# Вимоги

Назва файлу: Номер практичної роботи + \_ + Група + \_ + Прізвище + \_ + Ініціали.docx

В іменах файлів використовуємо тільки латинський алфавіт.

Приклад: 4\_IP96\_Nedashkivskyi\_YA.docx

Відповідь пишеться відразу після завдання курсивом. Питання не видаляються.

# Завдання:

Використовуючи MySQL в AWS, що було створено під час другої лабораторної роботи, або створивши новий екземпляр замість видаленого\*:

1. Відобразити назву продукту та його вартість. Якщо товар відсутній на складі – відобразити замість вартості N/A.

*select ProductName, if(UnitsInStock = 0, 'N/A', UnitPrice) as Price from Products;*

1. У дві колонки вивести співробітників, так їх керівників. Якщо керівник відсутній – повідомити про Self-managed.
2. Вивести номери замовлення та їх регіон відправки. Якщо такий відсутній – країну відправки. Якщо дати відправки немає – відобразити замість регіону фразу Not shipped.

*select OrderID,*

*case*

*when ShipRegion is not null*

*then ShipRegion*

*when ShipRegion is null*

*then ShipCountry*

*when OrderDate is null*

*then 'Not shipped'*

*end as Region*

*from Orders;*

1. Повернути з бази даних наступні дані: повне ім’я співробітника (однією колонкою), назву території, за яку він відповідає, до останньої через пробіл додати в дужках індикатор відповідного регіону (Nord, Sud, Est, Ovest). Приклад: Phoenix (Ovest).

*select concat(TitleOfCourtesy, ' ', FirstName, ' ', LastName) as FullName,*

*case*

*when r.RegionDescription = 'Eastern'*

*then concat(trim(t.TerritoryDescription), ' (Est)')*

*when r.RegionDescription = 'Westerns'*

*then concat(trim(t.TerritoryDescription), ' (Ovest)')*

*when r.RegionDescription = 'Northern'*

*then concat(trim(t.TerritoryDescription), ' (Nord)')*

*when r.RegionDescription = 'Southern'*

*then concat(trim(t.TerritoryDescription), ' (Sud)')*

*else trim(t.TerritoryDescription)*

*end as Territory*

*from Employees e*

*join EmployeeTerritories et on e.EmployeeID = et.EmployeeID*

*join Territories t on et.TerritoryID = t.TerritoryID*

*join Region r on t.RegionID = r.RegionID;*

1. Вивести по три найдешевших товару для кожної категорії.

*select CategoryID, ProductName, UnitPrice from (*

*select ProductName,*

*CategoryID,*

*UnitPrice,*

*row\_number() over (partition by CategoryID order by UnitPrice) as c\_rank*

*from Products*

*) ranks*

*where ranks.c\_rank <= 3;*

1. Вивести наступну інформацію: Країна, Ранг. Відсортувати за рангом. Ранжування провести по загальній вартості товарів відправлених в цю країну.

*select o.ShipCountry,*

*rank() over (order by sum(od.UnitPrice \* (1 - od.Discount) \* od.Quantity)) as rank\_place*

*from Orders o*

*join `Order Details` od on o.OrderID = od.OrderID*

*group by o.ShipCountry*

*order by rank\_place;*

1. Вивести окремими стовпчиками прізвище та ім’я співробітників Northwind та контактних осіб замовників, що мають посади спільні для обох таблиць. В якості третьої колонки вивести саму посаду.

*with title\_cte as (*

*select Title*

*from Employees*

*where Title in (*

*select ContactTitle from Customers*

*)*

*)*

*select FirstName, LastName, Title*

*from Employees*

*where Title in (*

*select \* from title\_cte*

*)*

*union*

*select substring\_index(ContactName, ' ', 1),*

*substring\_index(ContactName, ' ', -1),*

*ContactTitle*

*from Customers*

*where ContactTitle in (*

*select \* from title\_cte*

*);*

1. Вивести прізвище та ім’я співробітника, рік, номер замовлення в базі, та яким за рік стало це замовлення для конкретного співробітника (починаючи нумерацію з одиниці).

*select FirstName,*

*LastName,*

*OrderID,*

*row\_number() over (partition by o.EmployeeID, year(o.OrderDate) order by o.OrderID ) as OrderRank*

*from Orders o*

*join Employees e on o.EmployeeID = e.EmployeeID;*

1. Для кожного замовника знайти три замовлення з максимальною різницею між датою замовлення та датою відправлення.

*with all\_ranks\_cte as (*

*select c.CustomerID,*

*o.OrderID,*

*row\_number() over (partition by o.CustomerID order by datediff(o.OrderDate, o.ShippedDate) desc) as OrderRank*

*from Customers c*

*join Orders o on c.CustomerID = o.CustomerID*

*)*

*select \* from all\_ranks\_cte where OrderRank <= 3;*

1. Для кожного співробітника вивести другий десяток найбільш дорогих замовлень (тобто замовлення, що за загальною вартістю для конкретного співробітника будуть під номерами з 11 по 20).

*with employee\_total\_cost\_cte as*

*(*

*select e.EmployeeID,*

*o.OrderID,*

*sum(od.UnitPrice \* (1 - od.Discount) \* od.Quantity) as TotalCost*

*from Orders o*

*join Employees e on e.EmployeeID = o.EmployeeID*

*join `Order Details` od on o.OrderID = od.OrderID*

*group by e.EmployeeID, o.OrderID*

*)*

*select \* from (*

*select \*, row\_number() over (partition by EmployeeID order by TotalCost desc) as OrderRank*

*from employee\_total\_cost\_cte*

*) ranks*

*where OrderRank >= 11 and OrderRank <= 20;*

**Матеріали для повторення:**

[Cloud Databases (practice) pt4](https://youtu.be/s2ZPPiAFFqs) (NULL, JOIN, UNION, GROUP BY, агрегатні функції та текстові функції)

[Cloud Databases (practice) pt5](https://youtu.be/7pa9bP9zrnI) (INTERSECT, EXCEPT, ISNULL/IFNULL, COALESCE, IF, CASE, ROW\_NUMBER, RANK, DENSE\_RANK)

[Database Skill School 4](https://youtu.be/9R1mC3YQKQ8) (Агрегатні функції, GROUP BY, HAVING, текстові функції)

[Database Skill School 5](https://youtu.be/Zgm5qH9FhJU) (Вкладеність підзапитів, віконні функції ROW\_NUMBER, RANK)

[Database Skill School 6](https://youtu.be/9kbDpxBHEuc) (INTERSECT, EXCEPT, ISNULL/IFNULL, COALESCE, IF, CASE)

[Функції CASE, IF, IFNULL, COALESCE (рос.)](https://metanit.com/sql/mysql/6.4.php)

*\*будь-ласка впевніться, що ви використовуєте саме безкоштовний варіант (free tier).*