|  |
| --- |
| Name:Ro-el Jean Ryan Alocada |
| Email Address (gmail): |
| Date Taken: |

**Instruction: Answer the questions provided which are SQL related questions. Write your answer in the boxes provided.**

***Note: Don’t use any AI Tool (this will be subject to auto-fail)***

**Part I. Basic SQL Questions**

**1. Question:** What is the difference between IFNULL() and COALESCE()? **Points: 2/2**

Answer:

|  |
| --- |
| IFNULL() returns a specified value if the expression is NULL; otherwise, it returns the expression itself while COALESCE returns the first non-NULL value from a list of expressions. |

**2. Question:** What is wrong in this query? **Points: 2/2**

*SELECT UserId, AVG(Total) AS AvgOrderTotal*

*FROM Invoices*

*HAVING COUNT(OrderId) >= 1*

Answer:

|  |
| --- |
| There is no GROUP BY Clause . |

**3. Question:** Correct syntax query syntax to drop a column from a table is? **Points: 1/1**

1. DELETE COLUMN column\_name;
2. DROP COLUMN column\_name;
3. ALTER TABLE table\_name DROP COLUMN column\_name;
4. None is correct.

Answer:

|  |
| --- |
| C. |

**4. Question:** List 3 various Analytic aggregate functions **Points: 3/3**

Answer:

|  |
| --- |
| SUM , COUNT and AVG |

**5. Question:** What is the difference between UNION and UNION ALL? **Points: 2/2**

Answer:

|  |
| --- |
| UNION returns a unique record while UNION ALL returns all the records including duplicates. |

**6. Question:** What is the difference between RANK() and DENSE\_RANK()? **Points: 2/2**

Answer:

|  |
| --- |
| RANK() returns the ranking within your ordered partition while DENSE\_RANK() is the same as RANK() but the ranks are consecutive. |

**7. Question:** What is the difference between WHERE and HAVING? **Points: 2/2**

Answer:

|  |
| --- |
| WHERE clause is used to filter the records from a table that is based on a specified condition while HAVING clause is used to filter the record from the groups based on a specified condition. |

**8. Question:** Explain different types of JOIN with an example. **Points: 3/3**Answer:

|  |
| --- |
| INNER JOIN - returns all records that has matching values in both tables.  RIGHT JOIN - returns all records from the right table regardless of the matching values . LEFT JOIN - returns all records from the left table regardless of the matching values.  OUTER JOIN – returns all records regardless if there is a match in both tables. |

**Part II. Database questions (need to write SQL queries)**

**9. Use the table below to answer what is asked in Q9.1 and Q9.2:**

**Table 1 : user**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **User\_id** | **User\_name** | **Department** | **Salary** | **Manager\_id** |
| A | aa | X | 100,000 |  |
| B | bb | Y | 80,000 | A |
| C | cc | Y | 50,000 | B |
| D | dd | Y | 60,000 | B |
| ... | ... | ... | ... | ... |

***Q9.1:*** *Which manager (name) has the highest salary?* **Points: 5/5**

Answer:

|  |
| --- |
| SELECT User\_name  FROM user  WHERE user\_id IN ( SELECT DISTINCT Manager\_id FROM user)  ORDER BY Salary DESC  LIMIT 1 |

***Q9.2:*** *Which manager (name) has the highest average salary per reportee?* **Points: 5/5**

Answer:

|  |
| --- |
| SELECT user1.User\_name, AVG(user2.Salary) AS Average\_Salary  FROM user user1  JOIN user user2  ON user1.User\_id = user2.Manager\_id  GROUP BY user1.User\_name  ORDER BY Average\_Salary DESC  LIMIT 1; |

**10. Use the table below to answer what is asked in Q10.1 and Q10.2:**

**Table 2: billing**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **customer\_id** | **billing\_id** | **created\_date** | **country\_code** | **billing\_amount** |
| 1 | id\_11 | 2019-06-21 | US | $100 |
| 1 | id\_12 | 2020-05-11 | FR | $126 |
| 1 | id\_13 | 2019-12-28 | DE | $97 |
| 2 | id\_21 | 2019-07-01 | CN | $118 |
| 2 | id\_22 | 2020-08-10 | AT | $92 |
| 2 | id\_23 | 2019-07-04 | IN | $107 |
| 2 | id\_24 | 2021-01-03 | US | $71 |
| 3 | id\_31 | 2020-11-29 | CA | $89 |
| 3 | id\_32 | 2020-06-10 | US | $200 |
| 3 | id\_33 | 2020-10-10 | RU | $101 |
| 4 | id\_41 | 2020-04-08 | SG | $234 |
| 5 | id\_51 | 2019-09-07 | JP | $125 |

***Q10.1:*** *Find the most recent billing account creation date by each customer with the associated country code.* **Points: 5/5**

Answer:

|  |
| --- |
| SELECT b.customer\_id, b.billing\_id, b.created\_date, b.country\_code  FROM billing b  WHERE b.created\_date = (  SELECT MAX(b2.created\_date)  FROM billing b2  WHERE b2.customer\_id = b.customer\_id  ); |

***Q10.2****: Find the average billing amount per year from 2019 to 2021 for every customer, get $0 if the customer has no billing from any specific year.* **Points: 5/5**

Answer:

|  |
| --- |
| SELECT customer\_id, YEAR(created\_date) AS year, AVG(billing\_amount) AS average\_billing\_amount  FROM billing  WHERE YEAR(created\_date) BETWEEN 2019 AND 2021  GROUP BY customer\_id, YEAR(created\_date)  ORDER BY customer\_id, year; |