

E-COMMERCE DE ROUPAS

Adam Yoshida

BI3002543

Gustavo Shoji

BI3003973

Willian Gustavo

BI3003965



Diagrama pé de galinha

Diagrama "pé de galinha" - Loja de roupas online

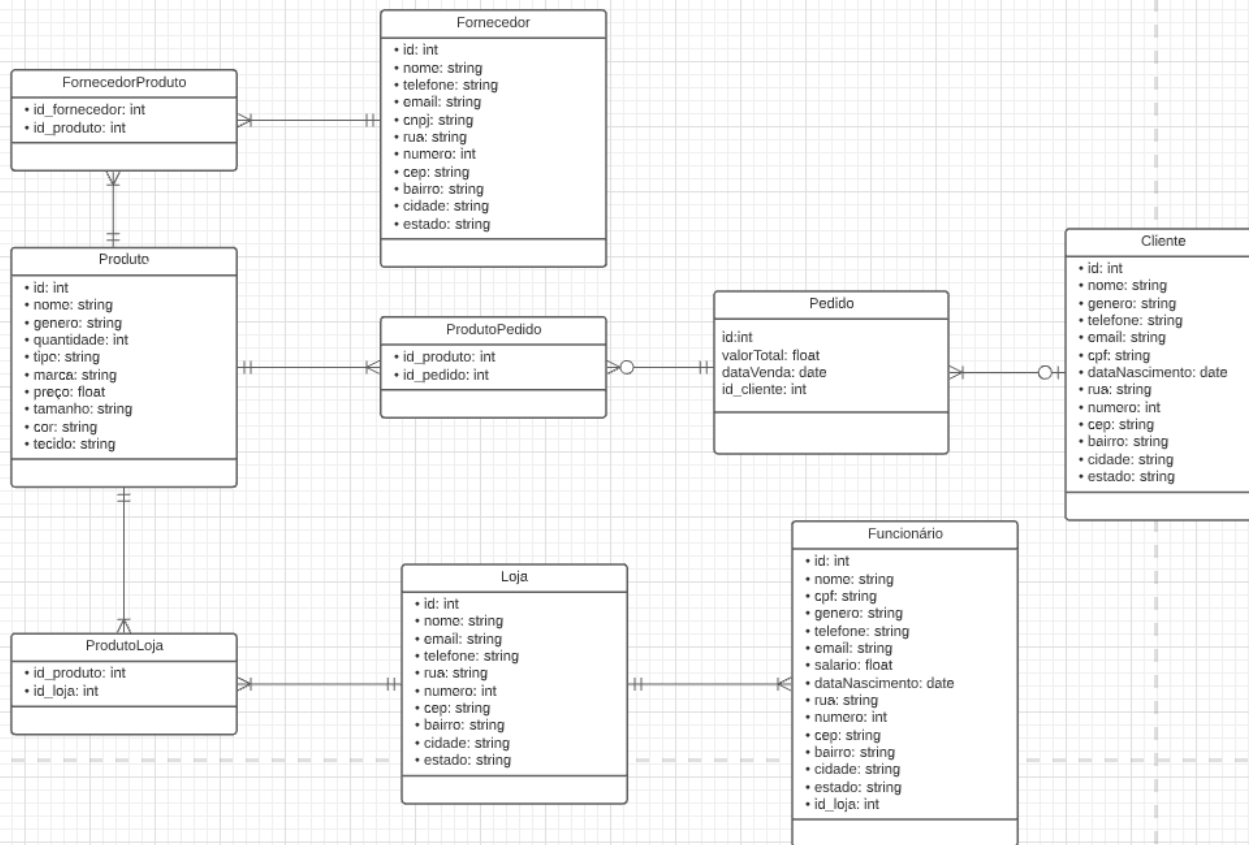


Tabela - Cliente

Nome da Coluna	Tipo de Dados	Anulável
ID	NUMBER	Não
NOME	VARCHAR2(50)	Não
GENERO	VARCHAR2(20)	Não
TELEFONE	VARCHAR2(25)	Não
EMAIL	VARCHAR2(50)	Não
DATANASCIMENTO	DATE	Não
RUA	VARCHAR2(40)	Não
NUMERO	NUMBER	Não
BAIRRO	VARCHAR2(40)	Não
CIDADE	VARCHAR2(30)	Não
ESTADO	VARCHAR2(20)	Não
CEP	VARCHAR2(10)	Não
CPF	VARCHAR2(15)	Não

Tabela - Fornecedor

Nome da Coluna	Tipo de Dados	Anulável
ID	NUMBER	Não
NOME	VARCHAR2(50)	Não
TELEFONE	VARCHAR2(20)	Não
EMAIL	VARCHAR2(50)	Não
CNPJ	VARCHAR2(25)	Não
RUA	VARCHAR2(40)	Não
NUMERO	NUMBER	Não
BAIRRO	VARCHAR2(40)	Não
CIDADE	VARCHAR2(30)	Não
ESTADO	VARCHAR2(20)	Não
CEP	VARCHAR2(10)	Não

Tabela - Funcionário

Nome da Coluna	Tipo de Dados	Anulável
ID	NUMBER	Não
NOME	VARCHAR2(50)	Não
CPF	VARCHAR2(20)	Não
GENERO	VARCHAR2(20)	Não
TELEFONE	VARCHAR2(20)	Não
EMAIL	VARCHAR2(50)	Não
SALARIO	FLOAT	Não
DATANASCIMENTO	DATE	Não
RUA	VARCHAR2(40)	Não
NUMERO	NUMBER	Não
BAIRRO	VARCHAR2(40)	Não
CIDADE	VARCHAR2(30)	Não
ESTADO	VARCHAR2(20)	Não
ID_LOJA	NUMBER	Não
CEP	VARCHAR2(10)	Não

Tabela - Loja

Nome da Coluna	Tipo de Dados	Anulável
ID	NUMBER	Não
NOME	VARCHAR2(50)	Não
TELEFONE	VARCHAR2(20)	Não
EMAIL	VARCHAR2(50)	Não
RUA	VARCHAR2(40)	Não
NUMERO	NUMBER	Não
BAIRRO	VARCHAR2(40)	Não
CIDADE	VARCHAR2(30)	Não
ESTADO	VARCHAR2(30)	Não
CEP	VARCHAR2(10)	Não

Tabela - Pedido

Nome da Coluna	Tipo de Dados	Anulável
ID	NUMBER	Não
VALORTOTAL	FLOAT	Não
DATAVENDA	DATE	Não
ID_CLIENTE	NUMBER	Não

Tabela - Produto

Nome da Coluna	Tipo de Dados	Anulável
ID	NUMBER	Não
NOME	VARCHAR2(50)	Não
GENERO	VARCHAR2(20)	Não
QUANTIDADE	NUMBER(5,0)	Não
PRECO	FLOAT	Não
TAMANHO	VARCHAR2(5)	Não
COR	VARCHAR2(20)	Não
TIPO	VARCHAR2(30)	Não
TECIDO	VARCHAR2(30)	Não
MARCA	VARCHAR2(30)	Não

Tabelas NxN

Nome da Coluna	Tipo de Dados	Anulável
ID	NUMBER	Não
ID_FORNECEDOR	NUMBER	Sim
ID_PRODUTO	NUMBER	Sim

Tabela FornecedorProduto

Nome da Coluna	Tipo de Dados	Anulável
ID	NUMBER	Não
ID_PRODUTO	NUMBER	Sim
ID_LOJA	NUMBER	Sim

Tabela ProdutoLoja

Nome da Coluna	Tipo de Dados	Anulável
ID	NUMBER	Não
ID_PRODUTO	NUMBER	Sim
ID_PEDIDO	NUMBER	Sim

Tabela ProdutoPedido

Constraints - Cliente

Constraint	Tipo	Condição de Pesquisa	Constraint Relacionada	Colunas
CLIENTE_CK_01	Verificação	NOME IS NOT NULL	-	-
CLIENTE_CK_02	Verificação	GENERO IS NOT NULL	-	-
CLIENTE_CK_03	Verificação	TELEFONE IS NOT NULL	-	-
CLIENTE_CK_04	Verificação	EMAIL IS NOT NULL	-	-
CLIENTE_CK_05	Verificação	DATANASCIMENTO IS NOT NULL	-	-
CLIENTE_CK_06	Verificação	RUA IS NOT NULL	-	-
CLIENTE_CK_07	Verificação	NUMERO IS NOT NULL	-	-
CLIENTE_PK	Primária	-	-	ID
CLIENTE_UK_01	Único	-	-	EMAIL
CLIENTE_UK_02	Único	-	-	TELEFONE

Constraints - Fornecedor

Constraint	Tipo	Condição de Pesquisa	Constraint Relacionada	Colunas
FORNECEDOR_CK_01	Verificação	NOME IS NOT NULL	-	-
FORNECEDOR_CK_02	Verificação	TELEFONE IS NOT NULL	-	-
FORNECEDOR_CK_03	Verificação	EMAIL IS NOT NULL	-	-
FORNECEDOR_CK_04	Verificação	CNPJ IS NOT NULL	-	-
FORNECEDOR_CK_05	Verificação	RUA IS NOT NULL	-	-
FORNECEDOR_CK_06	Verificação	NUMERO IS NOT NULL	-	-
FORNECEDOR_PK	Primária	-	-	ID
FORNECEDOR_UK_01	Único	-	-	CNPJ
FORNECEDOR_UK_02	Único	-	-	EMAIL

Constraints - Funcionário

Constraint	Tipo	Condição de Pesquisa	Constraint Relacionada	Colunas
FUNCIONARIO_FK_01	Estrangeira	-	FUNCIONARIO_FK_01 (WKSP_ADAMYS.LOJA)	ID_LOJA
FUNCIONARIO_PK	Primária	-	-	ID
FUNCIONARIO_UK_03	Único	-	-	CPF
FUNCIONARIO_UK_04	Único	-	-	TELEFONE
FUNCIONARIO_UK_05	Único	-	-	EMAIL
SYS_C00114206308	Verificação	"NOME" IS NOT NULL	-	-
SYS_C00114206341	Verificação	"CPF" IS NOT NULL	-	-
SYS_C00114206347	Verificação	"GENERO" IS NOT NULL	-	-
SYS_C00114206348	Verificação	"TELEFONE" IS NOT NULL	-	-
SYS_C00114206351	Verificação	"SALARIO" IS NOT NULL	-	-
SYS_C00114206357	Verificação	"EMAIL" IS NOT NULL	-	-
SYS_C00114206358	Verificação	"DATANASCIMENTO" IS NOT NULL	-	-
SYS_C00114206359	Verificação	"RUA" IS NOT NULL	-	-
SYS_C00114206360	Verificação	"NUMERO" IS NOT NULL	-	-
SYS_C00114206361	Verificação	"BAIRRO" IS NOT NULL	-	-
SYS_C00114206365	Verificação	"CIDADE" IS NOT NULL	-	-
SYS_C00114206368	Verificação	"ESTADO" IS NOT NULL	-	-
SYS_C00114206369	Verificação	"ID_LOJA" IS NOT NULL	-	-
SYS_C00114209007	Verificação	"CEP" IS NOT NULL	-	-

Constraints - Loja

Constraint	Tipo	Condição de Pesquisa	Constraint Relacionada	Colunas
LOJA_PK	Primária	-	-	ID
SYS_C00114206140	Verificação	"ID" IS NOT NULL	-	-
SYS_C00114206141	Verificação	"NOME" IS NOT NULL	-	-
SYS_C00114206142	Verificação	"TELEFONE" IS NOT NULL	-	-
SYS_C00114206143	Verificação	"EMAIL" IS NOT NULL	-	-
SYS_C00114206144	Verificação	"RUA" IS NOT NULL	-	-
SYS_C00114206145	Verificação	"NUMERO" IS NOT NULL	-	-
SYS_C00114206146	Verificação	"BAIRRO" IS NOT NULL	-	-
SYS_C00114206147	Verificação	"CIDADE" IS NOT NULL	-	-
SYS_C00114206148	Verificação	"ESTADO" IS NOT NULL	-	-
SYS_C00114209107	Verificação	"CEP" IS NOT NULL	-	-

Constraints - Pedido

Constraint	Tipo	Condição de Pesquisa	Constraint Relacionada	Colunas
PEDIDO_FK_01	Estrangeira	-	PEDIDO_FK_01 (WKSP_ADAMYS.CLIENTE)	ID_CLIENTE
PEDIDO_PK	Primária	-	-	ID
SYS_C00114206169	Verificação	"ID" IS NOT NULL	-	-
SYS_C00114206170	Verificação	"VALORTOTAL" IS NOT NULL	-	-
SYS_C00114206171	Verificação	"DATAVENDA" IS NOT NULL	-	-
SYS_C00114206172	Verificação	"ID_CLIENTE" IS NOT NULL	-	-

Constraints - Produto

Constraint	Tipo	Condição de Pesquisa	Constraint Relacionada	Colunas
PRODUTO_CK_01	Verificação	NOME IS NOT NULL	-	-
PRODUTO_CK_02	Verificação	GENERO IS NOT NULL	-	-
PRODUTO_CK_03	Verificação	QUANTIDADE IS NOT NULL	-	-
PRODUTO_CK_04	Verificação	PRECO IS NOT NULL	-	-
PRODUTO_CK_05	Verificação	TAMANHO IS NOT NULL	-	-
PRODUTO_CK_06	Verificação	COR IS NOT NULL	-	-
PRODUTO_CK_07	Verificação	TIPO IS NOT NULL	-	-
PRODUTO_CK_08	Verificação	TECIDO IS NOT NULL	-	-
PRODUTO_CK_09	Verificação	MARCA IS NOT NULL	-	-
PRODUTO_PK	Primária	-	-	ID

Constraints - FornecedorProduto

Constraint	Tipo	Condição de Pesquisa	Constraint Relacionada	Colunas
FORNECEDORPRODUTO_CK_01	Verificação	ID IS NOT NULL	-	-
FORNECEDORPRODUTO_CK_02	Verificação	ID_FORNECEDOR IS NOT NULL	-	-
FORNECEDORPRODUTO_CK_03	Verificação	ID_PRODUTO IS NOT NULL	-	-
FORNECEDORPRODUTO_FK_01	Estrangeira	-	FORNECEDORPRODUTO_FK_01 (WKSP_ADAMYS.FORNECEDOR)	ID_FORNECEDOR
FORNECEDORPRODUTO_FK_02	Estrangeira	-	FORNECEDORPRODUTO_FK_02 (WKSP_ADAMYS.PRODUTO)	ID_PRODUTO
FORNECEDORPRODUTO_PK	Primária	-	-	ID

Constraints - ProdutoLoja

Constraint	Tipo	Condição de Pesquisa	Constraint Relacionada	Colunas
PRODUTOLOJA_CK_01	Verificação	ID IS NOT NULL	-	-
PRODUTOLOJA_CK_02	Verificação	ID_PRODUTO IS NOT NULL	-	-
PRODUTOLOJA_CK_03	Verificação	ID_LOJA IS NOT NULL	-	-
PRODUTOLOJA_FK_01	Estrangeira	-	PRODUTOLOJA_FK_01 (WKSP_ADAMYS.PRODUTO)	ID_PRODUTO
PRODUTOLOJA_FK_02	Estrangeira	-	PRODUTOLOJA_FK_02 (WKSP_ADAMYS.LOJA)	ID_LOJA
PRODUTOLOJA_PK	Primária	-	-	ID

Constraints - ProdutoPedido

Constraint	Tipo	Condição de Pesquisa	Constraint Relacionada	Colunas
PRODUTOPEDIDO_CK_01	Verificação	ID IS NOT NULL	-	-
PRODUTOPEDIDO_CK_02	Verificação	ID_PRODUTO IS NOT NULL	-	-
PRODUTOPEDIDO_CK_03	Verificação	ID_PEDIDO IS NOT NULL	-	-
PRODUTOPEDIDO_FK_01	Estrangeira	-	PRODUTOPEDIDO_FK_01 (WKSP_ADAMYS.PRODUTO)	ID_PRODUTO
PRODUTOPEDIDO_FK_02	Estrangeira	-	PRODUTOPEDIDO_FK_02 (WKSP_ADAMYS.PEDIDO)	ID_PEDIDO
PRODUTOPEDIDO_PK	Primária	-	-	ID

Package - Cliente

Especificação:

Procedure de inclusão, alterar e excluir

Funções de busca para os tipos: number, varchar2 e date

Funções de verificar existência para os tipos: varchar2 e date

```
create or replace PACKAGE PKG_CLIENTE AS
vr_cliente  cliente%ROWTYPE;
PROCEDURE PR_INCLUIR      ( pr_cliente IN  cliente%ROWTYPE,
                           pv_erro    OUT varchar2
                           );
PROCEDURE PR_ALTERAR      ( pr_cliente IN  cliente%ROWTYPE );
PROCEDURE PR_EXCLUIR      ( pn_id       IN  number );
FUNCTION  FN_BUSCAR        ( pn_id       IN  number ) return cliente%ROWTYPE;
FUNCTION  FN_BUSCAR        ( pv_conteudo IN  varchar2, pn_tipoBusca IN number) return cliente%ROWTYPE;
FUNCTION  FN_BUSCAR        ( pn_data    IN  date ) return cliente%ROWTYPE;
FUNCTION  FN_VERIFICAREXISTE ( pv_conteudo IN  varchar2, pn_tipoBusca IN number) return boolean;
FUNCTION  FN_VERIFICAREXISTE ( pv_conteudo IN  date) return boolean;
END PKG_CLIENTE;
```

Package - Cliente

Corpo:

Procedure de inclusão

```
PROCEDURE PR_INCLUIR ( pr_cliente IN cliente%ROWTYPE,  
                       pv_erro   OUT varchar2  
                     ) AS  
  
    vb_existe  boolean;  
    vn_id      number;  
  
    BEGIN  
        pv_erro := '';  
        vb_existe := FN_VERIFICAREXISTE ( pv_conteudo => pr_cliente.email,  
                                           pn_tipoBusca => 1  
                                         );  
  
        if vb_existe then  
            pv_erro := 'Cliente já cadastrado para o Email informado';  
        else vb_existe := FN_VERIFICAREXISTE ( pv_conteudo => pr_cliente.telefone,  
                                               pn_tipoBusca => 2  
                                             );  
            if vb_existe then  
                pv_erro := 'Cliente já cadastrado para o telefone informado';  
            end if;  
        end if;  
    end if;
```

Parte 1 -
procedure de
inclusão

Package - Cliente

Corpo:

Procedure de inclusão

```
if pv_erro is null then
    select CLIENTE_SEQ_01.nextval
    into   vn_id
    from   dual;

    insert into cliente (id,   nome, genero, telefone, email, dataNascimento, rua, numero, bairro, cidade, estado, cpf)
    values (vn_id, pr_cliente.nome, pr_cliente.genero, pr_cliente.telefone, pr_cliente.email, pr_cliente.dataNascimento,
            pr_cliente.rua, pr_cliente.numero, pr_cliente.bairro, pr_cliente.cidade, pr_cliente.estado, pr_cliente.cpf);
end if;
END PR_INCLUIR;
```

Parte 2 - procedure de inclusão

Package - Cliente

Teste de inclusão:

```
declare
  vr_cliente cliente%ROWTYPE;
  vv_erro   varchar2(200);

begin
  vr_cliente.nome := 'Matheus Magalhaes Martins';
  vr_cliente.genero := 'Masculino';
  vr_cliente.telefone := '1893594354';
  vr_cliente.email := 'magal.magalhaes@gmail.com';
  vr_cliente.dataNascimento := '08-03-1999';
  vr_cliente.rua := 'Rua Augusta';
  vr_cliente.numero := 666;
  vr_cliente.bairro := 'Jardim Perola II';
  vr_cliente.cidade := 'Botucatu';
  vr_cliente.estado := 'Sao Paulo';
  vr_cliente.cep := '12413-423';
  vr_cliente.cpf := '321.321.321-12';

  PKG_CLIENTE.PR_INCLUIR(vr_cliente, vv_erro);
  if vv_erro is not null then
    raise_application_error (-20001, vv_erro);
  end if;
end;
```

Package - Cliente

Select após o teste:

ID	NOME	GENERO	TELEFONE	EMAIL	DATANASCIMENTO	RUA	NUMERO	BAIRRO	CIDADE	ESTADO	CEP	CPF
2	Jose	Masculino	18991304050	jose@gmail.com	21/09/90	Rua Benedito Faria Marques	52	Parque Maria Helena	Suzano	Sao Paulo	08683-000	268.916.340-32
21	Carlos	Masculino	18991865321	carlos@gmail.com	10/05/05	Boulevard São Caetano	222	Cerâmica	São Caetano do Sul	Sao Paulo	08683-000	834.800.540-48
6	Matheus Magalhaes Martins	Masculino	1893594354	magal.magalhaes@gmail.com	08/03/99	Rua Augusta	666	Jardim Perola II	Botucatu	Sao Paulo	12413-423	321.321.321-12

Package - Cliente

Corpo:

Procedure de alterar

```
PROCEDURE PR_ALTERAR ( pr_cliente IN cliente%ROWTYPE ) AS
BEGIN
    update cliente
    set     nome = pr_cliente.nome,
           genero = pr_cliente.genero,
           telefone = pr_cliente.telefone,
           email = pr_cliente.email,
           dataNascimento = pr_cliente.dataNascimento,
           rua = pr_cliente.rua,
           numero = pr_cliente.numero,
           bairro = pr_cliente.bairro,
           cidade = pr_cliente.cidade,
           estado = pr_cliente.estado,
           cpf = pr_cliente.cpf

    where id = pr_cliente.id;
END PR_ALTERAR;
```

Package - Cliente

Teste de alteração:

```
declare
    vr_cliente  cliente%ROWTYPE;

begin
    vr_cliente.id := 6;
    vr_cliente.nome := 'Matheus Magalhaes Martins';
    vr_cliente.genero := 'Masculino';
    vr_cliente.telefone := '1893594354';
    vr_cliente.email := 'magal.magalhaes@gmail.com';
    vr_cliente.dataNascimento := '08-03-1999';
    vr_cliente.rua := 'Rua Augusta dos Santos';
    vr_cliente.numero := 650;
    vr_cliente.bairro := 'Jardim Perola II';
    vr_cliente.cidade := 'Botucatu';
    vr_cliente.estado := 'Sao Paulo';
    vr_cliente.cep := '12413-423';
    vr_cliente.cpf := '321.321.321-12';

    PKG_CLIENTE.PR_ALTERAR(vr_cliente);

end;
```

Package - Cliente

Select após o teste:

ID	NOME	GENERO	TELEFONE	EMAIL	DATANASCIMENTO	RUA	NUMERO	BAIRRO	CIDADE	ESTADO	CEP	CPF
2	Jose	Masculino	18991304050	jose@gmail.com	21/09/90	Rua Benedito Faria Marques	52	Parque Maria Helena	Suzano	Sao Paulo	08683-000	268.916.340-32
21	Carlos	Masculino	18991865321	carlos@gmail.com	10/05/05	Boulevard São Caetano	222	Cerâmica	São Caetano do Sul	Sao Paulo	08683-000	834.800.540-48
6	Matheus Magalhaes Martins	Masculino	1893594354	magal.magalhaes@gmail.com	08/03/99	Rua Augusta dos Santos	650	Jardim Perola II	Botucatu	Sao Paulo	12413-423	321.321.321-12

Mudanças feitas: número e rua

Package - Cliente

Corpo:

Procedure de excluir

```
PROCEDURE PR_EXCLUIR ( pn_id IN number ) AS
BEGIN
    delete from cliente
    where id = pn_id;
END PR_EXCLUIR;
```

Package - Cliente

Teste de exclusão:

```
declare
    vr_cliente  cliente%ROWTYPE;
    vr_busca    NUMBER;

begin

    vr_busca := 6;

    PKG_CLIENTE.PR_EXCLUIR(vr_busca);

end;
```

Package - Cliente

Select após o teste:

ID	NOME	GENERO	TELEFONE	EMAIL	DATANASCIMENTO	RUA	NUMERO	BAIRRO	CIDADE	ESTADO	CEP	CPF
22	Gabriela	Feminino	18991854321	gabi123@gmail.com	10/03/01	Rua Argentina	456	Saudade	São Paulo	São Paulo	42683-500	436.432.540-48
2	Jose	Masculino	18991304050	jose@gmail.com	21/09/90	Rua Benedito Faria Marques	52	Parque Maria Helena	Suzano	Sao Paulo	08683-000	268.916.340-32
21	Carlos	Masculino	18991865321	carlos@gmail.com	10/05/05	Boulevard São Caetano	222	Cerâmica	São Caetano do Sul	Sao Paulo	08683-000	834.800.540-48
23	Fernanda	Feminino	11991756321	fernanda123@gmail.com	14/06/04	Rua Venezuela	532	Vila Estadio	Diadema	São Paulo	91387-420	126.672.574-48
4 linhas retornadas em 0,00 segundos Fazer Download												

```

r_cliente.email, vr_cliente.dataNascimento,
cliente.cidade, vr_cliente.estado, vr_cliente.cpf;

```

010

Package - Cliente

Teste de busca por id:

```
40 declare
41     vr_cliente  cliente%ROWTYPE;
42     vr_busca    NUMBER;
43
44 begin
45
46     vr_busca := 6;
47
48     vr_cliente := PKG_CLIENTE.FN_BUSCAR(vr_busca);
49     DBMS_OUTPUT.put_line(vr_cliente.nome);
50     DBMS_OUTPUT.put_line(vr_cliente.genero);
51 end;
```

Resultados	Explicação	Descrever	Instrução SQL Salva	Histórico
------------	------------	-----------	---------------------	-----------

Matheus Magalhaes Martins
Masculino

Instrução processada.

0,00 segundos

Package - Cliente

Corpo:

Função de buscar - varchar2

```
FUNCTION FN_BUSCAR ( pv_conteudo IN varchar2, pn_tipoBusca IN number) return cliente%ROWTYPE is

vr_cliente  Cliente%ROWTYPE;

cursor Busca_ClienteEmail is
select id, nome, genero, telefone, email, dataNascimento, rua, cep, numero, bairro, cidade, estado, cpf
from cliente
where upper(email) = upper(pv_conteudo);

cursor Busca_ClienteTelefone is
select id, nome, genero, telefone, email, dataNascimento, rua, cep, numero, bairro, cidade, estado, cpf
from cliente
where upper(telefone) = upper(pv_conteudo);
```

Parte 1 - Função de buscar - varchar2

Package - Cliente

Corpo:

Função de buscar - varchar2

```
BEGIN
  if
    pn_tipoBusca = 1 then
      open Busca_ClienteEmail;
      fetch Busca_ClienteEmail into vr_cliente.id, vr_cliente.nome, vr_cliente.genero, vr_cliente.telefone, vr_cliente.email, vr_cliente.dataNascimento,
                                         vr_cliente.rua, vr_cliente.cep, vr_cliente.numero, vr_cliente.bairro, vr_cliente.cidade, vr_cliente.estado, vr_cliente.cpf;
      if Busca_ClienteEmail%NotFound then
        vr_cliente := null;
      end if;
      close Busca_ClienteEmail;
    end if;

    if
      pn_tipoBusca = 2 then
        open Busca_ClienteTelefone;
        fetch Busca_ClienteTelefone into vr_cliente.id, vr_cliente.nome, vr_cliente.genero, vr_cliente.telefone, vr_cliente.email, vr_cliente.dataNascimento,
                                         vr_cliente.rua, vr_cliente.cep, vr_cliente.numero, vr_cliente.bairro, vr_cliente.cidade, vr_cliente.estado, vr_cliente.cpf;
        if Busca_ClienteTelefone%NotFound then
          vr_cliente := null;
        end if;
        close Busca_ClienteTelefone;
      end if;

  RETURN vr_cliente;
END FN_BUSCAR;
```

Parte 2 - Função de buscar - varchar2

Package - Cliente

Teste de busca por varchar2:

```
38 select * from cliente
39
40 declare
41     vr_cliente cliente%ROWTYPE;
42     vr_busca  VARCHAR2(200);
43
44 begin
45
46     vr_busca := 'magal.magalhaes@gmail.com';
47
48     vr_cliente := PKG_CLIENTE.FN_BUSCAR(vr_busca, 1);
49     DBMS_OUTPUT.put_line(vr_cliente.nome);
50     DBMS_OUTPUT.put_line(vr_cliente.genero);
51 end;
```

Resultados	Explicação	Descrever	Instrução SQL Salva	Histórico
Matheus Magalhaes Martins				
Masculino				
Instrução processada.				
0,01 segundos				

Package - Cliente

Corpo:

Função de buscar - data

```
FUNCTION FN_BUSCAR ( pn_data          IN date ) return cliente%ROWTYPE is

vr_cliente  Cliente%ROWTYPE;

cursor Busca_ClienteDataNascimento is
select id, nome, genero, telefone, email, dataNascimento, rua, cep, numero, bairro, cidade, estado, cpf
from cliente
where dataNascimento = pn_data;

BEGIN
  open   Busca_ClienteDataNascimento;
  fetch  Busca_ClienteDataNascimento into vr_cliente.id, vr_cliente.nome, vr_cliente.genero, vr_cliente.telefone, vr_cliente.email, vr_cliente.dataNascimento,
                                             vr_cliente.rua, vr_cliente.cep, vr_cliente.numero, vr_cliente.bairro, vr_cliente.cidade, vr_cliente.estado, vr_cliente.cpf;
  if     Busca_ClienteDataNascimento%NotFound then
    vr_cliente := null;
  end if;
  close  Busca_ClienteDataNascimento;

  RETURN vr_cliente;
END FN_BUSCAR;
```

Package - Cliente

Teste de busca por data

```
44 declare
45     vr_cliente  cliente%ROWTYPE;
46     vr_busca    DATE;
47
48 begin
49
50     vr_busca := '08-03-1999';
51
52     vr_cliente := PKG_CLIENTE.FN_BUSCAR(vr_busca);
53     DBMS_OUTPUT.put_line(vr_cliente.nome);
54     DBMS_OUTPUT.put_line(vr_cliente.genero);
55 end;
```

Resultados	Explicação	Descrever	Instrução SQL Salva	Histórico
Matheus Magalhaes Martins				
Masculino				
Instrução processada.				
0,01 segundos				

Package - Cliente

Corpo:

Verificar existência - varchar2

```
FUNCTION FN_VERIFICAREXISTE ( pv_contenido  IN  varchar2, pn_tipoBusca  IN  number) return boolean is

    vb_existe  boolean;
    vn_qtde    number;

    cursor Busca_ClienteEmail is
    select count(*)
    from   cliente
    where  upper(email) = upper(pv_contenido);

    cursor Busca_ClienteTelefone is
    select count(*)
    from   cliente
    where  upper(telefone) = upper(pv_contenido);
```

Parte 1 - Verificar existência - varchar2

Package - Cliente

Corpo:

Verificar existência - varchar2

```
BEGIN
  if    pn_tipoBusca = 1 then
    open  Busca_ClienteEmail;
    fetch Busca_ClienteEmail into vn_qtde;
    if    Busca_ClienteEmail%NotFound then
      vn_qtde := 0;
    end if;
    close Busca_ClienteEmail;
  end if;

  if    pn_tipoBusca = 2 then
    open  Busca_ClienteTelefone;
    fetch Busca_ClienteTelefone into vn_qtde;
    if    Busca_ClienteTelefone%NotFound then
      vn_qtde := 0;
    end if;
    close Busca_ClienteTelefone;
  end if;

  if    vn_qtde = 0 then
    vb_existe := false;
  else
    vb_existe := true;
  end if;

  return vb_existe;
END FN_VERIFICAREXISTE;
```

Parte 2 - Verificar existência - varchar2

Package - Cliente

*Teste de verificação de
varchar2:*

001

```
29 declare
30     vr_cliente cliente%ROWTYPE;
31     vr_busca  VARCHAR2(200);
32     vr_aux    BOOLEAN;
33
34 begin
35
36     vr_busca := 'magal.magalhaes@gmail.com';
37
38     vr_aux := PKG_CLIENTE.FN_VERIFICAREXISTE(vr_busca, 1);
39     DBMS_OUTPUT.PUT_LINE(sys.diutil.bool_to_int(vr_aux));
40 end;
41
42
43
44 declare
45     vr_cliente cliente%ROWTYPE;
```

Resultados	Explicação	Descrever	Instrução SQL Salva	Histórico
1	Instrução processada.			
0,02 segundos				

Package - Cliente

Corpo:

Verificar existência - data

```
FUNCTION FN_VERIFICAREXISTE ( pv_conteudo  IN  date) return boolean is

    vb_existe  boolean;

    vn_qtde    number;

    cursor Busca_ClienteDataNascimento is
    select count(*)
    from cliente
    where upper(dataNascimento) = upper(pv_conteudo);
```

Parte 1 - Verificar existência - data

Package - Cliente

Corpo:

Verificar existência - data

```
BEGIN

  open  Busca_ClienteDataNascimento;
  fetch Busca_ClienteDataNascimento into vn_qtde;
  if    Busca_ClienteDataNascimento%NotFound then
    vn_qtde := 0;
  end if;
  close Busca_ClienteDataNascimento;

  if    vn_qtde = 0 then
    vb_existe := false;
  else  vb_existe := true;
  end if;

  return vb_existe;
END FN_VERIFICAREXISTE;

END PKG_CLIENTE;
```

Parte 2 - Verificar existência - data

Package - Cliente

Teste de verificação de data:

```
28
29 declare
30     vr_cliente  cliente%ROWTYPE;
31     vr_busca    DATE;
32     vr_aux      BOOLEAN;
33
34 begin
35
36     vr_busca := '08-03-1999';
37
38     vr_aux := PKG_CLIENTE.FN_VERIFICAREXISTE(vr_busca);
39     DBMS_OUTPUT.PUT_LINE(sys.diutil.bool_to_int(vr_aux));
40 end;
41
42
43
```

Resultados	Explicação	Descrever	Instrução SQL Salva	Histórico
1	Instrução processada.			
0,01 segundos				

Package - Fornecedor

Especificação:

Procedure de inclusão, alterar e excluir

Funções de busca para os tipos: number, varchar2

Funções de verificar existência para os tipos: varchar2

Package - Funcionário

Especificação:

Procedure de inclusão, alterar e excluir

Funções de busca para os tipos: number, varchar2, date

Funções de verificar existência para os tipos: varchar2 e date

Package - Loja

Especificação:

Procedure de inclusão, alterar e excluir

Funções de busca para os tipos: number, varchar2

Funções de verificar existência para os tipos: varchar2

Package - Pedido

Especificação:

Procedure de inclusão, alterar e excluir

Funções de busca para os tipos: number, float e date

Funções de verificar existência para os tipos: number, float e date

Package - Tentativa de colocar CEP

Tentamos inserir automaticamente os dados rua, bairro, cidade e estado colocando apenas o CEP, a partir dos dados do site ViaCep, porém o APEX negou a solicitação.

```
declare
  vr_cliente cliente%ROWTYPE;
  vv_erro   varchar2(200);

begin
  vr_cliente.nome := 'Adam';
  vr_cliente.genero := 'Masculino';
  vr_cliente.telefone := '18996514588';
  vr_cliente.email := 'adamys@gmail.com';
  vr_cliente.datanascimento := '18-04-2001';
  vr_cliente.cpf := '413.243.788-31';
  vr_cliente.cep := '16020-025';

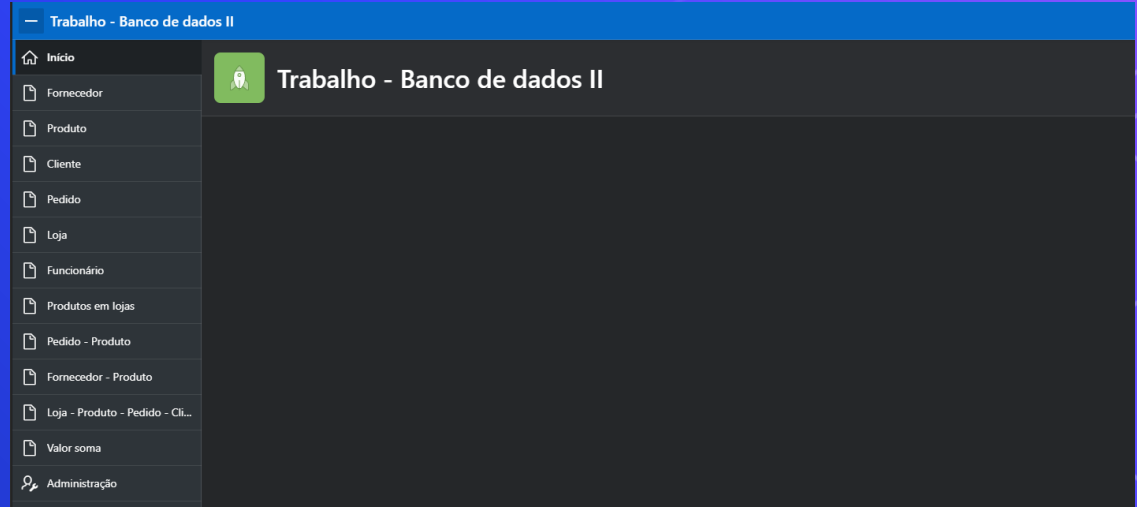
  PKG_CLIENTE.PR_INCLUIR(vr_cliente, vv_erro);
  if vv_erro is not null then
    raise_application_error (-20001, vv_erro);
  end if;
end;
```

```
ORA-29273: falha na solicitação HTTP
ORA-06512: em "SYS.UTL_HTTP", line 1530
ORA-24247: acesso à rede negado pela ACL (access control list)
ORA-06512: em "SYS.UTL_HTTP", line 380
ORA-06512: em "SYS.UTL_HTTP", line 1470
ORA-06512: em "WKSP_ADAMYS.PKG_CLIENTE", line 80
ORA-06512: em "WKSP_ADAMYS.PKG_CLIENTE", line 13
ORA-06512: em line 12
ORA-06512: em "SYS.DBMS_SQL", line 1721
```

```
1. declare
2.   vr_cliente cliente%ROWTYPE;
3.   vv_erro   varchar2(200);
```


Aplicativo

**Criado para mostrar as
selects feitas com uma
melhor interface
gráfica.**



Aplicativo - Fornecedor

Fornecedor

Nome	Telefone	Email	Cidade	Estado
Claudio	(18)991987654	claudinho@gmail.com	Aracatuba	Sao Paulo
Rogério	(18)991998844	rogerin@gmail.com	Aracatuba	Sao Paulo
1 - 2				

Aplicativo - Funcionário

Funcionário

Nome	Telefone	Email	Genero	ID Loja
Adam Yoshida	(18)991778899	adito@gmail.com	MASCULINO	5
Gustavo Shoji	(18)991759876	gusshoji@gmail.com	MASCULINO	3
Matheus Magalhaes	(18)991765489	magal@gmail.com	MASCULINO	23
Gabriela Ribeiro	(18)991223344	gabi@gmail.com	FEMININO	5
Willian Gustavo	(18)991987632	will@gmail.com	MASCULINO	3
1 - 5				

Aplicativo - Produto

Produto

Nome	Cor	Tamanho	Genero	Marca	Preco
Camiseta adidas Core 18	Preto	M	Masculino	Adidas	69,99
Camiseta Nike Legend 2.0	Laranja	M	Masculino	Nike	89,99
Camiseta Nike Dry Academy 21 Top	Azul	P	Masculino	Adidas	99,99
Camiseta Nike Sportswear Just Do It	Cinza Escuro	P	Masculino	Nike	84,99
Nike Dri-Fit 2.0	Preto	G	Masculina	Nike	109,99
Calcao Nike Dri-Fit Park III 21 NB K	Branco	G	Masculino	Nike	79,99
Calção Nike Dri-FIT	Preto	G	Masculino	Nike	179,99
Camiseta Dri-Fit Hoop Photo	Preto	G	Masculino	Nike	89,99
Camiseta Nike Tee Icon Futura	Amarelo	GG	Masculino	Nike	109,99
Meia Nike Cano Alto Everyday	Preto	GG	Masculino	Nike	19,99

1 - 10

Aplicativo - Cliente

Cliente

Nome	Genero	Telefone	Email
Gabriela	Feminino	18991854321	gabi123@gmail.com
Jose	Masculino	18991304050	jose@gmail.com
Carlos	Masculino	18991865321	carlos@gmail.com
Fernanda	Feminino	11991756321	fernanda123@gmail.com

1 - 4

Aplicativo - Pedido

Pedido

Valortotal	Datavenda	ID Cliente
5100	6/12/2021	21
12000	5/12/2021	2
1 - 2		

Aplicativo - Loja

Loja

Nome	Email	Telefone	Cidade
aditas2	aditas@gmail.com	18991807643	Marília
aditas3	aditas@gmail.com	18991874312	Sao Paulo
aditas1	torratorra@gmail.com	18991987654	Campinas
1 - 3			

INSERINDO REGISTROS - Clientes

```
insert into CLIENTE(NOME, GENERO, TELEFONE, EMAIL,CPF,DATANASCIMENTO,RUA,  
NUMERO,CEP,BAIRRO,CIDADE,ESTADO) values  
( 'Jose','Masculino','18991304050','jose@gmail.com','268.916.340-32' , '21-09-1990','Rua Benedito  
Faria Marques Filho',52,'08683-000','Parque Maria Helena','Suzano', 'Sao Paulo');
```


REPLACE - Fornecedor

SELECT

ID,
NOME,
TELEFONE,
EMAIL,
CNPJ,
RUA,
NUMERO,
CEP,
BAIRRO,
CIDADE,
ESTADO,

REPLACE(

REPLACE(

REPLACE(estado, 'Sao', 'S')
,',')
, 'Paulo', 'P')

as formatado

FROM FORNECEDOR



Substituindo os registros
“Sao Paulo” na coluna de
estado por “SP”

ID	NOME	TELEFONE	EMAIL	CNPJ	RUA	NUMERO	CEP	BAIRRO	CIDADE	ESTADO	FORMATADO
21	Claudio	(18)991987654	claudinho@gmail.com	38.222.506/0001-18	Franciscano de almeida	52	78735-741	Vila Estadio	Aracatuba	Sao Paulo	SP
25	Rogério	(18)991998844	rogerin@gmail.com	54.176.269/0001-52	Getulio Vargas	51	88708-309	Jardim Brasilia	Aracatuba	Sao Paulo	SP

2 linhas retornadas em 0,01 segundos [Fazer Download](#)

CONSULTA SQL- Pedido

```
SELECT sum(VAORTOTAL) as total_recebido FROM PEDIDO  
WHERE DATAVENDA between '01-11-2021' and '06-12-2021'
```



Consultando o lucro bruto de todas as lojas cadastradas em um período de tempo

TOTAL_RECEBIDO	
17100	
1 linhas retornadas em 0,01 segundos Fazer Download	

```
SELECT * FROM PEDIDO  
WHERE DATAVENDA between '02-10-2021' and '06-12-2021'  
ORDER BY id
```



Consultando os pedidos feitos em um período de tempo

ID	VALORTOTAL	DATAVENDA	ID_CLIENTE
1	5100	06/12/21	21
21	12000	05/12/21	2
2 linhas retornadas em 0,01 segundos Fazer Download			

CONSULTA SQL- Funcionário

```
SELECT * FROM PEDIDO WHERE VALORTOTAL >= 100
```



Pedidos onde foram gasto mais de R\$ 100,00

ID	VALORTOTAL	DATAVENDA	ID_CLIENTE
1	5100	06/12/21	21
21	12000	05/12/21	2

2 linhas retornadas em 0,01 segundos [Fazer Download](#)

```
SELECT * FROM FUNCIONARIO
```

ID	NOME	CPF	GENERO	TELEFONE	EMAIL	SALARIO
21	Adam Yoshida	833.802.840-17	MASCULINO	(18)991778899	adito@gmail.com	2900
1	Gustavo Shoji	004.110.640-70	MASCULINO	(18)991759876	gusschoji@gmail.com	1500
42	Matheus Magalhaes	346.557.440-06	MASCULINO	(18)991765489	magal@gmail.com	2000
41	Gabriela Ribeiro	004.153.640-70	FEMININO	(18)991223344	gabi@gmail.com	3500
23	Willian Gustavo	870.018.000-90	MASCULINO	(18)991987632	will@gmail.com	1200



Funcionarios registrados

CONSULTA SQL- Funcionário

```
SELECT * FROM FUNCIONARIO WHERE SALARIO >= 2000
```

ID	NOME	CPF	GENERO	TELEFONE	EMAIL	SALARIO
21	Adam Yoshida	833.802.840-17	MASCULINO	(18)991778899	adito@gmail.com	2900
42	Matheus Magalhaes	346.557.440-06	MASCULINO	(18)991765489	magal@gmail.com	2000
41	Gabriela Ribeiro	004.153.640-70	FEMININO	(18)991223344	gabi@gmail.com	3500



Funcionários com salários igual ou superior a R\$ 2000,00

UPDATE- Loja

Antes do update

ID	NOME	TELEFONE	EMAIL	RUA	NUMERO	BAIRRO	CIDADE	ESTADO	CEP
5	aditas2	18991807643	aditas@gmail.com	Rua José Capello	210	Osvaldo Fancelli	Marília	Sao Paulo	17512-722
23	aditas3	18991874312	aditas@gmail.com	Avenida Mercedes	120	Alto da Lapa	Sao Paulo	Sao Paulo	05081-060
3	Torra Torra	18991987654	torratorra@gmail.com	Rua Ângelo Bertazzoli	850	Loteamento Arboreto dos Jequitibás	Campinas	Sao Paulo	13105-688
3 linhas retornadas em 0,00 segundos Fazer Download									

UPDATE loja set NOME = 'aditas1' where nome = 'TORRA TORRA'

ID	NOME	TELEFONE	EMAIL	RUA	NUMERO	BAIRRO	CIDADE	ESTADO	CEP
5	aditas2	18991807643	aditas@gmail.com	Rua José Capello	210	Osvaldo Fancelli	Marília	Sao Paulo	17512-722
23	aditas3	18991874312	aditas@gmail.com	Avenida Mercedes	120	Alto da Lapa	Sao Paulo	Sao Paulo	05081-060
3	aditas1	18991987654	torratorra@gmail.com	Rua Ângelo Bertazzoli	850	Loteamento Arboreto dos Jequitibás	Campinas	Sao Paulo	13105-688
3 linhas retornadas em 0,00 segundos Fazer Download									

DELETE- Loja

Antes do delete

ID	NOME	TELEFONE	EMAIL	RUA	NUMERO	BAIRRO	CIDADE	ESTADO	CEP
5	aditas2	18991807643	aditas@gmail.com	Rua José Capello	210	Oswaldo Fancelli	Marília	Sao Paulo	17512-722
23	aditas3	18991874312	aditas@gmail.com	Avenida Mercedes	120	Alto da Lapa	Sao Paulo	Sao Paulo	05081-060
24	aditas4	18991987621	aditas@gmail.com	Rua General Câmara	290	Ipiranga	Ribeirão Preto	Sao Paulo	14055-080
3	aditas1	18991987654	torratorra@gmail.com	Rua Ângelo Bertazzoli	850	Loteamento Arboreto dos Jequitibás	Campinas	Sao Paulo	13105-688

4 linhas retornadas em 0,00 segundos [Fazer Download](#)

DELETE FROM loja WHERE nome = 'aditas4'

ID	NOME	TELEFONE	EMAIL	RUA	NUMERO	BAIRRO	CIDADE	ESTADO	CEP
5	aditas2	18991807643	aditas@gmail.com	Rua José Capello	210	Oswaldo Fancelli	Marília	Sao Paulo	17512-722
23	aditas3	18991874312	aditas@gmail.com	Avenida Mercedes	120	Alto da Lapa	Sao Paulo	Sao Paulo	05081-060
3	aditas1	18991987654	torratorra@gmail.com	Rua Ângelo Bertazzoli	850	Loteamento Arboreto dos Jequitibás	Campinas	Sao Paulo	13105-688

3 linhas retornadas em 0,01 segundos [Fazer Download](#)

Inner Join

```
SELECT loja.nome as loja, produto.nome as produto, produto.genero,  
produto.marca, produto.cor, produto.preco, produto.tecido, produto.quantidade  
FROM produtoloja  
INNER JOIN produto ON produto.id = produtoloja.id_produto  
INNER JOIN loja ON loja.id = produtoloja.id_loja  
WHERE loja.id=3;
```

Genero	Marca	Cor	Preco	Tecido	Loja	Produto
Masculino	Adidas	Preto	69,99	Poliester	Torra Torra	Camiseta adidas Core 18
Masculino	Nike	Laranja	89,99	Poliester	Torra Torra	Camiseta Nike Legend 2.0
Masculino	Nike	Cinza Escuro	84,99	Algodao	Torra Torra	Camiseta Nike Sportswear Just Do It
Masculina	Nike	Preto	109,99	Poliester	Torra Torra	Nike Dri-Fit 2.0
Masculino	Nike	Branco	79,99	Poliester	Torra Torra	Calcao Nike Dri-Fit Park III 21 NB K
Masculino	Nike	Preto	179,99	Algodao	Torra Torra	Calça Nike Dri-FIT

Inner join

```
SELECT fornecedor.nome as fornecedor, produto.id as id_produto, produto.nome as produto,  
produto.quantidade, produto.tecido  
FROM fornecedorproduto  
INNER JOIN fornecedor ON fornecedor.id = fornecedorproduto.id_fornecedor  
INNER JOIN produto ON produto.id = fornecedorproduto.id_produto  
WHERE fornecedor.id=21;
```

Fornecedor	Id Produto	Produto	Quantidade	Tecido
Claudio	1	Nike Dri-Fit 2.0	1500	Poliester
Claudio	2	Calcao Nike Dri-Fit Park III 21 NB K	2000	Poliester
Claudio	3	CalÃsa Nike Dri-FIT	2500	Algodao
Claudio	21	Camiseta Dri-FIT Hoop Photo	500	Algodao
Claudio	23	Camiseta Nike Tee Icon Futura	1200	Algodao

Inner join

```
SELECT pedido.id, pedido.datavenda, produto.nome, produto.marca, produto.cor,  
produto.preco,  
FROM produtopedido  
INNER JOIN produto ON produto.id = produtopedido.id_produto  
INNER JOIN pedido ON pedido.id = produtopedido.id_pedido  
WHERE pedido.id=1;
```

Id	Data venda	Nome	Marca	Cor	Preço
1	6/12/2021	Camiseta adidas Core 18	Adidas	Preto	69,99
1	6/12/2021	Nike Dri-Fit 2.0	Nike	Preto	109,99
1	6/12/2021	Calcao Nike Dri-Fit Park III 21 NB K	Nike	Branco	79,99
1	6/12/2021	Calça Nike Dri-FIT	Nike	Preto	179,99

Inner join

```
SELECT loja.nome as loja, produto.nome as produto, pedido.id, cliente.nome as cliente
FROM loja
INNER JOIN produtoloja ON produtoloja.id_loja = loja.id
INNER JOIN produto ON produto.id = produtoloja.id_produto
INNER JOIN produtopedido ON produtopedido.id_produto = produto.id
INNER JOIN pedido ON pedido.id = produtopedido.id_pedido
INNER JOIN cliente ON cliente.id = pedido.id_cliente
WHERE loja.id=3;
```

Loja	Produto	Id	Cliente
Torra Torra	Nike Dri-Fit 2.0	1	Carlos
Torra Torra	Calcao Nike Dri-Fit Park III 21 NB K	1	Carlos
Torra Torra	CalÃsa Nike Dri-FIT	1	Carlos
Torra Torra	Camiseta adidas Core 18	1	Carlos

Inner join

```
SELECT pedido.id, pedido.datavenda, sum(produto.preco) as valor_total  
FROM produtopedido  
INNER JOIN produto ON produto.id = produtopedido.id_produto  
INNER JOIN pedido ON pedido.id = produtopedido.id_pedido  
WHERE pedido.id=1  
GROUP BY pedido.id, pedido.datavenda;
```

Id	Datavenda	Valor Total
1	6/12/2021	439,96

XML

A	B	C	D	E	F	G	H	I
nome	email	celular	rua	numero	cep	bairro	cidade	estado
Aditas 1	torratorra@gmail.com	18991987654	Rua Ângelo Bertazzoli	850	13105-688	Loteamento Arboreto dos Jequitibás (Sousas)	Campinas	Sao Paulo
Aditas 2	espacoprincesa@gmail.com	18991807643	Rua José Capello	210	17512-722	Osvaldo Fanceli	Marília	Sao Paulo
Aditas 3	casadamoda@gmail.com	18991874312	Avenida Mercedes	120	05081-060	Sao Paulo	Sao Paulo	Sao Paulo
Aditas 4	modaoculta@gmail.com	18991987621	Rua General Câmara	290	14055-080	Ribeirão Preto	Ribeirão Preto	Sao Paulo
Aditas 5	shoxfashionweek@gmail.com	18991987654	Rua Argante Betarello	120	14401-020	Franca	Franca	Sao Paulo

XML

```
= "insert into LOJA (NOME, EMAIL, TELEFONE, RUA, NUMERO, CEP, BAIRRO, CIDADE, ESTADO)  
values ('&A2&', '&B2&', '&C2&', '&D2&', '&E2&', '&F2&', '&G2&', '&H2&', '&I2&')"
```

XML

```
insert into LOJA (NOME, EMAIL, TELEFONE, RUA, NUMERO, CEP, BAIRRO, CIDADE, ESTADO) values ('Aditas 1',  
'torratorra@gmail.com','18991987654','Rua Ângelo Bertazzoli', 850, '13105-688', 'Loteamento Arboreto dos Jequitibás  
(Sousas)','Campinas', 'Sao Paulo')
```

```
insert into LOJA (NOME, EMAIL, TELEFONE, RUA, NUMERO, CEP, BAIRRO, CIDADE, ESTADO) values ('Aditas 2',  
'espacoprincesa@gmail.com','18991807643','Rua José Capello', 210, '17512-722', 'Osvaldo Fanceli','Marília', 'Sao Paulo')
```

```
insert into LOJA (NOME, EMAIL, TELEFONE, RUA, NUMERO, CEP, BAIRRO, CIDADE, ESTADO) values ('Aditas 3',  
'casadamoda@gmail.com','18991874312','Avenida Mercedes', 120, '05081-060', 'Sao Paulo','Sao Paulo', 'Sao Paulo')
```

```
insert into LOJA (NOME, EMAIL, TELEFONE, RUA, NUMERO, CEP, BAIRRO, CIDADE, ESTADO) values ('Aditas 4',  
'modaoculta@gmail.com','18991987621','Rua General Câmara', 290, '14055-080', 'Ribeirão Preto','Ribeirão Preto', 'Sao  
Paulo')
```

```
insert into LOJA (NOME, EMAIL, TELEFONE, RUA, NUMERO, CEP, BAIRRO, CIDADE, ESTADO) values ('Aditas 5',  
'shoxfashionweek@gmail.com','18991987654','Rua Argante Betarello', 120, '14401-020', 'Franca','Franca', 'Sao Paulo')
```

OBIGADO!

Alguma pergunta?

