

## SQL

### SQL SELECT

```
SELECT
[DISTINCT | ALL] <column-list>
FROM <table-names>
[WHERE <condition>]
[ORDER BY <column-list>]
[GROUP BY <column-list>]
[HAVING <condition>]
```

[ ] - не е задължително    / - или

1

## SQL

### Псевдоними (Aliases)

- Псевдонимите преименуват колони или таблици за да
  - правят имената по-съдържателни
  - правят имената по-кратки и лесни за писане
  - решават проблема с еднаквите имена
- Две форми:
  - Псевдоним за колона  

```
SELECT column AS newName...
```
  - Псевдоним за таблица  

```
SELECT ... FROM table AS newName
```

2

## SQL

Пример:

```
SELECT
E.ID AS empID,
E.Name, W.Dep
FROM
Employee E
Working W
WHERE
E.ID = W.ID
```

Employee

ID	Name
1	Пенев
2	Ганчева

Working

ID	Dep
1	Маркетинг
2	Продажби
2	Маркетинг

3

## SQL

Пример:

```
SELECT
E.ID AS empID,
E.Name, W.Dep
FROM
Employee E
Working W
WHERE
E.ID = W.ID
```

empID	Name	Dep
1	Пенев	Маркетинг
2	Ганчева	Маркетинг
2	Ганчева	Продажби

4

## SQL

Псевдоними ...

- Псевдонимите могат да се използват за копиране на таблица и така таблицата да може да бъде комбинирана сама със себе си:

```
SELECT A.Name FROM Employee A, Employee B
WHERE A.Dept=B.Dept AND B.Name='Пенев'
```

Employee

Name	Dep
Пенев	Маркетинг
Ганчева	Маркетинг
Ганчева	Продажби
Петкова	Маркетинг

5

## SQL

```
SELECT A.Name FROM Employee A, Employee B
WHERE A.Dept=B.Dept AND B.Name='Пенев'
```

A

Name	Dep
Пенев	Маркетинг
Ганчева	Маркетинг
Ганчева	Продажби
Петкова	Маркетинг

B

Name	Dep
Пенев	Маркетинг
Ганчева	Маркетинг
Ганчева	Продажби
Петкова	Маркетинг

6

# SQL

**SELECT ... FROM Employee A, Employee B ...**

A.Name	A.Dep	B.Name	B.Dep
Пенев	Маркетинг	Пенев	Маркетинг
Ганчева	Маркетинг	Пенев	Маркетинг
Ганчева	Продажби	Пенев	Маркетинг
Петкова	Маркетинг	Пенев	Маркетинг
Пенев	Маркетинг	Ганчева	Маркетинг
Ганчева	Маркетинг	Ганчева	Маркетинг
Ганчева	Продажби	Ганчева	Маркетинг
Петкова	Маркетинг	Ганчева	Маркетинг
Пенев	Маркетинг	Ганчева	Продажби
Ганчева	...	...	...

7

# SQL

**SELECT ... FROM Employee A, Employee B  
WHERE A.Dep = B.Dep**

A.Name	A.Dep	B.Name	B.Dep
Пенев	Маркетинг	Пенев	Маркетинг
Ганчева	Маркетинг	Пенев	Маркетинг
Петкова	Маркетинг	Пенев	Маркетинг
Пенев	Маркетинг	Ганчева	Маркетинг
Ганчева	Маркетинг	Ганчева	Маркетинг
Петкова	Маркетинг	Ганчева	Маркетинг
Ганчева	...	...	...

8

# SQL

**SELECT ... FROM Employee A, Employee B  
WHERE A.Dep = B.Dep AND B.Name = 'Пенев'**

A.Name	A.Dep	B.Name	B.Dep
Пенев	Маркетинг	Пенев	Маркетинг
Ганчева	Маркетинг	Пенев	Маркетинг
Петкова	Маркетинг	Пенев	Маркетинг

9

# SQL

**SELECT A.Name FROM Employee A, Employee B  
WHERE A.Dep = B.Dep AND B.Name = 'Пенев'**

A.Name
Пенев
Ганчева
Петкова

Резултатът е – всички служители, които работят с Пенев!

10

# SQL

## Подзадачи

Един **SELECT** може да бъде използван в друга заявка  
**SELECT** – подзаявка

Резултатите от такава подзаявка се подават обратно на викащата заявка

Напр. Да се изведат всички служители, които работят с Пенев :

**SELECT Name FROM Employee WHERE Dep =  
(SELECT Dep FROM Employee WHERE Name='Пенев')**

11

# SQL

## Подзаявки

**SELECT Name FROM Employee WHERE Dep =  
(SELECT Dep FROM Employee WHERE Name='Пенев')**

Първо подзаявката се изчислява и връща стойност 'Маркетинг'

Резултатът се подава на главната заявка

**SELECT Name FROM Employee WHERE Dep =  
'Маркетинг'**

12

Name	Dep
Пенев	Маркетинг
Ганчева	Маркетинг
Ганчева	Продажби
Петкова	Маркетинг

## SQL

### Подзааявки

- Подзааявката може да върне не една, а няколко стойности
- Една стойност не може да се сравнява с множество стойности!!!
- Възможности:
  - IN** – проверява дали една стойност я има в множество стойности
  - EXISTS** – проверка дали едно множество е празно
  - ALL/ANY** - проверка дали една релация е за всеки член на множеството или за някои членове

13

## SQL

### (NOT) IN

- Използването на **IN** позволява проверка дали дадена стойност е в множество от стойности
- NOT IN** проверка дали дадена стойност не е в множество от стойности
- Множеството се задава изрично или чрез подзааявка:  
**SELECT <columns> FROM <tables> WHERE <value> IN <set>**

**SELECT <columns> FROM <tables> WHERE <value> NOT IN <set>**

14

## SQL

### (NOT) IN

**SELECT \* FROM Employee WHERE Dep IN ('Маркетинг', 'Продажби')**

#### Employee

Name	Dep	Manager	Name	Dep	Manager
Пенев	Маркетинг	Сомноев	Пенев	Маркетинг	Сомноев
Ганчева	Маркетинг	Сомноев	Ганчева	Маркетинг	Сомноев
Сомноев	Маркетинг	Петкова	Сомноев	Маркетинг	Петкова
Петкова	Продажби	Петкова	Петкова	Продажби	Петкова
Маринов	Мениджмънт				

15

## SQL

### (NOT) IN

**SELECT \* FROM Employee WHERE Name NOT IN (SELECT Manager FROM Employee)**

#### Employee

Name	Dep	Manager
Пенев	Маркетинг	Сомноев
Ганчева	Маркетинг	Сомноев
Сомноев	Маркетинг	Петкова
Петкова	Продажби	Петкова
Маринов	Мениджмънт	

16

## SQL

Първо подзааявката се изчислява:

**SELECT Manager FROM Employee**

Manager
Сомноев
Сомноев
Петкова
Петкова

17

## SQL

### (NOT) IN

Това е

**SELECT \* FROM Employee WHERE Name NOT IN ('Сомноев', 'Петкова')**

Name	Dep	Manager
Пенев	Маркетинг	Сомноев
Ганчева	Маркетинг	Сомноев
Маринов	Мениджмънт	

18

# SQL

**SELECT \* FROM Product WHERE Price**  
**IN (100,250,300)**

**SELECT \* FROM Product WHERE Price**  
**BETWEEN 100 AND 300**

Name	Price	Qty
Wheel	100	1020
Bike spec	150	38
Bike	250	450

19

# SQL

**(NOT) EXISTS**

- Използването на EXISTS е проверка дали има поне един елемент в множеството
- NOT EXISTS дава вятно (**true**) ако множеството е празно
- Множеството винаги се задава с подзаявка

**SELECT <columns> FROM <tables> WHERE EXISTS <set>**

**SELECT <columns> FROM <tables> WHERE NOT EXISTS <set>**

20

# SQL

**(NOT) EXISTS**

**SELECT \* FROM Employee AS E1**  
**WHERE EXISTS ( SELECT \* FROM Employee AS E2**  
**WHERE E2.Name = E1.Manager)**

Name	Dep	Manager	Name	Dep	Manager
Пенев	Маркетинг	Сомноев	Сомноев	Маркетинг	Петкова
Ганчева	Маркетинг	Сомноев	Петкова	Продажби	Петкова
Сомноев	Маркетинг	Петкова			
Петкова	Продажби	Петкова			
Маринов	Мениджмънт				

21

# SQL

**ANY и ALL**

- ANY и ALL сравняват една стойност с множество стойности
- Използват се с оператори за сравнение: =, >, <, <>, >=, <=
- **val = ANY (set) е истина (true)**, ако има поне един член на множеството, който да е равен на стойността val
- **val = ALL (set) е истина (true)**, ако всички членове на множеството са равни на стойността val

22

# SQL

**ALL**

Да се намерят имената на служителите (един или няколко), които получават най-високата заплата.

**SELECT Name FROM Employee WHERE Salary >=**  
**ALL (SELECT Salary FROM Employee)**

Name	Salary
Иванов	2,000
Петрова	5,000
Жечев	2,500
Стоянова	3,000

23

# SQL

**ANY**

Да се намерят имената на служителите (един или няколко), които получават повече от някой друг.

**SELECT Name FROM Employee WHERE Salary >**  
**ANY ( SELECT Salary FROM Employee)**

Name	Salary
Иванов	2,000
Петрова	5,000
Жечев	2,500
Стоянова	3,000

24

## SQL

### Търсене на символен низ (дума)

- Търсене на символен низ (дума)
  - Обикновено се използва при търсене на продукт в каталог
  - Желателно е да се търси по ключова дума
  - Желателно е за гъвкаво търсене - стоп думи
- Дадена база данни с книги
  - Търсене с "сгупт" ще върне
    - "Cryptonomicon" by Neil Stephenson
    - "Applied Cryptography" by Bruce Schneier

25

## SQL

### Търсене на символен низ (дума)

- За целта може да се пазят:
  - Таблица с продуктите, между които ще се търси
  - Таблица с ключови думи
  - Таблица, свързваща двете (кои ключови думи към кои продукти принадлежат)

Items    ItemID    ItemTitle    .....

Keywords    KeyID    KeyWord

ItemKey    ItemID    KeyID

26

## SQL

### Търсене на символен низ (дума)

За такива заявки се използват запитвания като:

```
SELECT * FROM Items WHERE itmID IN (
SELECT itmID FROM ItemKey WHERE keyID IN (
SELECT keyID FROM Keywords WHERE keyWord
LIKE 'crypt%'))
```

27

## SQL

### Търсене на символен низ (дума)

Търсене на множество от думи

За да намерите едно множество от думи се използва AND

За да намерите поне една от думите се използва OR

```
SELECT * FROM Items WHERE itmID IN (
SELECT itmID FROM ItemKey WHERE keyID IN (
SELECT keyID FROM Keywords WHERE keyWord LIKE
'word1%'))
```

```
AND itmID IN ( SELECT itmID FROM ItemKey
WHERE keyID IN ( SELECT keyID FROM Keywords
WHERE keyWord LIKE 'word2%'))
```

28

## БАЗИ ДАННИ

### За тази лекция:

Songs				
cID	Num	Name	Sec	aID
1	1	Момиче	336	1
1	2	Слънчев ден	152	1
1	3	Югата	233	1
2	1	Луната спи	128	2

CD		
cID	Name	Price
1	Всеки	10,00
2	Сингъл	12,00

Artist	
aID	Name
1	Младен
2	Нети

доц. Стоянова

29

## БАЗИ ДАННИ

- Намерете списък от имената на изпълнители, които имат изпълнения върху диск (CD) с име "Всеки"...

Забележка:

Въпросът го има и в миналата лекция. Решете сега задачата като използвате подзаявки и за един SELECT има само една таблица.

доц. Стоянова

30

