

ESCOLA SENAI SUÍÇO-BRASILEIRA PAULO ERNESTO TOLLE

Documentação da Atividade para e-commerce

Elder Santos Lima

Gabriel Sales de Albuquerque

Jhordy Alexis de la Cruz Rosillo

Trabalho feito para a aula de Banco de dados ministrado pelos professores Eduardo Inocêncio e Átila.

São Paulo, Brasil

2023

Tabela User

- **userId** (INT): Chave primária que identifica exclusivamente um usuário.
- **name** (VARCHAR(255)): Nome do usuário.
- **phoneNum** (VARCHAR(255)): Número de telefone do usuário.

Tabela Buyer

- **userId** (INT): Chave estrangeira referenciando a tabela **User**.

Tabela Seller

- **userId** (INT): Chave estrangeira referenciando a tabela **User**.

Tabela BankCard

- **cardNumber** (VARCHAR(255)): Chave primária que identifica exclusivamente um cartão bancário.
- **userId** (INT): Chave estrangeira referenciando a tabela **User**.
- **bank** (VARCHAR(255)): Nome do banco associado ao cartão.
- **expiryDate** (DATE): Data de validade do cartão.

Tabela CreditCard

- **cardNumber** (VARCHAR(255)): Chave primária que identifica exclusivamente um cartão de crédito.
- **userId** (INT): Chave estrangeira referenciando a tabela **User**.
- **organization** (VARCHAR(255)): Organização emissora do cartão de crédito.

Tabela DebitCard

- **cardNumber** (VARCHAR(255)): Chave primária que identifica exclusivamente um cartão de débito.

Tabela Store

- **sid** (INT): Chave primária que identifica exclusivamente uma loja.
- **name** (VARCHAR(255)): Nome da loja.
- **startTime** (TIME): Horário de abertura da loja.
- **customerGrade** (DECIMAL(3, 2)): Classificação do cliente para a loja.
- **streetAddr** (VARCHAR(255)): Endereço da rua onde a loja está localizada.
- **city** (VARCHAR(255)): Cidade onde a loja está localizada.
- **province** (VARCHAR(255)): Província onde a loja está localizada.

Tabela product

- **pid** (INT): Chave primária que identifica exclusivamente um produto.
- **sid** (INT): Chave estrangeira referenciando a tabela **Store**.
- **name** (VARCHAR(255)): Nome do produto.
- **brand** (VARCHAR(255)): Marca do produto.
- **type** (VARCHAR(255)): Tipo do produto.
- **amount** (INT): Quantidade disponível do produto.
- **price** (DECIMAL(10, 2)): Preço do produto.
- **colour** (VARCHAR(255)): Cor do produto.
- **customerReview** (DECIMAL(3, 1)): Avaliação do cliente para o produto.
- **modelNumber** (VARCHAR(255)): Número do modelo do produto.

Tabela orderItem

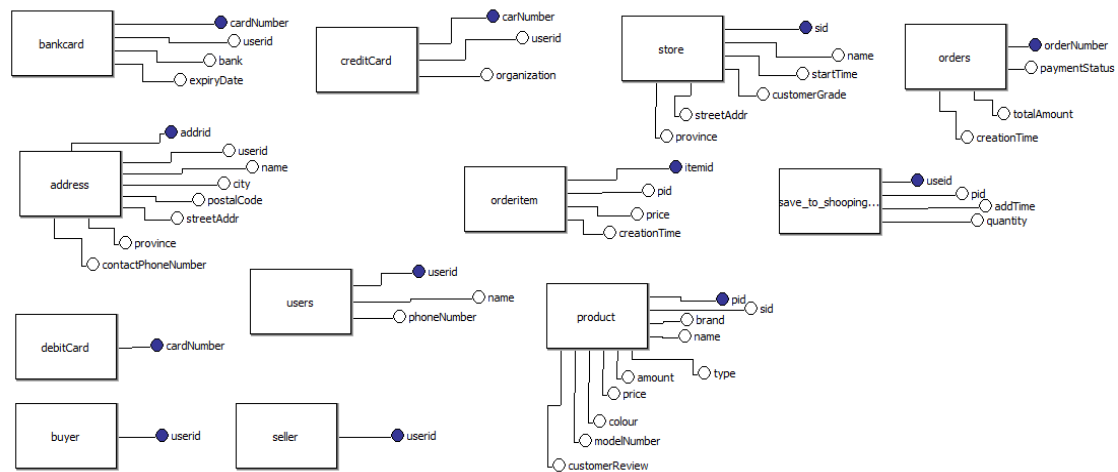
- **itemid** (INT): Chave primária que identifica exclusivamente um item de pedido.
- **pid** (INT): Chave estrangeira referenciando a tabela **Product**.
- **price** (DECIMAL(10, 2)): Preço do item.
- **creationTime** (DATETIME): Data e hora de criação do item.

Tabela orders

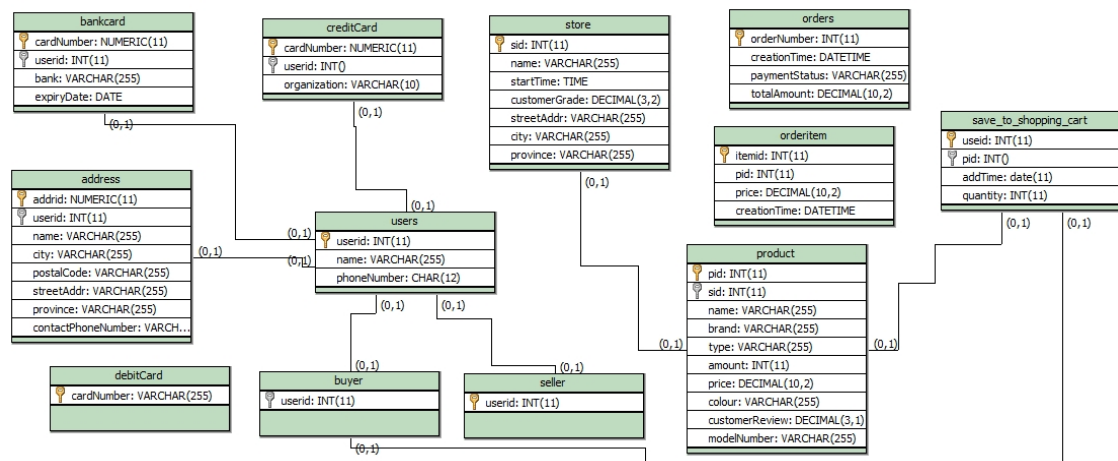
- **orderNumber** (INT): Chave primária que identifica exclusivamente um pedido.
- **creationTime** (DATETIME): Data e hora de criação do pedido.
- **paymentStatus** (VARCHAR(255)): Status de pagamento do pedido.
- **totalAmount** (DECIMAL(10, 2)): Valor total do pedido.

Tabela address

- **addrId** (INT): Chave primária que identifica exclusivamente um endereço.
- **userId** (INT): Chave estrangeira referenciando a tabela **User**.
- **name** (VARCHAR(255)): Nome associado ao endereço.
- **city** (VARCHAR(255)): Cidade do endereço.
- **postalCode** (VARCHAR(255)): Código postal do endereço.
- **streetAddr** (VARCHAR(255)): Endereço da rua.
- **province** (VARCHAR(255)): Província do endereço.
- **contactPhoneNumber** (VARCHAR(255)): Número de telefone de contato associado ao endereço.



Modelo Conceitual



Modelo Lógico

Aqui estão as tabelas que serão criadas no banco de dados "onlineShopping" utilizando o SQL:

```
CREATE TABLE user (
    userId INT PRIMARY KEY,
    name VARCHAR(255),
    phoneNum VARCHAR(255)
);
```

```
CREATE TABLE buyer (
    userId INT,
    FOREIGN KEY (userId) REFERENCES User(userId)
);
```

```
CREATE TABLE seller (  
    userId INT,  
    FOREIGN KEY (userId) REFERENCES User(userId)  
);
```

```
CREATE TABLE bankCard (  
    cardNumber VARCHAR(255) PRIMARY KEY,  
    userId INT,  
    bank VARCHAR(255),  
    expiryDate DATE,  
    FOREIGN KEY (userId) REFERENCES User(userId)  
);
```

```
CREATE TABLE creditCard (  
    cardNumber VARCHAR(255) PRIMARY KEY,  
    userId INT,  
    organization VARCHAR(255),  
    FOREIGN KEY (userId) REFERENCES User(userId)  
);
```

```
CREATE TABLE DebitCard (  
    cardNumber VARCHAR(255) PRIMARY KEY  
);
```

```
CREATE TABLE Store (  
    sid INT PRIMARY KEY,  
    name VARCHAR(255),  
    startTime TIME,  
    customerGrade DECIMAL(3, 2),  
    streetAddr VARCHAR(255),  
    city VARCHAR(255),  
    province VARCHAR(255)  
);
```

```
CREATE TABLE Product (  
    pid INT PRIMARY KEY,  
    sid INT,  
    name VARCHAR(255),  
    brand VARCHAR(255),  
    type VARCHAR(255),  
    amount INT,  
    price DECIMAL(10, 2),  
    colour VARCHAR(255),  
    customerReview DECIMAL(3, 1),  
    modelNumber VARCHAR(255),  
    FOREIGN KEY (sid) REFERENCES Store(sid)  
);
```

```
CREATE TABLE OrderItem (  
    itemid INT PRIMARY KEY,  
    pid INT,  
    price DECIMAL(10, 2),  
    creationTime DATETIME,  
    FOREIGN KEY (pid) REFERENCES Product(pid)  
);
```

```
CREATE TABLE Orders (  
    orderNumber INT PRIMARY KEY,  
    creationTime DATETIME,  
    paymentStatus VARCHAR(255),  
    totalAmount DECIMAL(10, 2)  
);
```

```
CREATE TABLE Address (  
    addrid INT PRIMARY KEY,  
    userId INT,  
    name VARCHAR(255),
```

```
city VARCHAR(255),  
postalCode VARCHAR(255),  
streetAddr VARCHAR(255),  
province VARCHAR(255),  
contactPhoneNumber VARCHAR(255),  
FOREIGN KEY (userId) REFERENCES User(userId)  
);
```

Criação das seguintes queries que permitam:

1. Consultar o endereço, hora de início (start time) e hora final (end time) dos pontos de serviço da mesma cidade que o usuário cujo ID é 5.

```
SELECT streetaddr, startTime, endTime  
  
FROM servicepoint  
  
WHERE city = (  
  
SELECT city  
  
FROM store  
  
WHERE sid = 8  
  
);
```

2. Consultar todos os produtos que é do tipo laptop.

```
SELECT *  
  
FROM product  
  
WHERE type = 'laptop';
```

3. Consultar a quantidade total de produtos que foram colocados no carrinho (shopping cart), considerando a loja com storeid (sid) igual a 8.

```
SELECT SUM(quantity) AS quantidade_total  
  
FROM save_to_shopping_cart  
  
WHERE pid IN (SELECT pid FROM product WHERE sid = 8);
```

4. Consultar os campos name, streetaddr e city (tabela address) de todos os pedidos que foram entregues em 17-02-2017.

```
SELECT a.name, a.streetAddr, a.city
```

```
FROM address a
```

```
JOIN deliver_to dt ON a.addrid = dt.addrid
```

```
JOIN orders o ON dt.orderNumber = o.orderNumber
```

```
WHERE dt.TimeDelivered >= '2017-02-17 00:00:00' AND  
dt.TimeDelivered <= '2017-02-17 23:59:59';
```

5. Consulte os comentários do produto 123456789;

```
SELECT *
```

```
FROM Comments
```

```
WHERE pid = 1;
```