04. DB-Basics-Data-Aggregation-Exercises

/\* 01RecordsCount\*/

SELECT COUNT(\*) AS [Count] FROM WizzardDeposits

/\* 02LongestMagicWand\*/

SELECT MAX(MagicWandSize) AS LongestMagicWand FROM WizzardDeposits

/\* 03LongestMagicWandPerDeositGroups\*/

SELECT DepositGroup, MAX(MagicWandSize) AS LongestMagicWand

FROM WizzardDeposits

GROUP BY DepositGroup

/\* 04SmallestDepositGroupPerMagicWandSize\*/

SELECT TOP(2) DepositGroup

FROM WizzardDeposits

GROUP BY DepositGroup

ORDER BY AVG(MagicWandSize)

/\* 05DepositsSum\*/

SELECT DepositGroup, SUM(DepositAmount) AS TotalSum

FROM WizzardDeposits

GROUP BY DepositGroup

/\* 06DepositsSumForOllivanderFamily\*/

SELECT DepositGroup, SUM(DepositAmount) AS TotalSum

FROM WizzardDeposits

WHERE MagicWandCreator = 'Ollivander family'

GROUP BY DepositGroup

--In Judge must be paste only one query

SELECT DepositGroup, SUM(DepositAmount) AS TotalSum

FROM WizzardDeposits

GROUP BY DepositGroup, MagicWandCreator

HAVING MagicWandCreator = 'Ollivander family'

/\* 07DepositsFilter\*/

SELECT DepositGroup, SUM(DepositAmount) AS TotalSum

FROM WizzardDeposits

WHERE MagicWandCreator = 'Ollivander family'

GROUP BY DepositGroup

HAVING SUM(DepositAmount) < 150000

ORDER BY TotalSum DESC

/\* 08DepositCharge\*/

SELECT DepositGroup, MagicWandCreator, MIN(DepositCharge) AS MinDepositCharge

FROM WizzardDeposits

GROUP BY DepositGroup, MagicWandCreator

ORDER BY MagicWandCreator, DepositGroup

/\* 09AgeGroups\*/

SELECT

CASE

WHEN w.Age BETWEEN 0 AND 10 THEN '[0-10]'

WHEN w.Age BETWEEN 11 AND 20 THEN '[11-20]'

WHEN w.Age BETWEEN 21 AND 30 THEN '[21-30]'

WHEN w.Age BETWEEN 31 AND 40 THEN '[31-40]'

WHEN w.Age BETWEEN 41 AND 50 THEN '[41-50]'

WHEN w.Age BETWEEN 51 AND 60 THEN '[51-60]'

WHEN w.Age >= 61 THEN '[61+]'

END AS AgeGroup,

COUNT(\*) AS WizzardCount

FROM WizzardDeposits AS w

GROUP BY CASE

WHEN w.Age BETWEEN 0 AND 10 THEN '[0-10]'

WHEN w.Age BETWEEN 11 AND 20 THEN '[11-20]'

WHEN w.Age BETWEEN 21 AND 30 THEN '[21-30]'

WHEN w.Age BETWEEN 31 AND 40 THEN '[31-40]'

WHEN w.Age BETWEEN 41 AND 50 THEN '[41-50]'

WHEN w.Age BETWEEN 51 AND 60 THEN '[51-60]'

WHEN w.Age >= 61 THEN '[61+]'

END

--In Judge must be paste only one query

SELECT g.AgeGroup, COUNT(\*) AS WizardCount FROM

(

SELECT

CASE

WHEN Age BETWEEN 0 AND 10 THEN '[0-10]'

WHEN Age BETWEEN 11 AND 20 THEN '[11-20]'

WHEN Age BETWEEN 21 AND 30 THEN '[21-30]'

WHEN Age BETWEEN 31 AND 40 THEN '[31-40]'

WHEN Age BETWEEN 41 AND 50 THEN '[41-50]'

WHEN Age BETWEEN 51 AND 60 THEN '[51-60]'

WHEN Age >= 61 THEN '[61+]'

END AS AgeGroup

FROM WizzardDeposits

) AS g

GROUP BY g.AgeGroup

/\* 10FirstLetter\*/

SELECT LEFT(FirstName, 1) AS FirstLetter

FROM WizzardDeposits

WHERE DepositGroup = 'Troll Chest'

GROUP BY LEFT(FirstName, 1)

ORDER BY FirstLetter

/\* 11AverageInterest\*/

SELECT DepositGroup, IsDepositExpired, AVG(DepositInterest) AS AverageInterest

FROM WizzardDeposits

WHERE DepositStartDate >= '01/01/1985'

GROUP BY DepositGroup, IsDepositExpired

ORDER BY DepositGroup DESC, IsDepositExpired

/\* 12RichWizardPoorWizard\*/

SELECT SUM(d.Diff) AS SumDifference

FROM (SELECT DepositAmount - (SELECT DepositAmount FROM WizzardDeposits WHERE Id = Host.Id + 1) AS Diff

FROM WizzardDeposits AS Host) AS d

--In Judge must be paste only one query

SELECT SUM(wd.Diff)

FROM (SELECT FirstName, DepositAmount,

LEAD(FirstName) OVER(ORDER BY Id) AS Wizard,

LAG(FirstName) OVER(ORDER BY Id) AS NextWizard,

LEAD(DepositAmount) OVER(ORDER BY Id) AS Deposit,

LAG(DepositAmount) OVER(ORDER BY Id) AS NextDeposit,

DepositAmount - LEAD(DepositAmount) OVER(ORDER BY Id) AS Diff

FROM WizzardDeposits)

AS wd

/\* 13DepartmentsTotalSalaries\*/

SELECT DepartmentID, SUM(Salary) AS TotalSalary

FROM Employees

GROUP BY DepartmentID

/\* 14EmployeesMinimumSalaries\*/

SELECT DepartmentID, MIN(Salary) AS MinimumSalary

FROM Employees

WHERE DepartmentID IN (2, 5, 7) AND HireDate > '01/01/2000'

GROUP BY DepartmentID

/\* 15EmployeesAverageSalaries\*/

SELECT \* INTO RichEmployees

FROM Employees

WHERE Salary > 30000

DELETE FROM RichEmployees

WHERE ManagerID = 42

UPDATE RichEmployees

SET Salary += 5000

WHERE DepartmentID = 1

SELECT DepartmentID, AVG(Salary) AS AverageSalary

FROM RichEmployees

GROUP BY DepartmentID

/\* 16EmployeesMaximumSalaries\*/

SELECT DepartmentID, MAX(Salary) AS MaxSalary

FROM Employees

GROUP BY DepartmentID

HAVING MAX(Salary) NOT BETWEEN 30000 AND 70000

/\* 17EmployeesCountSalaries\*/

SELECT COUNT(\*) AS [Count]

FROM Employees

WHERE ManagerID IS NULL

/\* 18ThirdHighestSalary\*/

SELECT s.DepartmentID, s.Salary

FROM

(

SELECT DepartmentID, Salary, DENSE\_RANK() OVER(PARTITION BY DepartmentID ORDER BY Salary DESC) AS Place

FROM Employees

GROUP BY DepartmentID, Salary

) AS s

WHERE Place = 3

GROUP BY s.DepartmentID, s.Salary

/\* 19SalaryChallenge\*/

SELECT TOP(10) FirstName, LastName, DepartmentID

FROM Employees AS e

WHERE Salary >

(

SELECT AVG(Salary)

FROM Employees AS e1

WHERE e.DepartmentID = e1.DepartmentID

GROUP BY DepartmentID

)