

## IFSC – EMI – IOT Web – StoredProcedures

Como foi visto até agora, os gatilhos (triggers) são pequenos códigos de programação em banco de dados que tem por objetivo realizar pequenas operações matemáticas em atributos de tabelas pai, tabelas que fornecem *códigos* para as tabelas dependentes, tabelas filho.

Os gatilhos são disparado através dos eventos *After* ou *Before*. Nos exemplos vistos até agora, somente os eventos *After* foram utilizados, porém em muitas situações é preciso realizar algum tipo de validação **antes de incluir** um registro em uma tabela. Nesta situação usa-se o gatilho no evento *Before*(antes) de inserir o registro.

A validação de algum atributo, deve ser programada na *StoredProcedure*, que nada mais é do que um pequeno código de programação que realiza esta validação.

A *StoredProcedure* é chamada através da função *CALL(nome\_da\_procedure,parametro1,parametro2,...)*. Esta função é inserida no Gatilho(trigger)da tabela em que será realizada a operação de inclusão no evento **Before**.


Vamos elaborar uma simulação de venda de produtos com a regra de negócio: *o valor de venda não pode ser menor que o valor de custo*.

**Obs.:** Mensagens de erro no MySQL. (<https://dev.mysql.com/doc/refman/8.0/en/signal.html>)

<https://dev.mysql.com/doc/refman/8.4/en/signal.html>

◆ Visão geral criada por IA

🔊 Ouvir

Em bancos de dados SQL, `SIGNAL SQLSTATE` é uma instrução usada para disparar um erro e retornar informações sobre ele, incluindo um código `SQLSTATE` e uma mensagem de erro. A instrução permite que os desenvolvedores controlem o comportamento da aplicação quando erros ocorrem, permitindo que eles sejam tratados de forma específica ou, se não houver tratamento, que a aplicação seja interrompida. 

### Validação de placa MAC para liberar/bloquear leituras no banco de dados

A procedure `validamacmiguel` recebe a variável `$idmac` para verificar no banco de dados se a placa está ativa(*Boolean*) para gravar dados através do código abaixo "BEGIN...END..." o valor será atribuído na variável de saída `$ativo`.

Detalhes

Nome das rotinas

Tipo **PROCEDURE**

	Direção	Nome	Tipo	Tamanho/Valores
Parâmetros	IN	<input type="text" value="\$idmac"/>	VARCHAR	17 <input type="button" value="✎"/>
	OUT	<input type="text" value="\$ativo"/>	TINYINT	<input type="button" value="✎"/>

```
BEGIN
select ativo into $ativo from mac
WHERE idmac=$idmac;
END
```

Após realizar a programação da Procedure, deve-se criar um **gatilho** na tabela Leitura no evento **Before – Insert**, e neste gatilho, chamar a procedure através do comando CALL validamacmiguel(New.idmac) para retornar a variável @ativo com o teste de bloqueio de gravação.

Detalhes

Nome do acionador

Tabela

Momento de ação

Evento

Definição

```
begin
CALL
validamacmiguel(new.mac_idmac,@$ativo);
if @$ativo=0
then SIGNAL SQLSTATE '45000' SET
MESSAGE_TEXT = 'Placa inativa';
end if;
end
```