

# Darbas su duomenimis

## 4. Eilučių filtravimas

# Komanda Where

Select, Update ir Delete užklausos gali turėti Where (filtravimo) komanda.

Vienintelė Insert komanda neturi Where komandos.

# Komandos where sąlygos

- Komanda turi turėti True arba False reikšmę.
- Jungiant filtravimo sąlygas mes naudojame AND ir OR komandomis.
- Jeigu turime daugiau nei 2 sąlygas, galime naudoti skliaustus.

Intermediate result	Final result	Intermediate result	Final result
WHERE true OR true	true	WHERE (true OR true) AND true	true
WHERE true OR false	true	WHERE (true OR false) AND true	true
WHERE false OR true	true	WHERE (false OR true) AND true	true
WHERE false OR false	false	WHERE (false OR false) AND true	false
		WHERE (true OR true) AND false	false
		WHERE (true OR false) AND false	false
		WHERE (false OR true) AND false	false
		WHERE (false OR false) AND false	false

# Komanda NOT

Komanda NOT padeda paneigti True.

```
WHERE NOT (first_name = 'STEVEN'  
OR  
last_name = 'YOUNG') AND  
create_date > '2006-01-01';
```

Pvz. Komandą NOT galime perrašyti tokiu būdu

```
WHERE first_name <> 'STEVEN' AND  
last_name <> 'YOUNG'  
AND create_date > '2006-01-01';
```

Intermediate result	Final result
WHERE NOT (true OR true) AND true	false
WHERE NOT (true OR false) AND true	false
WHERE NOT (false OR true) AND true	false
WHERE NOT (false OR false) AND true	true
WHERE NOT (true OR true) AND false	false
WHERE NOT (true OR false) AND false	false
WHERE NOT (false OR true) AND false	false
WHERE NOT (false OR false) AND false	false

# Iš ko galime sudaryti sąlygas

1. Skaičių
2. Stulpelių
3. Tekstinių eilučių (strings). Pvz. ‚Maple Street‘
4. Built-in – standartinių funkcijų. Concat(“mokausi“, “ “, “Sql“)
5. Subquery <sup>+, -, \*, and /</sup>
6. List of expressions – posakių sąrašo. (Boston, New York, Chicago)

Palyginimo operatoriai

- =, !=, <, >, <>, like, in, between

Aritmetiniai operatoriai

- +, -, \*, /

# Sąlygų tipai

- Lygybės sąlyga.
  - title = 'RIVER OUTLAW'
  - film\_id = (SELECT film\_id FROM film WHERE title = 'RIVER OUTLAW')
- Nelygybės sąlyga.
  - date(r.rental\_date) <> '2005-06-14'
  - amount <> 375.25

Populiariu modifikuoti duomenis naudojantis šiais sąlygų tipais

```
DELETE FROM rental
```

```
WHERE year(rental_date) = 2004;
```

# Sąlygų tipai

- Intervalo sąlyga.

```
SELECT customer_id, rental_date  
FROM rental  
WHERE rental_date <= '2005-06-16'  
AND rental_date >= '2005-06-14';
```

```
SELECT customer_id, rental_date  
FROM rental  
WHERE rental_date BETWEEN '2005-06-14' AND '2005-06-16';
```

Pabandykite komandą `select cast('2021-12-11' as datetime);`

# Eilučių intervalai

```
SELECT last_name, first_name  
FROM customer  
WHERE last_name BETWEEN 'FA' AND 'FR';
```



# Sąlygų tipai

- Priklausymo sąlyga

```
SELECT title, rating  
FROM film  
WHERE rating = 'G' OR rating = 'PG';
```

```
SELECT title, rating  
FROM film  
WHERE rating IN ('G','PG');
```

```
SELECT title, rating  
FROM film  
WHERE rating NOT IN ('PG-13','R', 'NC-17');
```

# Subqueries naudojimas

```
SELECT title, rating  
FROM film  
WHERE rating IN (SELECT rating FROM film WHERE title LIKE  
'%PET%');
```

# Sąlygų tipai.

- Teksto palyginimai

```
SELECT last_name, first_name  
FROM customer  
WHERE last_name LIKE '_A_T%S';
```

```
SELECT last_name, first_name  
FROM customer  
WHERE last_name LIKE 'Q%'  
OR last_name LIKE 'Y%';
```

Wildcard character	Matches
<code>_</code>	Exactly one character
<code>%</code>	Any number of characters (including 0)

Search expression	Interpretation
<code>F%</code>	Strings beginning with <i>F</i>
<code>%t</code>	Strings ending with <i>t</i>
<code>%bas%</code>	Strings containing the substring <code>'bas'</code>
<code>_ _t_</code>	Four-character strings with a <i>t</i> in the third position
<code>_ _ _ - _ - _ - _</code>	11-character strings with dashes in the fourth and seventh positions

# NULL

- Sąlyga gali būti NULL bet niekada nebus lygi NULL

```
SELECT rental_id, customer_id  
FROM rental  
WHERE return_date IS NULL;
```

```
SELECT rental_id, customer_id  
FROM rental  
WHERE return_date IS NOT NULL;
```

Palyginimui, kas bus jeigu įvykdysite šią komandą

```
SELECT rental_id, customer_id  
FROM rental  
WHERE return_date = NULL;
```

# Užduotys

- Atraskite išnuomotus filmus, kurie nebuvo gražinti 2005 metais nuo gegužės iki rugpjūčio.

# Užduotys

- Pažiūrėkite į lentelę dešinėje. Kurios eilutės bus grąžintos pritaikius sąlygą

- customer\_id <> 5 AND (amount > 8 OR date(payment\_date) = '2005-08-23')
- customer\_id = 5 AND NOT (amount > 6 OR date(payment\_date) = '2005-06-19')

payment_id	customer_id	amount	date(payment_date)
101	4	8.99	2005-08-18
102	4	1.99	2005-08-19
103	4	2.99	2005-08-20
104	4	6.99	2005-08-20
105	4	4.99	2005-08-21
106	4	2.99	2005-08-22
107	4	1.99	2005-08-23
108	5	0.99	2005-05-29
109	5	6.99	2005-05-31
110	5	1.99	2005-05-31
111	5	3.99	2005-06-15
112	5	2.99	2005-06-16
113	5	4.99	2005-06-17
114	5	2.99	2005-06-19
115	5	4.99	2005-06-20
116	5	4.99	2005-07-06
117	5	2.99	2005-07-08
118	5	4.99	2005-07-09
119	5	5.99	2005-07-09
120	5	1.99	2005-07-09

# Užduotys

- Sukurkite užklausą, kad iš payments lentelės būtų gražintos eilutės su sumomis 1.98, 7.98, arba 9.98.
- Sukurkit užklausą customers lentelei, kad jums būtų gražintos eilutės, kur kliento pavardės antra raidė yra A, o po jos bet kurioje pozicijoje būtų raidė W.