**Phase 2 TASK**

TABLE 1 1) Write a query to Retrieve the Top 20 Most Expensive Products 2) Write a query Calculate the Total Price for Each Category: 3) Write a query to Retrieve Products with a Price Range 2000 to 5000 4) Write a query to Calculate the Average Price for Each Brand . 5) Write a query to find Products with Unique Brand-Category Combinations

1.) Write a query to Retrieve the Top 20 Most Expensive Products

SELECT \*

FROM Table1

ORDER BY Price DESC

LIMIT 20;

2. Calculate the Total Price for Each Category:

SELECT Category, SUM(Price) AS TotalPrice

FROM Table1

GROUP BY Category;

3. Retrieve Products with a Price Range 2000 to 5000:

SELECT \*

FROM Table1

WHERE Price BETWEEN 2000 AND 5000;

4. Calculate the Average Price for Each Brand:

SELECT Brandname, AVG(Price) AS AveragePrice

FROM Table1

GROUP BY Brandname;

5. Find Products with Unique Brand-Category Combinations:

SELECT \*

FROM (

SELECT \*, COUNT(\*) OVER (PARTITION BY Brandname, Category) AS CombinationCount

FROM Table1

) AS subquery WHERE CombinationCount = 1;

Now write sql query for table1 by having followung questions for table2 Write the SQL queries 1) Write a query Order Rows by Material Type in Ascending Order . 2) Write a query to Calculate the Average Number of Reviews 3) Write a query to Retrieve the Top 30 Most Expensive Products by Weartype: 4) Write a query to Find Products with Similar Colors. 5) Write a query to Calculate the Total Price for Each Material Type

1. Order Rows by Material Type in Ascending Order:

SELECT \*

FROM Table2

ORDER BY Materialtype ASC;

1. Calculate the Average Number of Reviews:

SELECT AVG(Reviews) AS AverageReviews

FROM Table2;

1. Retrieve the Top 30 Most Expensive Products by Weartype:

SELECT \*

FROM (

SELECT \*, ROW\_NUMBER() OVER (PARTITION BY Weartype ORDER BY Price DESC) AS row\_num

FROM Table2

) AS ranked

WHERE row\_num <= 30;

1. Find Products with Similar Colors:

SELECT DISTINCT t1.\*

FROM Table2 t1

JOIN Table2 t2 ON t1.Sno != t2.Sno

AND (t1.color1 = t2.color1 OR t1.color2 = t2.color2 OR t1.color3 = t2.color3 OR t1.color4 = t2.color4);

1. Calculate the Total Price for Each Material Type:

SELECT Materialtype, SUM(Price) AS TotalPrice

FROM Table2

GROUP BY Materialtype;

Now write sql query for table1 by having followung questions for table3 Write the SQL queries Table 3 .csv 1) Write a query to Order Rows by Return Time in Ascending Order. 2) Write a query to find distinct country of origin . 3) Write a query to Calculate the Average Return Time by Country of Origin. 4) Write a query to Find Occasions with High Return Times. 5 ) Write a query to Retrieve Products with Multiple Sizes (Size1, Size2, Size3, Size4): JOIN QUERY using table1,table2 and table3 1) Write a query find Brandname, Category , Producttype ,ProductName with a listed price greater than the average listed price in table 1. 2) Write a query to list all Brandname, Category , Producttype ,ProductName along with their corresponding color information from Table2 3) Write a query to find the average reviews for each Producttype ,ProductName using table 1 and table 2 4) Write a query to find products with a Product name, material type that matches the most common material type in (use table1 and table 2) 5) Write a query to list all products Weartype , Material type along with their corresponding sizes from (use table 2 and table 3) 6) Write a query to find the average return time for each product type (use table1 and table 3) 7 Write a query to find Country of origin, return time ,occasion of each product type . (use table 1 and table 3)

1.Order Rows by Return Time in Ascending Order:

SELECT \*

FROM Table3

ORDER BY Returntime ASC;

2.Find distinct country of origin:

SELECT DISTINCT Countryoforigin

FROM Table3;

3.Calculate the Average Return Time by Country of Origin:

SELECT Countryoforigin, AVG(Returntime) AS AverageReturnTime

FROM Table3

GROUP BY Countryoforigin;

4. Find Occasions with High Return Times:

SELECT Occasion

FROM Table3

GROUP BY Occasion

HAVING AVG(Returntime) > (SELECT AVG(Returntime) FROM Table3);

1. Retrieve Products with Multiple Sizes (Size1, Size2, Size3, Size4):

SELECT \*

FROM Table3

WHERE Size1 IS NOT NULL AND Size2 IS NOT NULL AND Size3 IS NOT NULL AND Size4 IS NOT NULL;

Q)For the JOIN queries using Tablwe1, Table2 and Table3:

Find Brandname , Category , Producttype, ProductName a listed price greater than the average listed in TBLE 1:

ANS) SELECT t1.Brandname, t1.Category, t1.Producttype, t1.ProductName

FROM Table1 t1

WHERE t1.Price > (SELECT AVG(Price) FROM Table1);

1. List all Brandname, Category, Producttype, ProductName along with their corresponding color information from Table2:

SELECT t1.Brandname, t1.Category, t1.Producttype, t1.ProductName, t2.color1, t2.color2, t2.color3, t2.color4

FROM Table1 t1

JOIN Table2 t2 ON t1.Sno = t2.Sno;

1. Find the average reviews for each Producttype, ProductName using Table1 and Table2:

SELECT t1.Producttype, t1.ProductName, AVG(t2.Reviews) AS AverageReviews

FROM Table1 t1

JOIN Table2 t2 ON t1.Sno = t2.Sno

GROUP BY t1.Producttype, t1.ProductName;

4.Find products with a Product name, material type that matches the most common material type:

SELECT t1.\*

FROM Table1 t1

JOIN (

SELECT Materialtype, COUNT(\*) AS Count

FROM Table2

GROUP BY Materialtype

ORDER BY Count DESC

LIMIT 1

) AS most\_common ON t1.Materialtype = most\_common.Materialtype;

1. List all products Weartype, Material type along with their corresponding sizes:

SELECT t2.Weartype, t2.Materialtype, t3.Size1, t3.Size2, t3.Size3, t3.Size4

FROM Table2 t2

JOIN Table3 t3 ON t2.Sno = t3.Sno;

1. Find the average return time for each product type:

SELECT t1.Producttype, AVG(t3.Returntime) AS AverageReturnTime

FROM Table1 t1

JOIN Table3 t3 ON t1.Sno = t3.Sno

GROUP BY t1.Producttype;

1. Find Country of origin, return time, occasion of each product type:

SELECT t1.Producttype, t3.Countryoforigin, t3.Returntime, t3.Occasion

FROM Table1 t1

JOIN Table3 t3 ON t1.Sno = t3.Sno;