

Departamento de TI, UniProjeção

Orientador: Prof. Ms. Juan Toquica.

Autores: Nádio Dib (201918579), Caio Belmont (202014661), Patrick Mota (201919448), Mateus Silva (202015901) e Matheus Veríssimo (201812555).

Projeto 1: relatório de atividades da disciplina de Tópicos Avançados em Banco de Dados para operações através de Álgebra Relacional em base de dados relacional SQL com as operações de União, Seleção e Interseção.

Nome do sistema – “Sistema Taxi”.

A estrutura do banco de dados utilizado está disponível no repositório público na GitHub[1] e pode ser observado no diagrama EER gerado pelo SGBD MySQL Workbench na **figura 1**.

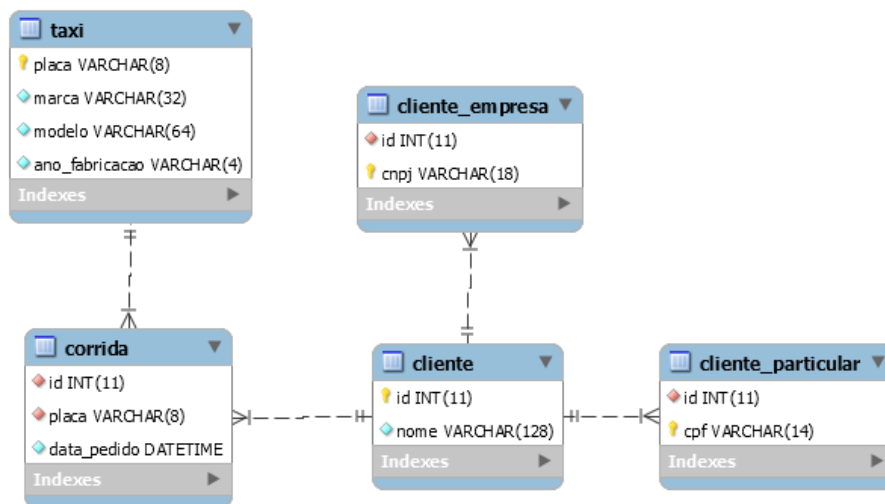


Figura 1: diagrama EER gerado pelo MySQL Workbench para o esquema *sistema_taxi.sql*.

Das operações em Álgebra Relacional, será utilizado na seguinte ordem de execução:

- **União (UNION)**

Operação 1: correlacionar os dados oriundos das tabelas **cliente**, **cliente_particular** e **cliente_empresa** de modo a unificar as informações contidas em cada tabela com os dados cadastrados; e

Operação 2: correlacionar os dados oriundos das tabelas **corrida**, **taxi** e **cliente** de modo a unificar as informações contidas em cada tabela com os dados cadastrados.

- **Seleção (SELECT)**

Operação 1: realizar uma consulta nas tabelas **cliente** e **cliente_particular** de modo a concatenar os dados cadastrados; e

Operação 2: realizar uma consulta na tabela **taxi** para identificar um campo comum entre os dados cadastrados.

- **Interseção ($R \cap S$)**

Operação 1: realizar uma consulta nas tabelas **taxi** e **corrida** para detectar quais dados concatenados não estão sendo utilizados; e

Operação 2: realizar uma consulta nas tabelas **corrida** e **cliente** para detectar quais dados concatenados não estão sendo utilizados.

União (UNION) [4][5][6]

Operação 1:

```
-- Operação 1:
-- correlacionar os dados oriundos das tabelas
-- cliente, cliente_particular e cliente_empresa
-- de modo a unificar as informações contidas
-- em cada tabela com os dados cadastrados
SELECT
  `cliente`.*,
  `cliente_empresa`.`cnpj`,
  `cliente_particular`.`cpf`
FROM `cliente`
  LEFT JOIN `cliente_empresa` ON `cliente`.`id`=`cliente_empresa`.`id`
  LEFT JOIN `cliente_particular` ON `cliente`.`id`=`cliente_particular`.`id`;
```

Figura 2: query SQL para realização da operação de união entre as tabelas **cliente**, **cliente_particular** e **cliente_empresa**.

id	nome	cnpj	cpf
1	Agatha Benedita Farias	09.679.620/0001-73	NULL
2	Márcia Teresinha Galvão	NULL	283.722.507-07
3	Kevin Igor Sebastião Barros	37.053.767/0001-99	NULL
4	Silvana Natália Alice Araújo	NULL	647.181.256-69
5	Otávio Cauê Fernandes	91.006.317/0001-90	NULL

Figura 3: resultado após a realização da query SQL para a operação de união entre as tabelas **cliente**, **cliente_particular** e **cliente_empresa**.

Operação 2:

```
-- Operação 2:
-- correlacionar os dados oriundos das tabelas
-- corrida, taxi e cliente de modo a unificar
-- as informações contidas em cada tabela com
-- os dados cadastrados
SELECT
  `corrida`.`id`,
  `cliente`.`nome`,
  `corrida`.`placa`, `corrida`.`data_pedido`,
  `taxi`.`marca`, `taxi`.`modelo`, `taxi`.`ano_fabricacao`
FROM `corrida`
  LEFT JOIN `taxi` ON `corrida`.`placa`=`taxi`.`placa`
  LEFT JOIN `cliente` ON `corrida`.`id`=`cliente`.`id`;
```

Figura 4: query SQL para realização da operação de união entre as tabelas **corrida**, **taxi** e **cliente**.

id	nome	placa	data_pedido	marca	modelo	ano_fabricacao
1	Agatha Benedita Farias	JKD-4423	2021-03-24 22:22:38	Troller	T-4 4x4 3.2 TGV tdi Cap. R	2013
5	Otávio Cauê Fernandes	JKF-3682	2021-03-24 22:22:38	Fibravan	Buqqy Vip 1.8 8V	2010

Figura 5: resultado após a realização da query SQL para a operação de união entre as tabelas **corrida**, **taxi** e **cliente**.

Seleção (SELECT) [4]

Operação 1:

```
-- Operação 1:
-- realizar uma consulta nas tabelas cliente
-- e cliente_particular de modo a concatenar
-- os dados cadastrados
SELECT
    `cliente`.*,
    `cliente_particular`.`cpf`
FROM `cliente`
INNER JOIN `cliente_particular`
ON `cliente`.`id`=`cliente_particular`.`id`;
```

Figura 6: query SQL para realização da operação de seleção entre as tabelas **cliente** e **cliente_particular**.

id	nome	cpf
2	Márcia Teresinha Galvão	283.722.507-07
4	Silvana Natália Alice Araújo	647.181.256-69

Figura 7: resultado após a realização da query SQL para a operação de seleção entre as tabelas **cliente** e **cliente_particular**.

Operação 2:

```
-- Operação 2:
-- realizar uma consulta na tabela taxi
-- para identificar um campo comum entre
-- os dados cadastrados
SELECT * FROM `taxi`
WHERE `taxi`.`modelo`
LIKE '%4x4%';
```

Figura 8: query SQL para realização da operação de seleção entre a tabela **taxi** de modo a retornar somente valores comuns que contenham as características do texto especificado.

placa	marca	modelo	ano_fabricacao
JKD-4423	Troller	T-4 4x4 3.2 TGV tdi Cap. R	2013
JKJ-0823	Cross Lander	CL-244 2.8 132cv 4x4 TB Int. Diesel	2018

Figura 9: resultado após a realização da query SQL para a operação de seleção entre a tabela **taxi** de modo a retornar somente valores comuns que contenham as características do texto especificado.

Interseção ($R \cap S$) [3]

Operação 1:

```
-- Operação 1:  
-- realizar uma consulta nas tabelas taxi e  
-- corrida para detectar quais dados concatenados  
-- não estão sendo utilizados  
SELECT * FROM `taxi`  
WHERE NOT EXISTS (  
    SELECT * FROM `corrida`  
    WHERE `corrida`.`placa`=`taxi`.`placa`);
```

Figura 10: query SQL para realização da operação de interseção entre as tabelas **taxi** e **corrida**.

id	placa	data_pedido
1	JKD-4423	2021-03-24 22:22:38
5	JKF-3682	2021-03-24 22:22:38

Figura 11: resultado após a realização da query SQL para a operação de interseção entre as tabelas **taxi** e **corrida**.

Operação 2:

```
-- Operação 2:  
-- realizar uma consulta nas tabelas corrida e  
-- cliente para detectar quais dados concatenados  
-- não estão sendo utilizados  
SELECT * FROM `cliente`  
WHERE NOT EXISTS (  
    SELECT * FROM `corrida`  
    WHERE `corrida`.`id`=`cliente`.`id`);
```

Figura 12: query SQL para realização da operação de interseção entre as tabelas **cliente** e **corrida**.

id	nome
2	Márcia Teresinha Galvão
3	Kevin Igor Sebastião Barros
4	Silvana Natália Alice Araújo

Figura 13: resultado após a realização da query SQL para a operação de interseção entre as tabelas **cliente** e **corrida**.

Referências bibliográficas

- [1]. “Project 1: Sistema Taxi”, disponível em https://github.com/Devwarlt/uni-db-projects/tree/main/tabd/project_1 ;
- [2]. “SQL LIKE Operator”, disponível em https://www.w3schools.com/sql/sql_like.asp ;
- [3]. “Example - With Multiple Expressions”, disponível em <https://www.techonthenet.com/mysql/intersect.php> ;
- [4]. “SQL INNER JOIN Keyword”, disponível em https://www.w3schools.com/sql/sql_join_inner.asp ;
- [5]. “Cláusulas INNER JOIN, LEFT JOIN e RIGHT JOIN no SQL Server”, disponível em <https://www.devmedia.com.br/clausulas-inner-join-left-join-e-right-join-no-sql-server/18930> ;
- [6]. “SELECT data FROM two tables in MySQL”, disponível em <https://stackoverflow.com/a/15113603> .