

WESLEY PRESTES PEREIRA

Estrutura da base de dados:

```
drop database if exists sql_test;
create database sql_test
default character set utf8
default collate utf8_general_ci;

use sql_test;

create table lojas(
  loj_prod int(8) auto_increment,
  desc_loj varchar(40) not null,
  primary key(loj_prod)
)ENGINE=innodb default charset=utf8;

create table produtos(
  cod_prod int(8) auto_increment,
  loj_prod int(8) not null,
  desc_prod Varchar(40) not null,
  dt_inclu_prod date not null,
  preco_prod decimal(8,2) not null,
  primary key(cod_prod),
  foreign key(loj_prod) references lojas(loj_prod)
)ENGINE=innodb default charset=utf8;

create table estoque(
  cod_prod int(8) auto_increment,
  loj_prod int(8) not null,
  qtd_prod int(15) not null,
  primary key (cod_prod),
  foreign key(loj_prod) references lojas(loj_prod)
)ENGINE=innodb default charset=utf8;
```

Inserção de dados, mas tabelas **Lojas**, **Produtos** e **Estoque**:

```
insert into lojas (desc_loj) values ('Loja america');
insert into lojas (desc_loj) values ('Super mercado');
insert into lojas (desc_loj) values ('Hiper mercado');
insert into lojas (desc_loj) values ('varejo');
insert into produtos (cod_prod, loj_prod, desc_prod, dt_inclu_prod, preco_prod) values
('190', '1', 'INSETICIDA', '2010-03-30', '45.30');
insert into produtos (cod_prod, loj_prod, desc_prod, dt_inclu_prod, preco_prod) values
('170', '2', 'LEITE CONDESADO MOCOCA', '2010-12-30', '45.40');
insert into produtos (cod_prod, loj_prod, desc_prod, dt_inclu_prod, preco_prod) values
('180', '3', 'LEITE SEMIDESNATADO', '2015-12-30', '160.00');
insert into produtos (cod_prod, loj_prod, desc_prod, dt_inclu_prod, preco_prod) values
('160', '4', 'PILHA AAA', '2020-12-30', '258.00');
insert into produtos (cod_prod, loj_prod, desc_prod, dt_inclu_prod, preco_prod) values
('200', '1', 'CAFE', '2015-12-20', '160.00');
insert into estoque (cod_prod, loj_prod, qtd_prod) values ('160', '1', '150');
insert into estoque (cod_prod, loj_prod, qtd_prod) values ('170', '2', '12');
insert into estoque (cod_prod, loj_prod, qtd_prod) values ('180', '3', '48');
insert into estoque (cod_prod, loj_prod, qtd_prod) values ('190', '4', '73');
insert into estoque (cod_prod, loj_prod, qtd_prod) values ('110', '4', '70');
```

Teste de SQL

Considere a seguinte tabela:

Tabela de produtos

Campo	Tipo de Campo	Chave
cod_prod	Integer (8)	X
loj_prod	Integer (8)	X
desc_prod	Char (40)	
dt_inclu_prod	Data (dd/mm/yyyy)	
preco_prod	decimal (8,3)	

Com base na tabela de “produtos” acima favor inserir um registro na referida tabela passando os seguintes valores : cod_prod=170, loj_prod=2, desc_prod=LEITE CONDESADO MOCOCA, dt_inclu_prod=30/12/2010 e preço_prod = R\$45,40.

O Índice da tabela de “produtos” é o cód_prod e a loj_prod, com base no referido índice faça a alteração do preço do produto para R\$95,40, lembrando que o cod_prod =170 e a loj_prod=2:

R.: Update produtos set preco_prod= '95.40' where cod_prod=170 and loj_prod=2;

Com base na tabela de “produtos” monte um select trazendo todos os registros da loja 1 e 2:

R.: Select * from produtos where loj_prod=1 or loj_prod=2;

Com base na tabela de “produtos” monte um select para trazer a maior e a menor data de inclusão do produto “dt_inclu_prod”:

R.: Select max(dt_inclu_prod) as maior_data, min(dt_inclu_prod) as menor_data from produtos;

Com base na tabela de “produtos” monte um select para trazer a quantidade total de registros existentes na tabela de “produtos”:

R.: Select count(cod_prod) as Total_registros from produtos;

Com base na tabela de “produtos” monte um select para trazer todos os produtos que comecem com a letra “L” na tabela de “produtos”:

R.: Select * from produtos where desc_prod like 'L%';

Com base na tabela de “produtos” monte um select para trazer a soma de todos os preços dos produtos totalizado por loja:

R.: Select loj_prod as codigo_loja, sum(preco_prod) as soma_precos from produtos group by loj_prod order by loj_prod asc;

Com base na tabela de “produtos” monte um select para trazer a soma de todos os preços dos produtos totalizados por loja que seja maior que R\$100.000

R.: Select loj_prod as codigo_loja, sum(preco_prod) as soma_precos from produtos group by loj_prod having sum(preco_prod)>100.00 order by loj_prod asc;

Observe as Tabelas Abaixo:

Tabela de Produtos

Campo	Tipo de Campo	Chave	Comentário
Cód_prod	Integer (8)	X	Código do Produto
loj_prod	Integer (8)	X	Código da Loja
desc_prod	Char (40)		Descrição do Produto
Dt_inclu_prod	Data (dd/mm/yyyy)		Data de Inclusão do Produto
preco_prod	decimal (8,3)		Preço do Produto

Tabela de Estoque

Campo	Tipo de Campo	Chave	Comentário
Cód_prod	Integer (8)	X	Código do Produto
loj_prod	Integer (8)	X	Código da Loja
qtd_prod	decimal(15,3)		Quantidade em Estoque do Produto

Tabela de Lojas

Campo	Tipo de Campo	Chave	Comentário
loj_prod	Integer (8)	X	Código da Loja
desc_loj	Char (40)		Descrição da Loja

A)Montar um unico select para trazer os seguintes campos: o código da loja do produto, a descrição da loja, código do produto, a descrição do produto, o preço do produto, a quantidade em estoque do produto. Considere que o código da loja para esta consulta seja igual a 1.

R.:

select produtos.loj_prod as cod_loja_produtos, lojas.desc_loj as descrição_loja, produtos.cod_prod as cod_produto_produtos, produtos.desc_prod as descrição_produto, produtos.preco_prod as preco_produto, estoque.qtd_prod as quantidade_produtos_estoque from ((lojas inner join produtos on lojas.loj_prod = produtos.loj_prod) inner join estoque on lojas.loj_prod=estoque.loj_prod) where produtos.loj_prod=1;

B)Observe a estrutura da tabela de estoque e da tabela de produtos, monte um select para trazer todos os produtos que existem na tabela de produtos que não existem na tabela de estoque.

R.: select * from produtos left join estoque on produtos.cod_prod = estoque.cod_prod where estoque.cod_prod is null;

C)Observe a estrutura da tabela de estoque e da tabela de produtos, monte um select para trazer todos os produtos que existem na tabela de estoque que não existem na tabela de produtos.

R.: select * from produtos right join estoque on produtos.cod_prod = estoque.cod_prod where produtos.cod_prod is null;