**Academic Prompts**

**Instructions**: Answer the following problems below. Show the SQL statements to address what is being asked in the problem. Briefly discuss the answer.

**Problem 1:**

You have a table named PlayerStat contains the following fields:

|  |  |  |
| --- | --- | --- |
| **Field** | **Data Type** | **Allow Nulls** |
| PlayerID | INT | FALSE |
| TeamID | INT | FALSE |
| GameDate | DATETIME | TRUE |
| Points | INT | TRUE |

What SQL command to display the total number of points per player on the team whose ID is 1.

*SELECT PlayerID, SUM(Points) FROM PlayerStat WHERE ID=1 GROUP BY PlayerID*

The query select PlayerID column and SUM(Points) for getting the total number of points per PlayerID and group it by PlayerID and filter the record by ID that is equal to 1.

**Problem 2:**

The following table name Building stores data about buildings and their most recent inspection dates.

|  |  |  |
| --- | --- | --- |
| **Field** | **Data Type** | **Allow Nulls** |
| BuildingID | INT | FALSE |
| Address | VARCHAR(100) | FALSE |
| InspectionID | CHAR(3) | TRUE |
| InspectionDate | DATETIME | TRUE |

The Address field store the buildings address. The InspectionDate field stores the most recent inspection date. A value of NULL in the InspectionDate field means that the building has not yet been inspected.

What SQL command to display the top 10 address with inspection date and arranged according to date?

|  |
| --- |
| *SELECT InspectionDate , Address FROM Building WHERE InspectionDate IS NOT NULL GROUP BY InspectionDate ORDER BY InspectionDate DESC LIMIT 10*  The query select InspectionDate and Address column from table Building and filter the table by InspectionDate where value in not null and group the rows by InspectionDate and sort the table by InspectionDate in descending order and specify only 10 records shown. |

**Problem 3:**

Given is the Customers table below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | FirstName | LastName | PhoneNumber | Extension |
| 1 | Hope | Ragabash | (123) 555-0111 | NULL |
| 2 | Luna | Faltor | (123) 555-0112 | NULL |
| 3 | Mickey | Sassafras | (123) 555-0113 | 12 |
| 4 | Minnie | Hemingway | (123) 555-0114 | 77 |
| 5 | Sherlock | Steam | (123) 555-0115 | NULL |

You need to create a query that returns a result set that contains the LastName, PhoneNumber and Extension for extensions that are valid numbers. The result set should be sorted by the customer ‘s last name.

*SELECT LastName, PhoneNumber, Extension FROM Customers WHERE Extension REGEXP '^[0-9]+$' OR Extension IS NOT NULL ORDER BY LastName ASC*

The query select LastName, PhoneNumber and Extension in a table and filter the result by column Extension using regular expression validate value as valid number and “OR” operator to validate value is not null

**Problem 4:**

The Products table contains the following data:

|  |  |  |  |
| --- | --- | --- | --- |
| **ItemNumber** | **ItemName** | **ItemDescription** | **Price** |
| 1 | Bonbon Box | Chocolate Truffles, Black Forest Truffles, and Cherry Bonbons | 24.95 |
| 2 | Brownie Mini Bites | Caramel Nut, German Chocolate,  Chocolate Swirl, and Double Dutch Brownies | 14.95 |
| 3 | Cappuccino  Companion  Collection | Tasty treats to accompany your favorite java. Includes Strawberry  Thumbprints, Chocolate Chip  Cookies, and Biscotti | 21.50 |
| 4 | Citrus Cooler Gilt Box | Refreshing citrus cookies, including  Orange Chiffon Mellaways, Key Lime  Bars, Lemon Collers and Tangerine  Coconut Smoothies | 19.99 |
| 5 | Fruit Jewels Assortment | Fruity Favorites, including Apricot  Jewels, Cherry Bonbons, Coconut  Patties, and Strawberry Thumbprints | 29.99 |

What query statement to execute that will increase the price of item 1 by 6 percent?

*UPDATE Products SET Price=((Price\*6)/100) WHERE ItemNumber = 1*

The query update the value of column Price by multiplying Price by 6 and divide it by 100 to get the 6 percent and filter it with ItemNumber whose value is 1.

**Problem 5:**

Given is the ItemOnOrder table below:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ItemNumber | ID | Quantity | UnitPrice | LineItemTotal |
| 100 | 1 | 10 | 24.95 | 249.50 |
| 100 | 2 | 25 | 14.94 | 373.75 |
| 100 | 3 | 25 | 19.99 | 499.75 |
| 101 | 4 | 10 | 25.00 | 250.00 |
| 102 | 5 | 10 | 29.99 | 299.00 |

You need to create a query that displays the total number of orders, the average line item total, the highest line item total, and the grand total of all items ordered.

LineItemTotal represents the line item total

What query should you execute?

|  |
| --- |
| *SELECT SUM(Quantity), AVG(LineItemTotal), MAX(LineItemTotal), SUM(LineItemTotal) FROM ItemOnOrder*  The query uses SUM function to get the total of quantity, AVG function to get the average of LineItemTotal, MAX function to get the highest LineItemTotal and SUM LineItemTotal to get the grand total |