



## Banco de Dados SQL

Bloco 20 - Aula 20.4



# Roadmap da Aula



- Comando Insert;
- Comando Insert com Select;
- Comando Update com WHERE;
- Comando Update com CASE e WHEN;
- Comando Update com WHERE e comandos;
- Comando Delete com WHERE;
- Comando Delete e suas restrições;
- Comando Truncate;
- Atividade prática;



# INSERT



- **Faça a inserção um registro na tabela staff**
- **Faça a inserção de três registros na tabela city**



# INSERT



- **Faça a inserção um registro na tabela staff**

- `INSERT INTO `sakila`.`staff` (first_name, last_name, address_id, email, store_id, active, username, password) VALUES ('Felipe', 'Nunes', 1, 'felipe.nunes@betrybe.com', 1, 1, 'nunesfb', '12345678');`

- **Faça a inserção de três registros na tabela**

- `INSERT INTO `sakila`.`city` (`city`, `country_id`) VALUES ('São Paulo', 15), ('Rio de Janeiro', 15), ('Bahia', 15);`
- `SELECT * FROM sakila.city WHERE country_id = 15;`



# INSERT



- **Insira um registro na tabela staff**



# INSERT



- **Insira um registro na tabela staff**

- `INSERT INTO `sakila`.`staff` (`first_name`, `last_name`, `address_id`, `email`, `store_id`, `active`, `username`, `password`) VALUES ('julia', 'souza', 11, 'julia.souza@teste.com', 2, 1, 'julia', '12345678');`
- `SELECT * FROM sakila.staff WHERE first_name = 'julia';`



# INSERT



- **Busque por todas pessoas na tabela customer cujo primeiro nome inicia com a letra B e insira o primeiro e último nome na tabela actor**



# INSERT



- **Busque por todas pessoas na tabela customer cujo primeiro nome inicia com a letra B e insira o primeiro e último nome na tabela actor**
  - `INSERT INTO sakila.actor (first_name, last_name) SELECT first_name, last_name FROM sakila.customer WHERE first_name LIKE 'b%';`
  - `SELECT first_name, last_name FROM sakila.actor WHERE first_name LIKE 'b%';`





# UPDATE



- **Atualize o `release_year` para 2009 e o `language_id` para 3 onde o `length` é maior que 150, o `rating` é PG-13 e o `replacement_cost` é maior que 25**





- **Atualize o release\_year para 2009 e o language\_id para 3 onde o length é maior que 150, o rating é PG-13 e o replacement\_cost é maior que 25**
  - `UPDATE sakila.film SET release_year = 2009, language_id = 3 WHERE length > 150 AND rating = 'PG-13' AND replacement_cost > 25;`
  - `SELECT * FROM sakila.film WHERE length > 150 AND rating = 'PG-13' AND replacement_cost > 25;`

# UPDATE



- **Faça o update do replacement cost para 30.99 quando o rental\_rate estiver entre 1 e 3, quando o length for maior que 150, mude para 35.99 e quando o length for menor ou igual a 82, mude para 20.33**



# UPDATE



- **Faça o update do replacement cost para 30.99 quando o rental\_rate estiver entre 1 e 3, quando o length for maior que 150, mude para 35.99 e quando o length for menor ou igual a 82, mude para 20.33**

```
UPDATE sakila.film
SET replacement_cost = (
    CASE
        WHEN rental_rate BETWEEN
1 AND 3 THEN 30.99
        WHEN length > 150 THEN
35.99
        WHEN length <= 82 THEN
20.33
        ELSE replacement_cost
    END
);
```

```
SELECT * FROM sakila.film WHERE rental_rate
BETWEEN 1 AND 3 OR length > 150 OR length <= 82;
```



# UPDATE



- **Faça a atualização da senha na tabela staff para os 5 últimos registros que sofreram atualização e que estão com cadastro ativo**



# UPDATE



- **Faça a atualização da senha na tabela staff para os 5 últimos registros que sofreram atualização e que estão com cadastro ativo**

```
UPDATE sakila.staff  
SET password = 'FavorResetarSuaSenha123'  
WHERE active = 1  
ORDER BY last_update  
LIMIT 5;
```



# DELETE



- **Delete todas pessoas com nome JULIA BARRYMORE da tabela actor**



# DELETE



- **Delete todas pessoas com nome JULIA BARRYMORE da tabela actor**

```
DELETE FROM sakila.actor  
WHERE first_name = 'JULIA BARRYMORE';
```

```
SELECT * FROM sakila.film_actor  
WHERE actor_id = 47;
```

```
DELETE FROM sakila.film_actor  
WHERE actor_id = 47;
```

```
DELETE FROM sakila.actor  
WHERE first_name = 'JULIA' AND  
last_name = 'BARRYMORE';
```



# DELETE



- **Delete todas cidades que tenham pa em algum lugar do nome**



# DELETE



- **Delete todas cidades que tenham pa em algum lugar do nome**

```
SELECT * FROM sakila.film_text  
WHERE title LIKE 'a%';
```

```
DELETE FROM sakila.film_text  
WHERE title LIKE 'a%';
```



# DELETE



- **Delete todos dados da tabela film\_text**



# DELETE



- **Delete todos dados da tabela film\_text**

```
SELECT * FROM sakila.film_text;
```

```
TRUNCATE sakila.film_text;
```



# DELETE CASCADE



- **Delete com CASCADE**
  - **Vamos deletar o registro 1 de customer;**



# DELETE CASCADE



- **Delete com CASCADE**

```
DELETE FROM sakila.payment WHERE customer_id = 1; -- Deletar as referências em payment  
DELETE FROM sakila.rental WHERE customer_id = 1; -- Deletar as referências em rental  
DELETE FROM sakila.customer WHERE customer_id = 1; -- Então, deletar o customer
```

```
SELECT * FROM sakila.customer;  
SELECT * FROM sakila.address_id;  
DELETE FROM address WHERE address_id = 5;
```



# Atividade de Fixação



**Vamos exercitar em grupo  
o que aprendemos!**



# Atividade de Fixação



1. **Adicione os dois principais atores do novo filme Matrix na tabela actor;**
2. **Adicione o filme Matrix 4 na tabela film;**
3. **Faça a ligação na tabela film\_actor destes registros inseridos anteriormente;**
4. **Insira na tabela inventory um registro com o id deste filme para a loja com id igual a 1;**
5. **Crie você como pessoa usuária na tabela customer (faça a associação do seu cadastro com um registro já existente na tabela address);**
6. **Insira os dados de aluguel na tabela rental deste filme, como sendo realizado a retirada do filme no dia 25/04/2022;**





# Atividade de Fixação



- 7. Faça a atualização com a devolução do filme no dia de hoje (26/04/2022) na tabela rental, no registro que foi inserido anteriormente (return\_date);**
- 8. Faça o delete lógico (update) do seu usuário na tabela customer (mudar o status de active para 0);**
- 9. Bônus: deletar todos registros inseridos anteriormente;**

