Mid-Term Exam

Classroom Online

Assignment Points: 20 points

Name: Michael Ghattas

Exam rules:

- You must submit this mid-term by **3/23/2023, 11:59 pm**. No late submission.
- Submission: submit in Canvas in pdf or word doc.
- This is open book exam, and any kind of resource materials are allowed.
- Collaborations and consultations are NOT allowed. Do your own work.

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

Total Points: 5 (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)
   A. enter into D (a, b, c);
   B. insert into D values (a, b, c);
   C. insert into D (a, b, c);
   D. insert (a, b, c) values into D;

2. When the table is joined with itself, the type of join is called
   A. Union
   B. Right Outer Join
   C. Left Outer Join
   D. Self-Join

3. The *address* field of a person table should not be part of the primary key since it is likely
   A. Dependent
   B. Changed
   C. Text
   D. Too long

4. The term *attribute* refers to a _____ of a table.
   A. Record
   B. Column
   C. Tuple
   D. Key

5. The term _____ is used to refer to a row.
    A. Attribute
    B. Tuple
    C. Field
    D. Instance

6. A relational database consists of a collection of
    A. Tables
    B. Fields
    C. Records
    D. Keys

7. CREATE TABLE employee is part of
    A. DML
    B. DDL
    C. VIEW
    D. Integrity constraint

8. The maximum value for data type Decimal (3, 2) is
    A. 9.99
    B. 99.99
    C. 999.99
    D. All of the above

9. Duplicate records will be eliminated when a query uses
    A. Select Only Clause
    B. Where Distinct Clause
    C. Select Distinct Clause
    D. From Distinct Clause

10. Which of the following is similar to "*HAVING*" clause in SQL statement?
    A. SELECT
    B. WHERE
    C. FROM
    D. None of the mentioned

11. INSERT INTO *Instructor* VALUES (10211, 'Smith', 'Biology', 66000);
    What type of statement is this?
    A. Query
    B. DML
    C. Relational
    D. DDL

12. What is the meaning of *"GROUP BY"* clause in SQL statement?
    A. Group data by column values
    B. Group data by row values
    C. Group data by column and row values
    D. None of the mentioned

13. Which among the following belongs to an *aggregate function*? Select all that apply.
    A. COUNT
    B. TOTAL
    C. LOWER
    D. All of the above

14. An artificially generated key that can be used for primary key when there are no other good candidate keys is called
    A. Surrogate Key
    B. Foreign Key
    C. Natural Key
    D. Composite Key

15. 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;
    A. The query is missing "Having" clause
    B. The query is syntactically incorrect
    C. The query is syntactically correct
    D. The query contains incorrect join.

16. A domain is *atomic* if elements of the domain is considered
    A. Different
    B. Indivisible
    C. Constant
    D. Divisible

17. Person table has PK personid with values of 1, 2, 3 and 4. “DELETE from Person WHERE personid = 2”. How many rows will be deleted when you run above SQL?
    A. 0
    B. 1
    C. 2
    D. None of the above

18. Which of the following clause must be present with 'HAVING' clause in SQL?
    A. Group by
    B. Where
    C. Order by
    D. None of the above

19. What column names are displayed when this SQL command is executed?
    SHOW COLUMNS FROM TableA LIKE '%name' ;
    A. first_name
    B. store_name
    C. company_name
    D. all of the above

20. What is xyz in the following statement?
    SELECT abc FROM xyz;
    A. row name
    B. column name
    C. table name
    D. database name

Section 2: Fill in the blanks

Total Points: 6 (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;

2. 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;

3. ***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.Customerid;

4. Update TableA to add 100 on *salary* for primary key *emp_id* = 10

   UPDATE TableA
   SET salary = salary + 100
   WHERE emp_id = 10;

5. 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, City;

6. Add FK on *child_table* (column1) referencing from *parent_table* (column1).

   ALTER TABLE *child_table*
   ADD CONSTRAINT fk_name
   FOREIGN KEY (column1) REFERENCES parent_table(column1);

Section 3: Write SQL statements

Total Points: 6 (All questions are equally weighted)

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

Customers

customer_id (PK)	first_name	last_name	job_title
C001	John	Kelly	DBA
C002	Amelia	Cruze	DBA
C003	Sohpia	Henry	Cashier
C004	Tom	Smith	QA
C005	Mia	Stark	Cashier

Orders

order_id (PK)	*customer_id (FK)*	order_date	shipping_company
1	C001	9/27/2019	FedEx
2	C002	9/30/2019	UPS
3	C002	8/15/2019	UPS
4	C005	8/20/2019	FedEx
5	C005	9/15/2019	UPS

1. Select full name (i.e., first_name and last_name, e.g., John Kelly) and job_title whose customers records exists in Customers table but NOT in Orders table **using sub-query**.

```
SELECT CONCAT(first_name, ' ‘, last_name) AS full_name, job_title
FROM Customers
WHERE customer_id NOT IN (SELECT customer_id FROM Orders);
```

2. 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 Customers c
JOIN Orders o ON c.customer_id = o.customer_id
WHERE o.order_date > ‘2019-08-31';
```

3. 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) AS rank
FROM Orders;
```

4. 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 Customers c
JOIN Orders o ON c.customer_id = o.customer_id
WHERE c.first_name LIKE '%a'
ORDER BY o.order_date DESC;
```

5. Write a SQL statement to find shipping_company and their count whose count is greater than 2.

```
SELECT shipping_company, COUNT(*) AS count
FROM Orders
GROUP BY shipping_company
HAVING COUNT(*) > 2;
```

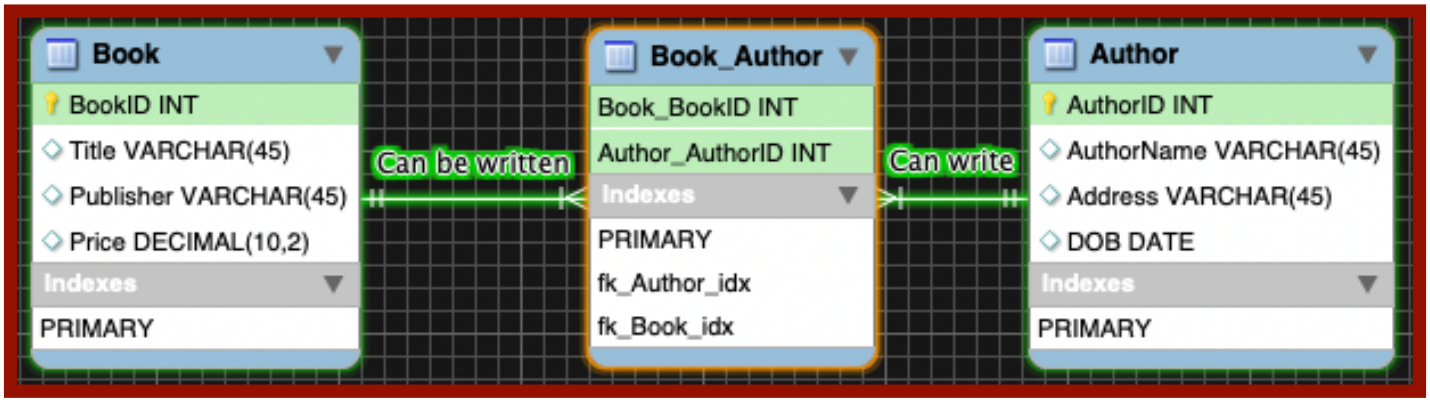
6. Write a SQL statement to find total count (all records) of job_title, unique (distinct) count of job_title and the difference between those two counts from the Customers table. Display results as total_count, unique_count and difference.

```
SELECT
   COUNT(*) AS total_count,
   COUNT(DISTINCT job_title) AS unique_count,
   COUNT(*) - COUNT(DISTINCT job_title) AS difference
FROM Customers;
```

Section 4: Create relationship for below tables, use proper symbols, lines and captions
Total Points: 3

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.



☺☺ The END ☺☺☺