Aula 4

ALTERANDO A ESTRUTURA DA TABELA

**CREATE DATABASE cavaleiros**

**DEFAULT CHARACTER SET utf8mb4**

**DEFAULT COLLATE utf8mb4\_general\_ci;**

**USE cavaleiros;**

**CREATE TABLE bronze(**

**id int NOT NULL AUTO\_INCREMENT,**

**nome varchar(30) NOT NULL,**

**nascimento date,**

**sexo enum('M', 'F'),**

**peso decimal(5,2),**

**altura decimal(3,2),**

**nacionalidade varchar(20) DEFAULT 'Brasil',**

**PRIMARY KEY (id)**

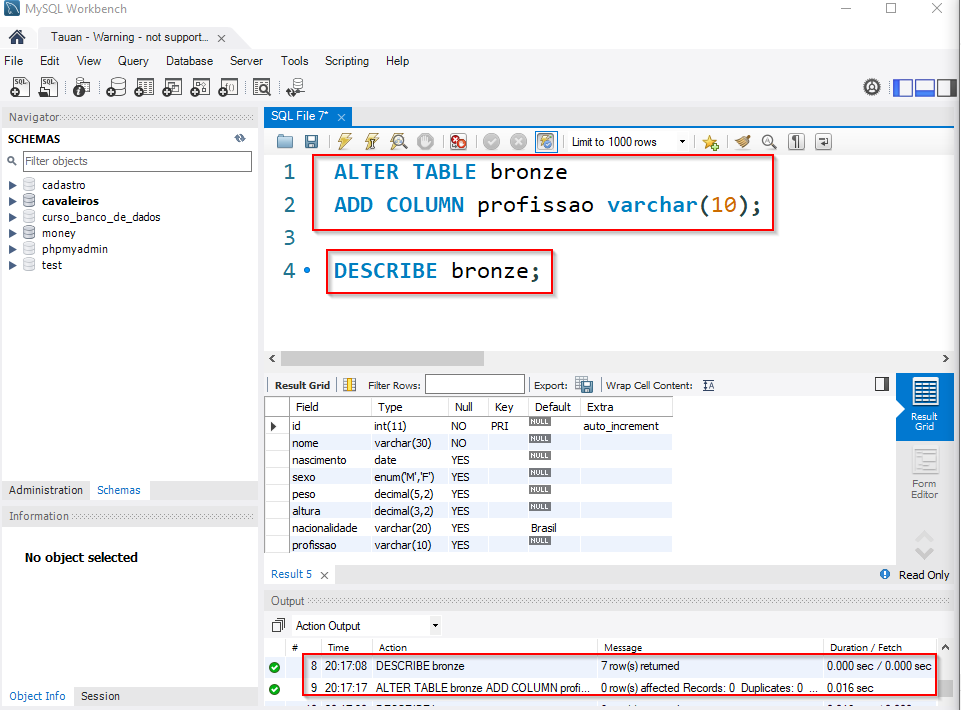
**)DEFAULT CHARSET = UTF8mb4;**

1. ADICIONANDO UMA NOVA COLUNA:

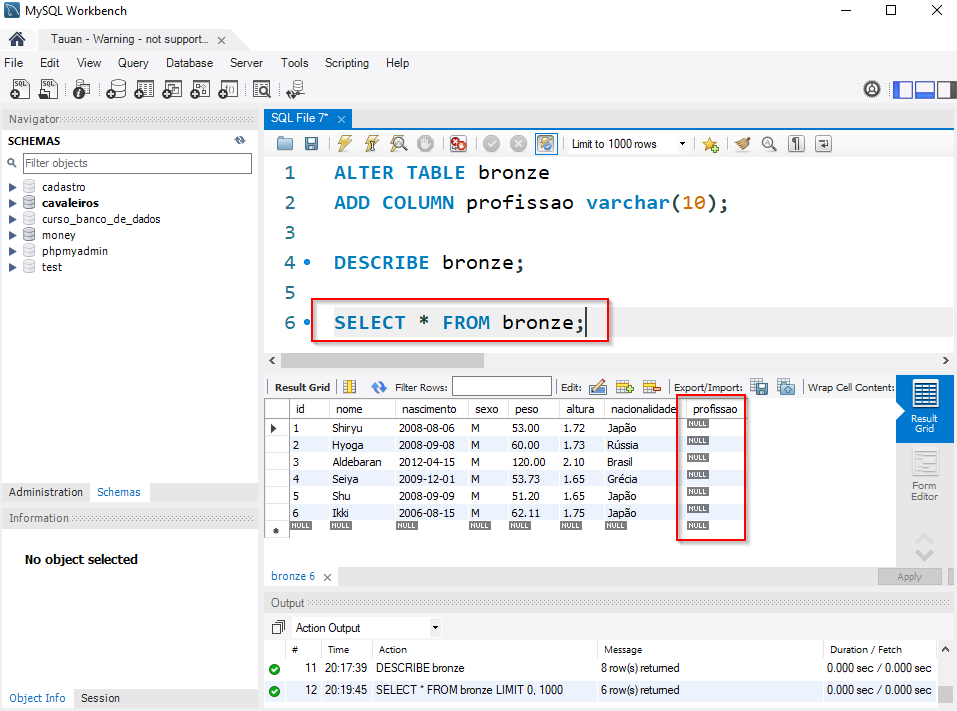
***COLUNAS = CAMPOS***

**ALTER TABLE bronze**

**ADD COLUMN profissao varchar (10);**



* Olhar o que aconteceu com as pessoas que já foram cadastradas.

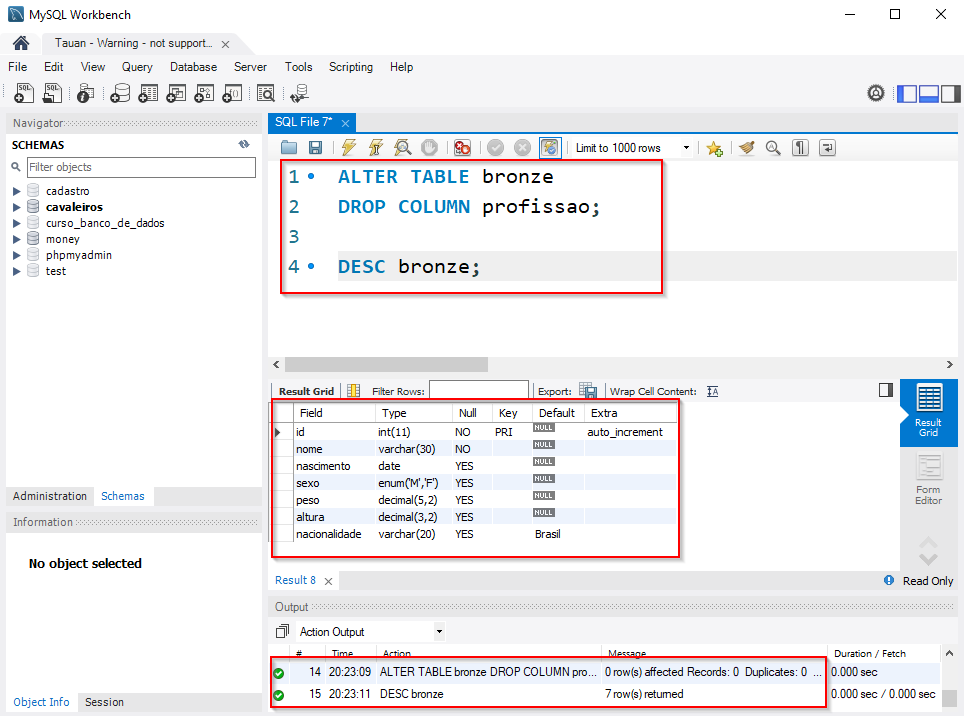


* Ao adicionar uma nova coluna, ela automaticamente vai para o final dos campos. Ou seja, como última coluna.
* Como colocar ela em outra coluna? Bem, antes é preciso eliminar ela.

1. REMOVENDO UMA COLUNA:

**ALTER TABLE bronze**

**DROP COLUMN profissao;**

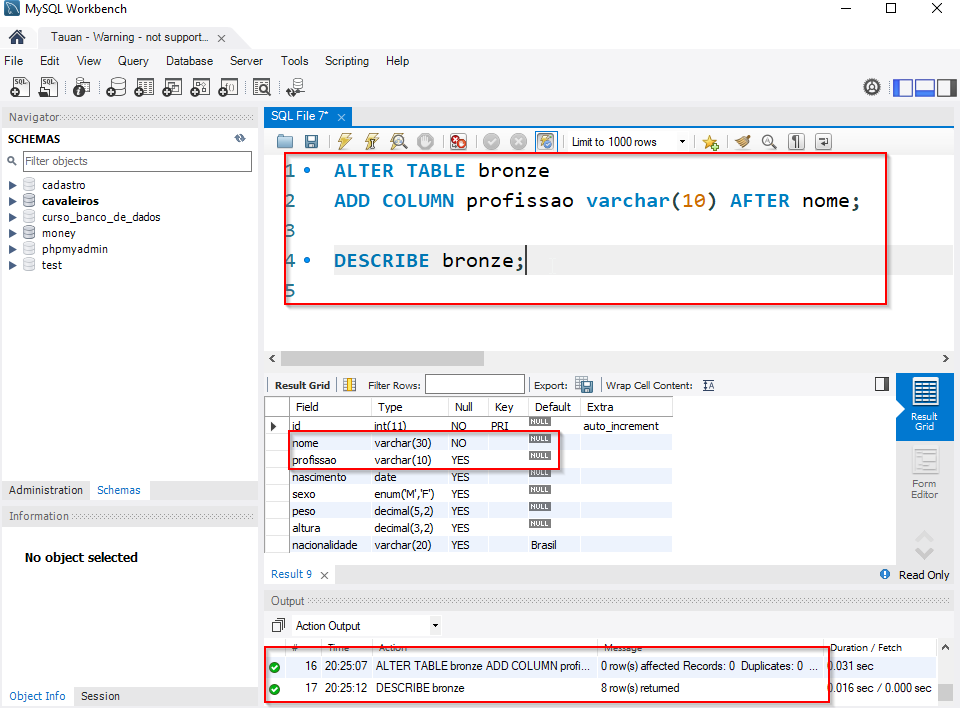


1. ESCOLHENDO A POSIÇÃO DA COLUNA:

**ALTER TABLE bronze**

**ADD COLUMN profissao varchar(10) AFTER nome;**

# ADD COLUMN profissao varchar(10) NOT NULL DEFAULT '' AFTER nome; #

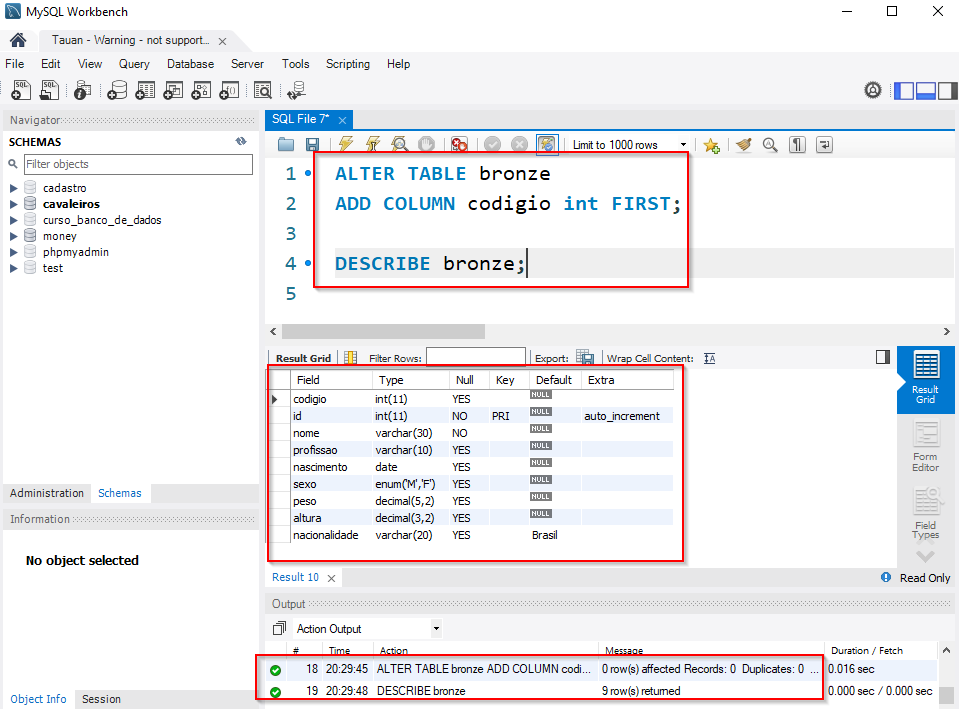


* NÃO existe um commando “BEFORE”. Então como adicionar antes do primeiro campo?

1. ADICIONANDO UMA COLUNA NA PRIMEIRA POSIÇÃO:

**ALTER TABLE bronze**

**ADD COLUMN codigo int FIRST;**



Ou seja:

* Para o PRIMEIRO posicionamento, coloque FIRST!;
* Para qualquer outro posicionamento, coloque AFTER <campo>;
* Se não colocar nada, o campo será automaticamente na última posição.

**[Exercício 1]** Escreva os commandos para adicionar o campo “Codigo” em diferentes posições na tabela. Ou seja, no começo, em algum lugar no meio da tabela e depois no final. [Mostre para o professor]

1. MODIFICANDO DEFINIÇÕES

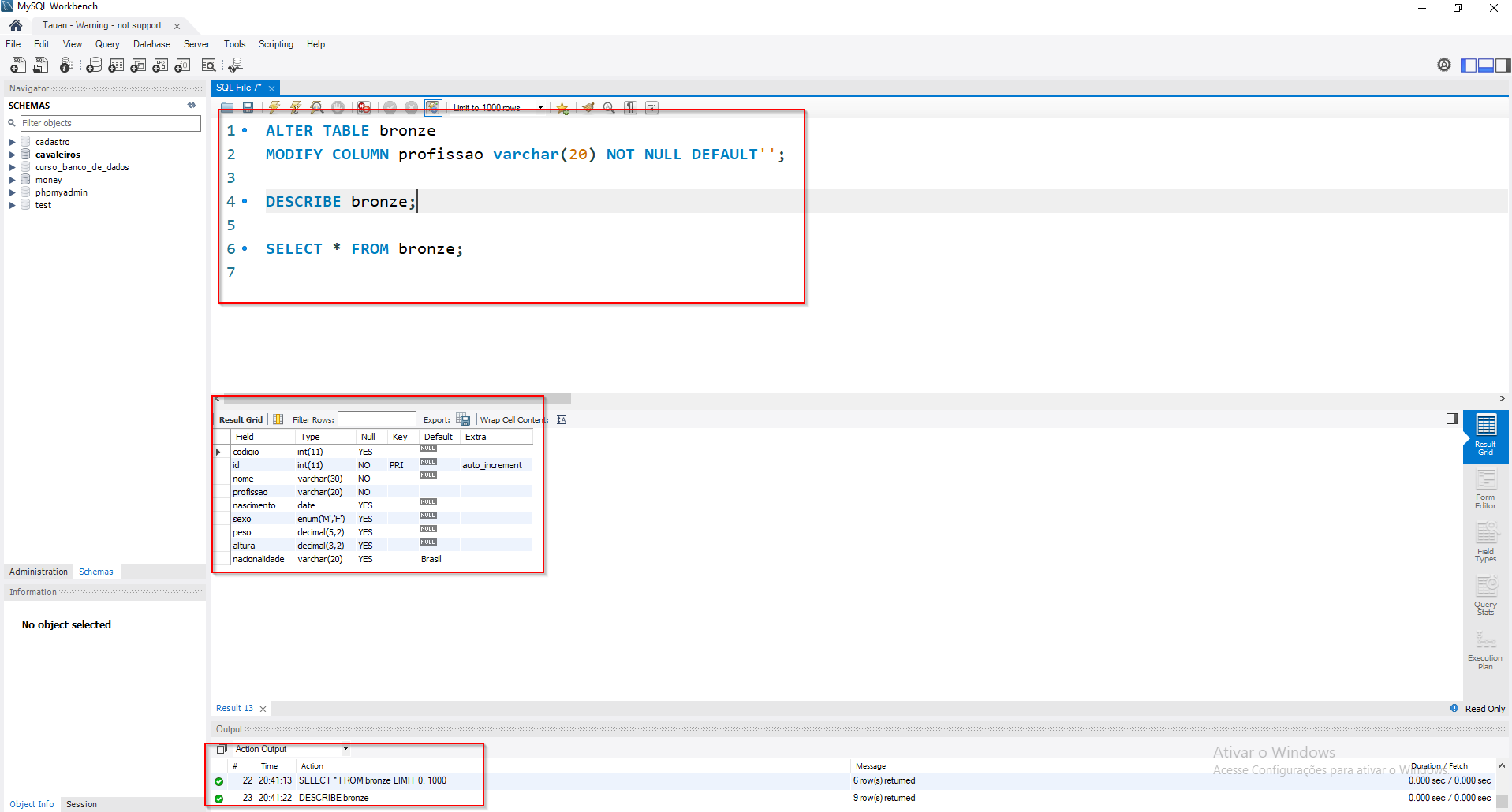
**ALTER TABLE bronze**

**MODIFY COLUMN profissao varchar(20) NOT NULL DEFAULT'';**

# MODIFY COLUMN profissao varchar(20) NOT NULL DEFAULT'';

O comando MODIFY:

* Permite alterar o tipo primitivo do campo (mudar de INT para CHAR, por exemplo);
* Permite alterar todas as contraints (mudar de NULL para NOT NULL, por exemplo);
* Não permite renomear os campos.



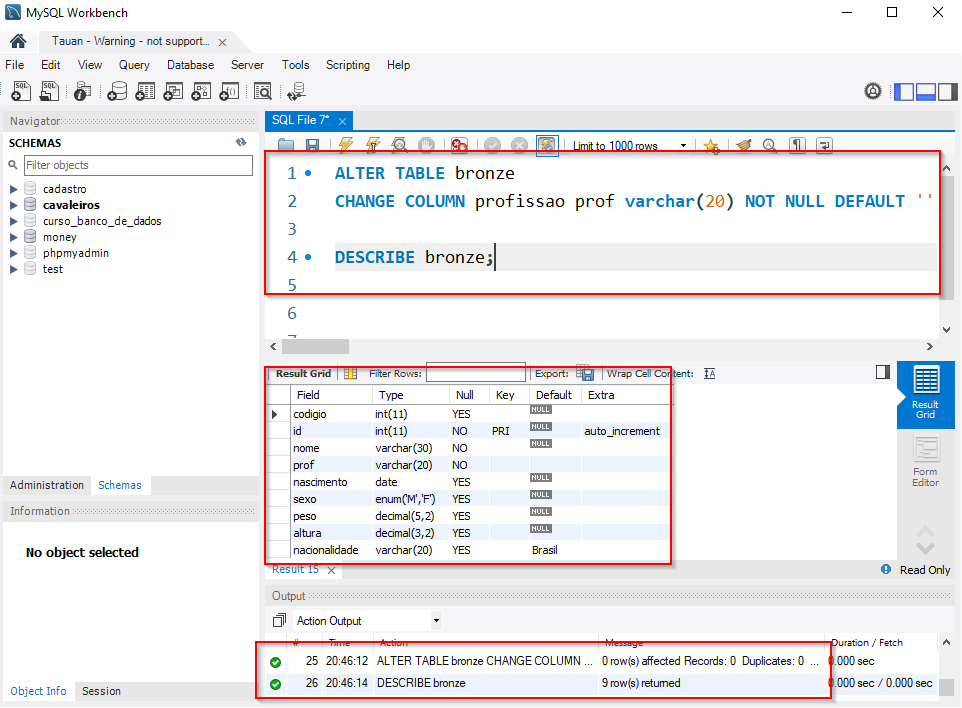
1. RENOMEANDO COLUNA

**ALTER TABLE pessoas**

**CHANGE COLUMN profissao prof varchar(20) NOT NULL DEFAULT '';**

O comando CHANGE:

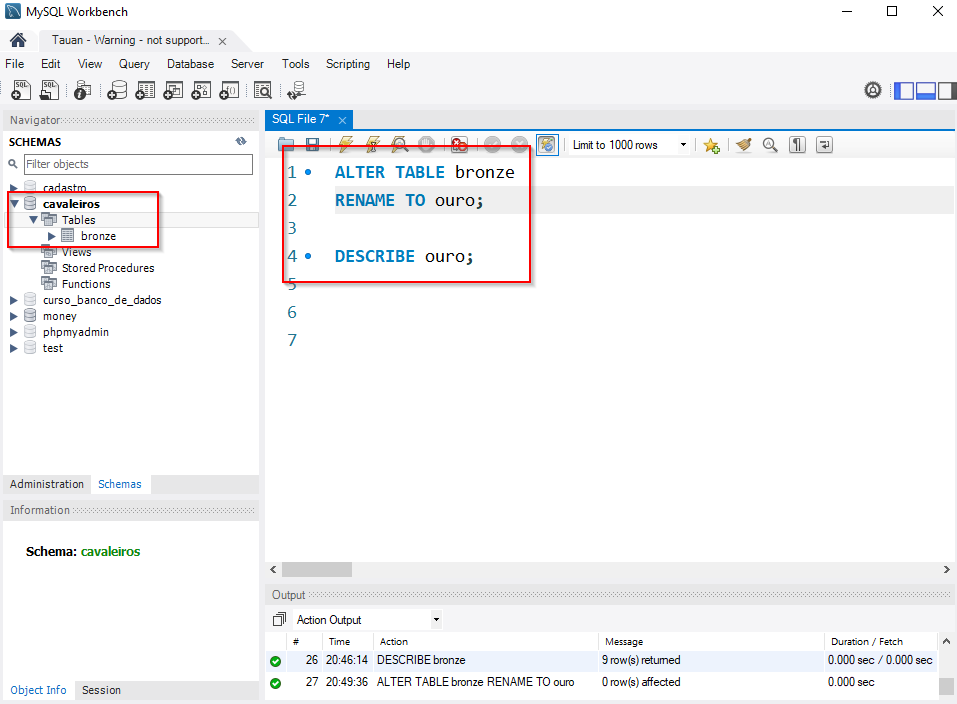
* Permite alterar o tipo primitivo do campo (mudar de INT para CHAR, por exemplo);
* Permite alterar todas as contraints (mudar de NULL para NOT NULL, por exemplo);
* Permite renomear os campos.

****

1. RENOMEANDO TABELA:

**ALTER TABLE bronze**

**RENAME TO ouro;**



MODIFY e CHANGE <- COLUMNS

RENAME TO <- TABLE

1. CRIANDO MAIS UMA TABELA:

**CREATE TABLE IF NOT EXISTS cursos (**

**nome varchar(30) NOT NULL UNIQUE,**

**descricao text,**

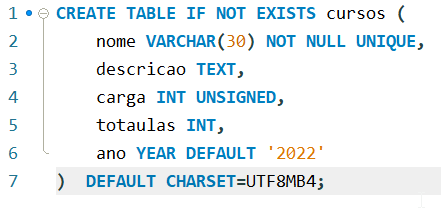
**carga int UNSIGNED,**

**totaulas int,**

**ano year DEFAULT '2022'**

**) DEFAULT CHARSET = utf8mb4;**

* **UNIQUE** = Indica que o valor daquele campo será único e não poderemos cadastrar outro campo com aquele mesmo valor;
* **PRIMARY KEY**, além de também ser UNIQUE, também exerce outras funções;

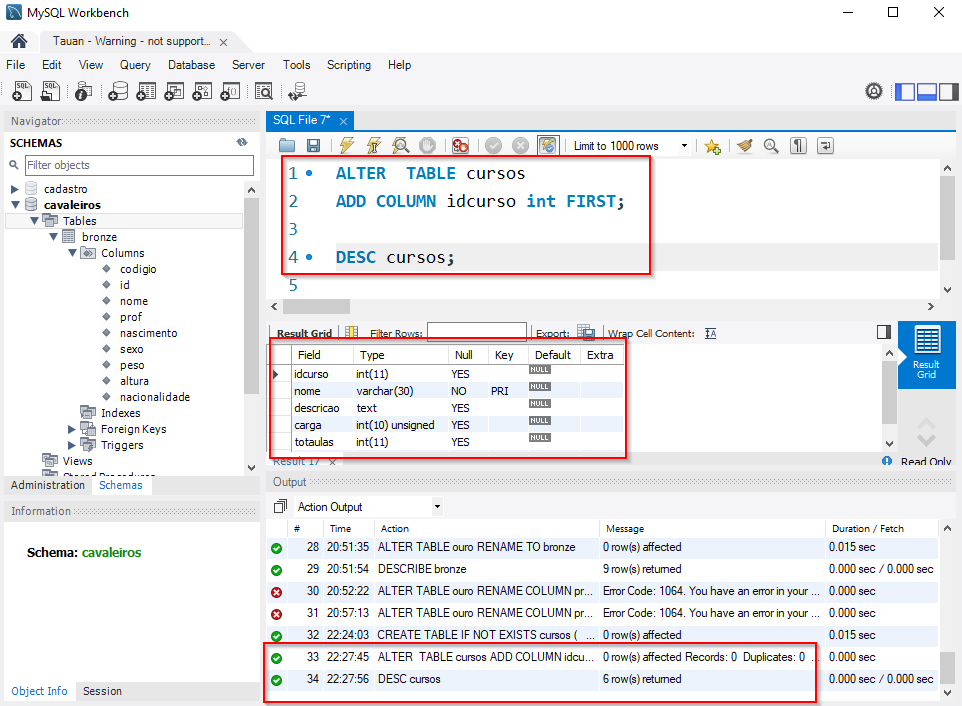
****

1. ADICIONANDO A CHAVE PRIMÁRIA:

9.1) Adiciona uma nova coluna:

**ALTER TABLE cursos**

**ADD COLUMN idcurso int FIRST;**

****

9.2) Indica que essa nova coluna é a chave primária:

**ALTER TABLE cursos**

**ADD PRIMARY KEY(idcurso);**

