
una
O MELHOR
CENTRO UNIVERSITÁRIO
PRIVADO DE BH
Fonte: MEC

una

O MELHOR
CENTRO UNIVERSITÁRIO
PRIVADO DE BH
Fonte: MEC

MySQL®

Professor Leonardo Larback

Views

Uma **view** é um recurso para a criação de consultas customizadas. É similar a uma entidade normal, porém é derivada de uma ou mais entidades e seus dados não são fisicamente armazenados.

A principal vantagem na utilização de **views** é deixar consultas previamente definidas para a manipulação dos dados, retirando dos projetistas e desenvolvedores a tarefa de construir consultas.

Views

Para criar uma view utilizamos o comando CREATE VIEW <identificador_da_view> AS SQL_ORIGEM:

- *create view v_fornecedores_mg as select nome, cnpj from fornecedores where uf='MG';*
- *create view v_produtos_sem_estoque as select * from produtos where estoque=0;*

Para executarmos uma view basta selecioná-la de maneira idêntica ao que faríamos com uma entidade:

*select * from v_produtos;*

As views serão exibidas juntamente com as tabelas ao executarmos o comando show tables;

Triggers

Um **Trigger** é um gatilho acionado automaticamente com base em uma premissa. Ele é executado antes ou depois de um evento na tabela.

O **Trigger** foi incorporado ao MYSQL na versão 5.0.2 em resposta aos “concorrentes” Oracle e Firebird que já utilizavam o recurso.

Triggers

As opções de evento para realização de um Trigger são:

- Before insert
- Before update
- Before delete
- After delete
- After insert
- After update

Triggers

Para exemplificar, criemos um trigger que salve os estados excluídos da tabela **uf**

Field	Type	Null	Key	Default	Extra
sigla	char(2)	NO	PRI	NULL	
nome	varchar(200)	YES		NULL	

na tabela **uf_excluido**

Field	Type	Null	Key	Default	Extra
sigla	char(2)	YES		NULL	
descricao	varchar(200)	YES		NULL	
data_exclusao	timestamp	NO		CURRENT_TIMESTAMP	on update CURRENT_TIMESTAMP

Triggers

1. delimiter ?>
2. create trigger ad_salva_uf
3. after delete on uf for each row
4. begin
5. insert into uf_excluido values (OLD.sigla, OLD.nome, now());
6. end;
7. ?>

Na linha 1, mudamos o delimitador de comandos, assim o mysql vai esperar até encontrar ?> para executar os comandos (o padrão é ponto-e-vírgula e não podemos usar pois dentro da trigger existirão pontos-e-vírgulas)

Triggers

Na linha 2, iniciamos a criação da trigger especificando seu nome.

Na linha 3, identificamos o evento que irá disparar o gatilho bem como a tabela onde este evento será realizado.

A linha 4 mara o inicio do corpo da trigger.

A linha 5 é o código da trigger propriamente dito, nela é feita a inclusão do registro na tabela auxiliar. O parâmetro OLD nesse caso refere-se ao dado que foi excluído da tabela. Além do OLD, existe também a variável NEW. O OLD representa um registro que existe na tabela antes de ser alterado ou excluído. O NEW representa um registro que está sendo inserido ou atualizado.

A linha 6 finaliza o corpo da trigger

Na linha 7 introduzimos o delimitador para que nossa sentença seja executada pelo mysql.