select \* from table\_1 order by Department, c2

select count (\*) as Количество from table\_1 where department='Маркетинг'

select avg (c2) as Количество from table\_1 where department='Маркетинг'

select department, avg (c2) as Количество from table\_1 group by department

1. SELECT model, COUNT(model) AS Qty\_model,
2. AVG(price) AS Avg\_price
3. FROM PC
4. GROUP BY model
5. HAVING AVG(price) < 800; --условие на агрегатные значения

Ниже приведен порядок обработки предложений в операторе **SELECT**:

1. FROM
2. WHERE
3. GROUP BY
4. HAVING
5. SELECT
6. ORDER BY

Between

SELECT e.FirstName, e.LastName, ep.Rate

FROM HumanResources.vEmployee e

JOIN HumanResources.EmployeePayHistory ep

ON e.BusinessEntityID = ep.BusinessEntityID

WHERE ep.Rate BETWEEN 27 AND 30

ORDER BY ep.Rate;

SELECT e.FirstName, e.LastName, ep.Rate

FROM HumanResources.vEmployee e

JOIN HumanResources.EmployeePayHistory ep

ON e.BusinessEntityID = ep.BusinessEntityID

WHERE ep.Rate > 27 AND ep.Rate < 30

ORDER BY ep.Rate;

SELECT e.FirstName, e.LastName, ep.Rate

FROM HumanResources.vEmployee e

JOIN HumanResources.EmployeePayHistory ep

ON e.BusinessEntityID = ep.BusinessEntityID

WHERE ep.Rate NOT BETWEEN 27 AND 30

ORDER BY ep.Rate;

IN

SELECT p.FirstName, p.LastName, e.JobTitle

FROM Person.Person AS p

JOIN HumanResources.Employee AS e

ON p.BusinessEntityID = e.BusinessEntityID

WHERE e.JobTitle = 'Design Engineer'

OR e.JobTitle = 'Tool Designer'

OR e.JobTitle = 'Marketing Assistant';

SELECT p.FirstName, p.LastName, e.JobTitle

FROM Person.Person AS p

JOIN HumanResources.Employee AS e

ON p.BusinessEntityID = e.BusinessEntityID

WHERE e.JobTitle IN ('Design Engineer', 'Tool Designer', 'Marketing Assistant');

SELECT p.FirstName, p.LastName

FROM Person.Person AS p

JOIN Sales.SalesPerson AS sp

ON p.BusinessEntityID = sp.BusinessEntityID

WHERE p.BusinessEntityID IN

(SELECT BusinessEntityID

FROM Sales.SalesPerson

WHERE SalesQuota > 250000);

SELECT p.FirstName, p.LastName

FROM Person.Person AS p

JOIN Sales.SalesPerson AS sp

ON p.BusinessEntityID = sp.BusinessEntityID

WHERE p.BusinessEntityID NOT IN

(SELECT BusinessEntityID

FROM Sales.SalesPerson

WHERE SalesQuota > 250000);

LIKE

|  |  |  |
| --- | --- | --- |
| **Символ-шаблон** | **Описание** | **Пример** |
| % | Любая строка длиной от нуля и более символов. | Инструкция WHERE Название LIKE '%компьютер%' выполняет поиск и выдает все названия книг, содержащие слово «компьютер». |
| \_ (подчеркивание) | Любой одиночный символ. | Инструкция WHERE фамилия\_автора LIKE '\_етров' выполняет поиск и выдает все имена, состоящие из шести букв и заканчивающиеся сочетанием «етров» (Петров, Ветров и т.п.). |
| [ ] | Любой одиночный символ, содержащийся в диапазоне ([a-f]) или наборе ([abcdef]). | Инструкция WHERE Фамилия\_автора LIKE '[Л-С]омов' выполняет поиск и выдает все фамилии авторов, заканчивающиеся на «омов» и начинающиеся на любую букву в промежутке от «Л» до «С», например Ломов, Ромов, Сомов и т.п. При выполнении операции поиска в диапазоне символы, включенные в диапазон, могут изменяться в зависимости от правил сортировки параметров сортировки. |
| [^] | Любой одиночный символ, не содержащийся вдиапазоне ([^a-f]) или наборе ([^abcdef]). | Инструкция WHERE Фамилия\_автора LIKE 'ив[^а]%' выполняет поиск и выдает все фамилии, начинающиеся на «ив», в которых третья буква отличается от «а». |

SELECT [Company Name], [Contact Name], Phone

FROM Customers

WHERE (Phone LIKE '9%')

ORDER BY [Contact Name]