

Disciplina: Programação para Internet B

Esta prova é composta por 3 fases, sendo que cada fase vale 10 pontos de um total de 30 pontos. A prova prática deve ser executada da seguinte forma:

- a) O aluno vai implementar o código + banco de dados referentes a descrição de cada parte. O aluno poderá consultar a internet e usar códigos de exemplos feitos em sala de aula.
- b) O enunciado de cada fase da prova estará disponível 10 minutos antes do início da aula.
- c) A tarefa no moodle para postagem do projeto web “COMPACTADO” de cada fase estará disponível até 5 dias após o termino da respectiva aula. Será avaliada essa postagem como cumprimento dos requisitos da prova.

Essa é a fase final da prova, portanto é desejável que o aluno poste um sistema web completo com as outras fases integradas.

Questões técnicas: O aluno deve criar um projeto web no eclipse com a seguinte nomenclatura: <nome Do Aluno>Parte3. ex. alencarMachadoParte3

Para essa fase, o aluno deve criar um arquivo LEIAME informando se tem interesse que essa fase seja avaliada com todos os pontos da fase anterior no seguinte formato:

Ex:

Fase 1: rejeitar

Fase 2: rejeitar

Fase 3: avaliar

PARTE 3

Para essa última fase da prova, o aluno terá 5 dias a partir do horário do início da prova para postar a página de vendas online que é descrita abaixo.

Obs: o layout dessa página deve seguir a imagem em anexo.

Questão 1: Deve-se desenvolver um JSP para a Venda de produtos e as respectivas classes e tabelas no banco de dados.

- a- Para que seja possível fazer a venda, o arquivo jsp (pagina web) deve disponibilizar para o usuário a possibilidade de selecionar um cliente, informar a data da venda, e incluir produtos que serão vendidos (item_venda)
- b- Cada item de uma venda é formado por um produto, a quantidade vendida e o total desse item. Para calcular o total do item, deve-se obter o preço do produto e multiplicar pela quantidade informada pelo usuário.
- c- Esse JSP deve mostrar o total da venda antes que essa venda seja finalizada, assim permitindo que o usuário tome a decisão mais apropriada sobre o valor vendido.

Diagrama de Classe

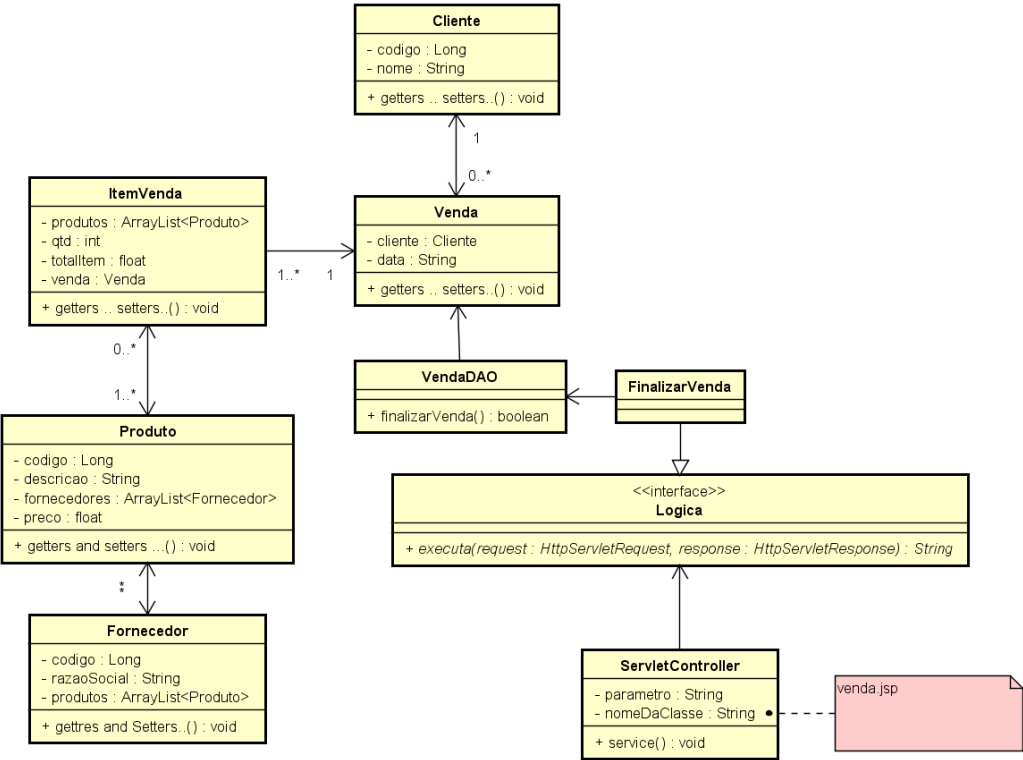
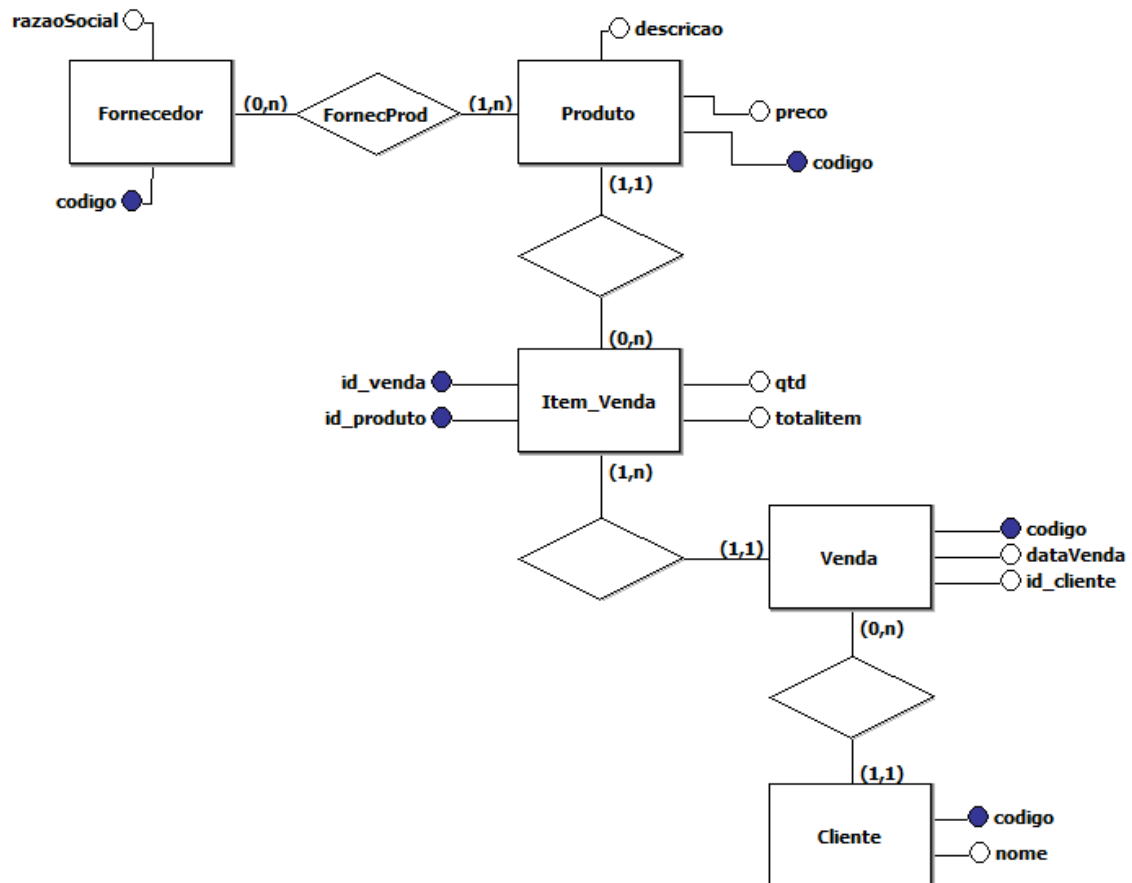


Diagrama Entidade-Relacionamento



SQL da Parte 1 e 2

```
create database prova1
```

```
create table produto ( codigo serial not null, descricao varchar(40) not null, preco float not null, primary key (codigo))
```

```
create table fornecedor ( codigo serial not null, razaoSocial varchar(40) not null, primary key (codigo))
```

```
create table fornecprod (codigoprod integer, codigofornec integer, primary key (codigoprod, codigofornec), FOREIGN KEY(codigoprod) REFERENCES produto(codigo), FOREIGN KEY(codigofornec) REFERENCES fornecedor(codigo))
```

INSERINDO DADOS

insert into produto (descricao, preco) values ('vinho tinto cabernet sauvignon', 21.75)

insert into produto (descricao, preco) values ('vinho tinto carmenere', 13.40)

insert into produto (descricao, preco) values ('carvão 5kg', 8.50)

insert into produto (descricao, preco) values ('carvão 3kg', 5.70)

insert into fornecedor (razaosocial) values ('FBI bebidas')

insert into fornecedor (razaosocial) values ('Carva Fornos LTDA')

insert into fornecprod (codigoprod, codigofornec) values (1,1)

insert into fornecprod (codigoprod, codigofornec) values (2,1)

insert into fornecprod (codigoprod, codigofornec) values (3,2)

insert into fornecprod (codigoprod, codigofornec) values (4,2)

SQL EXEMPLO ... retorna todos os produtos relacionados com o fornecedor 'FBI bebidas'

select p.codigo, p.descricao from fornecedor f, produto p, fornecprod fp

where f.codigo = fp.codigofornec and p.codigo = fp.codigoprod

and f.razaosocial = 'FBI bebidas'

Novas tabelas ..

create table cliente (codigo serial not null, nome varchar(40) not null, primary key (codigo))

insert into cliente (nome) values ('jocelaine')

insert into cliente (nome) values ('nego nelson')

insert into cliente (nome) values ('cicatriz')

create table venda (codigo serial not null, id_cliente integer not null, dataVenda varchar(11),
primary key (codigo))

insert into venda (id_cliente, dataVenda) values (1,'03/04/2016')

insert into venda (id_cliente, dataVenda) values (2,'03/04/2016')

create table item_venda (id_venda integer not null, id_produto integer not null,
qtd integer not null, totalitem float not null, primary key (id_venda, id_produto),
FOREIGN KEY(id_produto) REFERENCES produto(codigo),
FOREIGN KEY(id_venda) REFERENCES venda(codigo))

insert into item_venda (id_venda, id_produto, qtd, totalitem) values (1,1,2,43.5)

insert into item_venda (id_venda, id_produto, qtd, totalitem) values (1,2,1,14.4)

insert into item_venda (id_venda, id_produto, qtd, totalitem) values (2,2,10,140.4)

SQL EXEMPLO ... retorna todos os produtos da venda 1

```
select p.descricao, i.qtd, i.totalitem from produto p, item_venda i, venda v where i.id_produto = p.codigo and i.id_venda = v.codigo and v.codigo = 1
```

SQL EXEMPLO ... retorna o total da venda de código 1

```
select sum(i.totalitem) from item_venda i, venda v where i.id_venda = v.codigo and v.codigo = 1
```

LAYOUT...



Sistema de Vendas WEB

Nome do Usuário

Cliente:

Data

Produto:

Descrição

Quantidade

Preço Unitário

Total

Adicionar

Itens:

Produto	Descrição	Quantidade	Total
P3	P2	P1	P5
P3	P2	P1	P5
P3	P2	P1	P5
P3	P2	P1	P5
P3	P2	P1	P5

Total da Venda:

Finalizar Venda