Конспект по предмету Базы данных

Базы данных – это информационная панель

Сетевая база данных отличается от иерархической тем, что у дочернего элемента может быть несколько родителей.

**Реляционные базы данных**

Объекты в таких базах данных храниться

Select \* from <table> where <col=value>;

**Работа с 2 таблицами**

|  |  |  |  |
| --- | --- | --- | --- |
| **Покупатель** | **Товар** | **Цена** | **Цвет** |
| Иванов | iPhone | 80000 | Черный |
| Александров | Xiaomi | 30000 | Золотой |
| Шиммич | Samsung | 60000 | Синий |
| Федотов | Sony | 35000 | Белый |
| Шишкин | iPhone | 60000 | Синий |

|  |  |  |  |
| --- | --- | --- | --- |
| **Покупатель** | **Товар** | **Цена** | **Цвет** |
| Степнов | Honor | 20000 | Серебристый |
| Федотов | Sony | 25000 | Синий |
| Шиммич | Samsung | 60000 | Черный |
| Шитов | Samsung | 40000 | Черный |
| Федотов | Samsung | 50000 | Золотой |
| Шитов | iPhone | 70000 | Белый |
| Шиммич | iPhone | 80000 | Белый |

Операции Алгебры БД

1. Объединение
2. Пересечение
3. Разность
4. Симметрическая разность

Симметрическая разность – это обратная операция к пересечению, либо объединение 2 видов разности.

Эти операции можно выполнить над таблицами с одинаковой структурой.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № заказа | Покупатель | Товар | Кол-во | Цена | Общ.cумма |
| 1011 | Иванов | iPhone | 1 | 80000 | 80000 |
| 1012 | Сидоров | Xiaomi | 3 | 20000 | 60000 |
| 1013 | Алексеев | Sony | 2 | 30000 | 60000 |

Нормальная форма:

Связь один-ко-многим (через номер заказа).

Пользователи

|  |  |
| --- | --- |
| Id | Фамилия |
| 1 | Иванов |
| 2 | Сидоров |
| 3 | Алексеев |

Таблица заказов:

|  |  |
| --- | --- |
| Id | Заказы |
| 1 | 1011 |
| 2 | 1012 |
| 3 | 1013 |

Товары

|  |  |  |  |
| --- | --- | --- | --- |
| Id товара | Товар | Цена за шт | На складе |
| 1 | iPhone | 80000 | 10 |
| 2 | Xiaomi | 20000 | 15 |
| 3 | Sony | 30000 | 20 |

Информация заказов:

|  |  |
| --- | --- |
| № заказа | Пользователь |
| 1011 | 1 |
| 1012 | 2 |
| 1013 | 3 |

Таблица инверторя заказа

|  |  |  |
| --- | --- | --- |
| № заказа | Товар | Кол-во |
| 1011 | iPhone | 1 |
| 1012 | Xiaomi | 3 |
| 1013 | Sony | 2 |

Есть 2 вида фильтрации:

**По столбцам** (когда требуется выбрать определенные столбцы).

SELECT au\_lname + ' ' + au\_fname FROM authors

В предложении SELECT указываются имена столбцов ( в каком порядке укажете – в таком порядке и выведет), если нужны все столбцы ставится \*.

Так же столбцам можно давать свои имена

SELECT au\_lname AS 'Фамилия автора'

FROM authors -- откуда брать данные (название таблицы)

**По строкам**

SELECT ContactName FROM Customers

WHERE City='London'

SELECT LEN('Kolledge and FU');

SELECT REVERSE('Kolledge');

SELECT LEFT('Kolledge', 3)

SELECT RIGHT('Kolledge',3);

SELECT ContactName

FROM Customers

WHERE LEFT(ContactName,1) = 'A'

OR LEFT(ContactName,1) = 'D'

OR LEFT(ContactName,1) = 'X'

Или

SELECT ContactName

FROM Customers

WHERE LEFT(ContactName,1) IN ('A','D','X')

SELECT LastName

FROM Employees

WHERE LEFT(LastName,1) BETWEEN 'A' AND 'G'

--IN ('A','B','C','D','E','F','G')

SELECT ContactName

FROM Customers

WHERE ContactName LIKE 'A\_n%'

SELECT GETDATE();

SELECT DATEDIFF(SECOND,'04.04.2000',GETDATE())

SELECT YEAR(GETDATE())

SELECT \* FROM ORDERS WHERE ORDERDATE BETWEEN '01.03.1997' AND '31.05.1997'

SELECT \* FROM ORDERS WHERE YEAR(OrderDate) = 1997 AND MONTH(OrderDate) IN (3,4,5)

SELECT \* FROM ORDERS WHERE OrderDate>='19970301' AND OrderDate<='19970531'

Конверт типов

SELECT CAST(1 AS varchar) + '1'

SELECT 1 + CAST('1' AS INT)

SELECT CONVERT(VARCHAR, 1) + '1'

SELECT 1 + CONVERT(INT,'1')

SELECT CONVERT(DATE,GETDATE(),2)

SELECT CompanyName+ ' | ' + City+ ' | ' + (ISNULL(CAST(Fax AS VARCHAR),'(No Fax)'))

FROM Customers WHERE City = 'London'

SELECT CompanyName+ ' | ' + City+ ' | ' + (ISNULL(Fax,'(No Fax)'))

FROM Customers WHERE City = 'London'

ORDER BY Name\_and\_Fax DESC -- по убыванию

ORDER BY Name\_and\_Fax ASC -- по возрастанию

ORDER BY Name\_and\_Fax -- по возрастанию

SELECT TOP (5) \* FROM titles ORDER BY price DESC

Самостоятельная работа №1

1)

SELECT \* FROM ORDERS WHERE Odate >=‘20001003’AND Odate <= ‘20001004’

2)

SELECT \* FROM CUSTOMERS WHERE LEFT(Cname,1) BETWEEN ‘A’ AND ‘G’

3)

SELECT \* FROM CUSTOMERS WHERE LEFT(Cname,1) IN (‘C’,’c’)

4)

SELECT \* FROM ORDERS WHERE (Amt is NULL) OR (Amt = 0)

1)

SELECT Onum, Amt, Odate FROM ORDERS

2)

SECECT \* FROM CUSTOMERS WHERE Snum = 1001

3)

SELECT City, Sname, Snum, Comm FROM Customers

4)

SELECT DISTINCT Snum FROM Orders

5)

SELECT Sname FROM Salespeople WHERE (Comm > 0.10) AND (City = ‘London’)

6)

SELECT \* FROM Customers WHERE (Rating > 100) OR (City = ‘Rome’)

7)

3001

3003

3007

3009

3008

3010

3011

8)

3001

3003

3006

3009

3007

3008

3010

3011

9)

SELECT \* FROM Salespeople

SELECT CategoryID, Count(\*) AS 'Кол-во' FROM Products GROUP BY CategoryID

Групбай переходит от списка одних объектов к списку других объектов

SELECT CategoryID, COUNT(\*) AS kol\_vo, AVG(UnitPrice) AS AvgPrice, MAX(UnitPrice) AS MaxPrice

FROM Products

GROUP BY CategoryID

SELECT DISTINCT ShipCity

FROM Orders

WHERE YEAR(ShippedDate) = 1996

SELECT TOP(5) WITH Ties ShipCity

FROM Orders

WHERE YEAR(OrderDate) = 1997 AND MONTH(OrderDate) IN (3,4,5)

ORDER BY OrderDate DESC

SELECT CategoryID, SUM(UnitPrice \* UnitsOnOrder) as Earning

FROM Products

GROUP BY CategoryID

SELECT CategoryID,(

SELECT CategoryName FROM Categories

WHERE CategoryID = Products.CategoryID

) as CategoryName,

COUNT(ProductName) as NumOfGoods

FROM Products GROUP BY CategoryID

-- Сколько заказов отправил каждый продавец в Лондон в 1997 году?

SELECT EmployeeID, (

SELECT LASTNAME FROM Employees

WHERE EmployeeID=Orders.EmployeeID),

COUNT(\*)

FROM Orders

WHERE ShipCity = 'London' AND YEAR(ShippedDate) = 1997

GROUP BY EmployeeID

SELECT EmployeeID, LastName, (

SELECT COUNT(\*) FROM Orders

WHERE EmployeeID=Employees.EmployeeID AND ShipCity = 'London' AND YEAR(ShippedDate) = 1997

) AS CountOrders FROM Employees

ORDER BY EmployeeID

Одинаковые запросы, 2 более оптимизированный.

Прежде чем писать запрос содержащий вложенный, нужно ответить на вопрос список чего требуется получить, и тогда внешний запрос пишется к той таблице которая содержит список этих объектов.

-- В каких категориях средняя цена товара больше 30?

SELECT CategoryID, CategoryName, (

SELECT AVG(UnitPrice)

FROM Products

WHERE CategoryID = Categories.CategoryID

) as AvgPrice

FROM Categories

WHERE (

SELECT AVG(UnitPrice)

FROM Products

WHERE CategoryID = Categories.CategoryID

) > 30