SQL Syntax and Programming (22 Questions)

An example table to relate Social Security Number, Name, and Address:

EMPLOYEE					
SSN	**FirstName**	**LastName**	**Address**	**City**	**State**
512687458	Joe	Smith	83 First Street	Howard	Ohio
758420012	Mary	Scott	842 Vine Ave.	Losantiville	Ohio
102254896	Sam	Jones	33 Elm St.	Paris	New York
876512563	Sarah	Ackerman	440 U.S. 110	Upton	Michigan

Question 1

Create a SQL query to list all columns from the table above.

SELECT * FROM EMPLOYEE

Let's look at a new example table below:

EMPLOYEE_STATISTICS			
Employee_No	**Salary**	**Benefits**	**Position**
010	75000	15000	Manager
105	65000	15000	Manager
152	60000	15000	Manager
215	60000	12500	Manager
244	50000	12000	Staff
300	45000	10000	Staff
335	40000	10000	Staff
400	32000	7500	Entry-Level
441	28000	7500	Entry-Level

Question 2

WSWWCreate a SQL query to retrieve all employee no's that earn a salary of $50,000 and above.

SELECT Employee_No FROM EMPLOYEE_STATISTICS WHERE Salary = 50000

Question 3

Create a SQL query to show employee no and salary for all Managers.

SELECT Employee_No, Salary FROM EMPLOYEE_STATISTICS WHERE Position = "Manager"

Question 4

Create a query to display all employee no for staff position that are earning more than $40,000.

SELECT Employee_No FROM EMPLOYEE_STATISTICS WHERE Position = "Staff" AND Salary > 40000

Question 5

Create a query to see all employees who earn less than $40,000 or have less than $10,000 in benefits, listed together.

SELECT Employee_No FROM EMPLOYEE_STATISTICS WHERE Salary < 40000 OR Benefits < 10000

Question 6

Create a query to see all Managers that earn more than $60,000 or have more than $12,000 in benefits.

SELECT Position FROM EMPLOYEE_STATISTICS WHERE Salary > 60000 OR Benefits > 12000

Question 7

Create a SQL query to see a list of employees earning more than $50,000 or have a large benefit package of more than $10,000 and that happens to be a manager.

SELECT Employee_No FROM EMPLOYEE_STATISTICS WHERE Salary > 50000 OR Benefits > 10000 AND Position = "Manager"

Question 8

Using **IN**, create a SQL query to list all managers and staff:

SELECT Employee_No FROM EMPLOYEE_STATISTICS WHERE Position IN (SELECT Position FROM EMPLOYEE_STATISTICS WHERE Position = "Manager" AND Position = "Staff"

Question 9

Create a SQL query to list those earning greater than or equal to $30,000, but less than or equal to $50,000.

SELECT Employee_No FROM EMPLOYEE_STATISTICS WHERE Salary >= 30000 AND Salary <= 50000

Question 10

Create a SQL query to list everyone not in Question 5:

SELECT Employee_No FROM EMPLOYEE_STATISTICS WHERE NOT IN Salary < 40000 OR Benefits < 10000

Question 11

Using the EMPLOYEE table, create a SQL query to list all people whose last names started with "S".

SELECT * FROM EMPLOYEE WHERE LastName LIKE '%S'

Question 12

Please refer back to **EMPLOYEE_STATISTICS** table.
Create a query to list all available position with the sum of all their salaries.

SELECT * FROM EMPLOYEE_STATISCTICS WHERE Position IN (“Manager”, “Staff”, “Entry-Level”) AND COUNT(Salary)

Question 13

Create a query to show the number of employees that are earning more than $50,000.

SELECT COUNT (Employee_No) AS Jumlah_Karyawan FROM EMPLOYEE_STATISTICS WHERE Salary > 50000

Question 14 (Programming test - 3 Points)

Using any programming language that you experienced, write a simple if-else and loop statements to display Sum of salary for Manager position to user:
Expected results: Sum of Salary for Manager = 260000

Programming Language:
Codes:

Take a look at these example tables:

ANTIQUEOWNERS

OwnerID	OwnerLastName	OwnerFirstName
01	Jones	Bill
02	Smith	Bob
15	Lawson	Patricia
21	Akins	Jane
50	Fowler	Sam

ORDERS

OwnerID	ItemDesired
02	Table
02	Desk
21	Chair
15	Mirror

ANTIQUES

SellerID	BuyerID	Item
01	50	Bed
02	15	Table
15	02	Chair
21	50	Mirror
50	01	Desk
01	21	Cabinet
02	21	Coffee Table
15	50	Chair
01	15	Jewelry Box
02	21	Pottery
21	02	Bookcase
50	01	Plant Stand

Question 15

What would be the primary key for the table **AntiqueOwners**?

OwnerID

Question 16

Create a SQL query to find the names of those who bought a chair.

SELECT OwnerFirstName, OwnerLastName FROM AntiqueOwners JOIN Orders ON Orders.OwnerID = AntiqueOwners.Owner ID WHERE ItemDesired = "Chair"

Question 17

Create a SQL query to list the ID and names of only those people who have sold an antique. In addition to that, the list must be sorted by LastName, then by FirstName.

SELECT OwnerID, OwnerLastName, OwnerFirstName FROM AntiqueOwners JOIN Antiques ON Antiques.SellerID = AntiqueOwners.OwnerID AND Antiques.BuyerID = AntiqueOwners.OwnerID ORDER BY (OwnerLastName, OwnerFirstName) DESC

Question 18

Create a SQL query that lists the last name of those owners who have placed an order and what the order is, only listing those orders which can be filled (that is, there is a buyer who owns that ordered item).

SELECT AntiqueOwners.OwnersLastName, Antiques.Item FROM AntiquesOwners JOIN Antiques ON Antiques.BuyerID = AntiqueOwners.OwnerID WHERE IN BuyerID (SELECT Item FROM Antiques)

Question 19

Please refer back to **ORDERS** table.
Provide a SQL syntax to create a view called V_ORDERS base on the ItemDesired column in the ORDERS table.

CREATE VIEW V_Orders AS SELECT ItemDesired FROM ORDERS

Question 20

Please refer back to **ANTIQUES** table.
SellerID = 21
BuyerID = 01
Item = 'Table Lamp'
Base on the data above, write an insert statement to insert this new record into the **ANTIQUES** table.

INSERT INTO Antique (SellerID, Buyer ID, Item) Values (21, 01, "Table Lamp")

Question 21

Create a SQL statement to delete all Bookcase items in the **ANTIQUES** table.

DELETE FROM Antiques WHERE item = "Bookcase"

Question 22

Create a SQL statement to change the OwnerLastName of 'Atkins' to 'Atkinson' in the **ANTIQUEOWNERS** table.

UPDATE AntiqueOwners SET OwnerLastName = "Atkinson" WHERE OwnerID = 21