**Mid-Term Exam**

Class Room Online

Assignment Points: 15 points

Thursday 10/29/2020

**Name: Felipe Lima - 109290055**

**Exam rules:**

* You must submit this mid-term by today, **10/29/2020, 11:59 pm**.
* Submit your assignment in PDF format in Canvas. You can use word, excel or similar tools and convert into pdf.
* This is open book exam and any kind of resource materials are allowed.
* Collaboration and consultation is NOT allowed. Do your own work.

**Section 1: Multiple choice questions (use X mark or highlight your answer)**

**Total Points: 4 (All questions are equally weighted)**

1. What is the syntax to load data into the table? (Consider D as a table and a, b, c as data)
2. enter into D (a, b, c);
3. insert into D values (a, b, c);
4. insert into D (a, b, c);
5. insert (a, b, c) values into D;
6. When columns are join from the same table, the type of join is called ………….?
   1. Union
   2. Right Outer Join
   3. Left Outer Join
   4. Self-Join
7. The *address* field of a person table should not be part of the primary key since it is likely
8. Dependent
9. Changed
10. Text
11. Too long
12. The term *attribute* refers to a \_\_\_\_\_\_\_\_\_\_\_ of a table.
13. Record
14. Column
15. Tuple
16. Key
17. The term \_\_\_\_\_\_\_ is used to refer to a row.
18. Attribute
19. Tuple
20. Field
21. Instance
22. A relational database consists of a collection of
23. Tables
24. Fields
25. Records
26. Keys
27. CREATE TABLE employee is part of
28. DML
29. DDL
30. VIEW
31. Integrity constraint
32. The maximum value for data type Decimal (3, 2) is
33. 9.99
34. 99.99
35. 999.99
36. All of the above
37. Duplicate records will be eliminated, when a query uses
38. Select Only Clause
39. Where Distinct Clause
40. Select Distinct Clause
41. From Distinct Clause
42. Which clause is similar to “*HAVING*” clause in SQL statement?
43. SELECT
44. WHERE
45. FROM
46. None of the mentioned
47. INSERT INTO *Instructor* VALUES (10211, ’Smith’, ’Biology’, 66000);

What type of statement is this?

1. Query
2. DML
3. Relational
4. DDL
5. What is the meaning of *“GROUP BY”* clause in SQL statement?
6. Group data by column values
7. Group data by row values
8. Group data by column and row values
9. None of the mentioned
10. Which among the following belongs to an *aggregate function*?
11. COUNT
12. TOTAL
13. LOWER
14. All of the above
15. *Character* data can be stored as
16. Fixed length string
17. Variable length string
18. Either Fixed or Variable length string
19. None of the mentioned
20. SELECT a.branch\_name, COUNT (d.customer\_name) AS count

FROM account a, depositor d

WHERE a.account\_number = d.account\_number

GROUP BY a.branch\_id;

1. The query is syntactically correct but missing having clause
2. The query is syntactically incorrect
3. The query is syntactically correct
4. The query contains incorrect join.
5. A domain is *atomic* if elements of the domain are considered to be \_\_\_\_\_\_\_\_\_\_\_\_ units.
6. Different
7. Indivisible
8. Constant
9. Divisible
10. “DELETE from person WHERE person\_id = 2” where person\_id is a primary key in person table with values of 1, 2, 3 and 4.

How many rows will be deleted when you run above SQL?

1. 0
2. 1
3. 2
4. B and C both
5. Which of the following clause must be present with 'HAVING' clause in SQL?
6. Group by
7. Where
8. Order by
9. None of the above
10. What column names are displayed when this command is executed?

SHOW COLUMNS FROM TableA LIKE '%name' ;

1. first\_name
2. store\_name
3. company\_name
4. all of the above
5. What is xyz in the following statement?

SELECT abc FROM xyz;

1. row name
2. column name
3. table name
4. database name

**Section 2: Fill in the blanks**

**Total Points: 4 (All questions are equally weighted)**

1. **Item** table has primary key I**temID** AUTO\_INCREMENT and 10 rows of data inserted.

Change AUTO\_INCREMENT to start from 100.

ALTER TABLE Item

AUTO\_ INCREMENT = 100;

1. Table *Employee* has columns (empid, name and managerid). Complete to find employees who are also managers.

SELECT e. name

FROM employee e

INNER JOIN employee m

ON (e.empid = m.managerid);

…………………………………………………………………………………………………………………………

1. ***Customerid*** is key in both **Orders O** and **Customers C** tables. Complete below to select records that exists in both tables.

SELECT O.orderid, O.desc, C.name

FROM Orders O

INNER JOIN Customers C

ON (O.Customerid = C.Custoemrid);

………………………………………………………………..………………………………………………………….

1. Update TableA to add 100 on *salary* for primary key *emp\_id* = 10

UPDATE TableA

SET salary = salary + 100

WHERE emp\_id = 10;

…………………………………………………………………………………………………………………………..

1. Complete below SQL statement to find count of records from Customers table.

SELECT Country, State, City, Count(\*) AS Count

FROM Customers

GROUP BY Country, State;

……………………………………………………………………………………………………………..…………………..

1. Add FK on *child\_table* (column1) refrencingfrom *parent\_table* (column1).

ALTER TABLE *child\_table*

ADD FOREIGN KEY (column1) REFERENCES *parent\_table* (column1);

………………………………………………………………………………………………………………………..………….

**Section 3: Write SQL statements**

**Total Points: 5 (All questions are equally weighted)**

Please answer all question based on below tables. Make sure to use table aliases:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Customer (C)** |  |  |  |  | **Order (O)** |  |  |  |
| *customer\_id (PK)* | first\_name | last\_name | job\_title |  | *order\_id (PK)* | *customer\_id (FK)* | order\_date | shipping\_company |
| C001 | John | Kelly | DBA |  | 1 | C001 | 9/27/2019 | FedEx |
| C002 | Amelia | Cruze | DBA |  | 2 | C002 | 9/30/2019 | UPS |
| C003 | Sohpia | Henry | Cashier |  | 3 | C002 | 8/15/2019 | UPS |
| C004 | Tom | Smith | QA |  | 4 | C005 | 8/20/2019 | FedEx |
| C005 | Mia | Stark | Cashier |  | 5 | C005 | 9/15/2019 | UPS |

1. Select full name (i.e. first\_name and last\_name) and job\_title whose customers records exists in customers table but NOT in orders table **using sub-query**.

SELECT C.first\_name, C.last\_name, C.job\_title

FROM Customer C

WHERE C.customer\_id NOT IN (SELECT O.customer\_id FROM Order O);

1. Select first\_name, last\_name, shipping\_company and order\_date for all records from Customers table but ONLY matching records from Orders table for order\_date after August 31st 2019.

SELECT C.first\_name, C.last\_name, O.shipping\_company, O.order\_date

FROM Customer C

INNER JOIN Orders O ON (C.customer\_id = O.customer\_id)

WHERE O.order\_date > '2019/08/31';

1. Write a SQL statement selecting shipping\_company, order\_date and their rank with most recent order\_date rank first and so on.

SELECT shipping\_company, order\_date,

RANK () OVER (

ORDER BY order\_date DESC

) ranking

FROM Orders;

1. Select first\_name, last\_name and shipping\_company for matching records from both tables for customers first\_name **ends** with **a** and sort by most recent order\_date first.

SELECT C.first\_name, C.last\_name, O.shipping\_company

FROM Customer C

INNER JOIN Orders O ON (C.customer\_id = O.customer\_id)

WHERE C.first\_name LIKE '%a';

1. Write a SQL statement to find shipping\_company and their count whose count is greater than 2.

SELECT shipping\_company, COUNT(shipping\_company) AS count

FROM Orders

GROUP BY shipping\_company

HAVING count > 2;

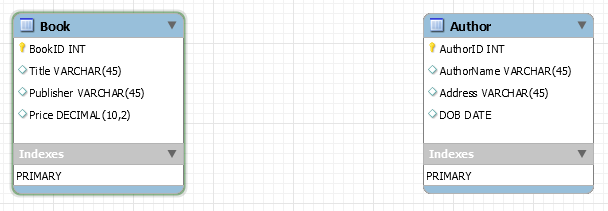
**Section 4: Create relationship for below tables, use proper symbols, lines and captions**

**Total Points: 2**

**Note: Create Book and Author tables as below and solve relationship using MySQL Workbench Data Model (ERD).**

**DDL generation NOT NEEDED.**

1. A Book **can be written** by several Authors
2. An Author **can write** several Books

Assumption: Each book must have an author and each author must write a book. 

Diagram

Description automatically generated ☺ ☺ The END ☺ ☺ ☺