**Disciplina de Banco de Dados**

**Prof. @cassiocosta\_**

# Trabalho AP2 BD em Grupo máximo 3 integrantes.

Criar uma modelagem baseada em uma regra de negócio real.

Podemos pegar emprestado o trabalho de Engenharia de software, modelagem de software ou Laboratório de Simulação e Prototipação.

Requisitos:

* Fazer modelagem ER no WorkBench ou alguma ferramenta ao seu gosto
* Mínimo:
  + ~~6 tabelas~~
  + ~~3 Views~~
  + ~~4 Stored Procedures~~
  + ~~3 Triggers~~

Executar a modelagem em um SGBD de sua escolha (sugestão mysql ou postgres)

Mostrar os scripts:

Modelagem, views, Procedures e Triggers

Os integrantes devem explicar a modelagem, restrições de integridade e mostrar o funcionamento das views, procedures e triggers.

Toda a documentação deverá ser postada.

# Grupo

**Grupo Ecommerce:**

* Diogo Vinícius
* Luan Knevitz
* Luiz Carlos

**Regras de negócio:**

**Usuários(users)** cadastrados nunca serão deletados, caso um usuário deseje encerrar sua conta o campo user\_active tem que ser setado para null à fim de manter toda a base de dados consistente e manter os dados do usuário cadastrados caso ele deseje voltar a utilizar os serviços.

* ~~Criar procedure pra insert de usuário com todos os campos menos id\_user que é auto\_increment e user\_active e que deve ser setado como true já que o usuário está se cadastrando, além disso como é seu primeiro cadastro é preciso inserir junto um endereço passando o campo address e cep.~~
* ~~Criar View que mostra usuarios e seus endereços.~~

**Endereços(Address)** Cada usuario poderá ter mais de um endereço cadastrado, então é necessário uma tabela pra isso, cada endereço pertencerá a um usuário e irá referenciar a id dele para validar quais endereços são de quais usuários.

* ~~Criar uma procedure para inserir esses dados na tabela, caso o usuário queira criar novos endereços para sí mesmo.~~

**Compras(purchases)** essa tabela é responsável por criar compras dos usuários, como cada usuário, como a relação entre item e compra é muitos pra muitos teremos uma tabela auxiliar que garantirá que nossa base de dados seja coesa e precisa em relação às compras do cliente.

* ~~Criar procedure pra facilitar a inserção de compras, inserir na tabela de shippings um novo envio e atualizar a tabela auxiliar com os dados necessários, essa procedure precisa poder adicionar compras a uma compra já existente em items\_comprados.~~
* ~~Criar view que liste as compras com seus itens e o usuário.~~
* ~~Criar view que liste as compras com seus itens, usuário e endereço de entrega.~~

**Items(items) i** items inseridos precisam ter todos os campos setados e o campo in\_stock precisa ser setado como true caso a quantidade (stock) seja maior do que 0 e sempre que um item for vendido é preciso que seja reduzido do campo stock a quantidade que foi vendida, o inverso também é válido no caso de uma compra ser deletada.

* ~~Criar procedures pra inserir items passando todos os campos menos in\_stock que deve ser setado automaticamente na procedure.~~

**Entregas(shippings)** essa tabela é responsável por criar uma tupla que dirá quando e onde serão entregues os produtos comprados, a data de entrega será setada X dias depois do dia em que a compra foi feita usando como base o campo purchased\_date e o endereço será setado por parâmetro passando o id do endereço, pode haver validação caso faça sentido.

Criar trigger para fazer essas inserções e automatizar o preenchimento desses campos.

**Items Comprados(purchased\_items)** É a tabela auxiliar que fará com que seja possivel validar os items que foram vendidos em qual compra pois ele referenciará tanto a id da compra quanto a id do item da compra, isso será feito com uma trigger em que sempre que houver um **INSERT** em compras, automaticamente seja feito um insert em purchased\_items com os mesmos dados que foram passados nesse insert em purchased\_items e setar o campo total\_price como a quantity \* price que está em item.

* ~~Para facilitar isso provavelmente vale a pena criar uma view de purchases e items~~
* ~~Também criar a trigger que fará a logica de setar esses dados.~~
* ~~Criar triggers que escutem em compras quando houver um~~ **~~INSERT, DELETE~~** ~~ou~~ **~~UPDATE~~** ~~na tabela de compras\_items e fazer a ação necessária e atualizem o estoque de item de acordo com a ação.~~