.

Faculdade de Informática e Administração Paulista

**Database Application Development**

**Entrega nº2 Challenge**

**INTEGRANTES**

|  |  |
| --- | --- |
| **RM**  **(SOMENTE NÚMEROS)** | **NOME COMPLEMENTO**  **(SEM ABREVIAR)** |
| **85707** | **Kaue Augusto Miranda Santos** |
| **86405** | **Eric Luiz Campos** |
| **86351** | **Larissa Alves Da Silva** |
| **85817** | **Giovanna de Mello Leiva** |
| **85109** | **Daniel Sanchez Melero** |
| **84549** | **Henrique Neves Lagos** |
|  |  |
|  |  |

Sumário

[**1 – Descrição do Projeto e Regras de Negócio 5**](#_Toc67673069)

[**2 – Dicionário de Dados 6**](#_Toc67673070)

[**3 – Projeto Lógico do Banco de Dados**](#_Toc67673071) **10**

[**4 – Projeto Físico do Banco de Dados**](#_Toc67673072) **11**

[**5 – Data Definition Language – DDL**](#_Toc67673073) **12**

[**6 – Data Manupulation Language – DML (INSERT) 1**](#_Toc67673074)**8**

7 – Data Manupulation Language – DML (UPDATE / DELETE) 23

[**8 – Data Query Language – DQL (SELECT)**](#_Toc67673076) **24**

[**8.1 –**](#_Toc67673077) **Relatório simples que traz uma ocorrência especifica de cada tabela.**[**.**](#_Toc67673077) **24**

## **8.2 –** **Relatório simples que traz todas as ocorrência da tabela………………….25**

## **8.3 –** **Relatório simples com selects que trazem alguma ocorrência específ….26**

## **8.4 –** **Relatório simples com INNER JOIN envolvendo duas ou mais tabelas….28**

## **8.5 –** **Relatório simples com GROUP BY envolvendo duas ou mais tabelas…..30**

# 1 – Descrição do Projeto e Regras de Negócio

**Visamos em nosso projeto criar um aplicativo mobile que mude a experiência dos usuários no âmbito de comer em restaurantes, permitindo que o cliente tenha controle de vários aspectos do estabelecimento que, antes, não eram possíveis, ou, se possíveis, eram feitos manualmente ou de uma forma mais complicada.**

**De uma forma fácil e inovadora, o cliente poderá pedir seu pedido por meio do aplicativo (sem a chamada do garçom), controlar as músicas do estabelecimento, e até as luzes. Depois que terminar de comer, poderá deixar uma avaliação por extenso do que ele tiver achado interessante (combinações de pratos, harmonia de vinhos, serviço, lugar), e ajudar outras pessoas a escolher seu pedido e ter uma experiência mais rica no lugar.**

**Pro restaurante, as vantagens de utilizar nosso aplicativo serão por meio de possibilidade de** análise dos dados de pedidos de cada cliente, possibilitando aplicar data analytics, data mining e Machine Learning para gerar insights e tomadas de decisões para o negócio. Além da melhor ambientação e experiência em geral dos consumidores.

**Regras de negócio:**

**Teremos 8 tabelas. Uma delas irá controlar o registro dos estabelecimentos no nosso aplicativo (T\_CPP\_ESTABELECIMENTO), que estará ligada a T\_CPP\_ITEM\_CARDAPIO, que recebe a chave primária de T\_CPP\_ESTABELECIMENTO como chave estrangeira. Fizemos essa lógica para que cada item do cardápio tenha a referência direta ao estabelecimento de qual ele pertence, não fizemos com a chave estrangeira no estabelecimento, para evitar que seja um atributo que receba um array de objetos de itens do cardápio, melhorando assim a performance e rapidez do sistema.**

**A terceira tabela será a T\_CPP\_CLIENTE que servirá para registrar os clientes em nosso sistema, e todos os dados relacionados a eles que nos interessam, e essa tabela se relacionará com T\_CPP\_PEDIDO que armazena os dados do pedido.**

**A 5ª tabela é a T\_CPP\_PEDIDO\_ITEM\_CARDAPIO, que é uma tabela associativa entre T\_CPP\_PEDIDO e T\_CPP\_ITEM\_CARDAPIO, essa associativa recebeu o nome padrão da junção das 2 tabelas das quais recebe as chaves estrangeiras, e só contêm de atributo uma chave primária. Serve apenas para organizar os registros de quais itens do cardápio pertencem a qual pedido.**

**A 6ª tabela é a T\_CPP\_AVALIACAO, e serve para armazenar as avaliações dos clientes em relação aos estabelecimentos, conseguinte estará ligada com a tabela T\_CPP\_CLIENTE e T\_CPP\_ESTABELECIMENTO.**

**A 7ª tabela é a T\_CPP\_CATEGORIA que serve para armazenar as categorias que cada restaurante possui está relacionada a tabela T\_CPP\_ESTABELELCIMENTO e T\_CPP\_ITEM\_CARDAPIO.**

**A última tabela é a T\_CPP\_GERENTE que armazenará os dados do gerente do estabelecimento estando ligada a tabela T\_CPP\_ESTABELECIMENTO.**

**A sigla CPP se referencia a CardáPio Poágilers.**

## 2 – Dicionário de Dados

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Tabela** | T\_CPP\_ESTABELECIMENTO | | | | |
| **Descrição** | Tabela de Estabelecimentos. Relaciona-se com a tabela T\_CPP\_ITEM\_CARDAPIO, T\_CPP\_PEDIDO, T\_CPP\_AVALIACAO, T\_CPP\_GERENTE E T\_CPP\_CATEGORIA | | | | |
| **Coluna** | **Tipo de Dados** | **Tamanho** | **Constraint** | **Descrição** | |
| cd\_estabelecimento | NUMERIC | 3 | PK | Identificador único do estabelecimento | |
| ds\_cnpj | CHAR | 14 | NN | | CNPJ do estabelecimento |
| nm\_razao\_social | VARCHAR | 50 | NN | | Razão Social |
| nm\_fantasia | VARCHAR | 50 | NN | | Nome fantasia do estabelecimento |
| ds\_endereco | VARCHAR | 100 | NN | | Endereço do estabelecimento |
| nr\_telefone | CHAR | 11 | NN | | Número do telefone do estabelecimento |
| ds\_email | VARCHAR | 40 | NN | | E-mail do estabelecimento |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabela** | T\_CPP\_ITEM\_CARDAPIO | | | |
| **Descrição** | Tabela de itens do cardápio. Relaciona-se com a tabela T\_CPP\_PEDIDO\_ITEM\_CARDAPIO e T\_CPP\_ESTABELECIMENTO, T\_CPP\_CATEGORIA | | | |
| **Coluna** | **Tipo de Dados** | **Tamanho** | **Constraint** | **Descrição** |
| cd\_item\_cardapio | NUMERIC | 4 | PK | Identificador único do item do cardápio |
| cd\_estabelecimento | NUMERIC | 3 | FK | Identificador único do estabelecimento |
| cd\_categoria | NUMERIC | 2 | FK | Identificador único da categoria |
| st\_destaque | BOOLEAN | 1 | NN | Para ver se vamos destacar o item no cardápio. Destacado = 1, não destacado = 0 |
| nm\_item\_cardapio | VARCHAR | 50 | NN | Nome do item do cardápio |
| ds\_item\_cardapio | VARCHAR | 300 | NN | Descrição do item do cardápio. |
| vl\_item\_cardapio | NUMERIC | 7,2 | NN | Valor do item do cardápio |
| ds\_ingredientes | VARCHAR | 300 | NN | Descrição em extenso sobre os ingredientes presentes no item do cardápio. |
| ds\_categoria | VARCHAR | 15 | NN | Descrição da categoria do item. |
| vl\_calorico | NUMERIC | 4 | N | Valor calórico do item do cardápio |
| fl\_foto\_ic | BLOB |  | N | Foto do item do cardápio |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabela** | T\_CPP\_PEDIDO\_ITEM\_CARDAPIO | | | |
| **Descrição** | Tabela associativa. Relaciona-se com a tabela T\_CPP\_ITEM\_CARDAPIO e T\_CPP\_PEDIDO | | | |
| **Coluna** | **Tipo de Dados** | **Tamanho** | **Constraint** | **Descrição** |
| cd\_item\_cardapio | NUMERIC | 4 | FK | Identificador único da associativa |
| cd\_pedido | NUMERIC | 4 | FK | Identicador único do código |
| qt\_item\_cardapio | NUMERIC | 2 | NN | Quantidade de cada item no pedido |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabela** | T\_CPP\_PEDIDO | | | |
| **Descrição** | Tabela dos pedidos do nosso sistema. Relaciona-se com a tabela T\_CPP\_PEDIDO\_ITEM\_CARDAPIO, T\_CPP\_CLIENTE e T\_CPP\_ESTABELECIMENTO | | | |
| **Coluna** | **Tipo de Dados** | **Tamanho** | **Constraint** | **Descrição** |
| cd\_pedido | NUMERIC | 4 | PK | Identificador único do pedido |
| cd\_cliente | NUMERIC | 5 | FK | Identificador único do cliente |
| cd\_estabelecimento | NUMERIC | 3 | FK | Identificador único do estabelecimento |
| vl\_total\_pedido | NUMERIC | 8,2 | NN | Valor total do pedido |
| st\_pedido | VARCHAR | 15 | NN | Status do pedido |
| hr\_pedido | DATE | - | NN | Horário e data em que o pedido foi feito |

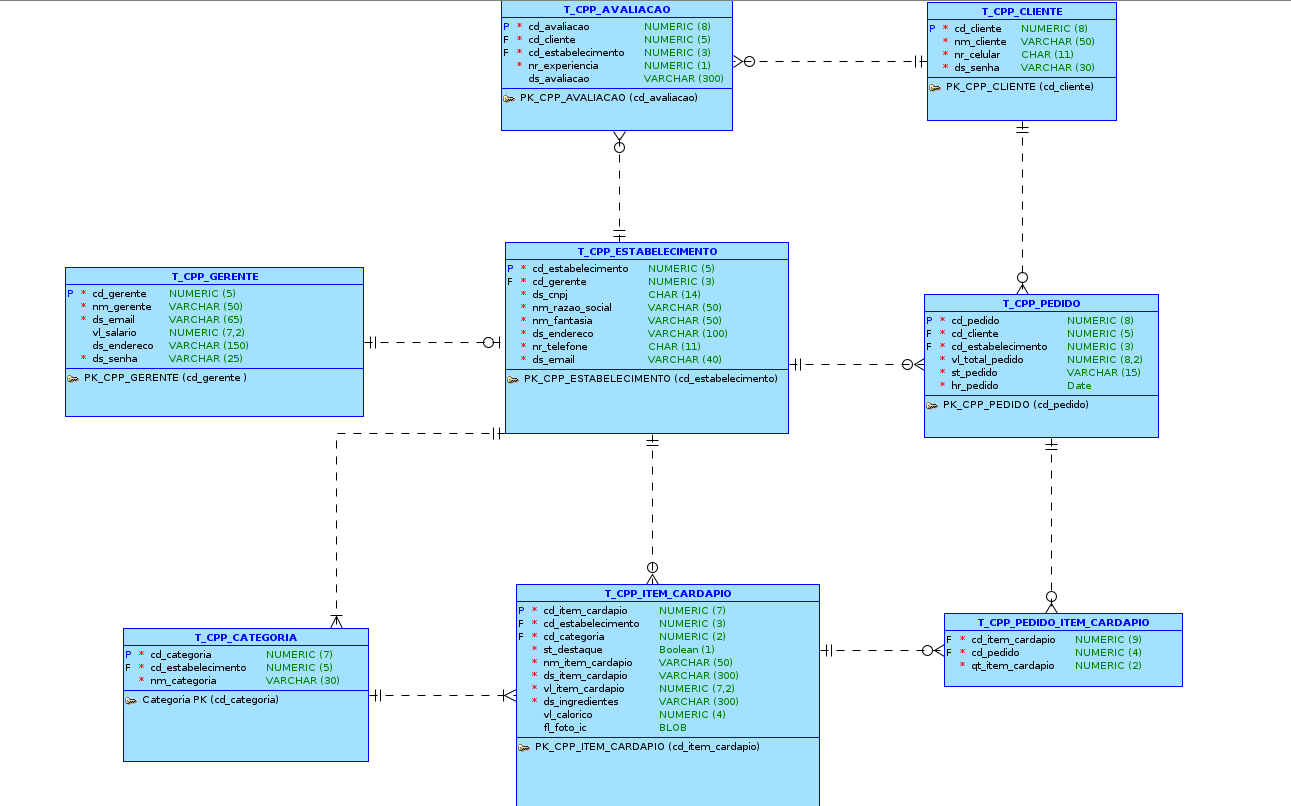
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabela** | T\_CPP\_CLIENTE | | | |
| **Descrição** | Tabela de clientes. Relaciona-se com a tabela T\_CPP\_PEDIDO e T\_CPP\_AVALIACAO | | | |
| **Coluna** | **Tipo de Dados** | **Tamanho** | **Constraint** | **Descrição** |
| cd\_cliente | NUMERIC | 5 | PK | Identificador único do cliente |
| nm\_cliente | VARCHAR | 50 | NN | Nome do cliente |
| nr\_celular | CHAR | 11 | NN | Número de celular do cliente |
| ds\_senha | VARCHAR | 30 | NN | Senha do cliente que dá acesso ao sistema |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabela** | T\_CPP\_AVALIACAO | | | |
| **Descrição** | Tabela de avaliação. Relaciona-se com a tabela T\_CPP\_CLIENTE | | | |
| **Coluna** | **Tipo de Dados** | **Tamanho** | **Constraint** | **Descrição** |
| cd\_avaliacao | NUMERIC | 4 | PK | Identificador único da avaliação |
| cd cliente | NUMERIC | 5 | FK | Identificador único do cliente |
| cd\_estabelecimento | NUMERIC | 3 | FK | Identificador único do estabelecimento |
| nr\_experiencia | NUMERIC | 1 | NN | Avaliação da experiência |
| ds\_avaliacao | VARCHAR | 300 | N | Descrição por extenso da avaliação |

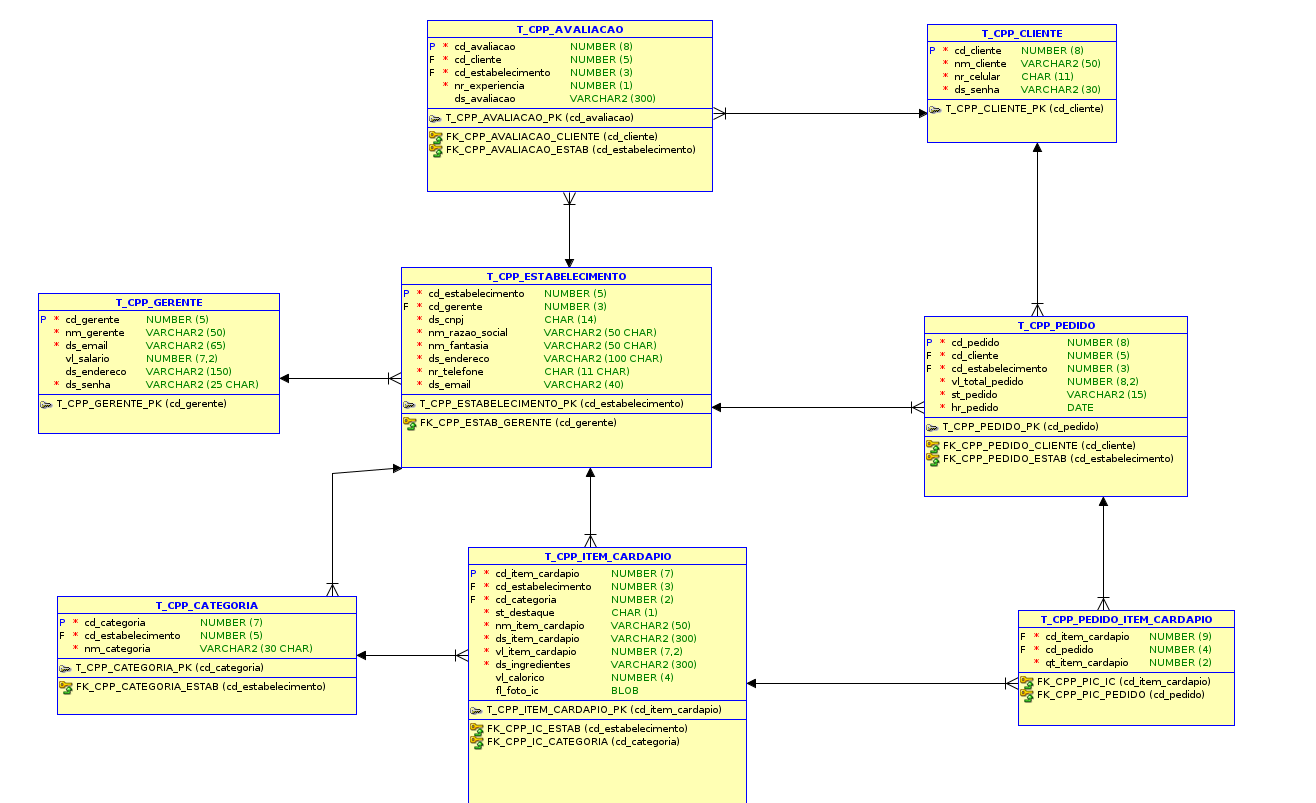
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabela** | T\_CPP\_GERENTE | | | |
| **Descrição** | Tabela de avaliação. Relaciona-se com a tabela T\_CPP\_ESTABELECIMENTO | | | |
| **Coluna** | **Tipo de Dados** | **Tamanho** | **Constraint** | **Descrição** |
| cd\_gerente | NUMERIC | 5 | PK | Identificador único da avaliação |
| nm\_gerente | VARCHAR | 50 | NN | Identificador único do cliente |
| ds\_email | VARCHAR | 65 | NN | E-mail do gerente |
| vl\_salario | NUMERIC | 7 | N | Salário do gerente |
| ds\_endereco | VARCHAR | 150 | N | Descrição do endereço gerente |
| ds\_senha | VARCHAR | 25 | NN | Senha do gerente para entrar na aplicação web |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Tabela** | T\_CPP\_CATEGORIA | | | |
| **Descrição** | Tabela de avaliação. Relaciona-se com a tabela T\_CPP\_ESTABELECIMENTO e T\_CPP\_ITEM\_CARDAPIO | | | |
| **Coluna** | **Tipo de Dados** | **Tamanho** | **Constraint** | **Descrição** |
| cd\_categoria | NUMERIC | 7 | PK | Identificador único da categoria |
| cd\_estabelecimento | NUMERIC | 5 | FK | Identificador único do estabelecimento |
| nm\_categoria | VARCHAR | 30 | NN | Nome da categoria |

## 3 – Projeto Lógico do Banco de Dados



## 4 – Projeto Físico do Banco de Dados



## 5 – Data Definition Language – DDL

|  |
| --- |
| CREATE TABLE t\_cpp\_avaliacao (  cd\_avaliacao NUMBER(8) NOT NULL,  cd\_cliente NUMBER(5) NOT NULL,  cd\_estabelecimento NUMBER(3) NOT NULL,  nr\_experiencia NUMBER(1) NOT NULL,  ds\_avaliacao VARCHAR2(300)  );  ALTER TABLE t\_cpp\_avaliacao ADD CONSTRAINT t\_cpp\_avaliacao\_pk PRIMARY KEY ( cd\_avaliacao );  ALTER TABLE t\_cpp\_avaliacao  ADD CONSTRAINT fk\_cpp\_avaliacao\_cliente FOREIGN KEY ( cd\_cliente )  REFERENCES t\_cpp\_cliente ( cd\_cliente );  ALTER TABLE t\_cpp\_avaliacao  ADD CONSTRAINT fk\_cpp\_avaliacao\_estab FOREIGN KEY ( cd\_estabelecimento )  REFERENCES t\_cpp\_estabelecimento ( cd\_estabelecimento ); |

Código-fonte 1 – DDL da tabela t\_cpp\_avaliacao

Fonte: Elaborado pelo grupo Poagilers (2021)

|  |
| --- |
| CREATE TABLE t\_cpp\_cliente (  cd\_cliente NUMBER(8) NOT NULL,  nm\_cliente VARCHAR2(50) NOT NULL,  nr\_celular CHAR(11) NOT NULL,  ds\_senha VARCHAR2(30) NOT NULL  );  COMMENT ON COLUMN t\_cpp\_cliente.cd\_cliente IS  'até 99 milhões ';  COMMENT ON COLUMN t\_cpp\_cliente.nr\_celular IS  '(11)922332233';  ALTER TABLE t\_cpp\_cliente ADD CONSTRAINT t\_cpp\_cliente\_pk PRIMARY KEY ( cd\_cliente ); |

Código-fonte 2 – DDL da tabela t\_cpp\_cliente

Fonte: Elaborado pelo grupo Poagilers (2021)

|  |
| --- |
| CREATE TABLE t\_cpp\_estabelecimento (  cd\_estabelecimento NUMBER(5) NOT NULL,  cd\_gerente NUMBER(3) NOT NULL,  ds\_cnpj CHAR(14) NOT NULL,  nm\_razao\_social VARCHAR2(50 CHAR) NOT NULL,  nm\_fantasia VARCHAR2(50 CHAR) NOT NULL,  ds\_endereco VARCHAR2(100 CHAR) NOT NULL,  nr\_telefone CHAR(11 CHAR) NOT NULL,  ds\_email VARCHAR2(40) NOT NULL  );  ALTER TABLE t\_cpp\_estabelecimento ADD CONSTRAINT t\_cpp\_estabelecimento\_pk PRIMARY KEY ( cd\_estabelecimento );  ALTER TABLE t\_cpp\_estabelecimento  ADD CONSTRAINT fk\_cpp\_estab\_gerente FOREIGN KEY ( cd\_gerente )  REFERENCES t\_cpp\_gerente ( cd\_gerente ); |

Código-fonte 3 – DDL da tabela t\_cpp\_estabelecimento

Fonte: Elaborado pelo grupo Poagilers (2021)

|  |
| --- |
| CREATE TABLE t\_cpp\_gerente (  cd\_gerente NUMBER(5) NOT NULL,  nm\_gerente VARCHAR2(50) NOT NULL,  ds\_email VARCHAR2(65) NOT NULL,  vl\_salario NUMBER(7, 2),  ds\_endereco VARCHAR2(150),  ds\_senha VARCHAR2(25 CHAR) NOT NULL  );  ALTER TABLE t\_cpp\_gerente ADD CONSTRAINT t\_cpp\_gerente\_pk PRIMARY KEY ( cd\_gerente ); |

Código-fonte 4 – DDL da tabela t\_cpp\_gerente

Fonte: Elaborado pelo grupo Poagilers (2021)

|  |
| --- |
| CREATE TABLE t\_cpp\_item\_cardapio (  cd\_item\_cardapio NUMBER(7) NOT NULL,  cd\_estabelecimento NUMBER(3) NOT NULL,  cd\_categoria NUMBER(2) NOT NULL,  st\_destaque CHAR(1) NOT NULL,  nm\_item\_cardapio VARCHAR2(50) NOT NULL,  ds\_item\_cardapio VARCHAR2(300) NOT NULL,  vl\_item\_cardapio NUMBER(7, 2) NOT NULL,  ds\_ingredientes VARCHAR2(300) NOT NULL,  vl\_calorico NUMBER(4),  fl\_foto\_ic BLOB  );  COMMENT ON COLUMN t\_cpp\_item\_cardapio.fl\_foto\_ic IS  'IC = item cardapio';  ALTER TABLE t\_cpp\_item\_cardapio ADD CONSTRAINT t\_cpp\_item\_cardapio\_pk PRIMARY KEY ( cd\_item\_cardapio );  ALTER TABLE t\_cpp\_item\_cardapio  ADD CONSTRAINT fk\_cpp\_ic\_categoria FOREIGN KEY ( cd\_categoria )  REFERENCES t\_cpp\_categoria ( cd\_categoria );  ALTER TABLE t\_cpp\_item\_cardapio  ADD CONSTRAINT fk\_cpp\_ic\_estab FOREIGN KEY ( cd\_estabelecimento )  REFERENCES t\_cpp\_estabelecimento ( cd\_estabelecimento ); |

Código-fonte 5 – DDL da tabela t\_cpp\_item\_cardapio

Fonte: Elaborado pelo grupo Poagilers (2021)

|  |
| --- |
| CREATE TABLE t\_cpp\_pedido (  cd\_pedido NUMBER(8) NOT NULL,  cd\_cliente NUMBER(5) NOT NULL,  cd\_estabelecimento NUMBER(3) NOT NULL,  vl\_total\_pedido NUMBER(8, 2) NOT NULL,  st\_pedido VARCHAR2(15) NOT NULL,  hr\_pedido DATE NOT NULL  );  ALTER TABLE t\_cpp\_pedido  ADD CONSTRAINT ck\_cpp\_pedido CHECK ( st\_pedido IN ( 'RECEBIDO', 'EM PREPARO', 'ENTREGUE', 'PAGO' ) );  ALTER TABLE t\_cpp\_pedido ADD CONSTRAINT t\_cpp\_pedido\_pk PRIMARY KEY ( cd\_pedido );  ALTER TABLE t\_cpp\_pedido  ADD CONSTRAINT fk\_cpp\_pedido\_cliente FOREIGN KEY ( cd\_cliente )  REFERENCES t\_cpp\_cliente ( cd\_cliente );  ALTER TABLE t\_cpp\_pedido  ADD CONSTRAINT fk\_cpp\_pedido\_estab FOREIGN KEY ( cd\_estabelecimento )  REFERENCES t\_cpp\_estabelecimento ( cd\_estabelecimento ); |

Código-fonte 6 – DDL da tabela t\_cpp\_pedido

Fonte: Elaborado pelo grupo Poagilers (2021)

|  |
| --- |
| CREATE TABLE t\_cpp\_pedido\_item\_cardapio (  cd\_item\_cardapio NUMBER(9) NOT NULL,  cd\_pedido NUMBER(4) NOT NULL,  qt\_item\_cardapio NUMBER(2) NOT NULL  );  ALTER TABLE t\_cpp\_pedido\_item\_cardapio  ADD CONSTRAINT fk\_cpp\_pic\_ic FOREIGN KEY ( cd\_item\_cardapio )  REFERENCES t\_cpp\_item\_cardapio ( cd\_item\_cardapio );  ALTER TABLE t\_cpp\_pedido\_item\_cardapio  ADD CONSTRAINT fk\_cpp\_pic\_pedido FOREIGN KEY ( cd\_pedido )  REFERENCES t\_cpp\_pedido ( cd\_pedido ); |

Código-fonte 7 – DDL da tabela t\_cpp\_pedido\_item\_cardapio

Fonte: Elaborado pelo grupo Poagilers (2021)

|  |
| --- |
| CREATE TABLE t\_cpp\_categoria (  cd\_categoria NUMBER(7) NOT NULL,  cd\_estabelecimento NUMBER(5) NOT NULL,  nm\_categoria VARCHAR2(30 CHAR) NOT NULL  );  ALTER TABLE t\_cpp\_categoria ADD CONSTRAINT t\_cpp\_categoria\_pk PRIMARY KEY ( cd\_categoria );  ALTER TABLE t\_cpp\_categoria  ADD CONSTRAINT fk\_cpp\_categoria\_estab FOREIGN KEY ( cd\_estabelecimento )  REFERENCES t\_cpp\_estabelecimento ( cd\_estabelecimento ); |

Código-fonte 8 – DDL da tabela t\_cpp\_categoria

Fonte: Elaborado pelo grupo Poagilers (2021)

|  |
| --- |
| CREATE SEQUENCE cdcategoria  INCREMENT BY 1  START WITH 1  MAXVALUE 9999999  NOCYCLE;  CREATE SEQUENCE cdcliente  INCREMENT BY 1  START WITH 1  MAXVALUE 99999999  NOCYCLE;  CREATE SEQUENCE cdestabelecimento  INCREMENT BY 1  START WITH 1  MAXVALUE 99999  NOCYCLE;  CREATE SEQUENCE cdavaliacao  INCREMENT BY 1  START WITH 1  MAXVALUE 99999999  NOCYCLE;  CREATE SEQUENCE cdgerente  INCREMENT BY 1  START WITH 1  MAXVALUE 99999  NOCYCLE;  CREATE SEQUENCE cditemcardapio  INCREMENT BY 1  START WITH 1  MAXVALUE 9999999  NOCYCLE; |

Código-fonte 9 – Sequence

Fonte: Elaborado pelo grupo Poagilers (2021)

## 6 – Data Manupulation Language – DML (INSERT)

|  |
| --- |
| insert into t\_cpp\_gerente values (cdgerente.nextval,'Jorge','pessoa.eric@gmail.com', 1900.00, 'Rua do grito agudo, 202', 'Bananinha123');  insert into t\_cpp\_gerente values (cdgerente.nextval,'Débora','Debora@gmail.com', 2200.00, 'Rua do grito grave, 103', 'GoiabadaAt77');  insert into t\_cpp\_gerente values (cdgerente.nextval,'João','Joao@gmail.com', 1500.00, 'Rua do abacate, 30', 'Mocoto99'); |

Script em SQL para carga inicial do banco de dados na tabela t\_cpp\_gerente

|  |
| --- |
| INSERT INTO t\_cpp\_estabelecimento VALUES(cdestabelecimento.nextval, 1, '12345678000108', 'Milk Shakespeare LTDA', 'Milk Shakespeare', 'Rua djuvidas, 197', '11986759980', ['milk@gmail.com](mailto:'milk@gmail.com)');  INSERT INTO t\_cpp\_estabelecimento VALUES(cdestabelecimento.nextval, 2, '11145555000109','Churros Store SA', 'Churros Store', 'Rua taí, 22', '33978964456', 'churros@gmail.com' );  INSERT INTO t\_cpp\_estabelecimento VALUES(cdestabelecimento.nextval, 3, '22245678000105', 'Primos e Cia', 'Restaurante dos Primos', 'Rua Melissa, 39', '21987558563', 'primos@gmail.com'); |

Script em SQL para carga inicial do banco de dados na tabela t\_cpp\_estabelecimento

|  |
| --- |
| insert into t\_cpp\_categoria values (cdcategoria.nextval, 1, 'Sobremesa');  insert into t\_cpp\_categoria values (cdcategoria.nextval, 1, 'Bebida');  insert into t\_cpp\_categoria values (cdcategoria.nextval, 2, 'Sobremesa');  insert into t\_cpp\_categoria values (cdcategoria.nextval, 2, 'Aperitivo');  insert into t\_cpp\_categoria values (cdcategoria.nextval, 2, 'Bebida');  insert into t\_cpp\_categoria values (cdcategoria.nextval, 3, 'Prato');  insert into t\_cpp\_categoria values (cdcategoria.nextval, 3, 'Lanche');  insert into t\_cpp\_categoria values (cdcategoria.nextval, 3, 'Bebida'); |

Script em SQL para carga inicial do banco de dados na tabela t\_cpp\_categoria

|  |
| --- |
| INSERT INTO t\_cpp\_item\_cardapio VALUES(cditemcardapio.nextval, 1, 1, '1', 'Sorvete de Chocolate', '100g de sorvete sabor chocolate', 5.00, 'Leite, chocolate em pó e açúcar', 700, utl\_raw.cast\_to\_raw('D:\Área de Trabalho\Kaue\husky'));  INSERT INTO t\_cpp\_item\_cardapio VALUES(cditemcardapio.nextval, 1, 1, '0', 'Milk Shake de Menta com Chocolate', '150ml de sorvete sabor menta e chocolate', 7.00, 'Leite, menta e chocolate em pó e açúcar', 750, utl\_raw.cast\_to\_raw('D:\Área de Trabalho\Kaue\husky'));  INSERT INTO t\_cpp\_item\_cardapio VALUES(cditemcardapio.nextval, 2, 3, '1', 'Churros de Chocolate', '1 unidade de churros sabor chocolate', 7.00, 'Leite, chocolate, farinha de trigo, água e açúcar', 850, utl\_raw.cast\_to\_raw('D:\Área de Trabalho\Kaue\husky'));  INSERT INTO t\_cpp\_item\_cardapio VALUES(cditemcardapio.nextval, 2, 3, '0', 'Churros de Doce de Leite', '1 unidade de churros sabor doce de leite', 7.00, 'Leite, doce de leite, farinha de trigo, água e açúcar', 800, utl\_raw.cast\_to\_raw('D:\Área de Trabalho\Kaue\husky'));  INSERT INTO t\_cpp\_item\_cardapio VALUES(cditemcardapio.nextval, 3, 6, '1', 'Arroz', '100g de arroz do tipo 1', 3.00, 'Arroz Tipo 1', 200, utl\_raw.cast\_to\_raw('D:\Área de Trabalho\Kaue\husky'));  INSERT INTO t\_cpp\_item\_cardapio VALUES(cditemcardapio.nextval, 3, 6, '0', 'Feijão', '100g de feijão carioca', 4.00, 'Feijão Carioca', 300, utl\_raw.cast\_to\_raw('D:\Área de Trabalho\Kaue\husky'));  insert into t\_cpp\_item\_cardapio values(cditemcardapio.nextval, 1, 1,'1', 'Milkshake unicórnio','Com MeMs', 8.00, 'Leite, chocolate, caramelo, chantilly e MeM', 500, utl\_raw.cast\_to\_raw('D:\Área de Trabalho\Kaue\husky'));  insert into t\_cpp\_item\_cardapio values(cditemcardapio.nextval, 1, 1, '0', 'Milkshake ovomaltine','O preferido da casa', 12.00, 'Leite, chocolate', 300, utl\_raw.cast\_to\_raw('D:\Área de Trabalho\Kaue\husky'));  insert into t\_cpp\_item\_cardapio values(cditemcardapio.nextval, 2, 5, '1', 'Cerveja pilsen','A melhor cerveja do MUNDO', 10.00, 'Lúpulo, cevada', 43, utl\_raw.cast\_to\_raw('D:\Área de Trabalho\Kaue\husky'));  insert into t\_cpp\_item\_cardapio values(cditemcardapio.nextval, 2, 4, '0', 'Amendoim sem casca','Um bom aperitivo, talvez o melhor de todos', 2.00, 'Amendoim', 30, utl\_raw.cast\_to\_raw('D:\Área de Trabalho\Kaue\husky'));  insert into t\_cpp\_item\_cardapio values(cditemcardapio.nextval, 3, 8, '0', 'Vinho tinto espanhol','Quer ostentar? então toma, ano de 1930', 10000.00, 'Vinho tinto espanhol', 0, utl\_raw.cast\_to\_raw('D:\Área de Trabalho\Kaue\husky'));  insert into t\_cpp\_item\_cardapio values(cditemcardapio.nextval, 3, 6, '1', 'Virada à paulista','Venha comer esse clássico!', 15.00, 'Arroz, feijão, Bife, ovo frito, couve refogada e tomate.', 400, utl\_raw.cast\_to\_raw('D:\Área de Trabalho\Kaue\husky'));  insert into t\_cpp\_item\_cardapio values(cditemcardapio.nextval, 3, 7, '1', 'Lanche Cheddar Duplo','200g de hambúrguer artesanal e 100g de cheddar,', 20.00, 'Hambúrguer, alface, cheddar, tomate.', 500, utl\_raw.cast\_to\_raw('D:\Área de Trabalho\Kaue\husky')); |

Script em SQL para carga inicial do banco de dados na tabela t\_cpp\_item\_cardapio

|  |
| --- |
| insert into t\_cpp\_cliente values(1, 'Lari', '11960205342', '123');  insert into t\_cpp\_cliente values(2, 'Eric', '11293847581', '1234');  insert into t\_cpp\_cliente values(3, 'Kaue', '11384713947', '12345');  insert into t\_cpp\_cliente values(4, 'Henrique', '1137284950', '123456');  insert into t\_cpp\_cliente values(5, 'Gi', '11372816481', '1234567');  insert into t\_cpp\_cliente values(6, 'Dan', '11846265749', '12345678'); |

Script em SQL para carga inicial do banco de dados na tabela t\_cpp\_cliente

|  |
| --- |
| INSERT INTO t\_cpp\_avaliacao values(1, 1, 1, 5, 'Melhor sorvete da região muito saboroso.');  INSERT INTO t\_cpp\_avaliacao values(2, 2, 3, 5, 'A comida estava saborosa e ainda chegou quentinha.');  INSERT INTO t\_cpp\_avaliacao values(3, 3, 2, 3, 'Churros estava queimado');  INSERT INTO t\_cpp\_avaliacao values(4, 4, 1, 4, 'Sorvete todo derretido.');  INSERT INTO t\_cpp\_avaliacao values(5, 5, 3, 5, 'Comida muito gostosa.');  INSERT INTO t\_cpp\_avaliacao values(6, 6, 2, 2, 'Churros queimado e com pouco recheio.'); |

Script em SQL para carga inicial do banco de dados na tabela t\_cpp\_avaliacao

# 7 – Data Manupulation Language – DML (UPDATE / DELETE)

|  |
| --- |
| UPDATE t\_cpp\_gerente  SET ds\_email = 'jorge@gmail.com'  WHERE cd\_gerente = 1;  UPDATE t\_cpp\_gerente  SET nm\_gerente = 'Jorge Mendes'  WHERE cd\_gerente = 1;  UPDATE t\_cpp\_gerente  SET nm\_gerente = 'Débora Leão'  WHERE cd\_gerente = 2;  UPDATE t\_cpp\_gerente  SET nm\_gerente = 'João Lima'  WHERE cd\_gerente = 3;  UPDATE t\_cpp\_cliente  SET ds\_senha = 'asdfpoiu89'  WHERE cd\_cliente = 1;  UPDATE t\_cpp\_cliente  SET ds\_senha = 'ericDoChurras'  WHERE cd\_cliente = 2;  UPDATE t\_cpp\_cliente  SET ds\_senha = 'qwermnbv45'  WHERE cd\_cliente = 4;  UPDATE t\_cpp\_cliente  SET ds\_senha = '280101'  WHERE cd\_cliente = 5;  UPDATE t\_cpp\_cliente  SET ds\_senha = 'henri45'  WHERE cd\_cliente = 6; |
| DELETE FROM t\_cpp\_categoria;  DELETE FROM t\_cpp\_avaliacao;  DELETE FROM t\_cpp\_pedido\_item\_cardapio;  DELETE FROM t\_cpp\_item\_cardapio;  DELETE FROM t\_cpp\_pedido;  DELETE FROM t\_cpp\_cliente;  DELETE FROM t\_cpp\_estabelecimento;  DELETE FROM t\_cpp\_gerente; |

Comandos de Atualização e de Deleção

# 8 – Data Query Language – DQL (SELECT)

## 8.1 – Relatório simples que traz uma ocorrência especifica de cada tabela.

Relatório para leitura da avaliação de código 1:

|  |
| --- |
| select \* from t\_cpp\_avaliacao where cd\_avaliacao = 1; |

Relatório 1

Relatório para leitura do cliente de código 1:

|  |
| --- |
| select \* from t\_cpp\_cliente where cd\_cliente = 1; |

Relatório 2

Relatório para leitura do estabelecimento de código 1:

|  |
| --- |
| select \* from t\_cpp\_estabelecimento  where cd\_estabelecimento = 1; |

Relatório 3

Relatório para leitura do gerente de código 1:

|  |
| --- |
| select \* from t\_cpp\_gerente where cd\_gerente = 1; |

Relatório 4

Relatório para leitura do item do cardápio de código 1:

|  |
| --- |
| select \* from t\_cpp\_item\_cardapio where cd\_item\_cardapio = 1; |

Relatório 5

Relatório para leitura do pedido de código 1:

|  |
| --- |
| select \* from t\_cpp\_pedido where cd\_pedido = 1; |

Relatório 6

Relatório para leitura do item do pedido de código 2:

|  |
| --- |
| select \* from t\_cpp\_pedido\_item\_cardapio where cd\_pedido = 2; |

Relatório 7

Relatório para leitura da categoria de código 2:

|  |
| --- |
| select \* from t\_cpp\_categoria where cd\_categoria = 2; |

Relatório 8

## 8.2 – Relatório simples que traz todas as ocorrência da tabela.

Relatório para leitura da tabela t\_cpp\_avaliacao:

|  |
| --- |
| select \* from t\_cpp\_avaliacao; |

Relatório 1

Relatório para leitura da tabela t\_cpp\_cliente:

|  |
| --- |
| select \* from t\_cpp\_cliente; |

Relatório 2

Relatório para leitura da tabela t\_cpp\_estabelecimento:

|  |
| --- |
| select \* from t\_cpp\_estabelecimento; |

Relatório 3

Relatório para leitura da tabela t\_cpp\_gerente:

|  |
| --- |
| select \* from t\_cpp\_gerente; |

Relatório 4

Relatório para leitura da tabela t\_cpp\_item\_cardapio:

|  |
| --- |
| select \* from t\_cpp\_item\_cardapio; |

Relatório 5

Relatório para leitura da tabela t\_cpp\_pedido:

|  |
| --- |
| select \* from t\_cpp\_pedido; |

Relatório 6

Relatório para leitura da tabela t\_cpp\_pedido\_item\_cardapio:

|  |
| --- |
| select \* from t\_cpp\_pedido\_item\_cardapio; |

Relatório 7

Relatório para leitura da tabela t\_cpp\_categoria:

|  |
| --- |
| select \* from t\_cpp\_categoria; |

Relatório 8

## 8.3 – Relatório simples com selects que trazem alguma ocorrência específica da aplicação de acordo com algum parâmetro.

Relatório para leitura dos itens nr\_celular e ds\_senha da tabela t\_cpp\_cliente:

|  |
| --- |
| select nr\_celular, ds\_senha from t\_cpp\_cliente; |

Relatório 1

Relatório para leitura do select de login de um cliente específico:

|  |
| --- |
| select nr\_celular, ds\_senha  from t\_cpp\_cliente  where nr\_celular = 11960205342 and ds\_senha = 'asdfpoiu89'; |

Relatório 2

Relatório para leitura do select do nome e descrição do item do cardápio pelo nome:

|  |
| --- |
| select nm\_item\_cardapio, ds\_item\_cardapio  from t\_cpp\_item\_cardapio  where nm\_item\_cardapio = 'Arroz'; |

Relatório 3

Relatório para leitura do código e nome do estabelecimento:

|  |
| --- |
| select cd\_estabelecimento, nm\_fantasia  from t\_cpp\_estabelecimento; |

Relatório 4

Relatório para leitura de um item em destaque:

|  |
| --- |
| select st\_destaque,  cd\_estabelecimento  from t\_cpp\_item\_cardapio  where cd\_estabelecimento = 2 and st\_destaque = '1'; |

Relatório 5

Relatório para leitura dos status de todos os pedidos:

|  |
| --- |
| select st\_pedido  from t\_cpp\_pedido; |

Relatório 6

Relatório para leitura do status um pedido em específico:

|  |
| --- |
| select cd\_pedido,  st\_pedido  from t\_cpp\_pedido  where cd\_pedido = 2; |

Relatório 7

## 8.4 – Relatório simples com INNER JOIN envolvendo duas ou mais tabelas.

Relatório para leitura de todos os itens do cardápio de dado estabelecimento:

|  |
| --- |
| select e.cd\_estabelecimento, e.nm\_fantasia, ic.cd\_item\_cardapio, ic.ds\_item\_cardapio  from t\_cpp\_estabelecimento e inner join t\_cpp\_item\_cardapio ic  on e.cd\_estabelecimento = ic.cd\_estabelecimento  where e.cd\_estabelecimento = 3; |

Relatório 1

Relatório para leitura dos pratos a partir da categoria SOBREMESA dos itens do cardápio:

|  |
| --- |
| select e.cd\_estabelecimento, e.nm\_fantasia, ic.nm\_item\_cardapio, cat.nm\_categoria  from t\_cpp\_estabelecimento e inner join t\_cpp\_item\_cardapio ic  on e.cd\_estabelecimento = ic.cd\_estabelecimento  inner join t\_cpp\_categoria cat  on e.cd\_estabelecimento = cat.cd\_estabelecimento  where e.cd\_estabelecimento = 1 and cat.nm\_categoria = 'Sobremesa'; |

Relatório 2

Relatório para leitura dos pratos a partir da categoria PRATO dos itens do cardápio:

|  |
| --- |
| select e.cd\_estabelecimento, e.nm\_fantasia, ic.nm\_item\_cardapio, cat.nm\_categoria  from t\_cpp\_estabelecimento e inner join t\_cpp\_item\_cardapio ic  on e.cd\_estabelecimento = ic.cd\_estabelecimento  inner join t\_cpp\_categoria cat  on e.cd\_estabelecimento = cat.cd\_estabelecimento  where e.cd\_estabelecimento = 3 and cat.nm\_categoria = 'Prato'; |

Relatório 3

Relatório para leitura dos itens do pedido 6:

|  |
| --- |
| select p.cd\_pedido, ic.ds\_item\_cardapio, ic.vl\_item\_cardapio  from t\_cpp\_pedido\_item\_cardapio pic inner join t\_cpp\_item\_cardapio ic  on pic.cd\_item\_cardapio = ic.cd\_item\_cardapio  inner join t\_cpp\_pedido p  on pic.cd\_pedido = p.cd\_pedido  where pic.cd\_pedido = 6; |

Relatório 4

Relatório para leitura da comanda do pedido 3:

|  |
| --- |
| select pic.cd\_pedido,  ic.nm\_item\_cardapio,  p.hr\_pedido,  ic.ds\_item\_cardapio,  ic.vl\_item\_cardapio,  e.nm\_fantasia  from t\_cpp\_pedido\_item\_cardapio pic  inner join t\_cpp\_item\_cardapio ic  on pic.cd\_item\_cardapio = ic.cd\_item\_cardapio  inner join t\_cpp\_pedido p  on pic.cd\_pedido = p.cd\_pedido  inner join t\_cpp\_estabelecimento e  on p.cd\_estabelecimento = e.cd\_estabelecimento  where pic.cd\_pedido = 3; |

Relatório 5

Relatório para leitura de um select que envolve todas as tabelas de um determinado estabelecimento ordenando em ordem pelo nome do item do cardápio:

|  |
| --- |
| SELECT E.cd\_estabelecimento  ,E.nm\_fantasia  ,G.nm\_gerente  ,A.nr\_experiencia  ,C.nm\_cliente  ,IC.nm\_item\_cardapio  ,IC.ds\_item\_cardapio  ,IC.vl\_item\_cardapio  ,CAT.nm\_categoria  FROM T\_CPP\_ESTABELECIMENTO E  INNER JOIN T\_CPP\_GERENTE G  ON G.cd\_gerente = E.cd\_gerente  INNER JOIN T\_CPP\_AVALIACAO A  ON E.cd\_estabelecimento = A.cd\_estabelecimento  INNER JOIN T\_CPP\_CLIENTE C  ON C.cd\_cliente = A.cd\_cliente  INNER JOIN T\_CPP\_ITEM\_CARDAPIO IC  ON IC.cd\_estabelecimento = E.cd\_estabelecimento  INNER JOIN T\_CPP\_CATEGORIA CAT  on E.cd\_estabelecimento = CAT.cd\_estabelecimento  WHERE E.cd\_estabelecimento = 1  order by IC.nm\_item\_cardapio; |

Relatório 6

## 8.5 – Relatório simples com GROUP BY envolvendo duas ou mais tabelas.

Relatório para leitura da média das avaliações do estabelecimento 3:

|  |
| --- |
| select e.nm\_fantasia, avg(a.nr\_experiencia) as Nota  from t\_cpp\_estabelecimento e inner join t\_cpp\_avaliacao a  on e.cd\_estabelecimento = a.cd\_estabelecimento  where e.cd\_estabelecimento = 3  group by nm\_fantasia; |

Relatório 1

Relatório para leitura do total de itens de um pedido:

|  |
| --- |
| select count(ic.nm\_item\_cardapio) as TotalItens,  p.vl\_total\_pedido,  p.cd\_pedido  from t\_cpp\_pedido p  inner join t\_cpp\_pedido\_item\_cardapio pic  on p.cd\_pedido = pic.cd\_pedido  inner join t\_cpp\_item\_cardapio ic  on ic.cd\_item\_cardapio = pic.cd\_item\_cardapio  where p.cd\_pedido = 2  group by p.vl\_total\_pedido, p.cd\_pedido; |

Relatório 2

Relatório para leitura do total de avaliações de todos os estabelecimentos:

|  |
| --- |
| select e.nm\_fantasia, count(a.nr\_experiencia) as "Total de Experiências"  from t\_cpp\_avaliacao a  inner join t\_cpp\_estabelecimento e  on e.cd\_estabelecimento = a.cd\_estabelecimento  inner join t\_cpp\_cliente c  on c.cd\_cliente = a.cd\_cliente  group by e.nm\_fantasia, a.nr\_experiencia; |

Relatório 3

Relatório para leitura do total de avaliações de todos os estabelecimentos com o nome do cliente:

|  |
| --- |
| select e.cd\_estabelecimento, e.nm\_fantasia, count(a.nr\_experiencia) as "Total de Experiências", c.nm\_cliente  from t\_cpp\_avaliacao a  inner join t\_cpp\_estabelecimento e  on e.cd\_estabelecimento = a.cd\_estabelecimento  inner join t\_cpp\_cliente c  on c.cd\_cliente = a.cd\_cliente  group by e.cd\_estabelecimento, e.nm\_fantasia, a.nr\_experiencia, c.nm\_cliente  order by e.nm\_fantasia; |

Relatório 4

Relatório para leitura do o total de categorias que cada restaurante tem registrado:

|  |
| --- |
| SELECT e.cd\_estabelecimento, e.nm\_fantasia, count(cat.cd\_categoria) as "Total de categorias do estabelecimento"  FROM t\_cpp\_estabelecimento e  inner join t\_cpp\_categoria cat  on e.cd\_estabelecimento = cat.cd\_estabelecimento  GROUP BY e.cd\_estabelecimento, e.nm\_fantasia; |

Relatório 5

Relatório para leitura do nome de todas as categorias registradas por um restaurante específico:

|  |
| --- |
| SELECT e.cd\_estabelecimento, e.nm\_fantasia, cat.nm\_categoria  FROM t\_cpp\_estabelecimento e  inner join t\_cpp\_categoria cat  on e.cd\_estabelecimento = cat.cd\_estabelecimento  where e.cd\_estabelecimento = 2; |

Relatório 6

Relatório para leitura de todos os restaurantes que contém algum tipo específico de categoria :

|  |
| --- |
| SELECT e.cd\_estabelecimento, e.nm\_fantasia, cat.nm\_categoria  FROM t\_cpp\_estabelecimento e  inner join t\_cpp\_categoria cat  on e.cd\_estabelecimento = cat.cd\_estabelecimento  where cat.nm\_categoria = 'Sobremesa'; |

Relatório 7

Relatório para leitura de todos os itens do cardápio de algum restaurante, que são de determinada categoria :

|  |
| --- |
| SELECT ic.cd\_item\_cardapio, cat.nm\_categoria, ic.nm\_item\_cardapio, e.nm\_fantasia, e.cd\_estabelecimento  FROM t\_cpp\_item\_cardapio ic  INNER JOIN t\_cpp\_estabelecimento e  on ic.cd\_estabelecimento = e.cd\_estabelecimento  INNER JOIN t\_cpp\_categoria cat  on ic.cd\_categoria = cat.cd\_categoria  where cat.nm\_categoria = 'Sobremesa' and e.nm\_fantasia = 'Churros Store'; |

Relatório 8

Relatório para leitura do nome fantasia do estabelecimento e o nome e e-mail do gerente pelo código do estabelecimento:

|  |
| --- |
| SELECT E.nm\_fantasia, G.nm\_gerente, G.ds\_email  FROM T\_CPP\_ESTABELECIMENTO E  INNER JOIN T\_CPP\_GERENTE G  ON G.cd\_gerente = E.cd\_gerente  WHERE E.cd\_estabelecimento = 1; |

Relatório 9

Relatório para leitura do nome do estabelecimento da razao social e o fantasia pelo código do estabelecimento:

|  |
| --- |
| SELECT E.cd\_estabelecimento  ,E.nm\_fantasia  ,E.nm\_razao\_social  FROM T\_CPP\_ESTABELECIMENTO E  WHERE E.cd\_estabelecimento = 2; |

Relatório 10