_\_\_\_ AND \_\_\_\_\_\_

a) Upper and lower limit

**b) Lower and upper limit**

c) Both a and b

d) None of the mentioned

615. What will be the output of the following query?

SELECT \* FROM employee WHERE start\_date >= ’2007-01-01’ AND Start\_date <= ’2005-01-01’

a) All employees between 2007 and 2005

b) All employees from 2007 to 2005

c) Empty set

**d) None of the mentioned**

616. What will be the output of the following query?

SELECT customer\_id, product\_id, avail\_balance FROM account WHERE avail\_balance BETWEEN 3000 AND 5000.

**a) Only those details will be shown whose as available balance form 3000 to 5000**

b) Only those details will be shown whose available balance is more than 3000

c) Only those details will be shown whose available balance is less than 5000

d) None of the mentioned

617. Which among the following operators is/are belongs to “Inequality conditions”?

a) <>

b) !=

c) =

**d) Both <> and !=**

618. Which among the following operators is/are belongs to “Equality conditions”?

a) <>

b) !=

**c) =**

d) >/<

619. Which among the following operators is/are belongs to “Range conditions”?

a) <>

b) !=

c) =

**d) >/<**

620. Find the error in the following query?

SELECT cust\_id, fed\_id FROM customer WHERE cust\_id = ’I’ AND fed\_id BETWEEN 5000-00-000 AND 9999-999-000;

a) Yes

**b) No**

c) Range too high to compare

d) None of the mentioned

621. What will be the output of the following query?

SELECT account\_id, product\_id, cust\_id FROM account WHERE product\_id IN (‘sav’, ‘chd’, ‘mm’);

**a) Only those values are selected whose product\_id is either ‘sav’, ‘chd’, ‘mm’**

b) Only those values are selected whose product\_id is either ‘sav’’

c) Only those values are selected whose product\_id is either ‘sav’, ‘chd’

d) All of the mentioned

622. What will be the output of the following query?

SELECT account\_id, product\_id, cust\_id FROM account WHERE product\_id NOT IN (‘sav’, ‘chd’, ‘mm’);

**a) Only those values are selected whose product\_id does not belong to either ‘sav’, ‘chd’, ‘mm’**

b) Only those values are selected whose product\_id does not belong to either ‘sav’’

c) Only those values are selected whose product\_id does not belong to either ‘sav’, ‘chd’

d) All of the mentioned

623. Consider a database table COLORS name whose attributes are ID (primary key), Color.

ID = {1, 2, 3, 4, 5, 6}

Color = {Red, Blue, Red, Blue, Green, Yellow}

What will be the output of the following query?

SELECT ID FROM COLORS WHERE Color IN (Green, Yellow);

**a) {5, 6}**

b) {1, 2, 3}

c) {3, 4}

d) None of the mentioned

624. Consider a database name COLORS whose attributes are ID (primary key), Color.

ID = {1, 2, 3, 4, 5, 6}

Color = {Red, Blue, Red, Blue, Green, Yellow}

What will be the output of the following query?

SELECT ID FROM COLORS WHERE Color not IN (Green, Yellow);

a) {5, 6}

b) {1, 2, 3}

c) {3, 4}

**d) None of the mentioned**

625. Consider a database name COLORS whose attributes are ID (primary key), Color.

ID = {1, 2, 3, 4, 5, 6}

Color = {Red, Blue, Red, Blue, Green, Yellow}

What will be the output of the following query?

SELECT ID FROM COLORS WHERE Color NOT IN (Green, Yellow, Blue);

a) {5, 6}

b) {1, 2, 3}

c) {3, 4}

**d) {1, 3}**

626. Consider a database name COLORS whose attributes are ID (primary key), Color.

ID = {1, 2, 3, 4, 5, 6}

Color = {Red, Blue, Red, Blue, Green, Yellow}

What will be the output of the following query?

SELECT ID FROM COLORS WHERE color IN (Green, Yellow, Blue);

a) {5, 6}

b) {1, 2, 3}

c) {3, 4}

**d) {2, 4, 5, 6}**

627. Consider a database name COLORS whose attributes are ID (primary key), Color.

ID = {1, 2, 3, 4, 5, 6}

Color = {Red, Blue, Red, Blue, Green, Yellow}

What will be the output of the following query?

SELECT ID FROM COLORS WHERE color IN (Green, Yellow, Blue, Red);

a) {5, 6}

b) {1, 2, 3}

**c) {1, 2, 3, 4, 5, 6}**

d) Empty set

628. Consider a database name COLORS whose attributes are ID (primary key), Color.

ID = {1, 2, 3, 4, 5, 6}

Color = {Red, Blue, Red, Blue, Green, Yellow}

What will be the output of the following query?

SELECT ID FROM COLORS WHERE color NOT IN (Green, Yellow, Blue, Red);

a) {5, 6}

b) {1, 2, 3}

c) {1, 2, 3, 4, 5, 6}

**d) Empty set**

629. What will be the output of the following query?

SELECT emp\_id, fname, lname FROM employee WHERE LEFT (lname, 1) = ’T’;

**a) Only those employees are selected whose last name started with ‘T’**

b) Only those employees are selected whose last name started with other than ‘T’

c) All of the mentioned

d) None of the mentioned

630. What will be the output of the following query?

SELECT emp\_id, fname, lname FROM employee WHERE LEFT (lname, 1) =’F’;

**a) Only those employees are selected whose last name started with ‘F’**

b) Only those employees are selected whose last name started with other than ‘F’

c) All of the mentioned

d) None of the mentioned

631. What will be the output of the following query?

SELECT emp\_id, fname, lname FROM employee WHERE LEFT (fname, 1) =’F’;

**a) Only those employees are selected whose first name started with ‘F’**

b) Only those employees are selected whose first name started with other than ‘F’

c) All of the mentioned

d) None of the mentioned

632. What will be the output of the following query?

SELECT emp\_id, fname, lname FROM employee WHERE LEFT (fname, 1) = ’F’ OR LEFT (lname, 1) = ’F’;

**a) Only those employees are selected whose first name and last name started with ‘F’**

b) Only those employees are selected whose first name started with ‘F’ but last name can be starts with any other letter

c) Only those employees are selected whose first name and last name started with any other letter except ‘F’

d) None of the mentioned

633. What will be the output of the following query?

SELECT \* FROM employee WHERE lname LIKE ‘\_a%e%’;

a) All employees whose last name start with any letter but second letter should be ‘a’

b) All employees whose last name start with any letter but contain at least one ‘e’ in his name

c) All employees whose last name should have letter ‘a’ in second position and at least one ‘e’ in his name

**d) All of the mentioned**

634. What will be the output of the following query?

SELECT \* FROM employee WHERE lname LIKE ‘F%’ OR lname LIKE ‘%T’;

a) All employees whose last name should started with ‘F’

b) All employees whose last name end with ‘T’

**c) All employees whose last name should started with ‘F’ and end with ‘T’**

d) None of the mentioned

635. What will be the output of the following query?

SELECT \* FROM employee WHERE lname LIKE ‘F%’ AND lname LIKE ‘%T’;

a) All employees whose last name should started with ‘F’ and end with ‘T’

b) All employees whose last name end with ‘T’ and start with ‘F’

**c) Both a and b**

d) None of the mentioned

636. What will be the output of the following query?

SELECT \* FROM employee WHERE lname LIKE %bas% OR fname LIKE %bbs%;

a) All employees whose last name should contain substring “bas”

b) All employees whose first name should contain substring “bbs”

**c) Both a and b**

d) None of the mentioned

637. Consider an EMPLOYEE relation whose attributes are ID (primary key), firstName, lastName, credit\_card, and salary.

EMPLOYEE = {ID int, firstName varchar (12), lastName varchar (12) credit\_card boolean, salary float (8, 2)}

What will be the output of the following query?

SELECT \* FROM E WHERE CREDIT\_CARD = FALSE;

1. All employee who have credit card
2. **All employee who doesn’t have credit card**
3. All of the mentioned
4. None of the mentioned

638. Consider an EMPLOYEE relation whose attributes are ID (primary key), firstName, lastName, credit\_card, and salary.

EMPLOYEE = {ID int, firstName varchar (12), lastName varchar (12) credit\_card boolean, salary float (8, 2)}

What will be the output of the following query?

SELECT \* FROM E WHERE CREDIT\_CARD = 0;

1. All employee who have credit card
2. **All employee who doesn’t have credit card**
3. All of the mentioned
4. None of the mentioned

639. Consider an EMPLOYEE relation whose attributes are ID (primary key), firstName, lastName, credit\_card, and salary.

EMPLOYEE = {ID int, firstName varchar (12), lastName varchar (12) credit\_card boolean, salary float (8, 2)}

What will be the output of the following query?

SELECT \* FROM E WHERE CREDIT\_CARD = true;

1. **All employee who have credit card**
2. All employee who doesn’t have credit card
3. All of the mentioned
4. None of the mentioned

640. Consider an EMPLOYEE relation whose attributes are ID (primary key), firstName, lastName, credit\_card, and salary.

EMPLOYEE = {ID int, firstName varchar (12), lastName varchar (12) credit\_card boolean, salary float (8, 2)}

What will be the output of the following query?

SELECT \* FROM E WHERE CREDIT\_CARD = 1;

1. **All employee who have credit card**
2. All employee who doesn’t have credit card
3. All of the mentioned
4. None of the mentioned

641. Consider an EMPLOYEE relation whose attributes are ID (primary key), firstName, lastName, credit\_card, and salary.

EMPLOYEE = {ID int, firstName varchar (12), lastName varchar (12) credit\_card boolean, salary float (8, 2)}

What will be the output of the following query?

SELECT \* FROM E WHERE CREDIT\_CARD = 2-2;

1. All employee who have credit card
2. **All employee who doesn’t have credit card**
3. Error in the statement
4. None of the mentioned

642. Consider an EMPLOYEE relation whose attributes are ID (primary key), firstName, lastName, credit\_card, and salary.

EMPLOYEE = {ID int, firstName varchar (12), lastName varchar (12) credit\_card boolean, salary float (8, 2)}

What will be the output of the following query?

SELECT \* FROM E WHERE CREDIT\_CARD = 2-1;

1. **All employee who have credit card**
2. All employee who doesn’t have credit card
3. Error in the statement
4. None of the mentioned

643. Which one is correct syntax for Where clause in SQL server?

a) SELECT WHERE "Condition" Col1, Col2 FROM "Table" ;

b) SELECT "Condition" Col1, Col2 FROM "Table" WHERE;

**c) SELECT Col1, Col2 FROM "Table" WHERE "condition";**

d) None of the above

644. What is the purpose of Order By Clause in SQL server?

**a) It is used to sort the result.**

b) It is used to change sequence order of columns

c) It can’ be used in SQL Server

d) None of the above

645. Order by can only be used by Where Clause, correct?

a) True

**b) False**

646. What needs to be added when user want to show results by Descending Order?

a) Descending order cannot be possible.

**b) User can add DESC with Order By clause**

c) User can add ‘<>ASC’ with Order by Clause.

d) None of the above

647. What is the default order of Order by Clause?

1. Descending
2. **Ascending**
3. Random
4. None of the above

648. Among the below Order By queries, which are correct ones?

1. SELECT \* FROM Table Order By Column;
2. SELECT \* FROM Table Order By Column ASC;
3. SELECT \* FROM Table Order By Column DESC;
4. **All of the above**

649. Is it possible to have both Orders (i.e. ASC/DESC) in a single query?

1. **True**
2. False

650. Can ‘IN’ operator be used with Where clause?

1. **True**
2. False

651. What is the meaning of “SELECT” clause in Mysql?

**a) Show me all Columns and rows**

b) Show me all columns

c) Show me all rows

d) None of the mentioned

652. Which of the following clause is evaluated in the last by database server?

**a) SELECT**

b) WHERE

c) FROM

d) None of the mentioned

53. What will be the output of a query given below?

SELECT \* FROM person;

**a) Show all rows and columns of table “person”**

b) Show all rows of table “person”

c) Show all columns of table “person”

d) None of the mentioned

654. What will be the output of a query given below?

SELECT person\_id, Fname, lname FROM person;

**a) Show only columns (person\_id, Fname, lname) and rows related to these columns**

b) Show only columns (person\_id, Fname, lname)

c) Show all rows

d) Show all columns except (person\_id, Fname, lname)

655. Can “SELECT” clause be used without the clause “FROM”?

a) YES

**b) NO**

c) DEPENDS

d) None

656. Find the error?

SELECT \*;

a) No Error

**b) No table mentioned**

c) Depends

d) None of the mentioned

657. What will be the output of a query given below?

SELECT \* FROM person WHERE person\_id=1;

**a) Show all columns but only those rows which belongs to person\_id=1**

b) Show all columns and rows

c) Shows only columns person\_id

d) None of the mentioned

658. What will be the output of a query given below?

SELECT person\_id, fname, lname FROM person WHERE person\_id=1;

**a) Show only columns (person\_id, fname, lname) but only those rows which belongs to person\_id=1**

b) Show all columns and rows

c) Shows only columns person\_id

d) None of the mentioned

659. Which clause is mandatory with clause “SELECT” in Mysql?

**a) FROM**

b) WHERE

c) Both FROM and WHERE

d) None of the mentioned

660. The DISTINCT keyword used along with the SELECT keyword retrieves \_\_\_\_\_\_?

1. Duplicate data depending on the column list
2. **Unique data depending on the column list**
3. Sorted data depending on the column list
4. None of the above

661. Which among the following can also be included with “SELECT” clause while writing query in Mysql?

a) Literals

b) Expressions

c) User defined functions

**d) All of the mentioned**

662. Is there any error in executing the following query?

SELECT emp\_id, ‘ACTIVE’, emp\_id \* 3.145, UPPER (lname) FROM Employee;

a) Yes

**b) NO**

c) Depends on condition

d) None of the mentioned

663. Is there any error in executing the following query?

SELECT USER (), VERSION (), DATABASE ();

a) Yes, “FROM” is not used

**b) No**

c) Depends

d) None of the mentioned

664. What is the need of “column Aliases” in “SELECT” clause?

a) To assign a new label to the column in result set

b) To overwrite the existing column name in result set

c) To modify the column name while using literals, Expression, built\_in functions with “SELECT clause

**d) All of the mentioned**

665. What will be the result of the query given below?

SELECT emp\_id, ‘ACTIVE’ STATUS, emp\_id \* 3.14 emp\_pi, UPPER (lname) last\_name FROM employee;

a) emp\_id, ACTIVE, emp\_id \* 314, UPPER(lname)

**b) emp\_id, Status, emp\_pi, last\_name**

c) Error

d) None of the mentioned

666. What will be the result of the query given below?

SELECT emp\_id, ‘ACTIVE’ AS STATUS, emp\_id \* 3.14 AS emp\_pi, UPPER (lname) AS last\_name FROM employee;

a) emp\_id, ACTIVE, emp\_id \* 314, UPPER(lname)

**b) emp\_id, Status, emp\_pi, last\_name**

c) Error

d) None of the mentioned

667. Which Keyword is used to remove duplicate rows in result set?

**a) DISTINCTS**

b) MODIFY

c) DISTINCT

d) All of the mentioned

68. If in Table “account”, a column “cust\_id” consists of {1,2,2,3,3,5,6,7,8,8} then what will be the output on executing the following query?

SELECT DISTINICT cust\_id FROM account;

a) {1, 2, 2, 3, 3, 5, 6, 7, 8, 8}

**b) {1, 2, 3, 5, 6, 7, 8}**

c) { }

d) None of the mentioned

669. If in Table “employee”, a column “emp\_id” consists of {1,2,2,3,3,5,6,7,8,8} then what will be the output on executing the following query?

SELECT DISTINICT emp\_id FROM employee;

a) {1,2,2,3,3,5,6,7,8,8}

**b) {1, 2, 3, 5, 6, 7, 8}**

c) { }

d) None of the mentioned

670. Which among the following is an optional Keyword?

a) DISTINICTS

b) ALL

c) AS

**d) Both AS and ALL**

671. Which operator performs pattern matching?

1. BETWEEN operator
2. **LIKE operator**
3. EXISTS operator
4. None of these

672. What operator tests column for the absence of data?

1. EXISTS operator
2. NOT operator
3. **IS NULL operator**
4. None of these

673. In MySQL, which command is used to SELECT only one copy of each set of duplicable rows

1. **SELECT DISTINCT**
2. SELECT UNIQUE
3. SELECT DIFFERENT
4. All of the above

674. In MySQL, which command is used to SELECT only one copy of each set of duplicable rows

1. **SELECT DISTINCTROW**
2. SELECT UNIQUE
3. SELECT DIFFERENT
4. All of the above

675. A command that lets you change one or more fields in a record is

1. Insert
2. **Modify**
3. Look-up
4. All of the above

676. A command that lets you change one or more fields in a record is

1. Insert …. Values ….
2. **Alter …. Modify ….**
3. Look…. up
4. All of the above

677. Which of the SQL statements is correct?

1. SELECT Username AND Password FROM Users
2. **SELECT Username, Password FROM Users**
3. SELECT Username, Password WHERE Username = 'user1'
4. None of these

678. The FROM SQL clause is used to...

1. **specify what table we are selecting or deleting data FROM**
2. specify range for search condition
3. specify search condition
4. None of these

679. Which MySQL keyword is used to retrieve only unique values?

1. DISTINCTIVE
2. UNIQUE
3. **DISTINCT**
4. DIFFERENT

680. Which MySQL keyword is used to retrieve only unique values?

1. DISTINCTIVE
2. UNIQUE
3. **DISTINCTROW**
4. DIFFERENT

681. Which of the following SQL commands is used to retrieve data?

1. DELETE
2. INSERT
3. **SELECT**
4. JOIN

682. Find all the cities whose humidity is 89

1. SELECT city WHERE humidity = 89;
2. **SELECT city FROM weather WHERE humidity = 89;**
3. SELECT humidity = 89 FROM weather;
4. SELECT city FROM weather;

683. Find the temperature in increasing order of all cities

1. SELECT city FROM weather ORDER BY temperature;
2. SELECT city, temperature FROM weather;
3. **SELECT city, temperature FROM weather ORDER BY temperature;**
4. SELECT city, temperature FROM weather ORDER BY city;

684. Find the names of these cities with temperature and condition whose condition is neither sunny nor cloudy.

1. **SELECT city, temperature, condition FROM weather WHERE condition NOT IN ('sunny', 'cloudy');**
2. SELECT city, temperature, condition FROM weather WHERE condition NOT BETWEEN ('sunny', 'cloudy');
3. SELECT city, temperature, condition FROM weather WHERE condition IN ('sunny', 'cloudy');
4. SELECT city, temperature, condition FROM weather WHERE condition BETWEEN ('sunny', 'cloudy');

685. Find the name of those cities with temperature and condition whose condition is either sunny or cloudy but temperature must be greater than 70oF.

1. SELECT city, temperature, condition FROM weather WHERE condition = 'sunny' AND condition = 'cloudy' OR temperature > 70;
2. SELECT city, temperature, condition FROM weather WHERE condition = 'sunny' OR condition = 'cloudy' OR temperature > 70;
3. **SELECT city, temperature, condition FROM weather WHERE condition = 'sunny' OR condition = 'cloudy' AND temperature > 70;**
4. SELECT city, temperature, condition FROM weather WHERE condition = 'sunny' AND condition = 'cloudy' AND temperature > 70;

686. Find all the tuples having temperature greater than 'Paris'.

1. **SELECT \* FROM weather WHERE temperature > (SELECT temperature FROM weather WHERE city = 'Paris')**
2. SELECT \* FROM weather WHERE temperature > (SELECT \* FROM weather WHERE city = 'Paris')
3. SELECT \* FROM weather WHERE temperature > (SELECT city FROM weather WHERE city = 'Paris')
4. SELECT \* FROM weather WHERE temperature > 'Paris' temperature

687. Find all the cities with temperature, condition and humidity whose humidity is in the range of 63 to 79

1. SELECT \* FROM weather WHERE humidity IN (63 to 79)
2. SELECT \* FROM weather WHERE humidity NOT IN (63 AND 79)
3. **SELECT \* FROM weather WHERE humidity BETWEEN 63 AND 79**
4. SELECT \* FROM weather WHERE humidity NOT BETWEEN 63 AND 79

688. Find the names of the countries whose condition is sunny.

1. SELECT country FROM location WHERE condition = 'sunny';
2. **SELECT country FROM location WHERE city IN (SELECT city FROM weather WHERE condition = sunny');**
3. SELECT country FROM location WHERE city NOT IN (SELECT city FROM weather WHERE condition = 'sunny');
4. SELECT country FROM location WHERE city UNION (SELECT city FROM weather WHERE condition = 'sunny');

689. Find the name of all cities with their temperature, humidity and countries.

1. SELECT city, temperature, humidity, country FROM location;
2. SELECT weather.city, temperature, humidity, country FROM weather, location;
3. **SELECT weather.city, temperature, humidity, country FROM weather, location WHERE weather.city = location.city;**
4. SELECT weather.city, temperature, humidity FROM weather SELECT country FROM location WHERE weather.city = location.city;

690. Find the name of cities with all entries whose temperature is in the range of 71 and 89

1. SELECT \* FROM weather WHERE temperature NOT IN (71 to 89);
2. SELECT \* FROM weather WHERE temperature NOT IN (71 and 89);
3. SELECT \* FROM weather WHERE temperature NOT BETWEEN 71 to 89;
4. **SELECT \* FROM weather WHERE temperature BETWEEN 71 AND 89;**

691. Which of the following query finds the names of the sailors who have reserved at least one boat?

1. **SELECT DISTINCT s.sname FROM sailors s, reserves r WHERE s.sid = r.sid;**
2. SELECT s.sname FROM sailors s, reserves r WHERE s.sid = r.sid;
3. SELECT DISTINCT s.sname FROM sailors, reserves WHERE s.sid = r.sid;
4. None of These

692. Which of the following query finds colors of boats reserved by "Dustin"?

1. SELECT DISTINCT b.color FROM boats b, sailors s WHERE s.sname = 'Dustin' AND s.sid = b.sid
2. **SELECT DISTINCT b.color FROM boats b, reserves r, sailors s WHERE s.sname = 'Dustin' AND s.sid = r.sid AND r.bid = b.bid;**
3. SELECT DISTINCT b.color FROM boats b, reserves r, sailors s WHERE s.sname = 'Dustin' AND s.sid = r.sid
4. SELECT DISTINCT b.color FROM boats b, reserves r, sailors s WHERE s.sname = 'Dustin' AND r.bid = b.bid

693. The SELECT statement SELECT 'Hi' FROM DUAL WHERE NULL = NULL; Output?

1. Hi
2. FLASE
3. TRUE
4. **Nothing**

694. If a query involves NOT, AND, OR with no parenthesis

1. **NOT will be evaluated first; AND will be evaluated second; OR will be evaluated last.**
2. NOT will be evaluated first; OR will be evaluated second; AND will be evaluated last.
3. AND will be evaluated first; OR will be evaluated second; NOT will be evaluated last.
4. The order of occurrence determines the order of evaluation.

695. Let the statement

SELECT column1 FROM myTable;

Return 10 rows. The statement

SELECT ALL column1 FROM myTable;

Will return

1. Less than 10 rows
2. More than 10 rows
3. **Exactly 10 rows**
4. None of these

696. The SQL keyword(s) \_\_\_\_\_\_\_\_ is used with wildcards.

1. NOT IN only
2. **LIKE only**
3. IN only
4. IN and NOT IN

697. In an SQL SELECT statement querying a single table, according to the SQL-92 standard the asterisk (\*) means that:

1. **All columns of the table are to be returned.**
2. All records meeting the full criteria are to be returned.
3. All records with even partial criteria met are to be returned.
4. None of the above is correct.

698. SQL can be used to:

1. Create database structures only.
2. Query database data only.
3. Modify database data only.
4. **All of the above can be done by SQL.**

699. The SQL keyword BETWEEN is used:

1. to limit the columns displayed.
2. **for ranges.**
3. as a wildcard.
4. None of these is correct.

700. Which of the following query is correct for using comparison operators in SQL?

1. SELECT name, course\_name FROM student WHERE age>50 and <80;
2. **SELECT name, course\_name FROM student WHERE age>50 and age <80;**
3. SELECT name, course\_name FROM student WHERE age>50 and WHERE age<80;
4. None of these

701. How to select all data from student table starting the name from letter 'r'?

1. **SELECT \* FROM student WHERE name LIKE 'r%';**
2. SELECT \* FROM student WHERE name LIKE '%r%';
3. SELECT \* FROM student WHERE name LIKE '%r';
4. SELECT \* FROM student WHERE name LIKE '\_r%';

702. Which of the following function below are the correct way to get the current time?

1. Time()
2. **Curtime()**
3. getTime
4. All of the above.

703. Which of the following ways below are the correct way to get the current date?

1. SELECT CURTIME();
2. **SELECT CURDATE();**
3. SLELCT CURRRENT\_TIME()
4. All of the above

704. What will be the output of the following statement?

SELECT ABS(-32);

1. 0
2. **32**
3. -32
4. Error

705. What will be the output of the following statement?

SELECT ASCII(NULL);

1. 0
2. **null**
3. empty
4. NaN

706. What will be the output of the following statement?

SELECT ASCII('a b c');

1. **97**
2. 97 98
3. 97 98 99
4. Will display an error

707. What will be the output of the following statement?

SELECT CEIL(1.23);

1. 1
2. 1.23
3. **2**
4. 0

708. What will be the output of the following statement?

SELECT FLOOR(1.23);

1. **1**
2. 1.23
3. 2
4. 0

709. What will be the output of the following statement?

SELECT INSTR('CHRUCH', 'ch');

1. 0
2. **1**
3. 2
4. 3

710. \_\_\_\_\_\_\_\_\_\_\_\_\_ command returns the default (current) database name as a string.

1. Show DATABASE();
2. Show DATABASES();
3. **SELECT DATABASE();**
4. SELECT DATABASES();

711. If there is no default database, DATABASE() returns?

1. **NULL**
2. Empty
3. undefined
4. Error will occur.

712. What will be the output of the following statement?

SELECT LEFT('CHRUCH', '2');

1. C
2. **CH**
3. UCH
4. None

713. What will be the output of the following statement?

SELECT RIGHT('CHRUCH', 2);

1. C
2. **CH**
3. UCH
4. None

714. What will be the output of the following statement?

SELECT LENGTH(1.1);

1. 0
2. 1
3. 2
4. **3**

715. What will be the output of the following statement?

SELECT LOWER('QUADRATIC ALLY');

1. **quadratic ally**
2. Quadratic ally
3. Quadratic Ally
4. QUADRATIC ALLY

716. What will be the output of the following statement?

SELECT UPPER('quadratic ally');

1. quadratic ally
2. Quadratic ally
3. Quadratic Ally
4. **QUADRATIC ALLY**

717. Current date is returned by method

1. system\_date
2. **current\_date**
3. machine\_date
4. today\_date

718. Current date is returned by method

1. system\_date()
2. **current\_date()**
3. machine\_date()
4. today\_date()

719. The SQL statement

SELECT SUBSTR ('123456789', INSTR ('abcabcabc', 'b'), 4);

1. 6789
2. **2345**
3. 1234
4. 456789

720. The SQL statement

SELECT SUBSTR ('abcdefghij', INSTR ('123321234', '2'), 2);

Prints

1. gh
2. 23
3. **bc**
4. ab

721. The SQL statement

SELECT ROUND (45.926, -1);

1. Is illegal
2. Prints garbage
3. Prints 045.926
4. **Prints 50**

722. The SQL statement

SELECT ROUND (45.926, -2);

1. is illegal
2. **Prints 0**
3. Prints 045.926
4. Prints 50

723. Which of the following must be enclosed in double quotes in MySQL?

1. Dates
2. Column Alias
3. Strings
4. **All of the above**

724. Which function returns NULL if expr1 = expr2?

a) CASE

b) IF()

c) IFNULL()

**d) NULLIF()**

725. The string function that returns the index of the first occurrence of substring is \_\_\_\_\_\_\_\_\_\_\_\_\_

a) INSERT()

**b) INSTR()**

c) INSTRING()

d) INFSTR()

726. What does the AUTO\_INCREMENT sequences begin at by default?

a) 0

**b) 1**

c) -1

d) 2

727. The one that is not optional is \_\_\_\_\_\_\_\_\_\_\_\_\_

SELECT select\_list FROM table\_list WHERE row\_constraint GROUP BY grouping\_columns;

**a) select\_list**

b) table\_list

c) row\_constraint

d) grouping\_columns

728. Which operator is used to compare a value to a specified list of values?

1. BETWEEN
2. ANY
3. **IN**
4. ALL

729. A SQL query automatically eliminates duplicates?

1. TRUE
2. **FALSE**

730. Which of the following is a comparison operator in SQL?

1. **=**
2. ==
3. ===
4. None of the above

731. Which of the following is a legal expression in SQL?

1. SELECT \* FROM SALES WHEN NULL
2. **SELECT NAME FROM SALES**
3. SELECT \* FROM SALES WHEN PRICE = NULL
4. SELECT # FROM SALES

732. The command to remove rows from a table ‘CUSTOMER’ is:

1. REMOVE FROM CUSTOMER …
2. DROP FROM CUSTOMER …
3. **DELETE FROM CUSTOMER WHERE …**
4. UPDATE FROM CUSTOMER …

733. The SQL WHERE clause:

1. Limits the column data that are returned.
2. **Limits the row data are returned.**
3. Both A and B are correct.
4. Neither A nor B are correct.

734. A CASE SQL statement is which of the following?

1. **A way to establish an IF-THEN-ELSE in SQL.**
2. A way to establish a loop in SQL.
3. A way to establish a data definition in SQL.
4. All of the above.

735.

736.

737.

738.

739.

740.

741. Result of the below query is:

SELECT SUBSTR('This is the test', null, 1);

1. 0
2. **null**
3. T
4. None of the above

742. Result of the below query is:

SELECT INSTR('This is the test for null', null);

1. 0
2. **null**
3. 22
4. None of the above

743. Result of the below query is:

SELECT INSTR('This is the test for null', 'null');

1. 0
2. null
3. **22**
4. None of the above

744. Which operator is used to check whether the expression is “NULL”?

**a) IS NULL**

b) NOT NULL

c) ON

d) None of the mentioned

745. Which operator is used to check the expression is not “NULL”?

a) IS NULL

**b) IS NOT NULL**

c) ON

d) None of the mentioned

746. What will be the output of the following query?

SELECT \* FROM person WHERE emp\_id IS NULL;

**a) Only those columns whose emp\_id is NULL**

b) Only those columns whose emp\_id is not NULL

c) No output

d) None of the mentioned

747. What will be the output of the following query?

SELECT \* FROM person WHERE emp\_id IS NOT NULL;

a) Only those columns whose emp\_id is NULL

**b) Only those columns whose emp\_id is not NULL**

c) No output

d) None of the mentioned

748. What will be the output of the following query?

SELECT \* FROM person WHERE emp\_id = NULL;

a) Only those columns whose emp\_id is NULL

b) Only those columns whose emp\_id is not NULL

**c) No output**

d) None of the mentioned

749. What will be the output of the following query?

SELECT fname FROM person WHERE emp\_id != 6;

**a) Only those names whose emp\_id is not equal to 6**

b) Only those names whose emp\_id is equal to 6

c) All of the mentioned

d) None of the mentioned

750. What will be the output of the following query?

SELECT fname FROM person WHERE emp\_id != 6 OR emp\_id IS NULL;

**a) Only those names whose emp\_id is not equal to 6 or emp\_id with NULL values**

b) Only those names whose emp\_id is not equal to 6

c) All of the mentioned

d) None of the mentioned

751. Comparisons between two null values (where null = null), returns.............

1. NULL
2. 0
3. **Empty set**
4. NONE OF ABOVE

752. Null values can be inserted into a column by...........

1. explicitly stating NULL in an INSERT
2. when adding a new column to an existing table by using the ALTER TABLE statement
3. by leaving a column out of an INSERT statement
4. **All of above**

753. Result of the below query is:

SELECT ('potato ' + NULL + 'chips')

1. **Potato Chips**
2. Potato
3. NULL
4. NONE OF ABOVE

754. Result of the below query is:

SELECT 10 + 10 + NULL;

1. 10 + 10
2. 20
3. **NULL**
4. NONE OF ABOVE

755. Result of the below query is:

SELECT COUNT(NULL);

1. **0**
2. EMPTY
3. NULL
4. NONE OF ABOVE

756. Result of the below query is:

SELECT MAX(NULL);

1. 0
2. EMPTY
3. **NULL**
4. NONE OF ABOVE

757. Result of the below query is:

SELECT MIN(NULL);

1. 0
2. EMPTY
3. **NULL**
4. NONE OF ABOVE

758. Result of the below query is:

SELECT IFNULL(NULL, TRUE);

1. **1**
2. EMPTY
3. NULL
4. NONE OF ABOVE

759. Result of the below query is:

SELECT IFNULL(NULL, FALSE);

1. **0**
2. EMPTY
3. NULL
4. NONE OF ABOVE

760. Result of the below query is:

SELECT UPPER('null');

1. 0
2. **'NULL'**
3. EMPTY
4. None of the above

761. Result of the below query is:

SELECT UPPER(null);

1. 0
2. **NULL**
3. EMPTY
4. None of the above