



# LAB\_02

## 2.1 → Criando DATABASE

1. Para que serve o comando SQL: **USE**?

Serve para selecionar uma Base de Dados do SGBD

2. Como devemos usar o comando **DROP** para eliminar totalmente o database LAB\_02 criado?

```
DROP DATABASE LAB_02;
```

3. Recrie o LAB\_02

```
CREATE DATABASE LAB_02;  
  
USE LAB_02;
```

## 2.2 → Criando TABELAS

```
CREATE TABLE Disciplina  
(  
  id_discip int    NOT NULL,  
  nome varchar(50) NOT NULL,  
  ementa text,  
  creditos int     NOT NULL,  
  periodo int      NOT NULL  
);  
  
ALTER TABLE disciplina  
ADD CONSTRAINT PRIMARY KEY (ID_discip);
```

1. O que significa quando não indicamos que um campo (atributo) é **NOT NULL**?

Significa que é um campo “não obrigatório”, que pode conter valor nulo.

### 2. Para que serve o comando **ALTER TABLE... ADD CONSTRAINT**?

Serve para mudar uma tabela e adicionar uma restrição a ela (PRIMARY KEY, FOREIGN KEY...).

### 3. O que significa a restrição PRIMARY KEY? Para que ela serve?

É uma chave primária. Ela é única, fazendo assim um papel de identificação dos registros na tabela.

---

## 2.3 Povoando TABELAS

```
INSERT INTO disciplina
VALUES (1, 'Banco de Dados', NULL, 4, 2);

INSERT INTO disciplina
VALUES (1, 'Redes', 'Básico de redes de computadores', 4, 3);
```

### 1. Dos comandos passados, qual não funcionou e como foi arrumado?

O segundo comando não deu certo, já que a chave primária foi repetida. Para arrumá-lo, basta trocar o primeiro valor “1” por outro número inteiro

### 2. Qual o comando para visualizar as inserções para ver se elas estão corretas?

```
SELECT * FROM disciplina;
```

### 3. Crie e execute um comando para inserir mais 5 registros/linhas na tabela disciplina

```
INSERT INTO disciplina
VALUES (3, 'Engenharia de Requisitos', NULL, 4, 2);

INSERT INTO disciplina
VALUES (4, 'Programação Web', 'Desenvolvendo sites', 4, 3);

INSERT INTO disciplina
VALUES (5, 'Lógica Matemática', 'Resolvendo problemas de computação com lógica', 4, 3);

INSERT INTO disciplina
VALUES (6, 'Interação Humano Computador', 'Interface x Interação', 4, 3);

INSERT INTO disciplina
VALUES (7, 'Ethical Hacking', 'Introdução ao Ethical Hacking', 4, 4);
```

```
SELECT * FROM disciplina;
```

#### 4. Apresente todos os registros inseridos na tabela disciplina

id_discip	nome	ementa	creditos	periodo
1	Banco de Dados	NULL	4	2
2	Redes	Básico de redes de computadores	4	3
3	Engenharia de Requisitos	NULL	4	2
4	Programação Web	Desenvolvendo sites	4	3
5	Lógica Matemática	Resolvendo problemas de computação com lógica	4	3
6	Interação Humano Computador	Interface x Interação	4	3
7	Ethical Hacking	Introdução ao Ethical Hacking	4	4

## NOVA TABELA

```
CREATE TABLE Professor (  
  id_prof INT AUTO_INCREMENT PRIMARY KEY,  
  nome VARCHAR(50) NOT NULL,  
  dt_nascimento DATE,  
  apelido VARCHAR(50) GENERATED ALWAYS AS (SUBSTRING_INDEX(nome, " ", 1))  
);  
  
SELECT DATE_FORMAT(curdate(), '%d/%m/%y') AS Data;  
  
INSERT INTO Professor (nome, dt_nascimento)  
VALUES ('Maria das Flores', STR_TO_DATE('23/12/1990', '%d/%m/%Y'));  
  
SELECT * FROM Professor;  
  
SELECT nome, dt_nascimento AS 'Data de Nascimento',  
TIMESTAMPDIFF(YEAR, dt_nascimento, CURDATE()) AS Idade  
FROM Professor;
```

#### 1. O que significa **AUTO\_INCREMENT PRIMARY KEY** e pra que ela serve?

É um comando que serve para o SGBD gerar automaticamente os valores de uma coluna, nesse caso, os valores da chave primária **id\_prof**.

#### 2. O que significa **GENERATED ALWAYS AS (SUBSTRING\_INDEX(nome, " ", 1))** e pra que ele serve?

É um comando que gera uma coluna que não armazena os valores no banco de dados, mas sim faz um cálculo com base em alguma expressão. Nesse caso, a expressão é **SUBSTRING\_INDEX(nome, " ", 1)**

### 3. Um atributo derivado é salvo em um disco? Qual sua utilidade?

Não é armazenado no disco. Um atributo derivado pode servir para cálculos em tempo real, resumos, transformação de dados, etc.

## Múltiplos INSERTs

```
INSERT INTO Professor (nome, dt_nascimento)
VALUES ('José da Silva', STR_TO_DATE('20/02/1985', '%d/%m/%Y')),
       ('Paulo Soares', STR_TO_DATE('10/12/1995', '%d/%m/%Y')),
       ('Ana Rita', STR_TO_DATE('20/02/2000', '%d/%m/%Y'));

SELECT nome, dt_nascimento AS 'Data de Nascimento',
TIMESTAMPDIFF(YEAR, dt_nascimento, CURDATE()) AS Idade
FROM Professor;
```

#### 1. O que significa STR\_TO\_DATE('20/02/1985', '%d/%m/%Y') e para que esse comando foi utilizado?

É um comando que transforma uma string de data para o formato de data do Banco de Dados

#### 2. O que significa AS e para que serve?

Traduzindo, a palavra AS significa “como”. Ele serve para renomear colunas ou tabelas temporariamente, sem alterar o nome verdadeiro (apelido).

#### 3. O que o comando TIMESTAMPDIFF(YEAR, dt\_nascimento, CURDATE()) está realizando?

Esse comando serve para calcular a diferença de anos entre duas datas, neste caso, calculando a idade do professor.

## 2.4 → Integridade Referencial

```
CREATE TABLE Turma (
  id_turma int AUTO_INCREMENT PRIMARY KEY,
  ano      int NOT NULL,
  semestre int NOT NULL,
  id_discip int NOT NULL,
  id_prof   int NOT NULL,
  CONSTRAINT CK_Sem CHECK (semestre BETWEEN 1 AND 2),
  CONSTRAINT UN_Ofeta UNIQUE (ano, semestre, id_discip, id_prof),
  CONSTRAINT FK_Prof FOREIGN KEY (id_prof) REFERENCES Professor (id_prof),
  CONSTRAINT FK_Discip FOREIGN KEY (id_discip) REFERENCES Disciplina (id_discip)
);

SELECT * FROM Turma;
```

1. O que é e para que serve:

a. **CHECK** (semestre BETWEEN 1 AND 2)

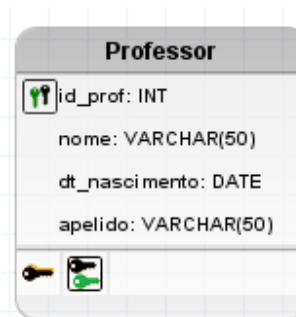
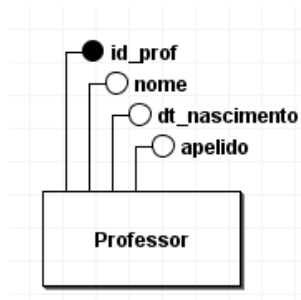
É usada para impor restrições em uma coluna. Nesse caso, garantindo que os valores de “semestre” estejam entre 1 e 2.

b. **UNIQUE** (ano, semestre, id\_discip, id\_prof)

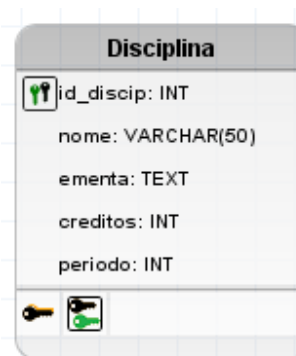
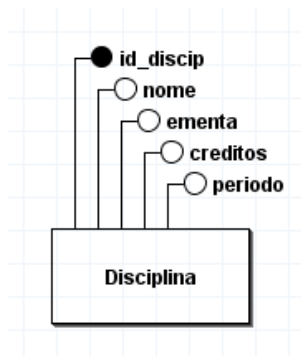
É usada para impor exclusividade nos valores das colunas especificadas.

2. Modelos Conceituais e Lógicos:

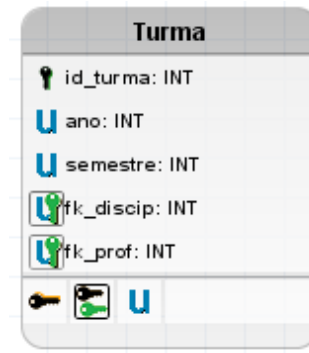
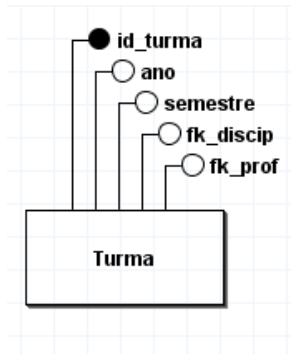
a. **Professor**



b. **Disciplina**



c. **Turma**



## POVOANDO A TABELA

```

-- Primeiro Comando
INSERT INTO Turma (ano, semestre, id_prof, id_discip)
VALUES (2020, 1, 2, 2),
       (2020, 2, 2, 2),
       (2020, 1, 3, 1);

-- Segundo Comando
INSERT INTO Turma (ano, semestre, id_prof, id_discip)
VALUES (2020, 1, 2, 2);

-- Terceiro Comando
INSERT INTO Turma (ano, semestre, id_prof, id_discip)
VALUES (2020, 5, 2, 2);

```

### 1. Quais comandos não funcionaram e como foram arrumados?

**Segundo** e **Terceiro** comando deram erro. O primeiro porque a combinação '2020, 1, 2, 2' (alterar para '2020, 1, 2, 1') já existia, já o segundo porque a coluna "semestre" não estava com um número entre 1 e 2 (mudar "semestre" para 2 e "id\_prof" para 4).

### 2. Apresente os registros das 3 tabelas.

#### Disciplina

id_discip	nome	ementa	creditos	periodo
1	Banco de Dados	NULL	4	2
2	Redes	Básico de redes de computadores	4	3
3	Engenharia de Requisitos	NULL	4	2
4	Programação Web	Desenvolvendo sites	4	3
5	Lógica Matemática	Resolvendo problemas de computação com lógica	4	3
6	Interação Humano Computador	Interface x Interação	4	3
7	Ethical Hacking	Introdução ao Ethical Hacking	4	4

## Professor

id_prof	nome	dt_nascimento	apelido
1	Maria das Flores	1990-12-23	Maria
2	José da Silva	1985-02-20	José
3	Paulo Soares	1995-12-10	Paulo
4	Ana Rita	2000-02-20	Ana
NULL	NULL	NULL	NULL

nome	Data de Nascimento	Idade
Maria das Flores	1990-12-23	32
José da Silva	1985-02-20	38
Paulo Soares	1995-12-10	27
Ana Rita	2000-02-20	23

## Turma

id_turma	ano	semestre	id_discip	id_prof
5	2020	1	1	2
3	2020	1	1	3
1	2020	1	2	2
2	2020	2	2	2
7	2020	2	2	4
NULL	NULL	NULL	NULL	NULL

### 3. Quais professores lecionam quais disciplinas e quando?

#### 1º Semestre:

José da Silva → Banco de Dados → Turma 5

Paulo Soares → Banco de Dados → Turma 3

José da Silva → Redes → Turma 1

#### 2º Semestre:

José da Silva → Redes → Turma 2

Ana Rita → Redes → Turma 7

## BUSCANDO DADOS NA TABELA

```
-- Primeiro Comando
SELECT *
FROM Turma, Professor, Disciplina

-- Segundo Comando
SELECT *
FROM Turma AS t, Professor AS p, Disciplina AS d
WHERE t.id_discip = d.id_discip AND p.id_prof = t.id_prof;

-- Terceiro Comando
SELECT t.ano, t.semestre, p.nome, d.nome
FROM Turma AS t, Professor AS p, Disciplina AS d
WHERE t.id_discip = d.id_discip AND
      p.id_prof = t.id_prof AND
      t.semestre = 1;

-- Quarto Comando
SELECT t.ano, t.semestre, p.nome, d.nome
FROM Turma AS t, Professor AS p, Disciplina AS d
WHERE t.id_discip = d.id_discip AND p.id_prof = t.id_prof
ORDER BY t.ano ASC, t.semestre DESC;

-- Quinto Comando
SELECT p.nome, d.nome, t.ano
FROM Turma AS t, Professor AS p, Disciplina AS d
WHERE t.id_discip = d.id_discip AND
      p.id_prof = t.id_prof AND
      p.nome LIKE 'j%';
```

1. Explique os códigos e mostre os resultado obtidos.

### Primeiro

id_turma	ano	semestre	id_discip	id_prof	id_prof	nome	dt_nascimento	apelido	id_discip	nome	ementa	creditos
5	2020	2	2	4	1	Maria das Flores	1990-12-23	Maria	1	Banco de Dados	NULL	4
5	2020	2	2	4	2	José da Silva	1985-02-20	José	1	Banco de Dados	NULL	4
5	2020	2	2	4	3	Paulo Soares	1995-12-10	Paulo	1	Banco de Dados	NULL	4
5	2020	2	2	4	4	Ana Rita	2000-02-20	Ana	1	Banco de Dados	NULL	4
2	2020	2	2	2	1	Maria das Flores	1990-12-23	Maria	1	Banco de Dados	NULL	4
2	2020	2	2	2	2	José da Silva	1985-02-20	José	1	Banco de Dados	NULL	4
2	2020	2	2	2	3	Paulo Soares	1995-12-10	Paulo	1	Banco de Dados	NULL	4
2	2020	2	2	2	4	Ana Rita	2000-02-20	Ana	1	Banco de Dados	NULL	4
1	2020	1	2	2	1	Maria das Flores	1990-12-23	Maria	1	Banco de Dados	NULL	4
1	2020	1	2	2	2	José da Silva	1985-02-20	José	1	Banco de Dados	NULL	4
1	2020	1	2	2	3	Paulo Soares	1995-12-10	Paulo	1	Banco de Dados	NULL	4
1	2020	1	2	2	4	Ana Rita	2000-02-20	Ana	1	Banco de Dados	NULL	4
3	2020	1	1	3	1	Maria das Flores	1990-12-23	Maria	1	Banco de Dados	NULL	4
3	2020	1	1	3	2	José da Silva	1985-02-20	José	1	Banco de Dados	NULL	4

O comando SELECT serve para mostrar algo em uma tabela, porém escrito desse jeito combina TODAS as linhas de todas as tabelas, gerando um quadro muito grande e repetitivo

### Segundo



id_turma	ano	semestre	id_discip	id_prof	id_prof	nome	dt_nascimento	apelido	id_discip	nome	ementa	creditos	period
4	2020	1	1	2	2	José da Silva	1985-02-20	José	1	Banco de Dados	NULL	4	2
3	2020	1	1	3	3	Paulo Soares	1995-12-10	Paulo	1	Banco de Dados	NULL	4	2
1	2020	1	2	2	2	José da Silva	1985-02-20	José	2	Redes	Básico de redes de computadores	4	3
2	2020	2	2	2	2	José da Silva	1985-02-20	José	2	Redes	Básico de redes de computadores	4	3
5	2020	2	2	4	4	Ana Rita	2000-02-20	Ana	2	Redes	Básico de redes de computadores	4	3

Feito também para mostrar a junção das tabelas, porém escrito do jeito certo, mantendo a organização na tabela

### Terceiro

	ano	semestre	nome	nome
►	2020	1	José da Silva	Banco de Dados
	2020	1	Paulo Soares	Banco de Dados
	2020	1	José da Silva	Redes

Mostra os professores e matérias do primeiro semestre

### Quarto

2020	2	José da Silva	Redes
2020	2	Ana Rita	Redes
2020	1	José da Silva	Banco de Dados
2020	1	Paulo Soares	Banco de Dados
2020	1	José da Silva	Redes

Mostra os professores e matérias de cada semestre, começando pelo segundo e indo para o primeiro

### Quinto

nome	nome	ano
José da Silva	Redes	2020
José da Silva	Redes	2020
José da Silva	Banco de Dados	2020

Mostra as matérias dadas pelo professor que começa com "J"

## 2. Explique a diferença entre o Primeiro e o Segundo comando.

O primeiro comando realizou uma Junção Cruzada, que combina todas as linhas da tabela Turma com todas as linhas das tabelas Professor e Disciplina, gerando resultados amplos e repetitivos.

O segundo Junção Interna, que juntam as linhas das tabelas com base em alguma restrição, deixando assim somente o necessário visível.

## 2.5 Integridade de Valores

```
CREATE TABLE Colaborador (  
  id_emp INT NOT NULL CONSTRAINT PK_emp PRIMARY KEY ,CONSTRAINT ID_val CHECK (id_emp BETWEEN 0 AND 1000),  
  nome VARCHAR(50) NOT NULL,  
  salario FLOAT NOT NULL CONSTRAINT SL_val CHECK (salario >= 1000));  
  
INSERT INTO Colaborador  
VALUES (2000, 'Josué', 1500.56);  
  
INSERT INTO Colaborador (id_emp, salario)  
VALUES (300, 3500.56);  
  
INSERT INTO Colaborador  
VALUES (400, 'Antônio', 350.56);
```

```
CREATE TABLE Colaborador (  
  id_emp INT NOT NULL CONSTRAINT PK_emp PRIMARY KEY ,CONSTRAINT ID_val CHECK (id_emp BETWEEN 0 AND 1000),  
  nome VARCHAR(50) NOT NULL,  
  salario FLOAT NOT NULL CONSTRAINT SL_val CHECK (salario >= 1000));
```

No MySQL, foi detectado um erro de sintaxe na construção da tabela, que eu não consegui resolver.