

2 Април, ЧЮНН КОНТРОЛНН

• как изглежда съдържавка (общ синтаксис)?

изрази  $\leftarrow \text{SELECT } * \left( \begin{array}{l} \{ \text{DISTINCT} \} \text{ column / expression [alias]} \\ \text{as } \text{именование} \end{array} \right) \dots \right) \text{FROM table}$

атрибути  $\left[ \text{WHERE condition} \right]$

нестандартизирана форма

което е идентично

на този език

некоето

от кой таблица

се извеждат

оттук

изпълнението

некоето

statement завърши с;

но като условие трябва да отвори

корелации

SELECT

включат в определен рег

FROM

некоето на таблиците трябва да са

From некоето изпълнението на SELECT

уникални.

Movie (title, year, length, studioname)

некоето да изведе title, year

това е

SELECT title, year

направота

FROM movie

значи ще изведе първо From на Movie  $\uparrow$  SELECT

WHERE length > 120

в таблицата на movie накато атрибути

направи за филм с > 120'

SELECT отреди кой което да изберат  
от на  $\uparrow$  атрибути.

(условието включват

title	year	length	name	movie
		120		

този регионе съдържащо #параметър {

съдържащо условието).

x

н атрибути

в SELECT накато в определената

m	title	year

select предизвика  
организацията атрибути

некоето на length-a да добави

m	title	year

$m \leq h$

5min.

#параметър
y < x

SELECT title, year, (length + 5) AS newlength, title, year ? Newlength : length

m	title	year	newlength
125		120	

FROM movie

ако накато или use as где на колони  
нам.

WHERE length > 120

AND studioname = 'Disney';

оператор 3d сравнете на низове по съдълост

" AND studioname LIKE '% Disney %'

които са с Disney но ако има между / след Disney

членове New Disney

'% Disney %'

които има между името и след Disney и го губи изваден  
в таблица

' Disney %'

има членове между Disney

NOT LIKE отрицанието на LIKE

= 'Disney' има да бъде изваден в резултатната таблица ако

е New Disney или Disneylength

LIKE '% Disney %' може да извади и New Disney и Disneylength.

LIKE '% [%] Disney %'

= '% Disney'

дали името на студията е в никое множество от студии

WHERE studioname IN ('MGM', 'Disney', 'Lucas'); | S | t | y | l | s  
да проверяваш което името на студията е в  
този множества

	t	y	l	s
S		1978	20	MGM
X		2010	180	Disney
Y		2020	210	Lucas

искам да извадя т филми между 120 - 180 min

WHERE length BETWEEN 120 AND 180

120 <= length <= 180

1 вариант

2 вариант

3d го това

'2024-02-20'

len

Ако len от стойностите не е известна

нр.

120

е очевидното T

null → няма, не е конкретна, не е 0

null

F

→ нр. ако нака null + 5 е так null upon newlength

→ при сравнение все е UNKNOWN (не знаям какът е T v F) и не е включено в резултата

→ ако искам да го включим нр.

WHERE length BETWEEN 10 AND  
OR length IS NULL

IS NOT NULL

(отрицателно)

→ True AND UNKNOWN → U

T OR U → T

Как да сортираме резултата по-icon начин  
SELECT

FROM ~~намерените с 1+ от length~~

ORDER BY title, year

ако не ~~като~~ поумолчанию по дефолт е descending

ascending descending

ORDER BY title ASC, year DESC

или с чисто га го кати ORDER BY ①ASC, ②DESC  
title year

Наш отговор USE movie;

3) ~~title~~

Семинар 2

27.02.2024

Movie ( title, year, prod# )

Movieexec ( name, cert# )

	Title	year	prod#	movieexec	name	cert#
i	mA	2000	123		prod1	123
	mb	2005	567		prod2	567
	mc	2020	123			

Некаче го извршаме тајкомбинација на movie и movieexec

резултатна таблици

Movie x Movieexec,  $m \times n$

Title	year	prod#	name	cert#
mA	2000	123	prod1	123
mA	2000	123	prod2	567
mB	2005	567	prod1	123
mB	2005	567	prod2	567
mc	2020	123	prod1	123
mc	2020	123	prod2	567

SELECT title, name

1. Изврши  
FROM movie CROSS JOIN movieexec ;  
2. Изврши  
FROM movie, movieexec.

1. Ако movie има редове  $\phi$  в резултатната таблици тие се  
име  $\phi$ . Таблици без кореспондентни

1. Ако искаме да извршим  $prod\# = cert\#$ . За тоа искаме да  
имаме свидетелство.

Movie  $\bowtie_{prod\# = cert\#}$  Movieexec

5 → INNER JOIN

SELECT name

FROM movie INNER JOIN movieexec.

ON prod# = cont# } условия для связывания  
WHERE title = 'mA';

SELECT name

FROM movie, movieexec

WHERE prod# = cont#

AND title = 'mA';

} No grouping operation

3d Movie JOIN

SELECT

FROM t<sub>1</sub> JOIN t<sub>2</sub> ON t<sub>1</sub>.a = t<sub>2</sub>.a

JOIN t<sub>3</sub> ON t<sub>3</sub>.a = t<sub>2</sub>.a

operation

execution order

rotate until

apply rule

запись

условие

условие для этого JOIN не know how to handle

без ON

More go связывание этого JOIN с реальным примером.

Movie (name, address)

name	address
mSA	a1
mSB	a2
mSC	a1

результаты на Таблица name | address

mSA | mSA a1

mSA | mSB a2

mSA | mSC a1

!

Узнайте что такое с единицей адреса Таблица B этого конфликта Таблица A

этого конфликта не в равенстве  $a \neq b$ .

SELECT moviestar.name, ms.name

FROM moviestar, moviestar AS ms2.

WHERE moviestar.address = ms2.address

Запрос для этого

msA | msA

msA | msC

msC | msA

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

msA | msC

msC | msA

msA | msC

msC | msC

WHERE moviestar.address = ms2.address  
 AND moviestar.name <> ms2.name

результатно

msA msc  
 msC msA

Не използване distinct защото distinct не избира атрибути с  $\neq$  корелации. Distinct се използва когато искат уникатни атрибути да са взети и чимато корелантите съвпадат.

## МНОЖЕСТВЕНИ ОПЕРАЦИИ

2 Таблици

	chan	int	chan
a	b	c	
a	1	2	
b	2	3	
c	3	4	

	d	e
	1	a
	5	d

се правят RUS, R ∩ S, R - S.

UNION, INTERSECT, EXCEPT

1<sup>o</sup> случаен не могат да бъдат обединени защото имат  $\neq$  колони, т.к. SELECT ги са включени в  $\neq$  колони.

Наколко колони с.

2<sup>o</sup> случаен. пегат на атрибутита Тривиално да е Такащ че типовете на данните да са съвместими.

a	b

(q1)

UNION

(q2) - 4 тип

SELECT a, b  
 FROM R  
 UNION

SELECT a, b  
 FROM R  
 WHERE

типовете на данните да са съвместими

(q2) : SELECT e, d  
 FROM S

SELECT e, d  
 FROM S  
 WHERE

a	b	резултатният
a	1	
b	2	таблица.
c	3	
a	1	
d	5	

дуплицираните корелации се премахват с UNION. Ако искате да ги използвате UNION ALL

INTERSECT RNS

a	b
d	1

EXCEPT R-S

a	b
b	2
c	3

Задача & решение на starmovie

(SELECT title, year  
FROM movie)

UNION

(SELECT movietitle, movieyear  
FROM starsin),  
ORDER BY title;

Ниже в запросах нет актеров, которых нет в кинорежиссерах

(SELECT name  
FROM moviestar  
WHERE gender = 'F')

INTERSECT

(SELECT name  
FROM movieexec);

Query 1

EXCEPT

→ also moviestar ТБО где есть режиссеры, которых нет в movieexec

Q2

HINTERSECT → moviestar & movieexec ТБО.

Q3

ПРОСТН ПОДЗАВКН

SELECT name

FROM movies JOIN movieexec ON cont# = producer#

WHERE title = 'Star Wars';

да изведен тонките на критичност

SELECT name

FROM movieexec

WHERE cont#  $\in$  (SELECT producer#

КОГАТО FROM movie

некое склонение WHERE title = 'Star Wars');

(! СТОЙНОСТ)

$\hookrightarrow$  око ика нове от 1 или о критичн

Трябва да използвате груп оператор от =  
Зададо лице върху грешка.

Нека S е склонение а R е унарна релация (с ! критично)

EXISTS R лице ВЪЗПРИЯТИЕ ИСТИНА ОДИ В R ИКА КОРЕКТИН

S IN R  $\rightarrow$  True лице S е = НОДЕ 1 СТОЙНОСТ от R

S > ALL R  $\rightarrow$  True лице S е > от + СТОЙНОСТИ ИЗЧИСЛЕННИ

от НЕГРАДИБЛОТО R

S > ANY R  $\rightarrow$  True лице S е > от ЕДИН ГЛАВНА СТОЙНОСТ от P  
ОПРЕДЕЛЕНО ТОЧКА

S NOT IN R (или S <> ALL R)

NOT S > ALL R

NOT S > ANY R

преп 1 име в network да изпълнете с име генерики network

SELECT name, networth

FROM movieexec

с несвръзан

WHERE networth >= ALL (SELECT networth

SELECT me.networth

FROM movieexec); FROM movies;

Мостите за всички се измълчват в резултата от корелативните многократни.

Пример 2: Записът на филм, кий на автор е роден на 1980

SELECT movietitle, starname

FROM starsin

WHERE starname IN (SELECT name

Търси кий на автор

FROM moviestar

SE

го е в този момент

WHERE birthdate LIKE '%1980')

Минимум 3 като корелативна мост за всичко

SELECT movietitle, starname

FROM starsin si

WHERE EXISTS (SELECT \*

от всички звезди в момента

Звездата кий

FROM moviestar

настоящ

WHERE birthdate LIKE '%1980'

AND ms.name = starname);

Магазинка в FROM дава ли producer когато е участък Harrison Ford

Минимум 4

SELECT name

me

във всички филми

FROM movieXfc JOIN

(SELECT producer#

join

FROM movie, starsin ON title = movietitle

това е мост

(1) AND year = movieyear

WHERE starname = 'Harrison Ford' ) AS (p)

ON me.centre# = p.producer#;

ако това е участник то го идентифицира

Независимо това е участък

Звездата не е корелативна

Корелативна подзапитка - запитка в която кий зависи от  
запитката в която участва своята магазинка.

написать на запросе B SELECT изъята  
номер 5 title, length от Movie, length то какое значение опущено  
SELECT title, length (SELECT TOP 1 length  
FROM movie

From movie  
ORDER BY length DESC); - length AS diff  
FROM movie ;

сортировка и изъятие первого

номер 6 title то программа и length то какое значение было опущено.

SELECT name, (SELECT TOP 1 length  
FROM movie  
WHERE period# = me. cent#  
ORDER BY length DESC) AS max#,  
FROM movieexer me;

12.03.2024

Семинар 4 (запоминание)

что скажет компилятор на различные способы записи

SELECT \*

FROM customer **CROSS JOIN** orden;

CUSTOMERID	COMPANYNAME	ORDERID	CUSTOMERID
------------	-------------	---------	------------

да изъедем и что это за кросс та и сколько в текстах написано  
но на разницу между **INNER JOIN** и **JOIN**  
2 ряда, 4 колонки

123	A	1	123
567	B	2	567

тогда и будет ли сбрасываться.

ВЪВ ВЪДЪЛЪГИТО Съединение - когато се извршват две от тези неравенства

се включват като редовете към съединение с редовете на другата

таблица Така и тези редове към не съответстват и са допълнителни

с NULL стойност.

**SELECT \***

FROM customers c left join orders o ON c.customerID = o.customerID  
от лявата таблица.

				$t_1$	$t_2$
123	A	1	123		
567	B	2	567		
987	C	NULL	NULL		

inner join } left join .

1) a   b   c			2) a   b   c			3) a   b   c		
1	10	1	1	10	1	1	10	1
3	12	3	2	11	null	2	11	null
			3	12	3	3	12	3
			4	12	null	4	12	null
						Null	Null	5

select \* from  $t_1$  join  $t_2$  on  $a=c$  and  $c=1$   
where  $c=1$   $\rightarrow$  както виждате

select \* from  $t_1$  left join  $t_2$  on  $a=c$  where  $c=1$  1 ред

select \* from  $t_1$  left join  $t_2$  on  $a=c$  and  $c=1$

1 10 1      и първите от всичките са включени

2 11 null , грешка null стойност там следва да отговаря

3 12 null      и третият е допълнителен

4 12 null      и четвъртият е допълнителен

select \* from  $t_1$  left join  $t_2$  on  $a>c$  and  $b=12$

a	b	c				
1	10	null	$\rightarrow$			
2	11	null				
3	12	1				
4	12	1				

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

$t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3

inner join  $\rightarrow$   $t_2$  on  $b=12$  в ON clause

to be null or we use rectangle

=)

1	10	null
2	11	null
3	12	1
4	12	1
u	12	3</

Select \* from t<sub>1</sub> left join t<sub>2</sub> on d > c or b = 12

d > c			upon b = 12		
a	b	c	a	b	c
1	10	null	1	10	null - Не убывает десконтная
2	11	1	2	11	1 $d > c$
3	12	1	3	12	1 $d > c$
4	12	1	3	12	3 $b = 12$
4	12	3	3	12	5 $b = 12$
			4	12	1 $d > c \& b = 12$
			4	12	3 $b = 12 \& d > c$
			4	12	5 $b = 12$

19.03.2024

## ЧИТАП5 функшн

### группови функшн

COUNT връща # погрешки

Count \* връща всички броя на погрешки от таблицата - не игнорира NULL стойност

Count (expression) # погрешки с не NULL стойност

Avg работи като с числа

Avg също игнорира Null стойности.

Expression може да е # типов данни : int, string т.н.

Count (DISTINCT starname) все броят погрешки като unique

Задача 1 Задача като извлича средна стойност на networth на Hollywood

SELECT AVG(networth) AS averageNet.

FROM movieexec;

Задача 2 SELECT COUNT(\*) всички погрешки от starsin

FROM starsin;

загадка 3

кои искате да превроят # пътешествие в starname  
 SELECT COUNT (starname)  
 FROM starsin;

загадка 4

Employee : id, name, job\_id, salary, commission, hire\_date, dep\_id  
 Department dep\_id dep\_name.

SELECT AVG (salary), MIN (salary), MAX (salary) <sup>най-крайна</sup>  
 FROM employee  
 WHERE job\_id = 'ref';

загадка 5

Најчесто срещани | Најчесто използвани  
 SELECT MIN (hire\_date), MAX (hire\_date)  
 FROM employee

загадка 6

SELECT COUNT (emp\_id) <sup>брой служители с id 50.</sup>  
 FROM employee  
 WHERE dep\_id = 50;

загадка 7

обичата комисия разделята на т. служителни (кои не им служат)  
 която не получават комисия (NULL)  
 SELECT AVG (ISNULL (commission, 0)) <sup>проверява дали commission</sup>  
 FROM employee; <sup>replacement & Null и ако е NULL то обичаи са</sup>  
<sup>expansion</sup>

\*

да разделим вортичите на таблицата в отделни групи			
dep_id	salary	dep_id	salary
90	1000	90	2000
90	2000	60	4500
90	3000	50	6000
60	4000		
60	5000		
50	6000		

## Създаване на груп

SELECT column<sub>1</sub>, group-function(column<sub>1</sub>), group-function(column<sub>2</sub>), ...  
FROM table  
WHERE condition  
GROUP BY column<sub>1</sub>, ...;

### Задача 8

\* нареди средната залогара на служителите в съответните депари  
SELECT AVG(salary) AS avg-salary, dep\_id <sup>ако добавим и деп.</sup>  
FROM employee, <sup>и показват само отгрупуваните</sup> т. Грешно Задача  
GROUP BY dep\_id; <sup>които групират</sup> 90 1000 <sup>emp\_id</sup> A  
90 2000 B -> р  
90 3000 C

### Задача 9

да извеждат average salary и сумата на заплатите 90 3000  
dep\_id job\_id salary dep\_id job\_id sum\_salary <sup>нако е голям</sup>  
90 rep 1000 90 rep 3000 <sup>този среден</sup>  
90 rep 2000 90 clerk 3000 <sup>заплатата</sup>  
90 clerk 3000 => 60 rep 9000 <sup>в нюанси</sup>  
60 rep 4000 50 clerk 6000

60 rep 5000

50 clerk 6000

от тази точка са в group by

SELECT SUM(salary) as sumSalary, dep\_id, job\_id  
FROM employee  
GROUP BY dep\_id, job\_id

Having <sup>крайца</sup>:

" WHERE <sup>-та може да използва групова ф-ня</sup>  
GROUP BY

HAVING group-condition <sup>→ на която трябва да отговарят отгрупуваните</sup>  
<sup>группи за които се включват нито за тези</sup>  
<sup>които не използват групова ф-ня</sup>  
условия за групова ф-ня

Задача 10 Некто на  $\frac{1}{3}$  продуцентните общи гранници на  $\frac{1}{3}$  продуцентните общи от Т.9.

```
SELECT SUM(length), name  
FROM movie JOIN movieexec ON prod-cont# = cont#  
GROUP BY name;
```

Задача 11 Огюра дзвінка та філмів та вских прогулянок net worth > 10'000'000  
globaline WHERE networth > 10'000'000  
GROUP BY name;

Обща грешка за тези производители които са най-малките единици  
 където  $Y < \text{min}$   
 давате  $HAVING MIN(year) < 1950$   
 давате  $the income sum$  на всички фирмии производител от  
 тези производители и са същите същите, това условие в where  
 която се използва тези първите.

Задача 12) NULL не является типом группировки.

$$R(A, B)$$

( $\downarrow$ null,  $\downarrow$ null)

SELECT A, COUNT(B)

FROM R

## GROUP BY A:

~~we negotiate.~~

SELECT A, SUM(B)

$$\Rightarrow (\text{null}, 0)$$

30/04/13

SELECT A, SUM(B)

FROM R

## Group By A

$\Rightarrow (\text{null}, \text{null})$

hull he gomphoceridae

Задача 14 Средняя честота  $\bar{n}$  пр.

## He un TPE6BD za cera

SELECT CONVERT(decimal(6,2), AVG(speed)) AS avgSpeed

FROM pc | decimal (Precision scale)  
BINARY INT GR (e NOTBEGRE)

В квартале где я остановился

## УНИТАР 6

## ПРЕГОВОР

5. SELECT

1. FROM

2. WHERE

3. GROUP BY

4. HAVING → условие за груп. групите които имат го това условие не се включват

6. ORDER BY

IN ( )

символът няма значение

a LIKE 'A%' да започва с A

a → здравен символ.

5 + NULL = NULL

незададена стойност

TRUE

FALSE

UNKNOWN → не се включва в крайния резултат

Задача работи с 2 + кортежи на 1 таблица:

FROM t1 as t<sub>1</sub>, t2 as t<sub>2</sub>

t <sub>1</sub>		t <sub>2</sub>	
a	b	c	d
1	2	5	8

декартово произведение

a	b	c	d
1	2	5	8

t<sub>1</sub> × t<sub>2</sub>

свръзният

На свръзте таблици

ако има разлика

a	b
1	2
5	8

на имената на атрибути  
имат ab cd взимаме атрибути  
от първо на второ имена ab.UNION ALL се включва и повторя ли се кортежи а UNION  
не ги премахва и се оставят само 1.

<име на таблица>, <атрибут>. ga провериете как кратът атрибут.

• SELECT name

name = вътрешни ид са

stardname свидетелство за то че го до времето това заместо е в неизвестен

Group BY stardname Bng.

• подзаписка в From Таблица → същно просто подзаписка

Не са част от базата. единствено трябва да имат един и същ

WHERE a IN (подзаписка)

→ тук употребяват се множества рефере

a
1
2

тук сравняват a с множество в таблица

WHERE a = (подзаписка)

тук се сравняват

a тук ! peg

a
1

t <sub>1</sub>	a   b
	1   2

t <sub>2</sub>	c
	1
	3

[INNER]

t<sub>1</sub> JOIN t<sub>2</sub> ON a=c

a   b   c
1   2   1

t<sub>2</sub> RIGHT JOIN t<sub>1</sub> ON a=c → тук редове от пустата Таблица са

първи

последни

не са свидетелстват са пустите

t<sub>1</sub> RIGHT JOIN t<sub>2</sub> ON a=c

a   b   c
1   2   1

пример: кон ред не е участник в никак null null 3

вие ? кон клиент не е направил поръчка?

WHERE a IS NULL

a IS NOT NULL

не използват оператор за сравняване.

COUNT

MIN

MAX

AVG

SUM

\* В select , не може в FROM clause да е многоставко  
и в where не , в having може да е order by

грумови функции която извеждат информация в отбор  
във. Трябва да има груп и да има идентификатор

игнорират NULL стойност с изключение на COUNT(\*)

COUNT(a) → брой на редове в таблица игнорирайки NULL

COUNT(DISTINCT) за да униколи редове.

a	b	c
1	3	8
1	2	1

SELECT DISTINCT a, b, c

0
1

SELECT DISTINCT a, b

a	b
1	3
1	2

записът DIST

работи върху  
уникални  
(рек)

Name dep-ID job-ID age

Subgr 1.1 Ivan dev BA 20 }

Subgr 2.1 Peter dev dev 25 }

Ani dev dev 30 }

Subgr 2.1 Georgi acc manager 30 }

Subgr 2.2 Svetla acc auditor 40 }

средната възраст на отбора  
и може ли?

→ Трябва да групирате по dep

Групирате по първото и  
второто в макаро групирате  
с второто

GROUP BY dep-ID, job-ID

AVG - AGE	
gr1	
gr2	

AVG	
gr1	
gr2	
gr3	
gr4	

което групирате с job-ID