--Project2Solution.mysql

--Q1

SELECT Empid, Lastname

FROM Employees;

--Q2

SELECT Empid, Lastname

FROM Employees

Order by Empid Asc;

--Q3

SELECT Country, Cast(hiredate as Date) Hiredate, Count(\*) NumberofEmployees

FROM Employees

WHERE Hiredate >= '20030101'

GROUP BY Hiredate

HAVING Count(\*) > 1

ORDER BY Country Asc;

/\* SELECT Country, Hiredate, Count(\*) NumberofEmployees

FROM Employees

WHERE Hiredate >= '20030101'

GROUP BY Hiredate

HAVING Count(\*) >= 1

ORDER BY Country Asc;\*/

--Q4

SELECT \* FROM Orders;

--(a)

SELECT Shipperid, SUM(freight) AS totalfreight

FROM Orders

WHERE Shipperid >= 1

GROUP BY shipperid

HAVING SUM(Freight) > 20000.00

ORDER BY SUM(Freight) DESC;

/\*"Having" Function is used to Filter Derived Aggregate Column, Whereas "Where" is

used on the other hand to Filter sets of Records that Need to be Selected.

The Schemas for Order Table Does not Contain Sales\*/

--(b)

SELECT shipperid, SUM(freight) AS totalfrieght

FROM Orders

GROUP BY Shipperid

HAVING SUM(freight) > 20000.00;

--Q5

SELECT Shipperid, Companyname, Phone

FROM Shippers;

--Q6

SELECT S.Shipperid, S.Companyname, S.Phone

FROM Shippers S;

--Q7

SELECT Empid, Firstname, Lastname, Country, Region, City

FROM Employees

WHERE Region = 'WA';

--Q8

SELECT Empid, Firstname, Lastname, Country, Region, City

FROM Employees

WHERE Region is NULL;

--Q9

SELECT Orderid, Cast(Orderdate As Date) Orderdate, Custid, Empid

FROM Orders

WHERE Orderdate BETWEEN '20080211' AND '20080213';

SELECT Orderid, CONVERT(Orderdate, Date) Orderdate, Custid, Empid

FROM Orders

WHERE Orderdate BETWEEN '20080211' AND '20080213';

--Q10

SELECT Empid, Firstname, Lastname, City

FROM Employees

WHERE City LIKE '%a%';

SELECT \* FROM Employees;

-- PROJECT 3 SOLUTION

-- Q1

SELECT COUNT(\*) AS TOTALEMPLOYEES

FROM employees;

-- Q2

SELECT REGION, COUNT(\*) AS `TOTAL EMPLOYEES`

FROM employees

group by REGION;

-- Q3

SELECT Orderid, SUM(freight) AS TOTALAMOUNT

FROM orders

group by Orderid

order by sum(freight) desc;

-- Q4

select ORDERID, convert(orderdate, date) orderdate, -- Cast(ORDERDATE as date),

CONCAT(SHIPCITY, SHIPREGION) AS LOCATION

FROM ORDERS;

-- Q5

select concat(FIRSTNAME, ' ', LASTNAME) AS employeename, city,

left(city, 3) abbreviatedcity

from employees;

-- Q6

select lastname, firstname,

length(lastname) as namelength

from employees;

-- Q7 Option 1

SELECT supplierid, contactname, region,

coalesce(Region, 'CA')

FROM suppliers;

-- Q7 option 2

SELECT supplierid, contactname, region,

ifnull(Region, 'CA')

FROM suppliers;

-- Q8

select lastname, firstname,

year(now()) - year(hiredate) YearsOfEmployment

from employees;

--SELECT LASTNAME, FIRSTNAME, DATEDIFF(YEAR, CURRENT\_TIMESTAMP, HIREDATE) YEAROFEMP

FROM Employees;

--SELECT LASTNAME, FIRSTNAME,

DATEDIFF(YEAR, HIREDATE, CONVERT (DATE, CURRENT\_TIMESTAMP) EMPLOYMENTYEAR

FROM Employees;

--SELECT LASTNAME, FIRSTNAME,

DATEDIFF(YEAR, HIREDATE, CAST(CURRENT\_TIMESTAMP AS DATE)) EMPLOYMENTYEAR

FROM Employees;

-- Q9

select lastname, firstname, title,

year(now()) - year(birthdate) age

from employees;

select lastname, firstname, title,

year(getdate()) - year(birthdate) age

from employees;

-- Q10

select lastname, firstname, title,

year(now()) - year(birthdate) age

from employees

where year(now()) - year(birthdate)

>= (select avg(year(now()) - year(birthdate)) from employees);

-- Q11

select D.n Days, S.n Shift

from nums D

cross join nums S

where D.n <= 7 AND S.n <= 3

Order by Days, Shift asc;

-- Q12

select s.companyname, s.contactname, p.productname, s.country

from suppliers s

inner join products p

on s.supplierid = p.supplierid

where s.country = 'japan';

select s.companyname, s.contactname, p.productname, s.country

from suppliers s, products p

where s.supplierid = p.supplierid

and s.country = 'japan';

-- Q13

select e.empid,

concat(e.firstname, ' ', e.lastname) employeefullname,

concat(m.firstname, ' ', m.lastname) managername

from employees e

inner join employees m

on e.empid = m.mgrid;

-- Q14

select s.companyname, s.contactname, p.productname, s.country

from suppliers s

right join products p

on s.supplierid = p.supplierid

where s.country = 'japan';

-- Q15

select country, region, city

from employees

union

select country,

region, city

from customers;

-- Q16

select country,

region, city

from employees

union all

select country,

region, city

from customers;

-- Q17

select c.custid, c.contactname, o.orderid

from customers c left join orders o

on c.custid = o.custid

where o.orderid is null;

-- Q18

select c.custid, c.companyname, o.orderdate, o.orderid

from customers c

left outer join orders o

on c.custid = o.custid

where o.orderdate >= '20080201' AND o.orderdate <= '20080301';

select c.custid, c.companyname, o.orderdate, o.orderid

from customers c

left outer join orders o

on c.custid = o.custid

where o.orderdate > '20080131' AND o.orderdate < '20080301'

order by o.orderdate asc;

select c.custid, c.companyname, o.orderdate, o.orderid

from customers c

left outer join orders o

on c.custid = o.custid

where o.orderdate >= '20080201' AND o.orderdate <= '20080228'

order by o.orderdate asc;

select c.custid, c.companyname, o.orderdate, o.orderid

from customers c

left outer join orders o

on c.custid = o.custid

where o.orderdate between '20080201' AND '20080228';

PROJECT4-SOLUTION

--CREATING DATABASE

CREATE DATABASE employeeDB;

--Creating Tables

Create table empinfoTable

(EmpID Varchar(1) Not Null,

Firstname Varchar(8) Not Null,

Lastname Varchar(8) Not Null ,

Department Varchar(7) Not Null,

Project Char(2) Not Null,

Address Varchar(15) Not Null ,

DOB Char(10) Not Null,

Gender Char(1) Not Null);

Create table positionTable

(EmpID Varchar(1) Not Null,

Position Varchar(10) Not Null,

DateOfJoining Char(10) Not Null,

Salary Varchar(6) Not Null);

Alter Table positionTable

Alter Column Salary Money Not Null;

Drop table positionTable;

--Inserting Data into the Tables

Insert into empinfoTable

(EmpID, Firstname, Lastname, Department, Project, Address, DOB, Gender)

Values

('1', 'Sanjay', 'Mehra', 'HR', 'P1', 'Hyderabad(HYD)', '01/12/1976', 'M'),

('2', 'Ananya', 'Mishra', 'Admin', 'P2', 'Delhi(DEL)', '02/05/1968', 'F'),

('3', 'Rohan', 'Diwan', 'Account', 'P3', 'Mumbai(BOM)', '01/01/1980', 'M'),

('4', 'Sonia', 'Kulkarni', 'HR', 'P1', 'Hyderabad(HYD)', '02/05/1992', 'F'),

('5', 'Ankit', 'Kapoor', 'Admin', 'P2', 'Delhi(DEL)', '03/07/1994', 'M');

Insert into positionTable

(EmpID, Position, DateOfJoining, Salary)

Values

('1', 'Manager', '01/05/2022', '500000'),

('2', 'Executive', '02/05/2022', '75000'),

('3', 'Manager', '01/05/2022', '90000'),

('2', 'Lead', '02/05/2022', '85000'),

('1', 'Executive', '01/05/2022', '300000');

Select \* from empinfoTable;

Select \* from positionTable;

--Q1

/\*1. Write a query to fetch the EmpFname from the EmployeeInfo

table in the upper case and use the ALIAS name as EmpName.\*/

Select Upper(Firstname) EmpFirstname from empinfoTable;

--Q2

--2. Write a query to fetch the number of employees working in the department ‘HR’.

Select Department, Count(Empid) numberofEmployees

From empinfoTable

Where Department = 'HR'

Group by Department;

Select Count(\*)

From empinfoTable

Where Department = 'HR';

--Q3

--3. Write a query to get the current date.

Select Cast(Getdate() as Date) Currentdate;

Select Convert(Date, Getdate()) Currentdate;

Select Cast(Current\_Timestamp as Date) Currentdate;

Select Convert(Date, Current\_Timestamp) Currentdate;

Select Getdate() Currentdate;

Select Current\_Timestamp Currentdate;

--Q4

/\*4. Write a query to retrieve the first four characters of

EmpLname from the EmployeeInfo table.\*/

Select Lastname, SUBSTRING(Lastname, 1, 4) Abbrv from empinfoTable;

Select Lastname, Left(Lastname, 4) Abbrv from empinfoTable;

--Q5

/\*5. Write a query to fetch only the place name(string before brackets)

from the Address column of EmployeeInfo table.\*/

Select Address, Substring(Address, 0, Charindex('(',Address)) AddressPlaceName

From empinfoTable;

Select \*, Substring(Address, 0, Charindex('(',Address)) AddressPlaceName

From empinfoTable;

--Q6

/\*6. Write a query to create a new table that consists of

data and structure copied from the other table.\*/

/\*Create Table newEmployees

As Select EmpID, Firstname, Lastname, Department, Gender

From empinfoTable;\*/ FOR MYSQL

Select EmpID, Firstname, Lastname, Department, Gender

into newEmployees from empinfoTable;

Select \* into newEmployees from empinfoTable;

Select \* from newEmployees;

--Q7

--7. Write a query to find all the employees whose salary is between 50000 to 100000.

Select \* from positionTable

Where Salary between 50000 AND 100000;

Select \* from positionTable;

--Q8

--8. Write a query to find the names of employees that begin with ‘S’

Select \* from empinfoTable

Where Firstname like 'S%';

--Q9

--9. Write a query to fetch top N records.

Select \* from empinfoTable

Where Firstname between 'A' and 'N';

--Select Top N \* From positionTable Order by Salary Desc;

--Q10

/\*10. Write a query to retrieve the EmpFname and EmpLname in a single column as

“FullName”. The first name and the last name must be separated with space.\*/

Select EmpID, Concat(Firstname, ' ', Lastname) Fullname

From empinfoTable;

Select EmpID, Concat(Firstname,' ',Lastname) Fullname

From empinfoTable;

--Q11

/\*11. Write a query to find number of employees whose DOB is between

02/05/1970 to 31/12/1975 and are grouped according to gender\*/

Select Gender, Count(\*) EmployeeCount

from empinfoTable

where DOB between '02/05/1970' AND '31/12/1975'

Group by Gender;

Select DOB, Count(\*) EmployeeCount

from empinfoTable

where DOB between '02/05/1970' AND '31/12/1975'

Group by DOB;

Select Gender, DOB, Count(\*) EmployeeCount

from empinfoTable

where DOB between '02/05/1970' AND '31/12/1975'

Group by DOB, Gender;

Select DOB, Gender, Project, Address, Department, Count(\*) EmployeeCount

From empinfoTable

where DOB between '02/05/1970' AND '31/12/1975'

Group by DOB, Gender, Project, Address, Department;

--Q12

/\*12. Write a query to fetch all the records from the EmployeeInfo table ordered by

EmpLname in descending order and Department in the ascending order\*/

Select \* From empinfoTable

Order by Lastname Desc, Department Asc;

--Q13

/\*13. Write a query to fetch details of employees whose EmpLname

ends with an alphabet ‘A’ and contains five alphabets.\*/

Select \* From empinfoTable

Where Lastname Like '\_\_\_\_a';

--Q14

/\*14. Write a query to fetch details of all employees excluding the employees

with first names, “Sanjay” and “Sonia” from the EmployeeInfo table.\*/

Select \* From empinfoTable

Where Firstname NOT IN ('Sanjay', 'Sonia');

Select \* From empinfoTable

Where EmpID != 1 And EmpID != 4;

--Q15

--15. Write a query to fetch details of employees with the address as “DELHI(DEL)”.

Select \* From empinfoTable

Where Address = 'DELHI(DEL)';

Select \* From empinfoTable

Where Address Like 'DELHI(DEL)%';

--Q16

--16. Write a query to fetch all employees who also hold the managerial position.

Select e.Empid, e.Firstname, e.Lastname, e.Department, p.Position

From empinfoTable e

Inner Join PositionTable P

On e.Empid = p.Empid

Where Position = 'Manager';

Select e.Empid, e.Firstname, e.Lastname, p.Position

From empinfoTable e

Inner Join PositionTable P

On e.Empid = p.Empid

Where Position = 'Manager';

Select e.Empid, e.Firstname, e.Lastname, p.Position

From empinfoTable e

Inner Join PositionTable P

On e.Empid = p.Empid and Position = 'Manager';

Select e.Empid, e.Firstname, e.Lastname, p.Position

From empinfoTable e

Inner Join PositionTable P

On e.Empid = p.Empid and Position in ('Manager');

--Q17

/\*17. Write a query to fetch the department-wise count of employees sorted by

department’s count in ascending order\*/

Select Department, Count(Empid) DepartmentalCount

From empinfoTable

Group by Department

Order by Department Asc;

Select Department, Count(\*) DepartmentalCount

From empinfoTable

Group by Department

Order by DepartmentalCount Asc;

Select Department, Count(\*) DepartmentalCount

From empinfoTable

Group by Department

Order by Department Asc;

--Q18

--18. Write a query to calculate the even and odd records from a table.

Select Empid, Sum(Salary) TotalSalary

from positionTable

Group by Empid;

Select Empid, Sum(Salary) TotalSalary

from positionTable

Group by Empid;

SELECT Empid

FROM (Select rowno, Empid From empinfoTable)

WHERE MOD (rowno, 2)=1;

--Q19

/\*19. Write a SQL query to retrieve employee details from EmployeeInfo table

who have a date of joining in the EmployeePosition table\*/

SELECT \* FROM empinfoTable e

WHERE EXISTS

(Select \* From positionTable p Where e.Empid = p.Empid);

Select e.Empid, e.Firstname, e.Lastname, e.Department, e.Project,

e.Address, e.DOB, e.Gender, p.DateOfJoining, Count(\*) EmployeeCount

From empinfoTable e

Inner Join PositionTable P

On e.Empid = p.Empid

Group by e.Empid, e.Firstname, e.Lastname, e.Department, e.Project,

e.Address, e.DOB, e.Gender, p.DateOfJoining;

Select e.Empid, e.Firstname, e.Lastname, e.Department, e.Project, e.Gender,

Count(Gender) EmployeeCount

From empinfoTable e

Inner Join PositionTable P

On e.Empid = p.Empid

Group by e.Empid, e.Firstname, e.Lastname, e.Department, e.Project, e.Gender;

--Q20

/\*20. Write a query to retrieve two minimum and maximum

salaries from the EmployeePosition table.\*/

Select \* From positionTable

Where Empid = '2'

Order by Salary Asc;

Select Top 2 \* From positionTable

Order by Salary Desc;

Select \* From positionTable

Where Empid = '1'

Order by Salary Asc;

Select Distinct Salary From positionTable P1

where 2 >= (Select Count(Distinct Salary) From positionTable P2

Where P1.Salary >= P2. Salary)

Order by Salary Desc;

Select Distinct Salary From positionTable P1

where 2 >= (Select Count(Distinct Salary) From positionTable P2

Where P1.Salary <= P2. Salary)

Order by Salary Desc;

--Q21

/\*21. Write a query to find the Nth highest salary from the table

without using TOP/limit keyword.\*/

Select \* From positionTable

Where Empid = 1

Order by Salary Desc;

Select \* From positionTable

Where Empid in (1);

--Q22

--22. Write a query to retrieve duplicate records from a table.

Select Department, Project, Address, Count(\*) EmployeeCount

From empinfoTable

Group by Department, Project, Address

Having Count(\*) > 1;

Select Department, Count(\*) EmployeeCount

From empinfoTable

Group by Department

Having Count(\*) > 1

Order by Department Asc;

Select Position, Count(\*) EmployeePositionCount

From positionTable

Group by Position

Having Count(\*) > 1;

Select e.Empid, e.Firstname, e.Lastname, e.Department, e.Project,

e.Address, e.DOB, e.Gender, p.DateOfJoining, Count(\*) EmployeeCount

From empinfoTable e

Inner Join PositionTable P

On e.Empid = p.Empid

Group by e.Empid, e.Firstname, e.Lastname, e.Department, e.Project,

e.Address, e.DOB, e.Gender, p.DateOfJoining

Having Count(\*) > 1;

--Q23

--23. Write a query to retrieve the list of employees working in the same department.

Select Distinct E1.Empid, E1.Firstname, E1.Lastname, E1.Department

From empinfoTable E1, empinfoTable E2

Where E1.Department = E2.Department AND E1.Empid != E2.Empid;

Select Distinct E1.Empid, E1.Firstname, E1.Department

From empinfoTable E1

Join empinfoTable E2

On E1.Department = E2.Department AND E1.Empid != E2.Empid;

Select \*

From empinfoTable E1, empinfoTable E2

Where E1.Department = E2.Department AND E1.Empid != E2.Empid;

--Q24

--24. Write a query to retrieve the last 3 records from the EmployeeInfo table.

Select \* From empinfoTable;

Select \* From empinfoTable

Where Empid > (Select Count(\*) From empinfoTable) - 3;

Select \* From empinfoTable

Where Empid >= (Select Count(\*) From empinfoTable) - 2;

Select \* From empinfoTable

Where Empid > 2;

Select \* From empinfoTable

Where Empid in (3, 4, 5);

--Q25

--25. Write a query to find the third-highest salary from the EmpPosition table.

Select Top 1 Salary From

(Select Top 3 Salary From positionTable Order by Salary Desc) emp

Order by Salary Asc;

Select Top 1 Salary From

(Select Top 3 Salary From positionTable Order by Salary Asc) As empTable

Order by Salary Desc;

Select TOP 1 e.Empid, e.Firstname, e.Lastname, e.Department, e.Project,

e.Address, p.Salary, p.Position,

Row\_Number() over (Partition by Department order by Salary asc) Salary\_Rank

From empinfoTable e

Inner Join PositionTable P

On e.Empid = p.Empid;

Select e.Empid, e.Firstname, e.Lastname, e.Department, e.Project,

e.Address, p.Salary, p.Position,

Row\_Number() over (Partition by e.Empid order by Salary asc) Salary\_Rank

From empinfoTable e

Inner Join PositionTable P

On e.Empid = p.Empid

Where e.Empid = 3;

Select Empid, Position, DateOfJoining, Salary,

Row\_Number() Over(Partition by Empid Order by Salary Desc) Salary\_Rank

From positionTable

Where Empid = 3;

With Salaries\_Ranks As (Select Empid, Position, DateOfJoining, Salary,

Row\_Number() Over(Partition by Empid Order by Salary Desc) Salary\_Rank

From positionTable)

Select \* From Salaries\_Ranks

Where Empid = 3;

--Q26

/\*26. Write a query to display the first and the last record

from the EmployeeInfo table.\*/

Select \* from empinfoTable

Where Empid IN ((Select Min(Empid) From empinfoTable),

(Select Max(Empid) From empinfoTable));

Select \* from empinfoTable

Where Empid = (Select Min(Empid) From empinfoTable);

Select \* from empinfoTable

Where Empid = (Select Max(Empid) From empinfoTable);

Select \* from empinfoTable

Where Empid in (1, 5);

--Q27

--27. Write a query to add email validation to your database.

--Q28

/\*28. Write a query to retrieve Departments who have less

than 2 employees working in it.\*/

Select \*

From empinfoTable

Where Department IN

(Select Department

from empinfoTable

Group by Department

Having Count(\*) < 2);

Select Empid, Firstname, Lastname, Department, Project

From empinfoTable

Where Department IN (Select Department from empinfoTable

Group by Department

Having Count(\*) < 2);

Select Department, Count(\*) As EmployeeCounts

From empinfoTable

Group by Department

Having Count(Department) < 2;

Select Empid, Firstname, Lastname, Department, Project,

Row\_Number() Over(Partition by Department Order by firstname Asc) RowNumb

From empinfoTable

Where Empid = 3;

--Q29

/\*29. Write a query to retrieve EmpPostion along with total

salaries paid for each of them.\*/

Select Position, Sum(Salary) totalSalary

From positionTable

Group by Position

Order by totalSalary Desc;

--Q30

--30. Write a query to fetch 50% records from the EmployeeInfo table.

Select TOP 50 Percent \*

From empinfoTable;

Select \* From empinfoTable

Where Empid <= (Select Count(Empid)/2 From empinfoTable);

Select \* From positionTable;

Select \* from empinfoTable;

JOIN ASSIGNMENT SOLUTION

Using the E-R Diagram, write the SQL statements for the following questions.

1. Get all the orders placed by a specific customer. CustomerID for this customer is MAGAA.

SELECT \*

FROM tblCustomers C

JOIN tblOrders O

ON C.CustID = O.CustID

WHERE C.CustID = 'MAGAA';

2. Show customers whose ContactTitle is not Sales Associate. Display CustomerID, CompanyName, Contact Name and ContactTitle

SELECT CustID, CompanyName, ContactName, ContactTitle

FROM tblCustomers

WHERE ContactTitle <> 'Sales Associate';

3. Show customers who bought products where the EnglishName includes the string “chocolate”. Display CustomerID, CompanyName, ProductID, ProductName and EnglishName.

SELECT C.CustID, C.CompanyName, P.ProductID, P.ProductName, P.EnglishName

FROM tblCustomers C

JOIN tblOrders O

ON C.CustID = O.CustID

JOIN tblOrderDetails D

ON O.OrderID = D.OrderID

JOIN tblProducts P

ON D.ProductID = P.ProductID

WHERE P.EnglishName LIKE '%chocolate%';

4. Show products which were bought by customers from “Italy or USA”. Display CustomerID, CompanyName, ShipCountry, ProductID, ProductName and EnglishName’

SELECT C.CustomerID, C.CompanyName, O.ShipCountry, D.ProductID, P.ProductName, P.EnglishName

FROM tblCustomers C

JOIN tblOrders O

ON C.CustID = O.CustID

JOIN tblOrderDetails D

ON O.OrderID = D.OrderID

JOIN tblProducts P

ON D.ProductID = P.ProductID

WHERE O.ShipCountry IN ('Italy', 'USA');

5. Show total price of each product in each order. Note that there is not a column named as total price. You should calculate it and create a column named as TotalPrice. Display OrderID, ProductID, ProductName, UnitPrice, Quantity, Discount, and TotalPrice.

SELECT OrderID, P.ProductID, P.ProductName, D.UnitPrice, D.Quantity, D.Discount,

CAST(SUM(D.UnitPrice \* D.Quantiy \* (1 - D.Discount)) AS FLOAT(12,2)) TotalPrice

FROM tblOrders O

JOIN tblOrderDetail D

ON O.OrderID = D.OrderID

JOIN tblProducts P

ON D.ProductID = P.ProductID

GROUP BY OrderID, P.ProductID, P.ProductName, O.UnitPrice, O.Quantity, O.Discount;

6. Show the total number of customers in each City. Display Country, City, TotalCustomers.

SELECT Country, City, Count(\*) TotalCustomers

FROM tblCustomers

GROUP BY Country, City;

7. Show the orders which were shipped late than the actual required date. Display OrderID, OrderDate, RequiredDate and ShippedDate.

SELECT OrderID, OrderDate, RequiredDate, ShippedDate

FROM tblOrders

WHERE ShippedDate > RequiredDate;