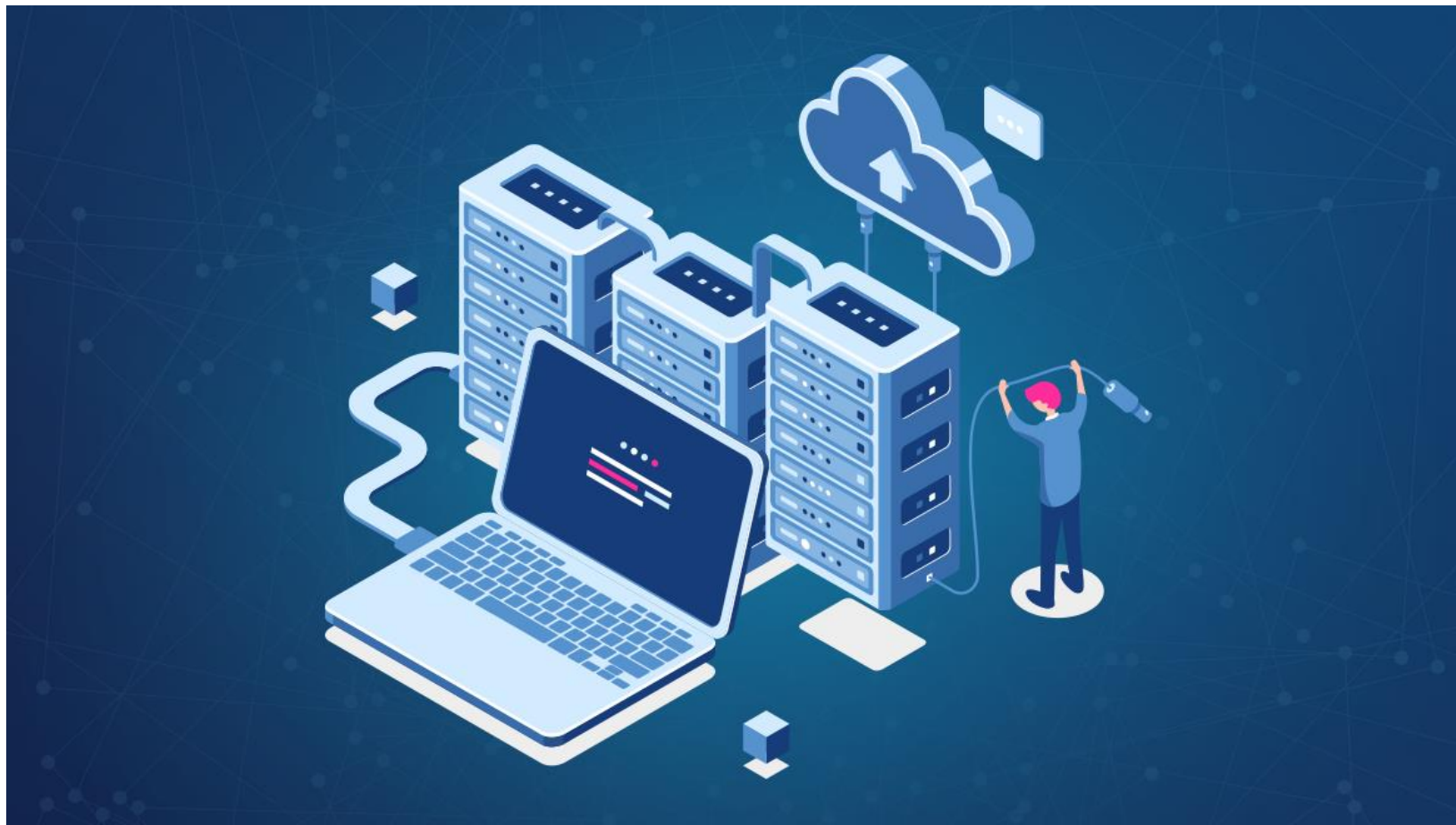




BANCO DE DADOS

AULA 3

DML-SQL

Data Manipulation Language

INSERT

```
INSERT INTO tabela (campo1, campo2, campo3, ...)
VALUES (valor1, valor2, valor3, ...);
```

```
INSERT INTO alunos (nome, tel, codCurso)
VALUES ('José da Silva', '(21) 3322-1212', '5'),
('Maria Socorro', '(21) 4321-0101', '3'),
('Paulo Pereira', '(21) 2134-5438', '3');
```

* campo *mat* é AUTO_INCREMENT

OBS: insert into tabela values (default, 'alex', 9999, 01)

UPDATE

```
UPDATE tabela  
SET coluna=valor,coluna=valor,...  
WHERE expressão;
```

```
UPDATE alunos  
SET nome='Maria do Socorro'  
WHERE mat='3';
```

```
UPDATE alunos  
SET cidade='São Paulo', codCurso='9'  
WHERE mat='10';
```

```
UPDATE alunos  
SET totMaterias='12'  
WHERE codCurso='1';
```

DELETE

DELETE FROM *tabela*
WHERE *expressão*;

DELETE FROM alunos
WHERE mat='171';

DELETE FROM alunos
WHERE cidade = 'São Paulo' AND codCurso = '4';

DELETE FROM alunos
WHERE *expressão*;



SELECT

```
SELECT coluna1,coluna2, ...  
FROM tabela  
WHERE expressão;
```

```
SELECT nome,cidade  
FROM alunos  
WHERE codCurso = '3';
```

```
SELECT *  
FROM alunos  
WHERE nome LIKE 'G%' AND cidade = 'Rio de Janeiro';
```

OPERADORES

< <= = >= > relacionais

<> != diferente

BETWEEN entre

LIKE semelhante

IN range de valores

DISTINCT

```
SELECT DISTINCT cidade  
FROM alunos;
```

```
SELECT DISTINCT codCurso  
FROM alunos  
WHERE nome LIKE 'A%';
```

```
SELECT DISTINCT codCurso  
FROM alunos  
WHERE mensalidade BETWEEN 100 AND 500;
```



ORDENAÇÃO

```
SELECT nome, codCurso  
FROM alunos  
WHERE cidade IN ('Rio de Janeiro', 'São Paulo')  
ORDER BY nome ASC
```

```
SELECT *  
FROM alunos  
WHERE tel LIKE '%31'  
ORDER BY codCurso DESC
```

PAGINAÇÃO

```
SELECT * FROM alunos  
ORDER BY nome  
LIMIT 10;
```

```
SELECT * FROM alunos  
WHERE nome LIKE '%S_ _ _a%'  
LIMIT 11, 10;
```

```
SELECT * FROM alunos  
WHERE nome LIKE '[A-C]%'  
LIMIT 110, 20;
```

UNION

```
SELECT cidade FROM alunos  
ORDER BY cidade  
UNION  
SELECT cidade FROM professores  
ORDER BY cidade
```

UNION ALL

```
SELECT cidade FROM alunos  
ORDER BY cidade  
UNION ALL  
SELECT cidade FROM professores  
ORDER BY cidade
```



FUNÇÕES DE AGREGAÇÃO

SELECT AVG(preco) FROM produtos;

SELECT COUNT(mat) FROM alunos;

SELECT COUNT(*) FROM fornecedores;

SELECT SUM(qtd) FROM produtos
WHERE tipo = "brinquedo";

SELECT MAX(salario) FROM professores;



54 – Sobre a utilização da sintaxe SQL, informe se é falso (F) ou verdadeiro (V) o que se afirma abaixo. A seguir, indique a opção com a sequência correta.

- () A instrução SELECT permite definir quais colunas de quais tabelas desejamos obter as fontes (tabelas) dos dados e os ordenamentos, entre outros.
 - () A instrução DELETE não permite definir uma condição, enquanto que o comando SELECT permite.
 - () A instrução SELECT é utilizada para retornar um conjunto de linhas a partir de uma ou mais tabelas do banco de dados.
 - () A instrução INSERT permite definir quais colunas terão valores a serem preenchidos, mas não permite inserir um conjunto de registros a partir de um comando SELECT.
 - () A instrução UPDATE permite definir uma condição nos mesmos moldes do comando SELECT, envolvendo, inclusive, mais de uma tabela.
-
- a) V - V - F - F - F
 - b) V - F - V - F - V
 - c) F - F - F - V - V
 - d) F - V - V - V - F

67 – Considere que está sendo utilizado o comando UPDATE da linguagem SQL, para alterar dados em um banco de dados. Considere ainda que a tabela e suas colunas são denominadas da seguinte forma:

Nome da tabela: produtos;

Valor do custo: custo;

Valor de venda: venda;

Código do fornecedor: fornecedor.

Utilizando a seguinte linha de código:

```
UPDATE produtos SET  custo  = custo * 1.2,  
                    venda = venda * 1.25  
                    WHERE fornecedor = 10;
```

Podemos então afirmar que

- a) a tabela produtos terá todos os seus valores de custo reajustados em 1.2; e os valores de venda em 1.25.
- b) a sintaxe do comando UPDATE está incorreta.
- c) o comando UPDATE permite alterar os valores das duas colunas (custo e venda) ao mesmo tempo.
- d) o fornecedor cujo código é “10” será excluído da tabela.

86 – Considere a instrução SQL a seguir:

```
INSERT INTO pessoas (codigo, nome) VALUES ('1','maria');
```

Verifique quais são as afirmações verdadeiras (V) e as falsas (F) a respeito da instrução acima e assinale a alternativa que apresenta a sequência correta.

- () O nome da tabela que está sendo usada é “nome”.
 - () O nome da tabela que está sendo usada é “pessoas”.
 - () Ao ser executada, a instrução irá atualizar o valor da coluna “codigo” para ‘1’.
 - () O comando INSERT atualiza dados em uma tabela.
 - () Utilizando esta instrução será inserida uma nova linha na tabela, contendo valores em duas colunas.
-
- a) V - F - F - F - V
 - b) F - V - F - F - V
 - c) V - F - V - F - F
 - d) F - V - V - V - F

68 – Considere a tabela abaixo chamada “produtos”, com a estrutura e os dados a seguir:

codigo_produto	nome_produto	preco
1	Camisa branca	49.50
2	Camisa verde	39.20
3	Camisa azul	30.60
4	Camisa vinho	45.90

Supondo-se a necessidade de um relatório que contenha todos os dados das camisas desta tabela que custem mais do que 40.02, assinale a alternativa que contém o comando SQL correto para essa finalidade.

- a) `SELECT codigo_produto, nome_produto, FROM produtos WHERE preco>40.02;`
- b) `SELECT codigo_produto, nome_produto, preco FROM produtos WHERE preco>40.02;`
- c) `SELECT codigo_produto, nome_produto, preco FROM produtos WHERE preco<40.02;`
- d) `SELECT nome_produto, preco FROM produtos WHERE preco<40.02;`

84 – Considere o comando a seguir, que utiliza a linguagem SQL. Assinale a alternativa que apresenta informação verdadeira a respeito desse comando.

```
INSERT INTO produtos  
(codigo_produto, nome_produto, preco, cod_categoria)  
VALUES (2, 'Camiseta branca', 50.03, 4);
```

- a) O campo codigo_produto receberá um valor “NULL”.
- b) Os valores já existentes na tabela serão substituídos pelos valores declarados, conforme a ordem definida.
- c) Os valores serão inseridos na tabela produtos, ocupando os campos, conforme a ordem definida.
- d) O campo cod_categoria receberá um valor 2.



93 — A seguir são apresentados comandos utilizados na linguagem SQL, que podem ser usados, por exemplo, no SGBD MySQL. Relacione as colunas e, em seguida, assinale a alternativa que contém a sequência correta.

- | | |
|---|--|
| 1 desc <nome_tabela>; | () Utilizado para incluir dados em uma tabela. |
| 2 DROP TABLE <nome_tabela1> [, <nome_tabela2>, ...]; | () Utilizado para visualizar a estrutura de determinada tabela. |
| 3 show tables; | () Utilizado para excluir tabelas do banco de dados. |
| 4 INSERT INTO <nome_tabela> VALUES (valor1, valor2, ..., valorn); | () Mostra o nome das tabelas criadas. |

- a) 1 – 3 – 2 – 4
- b) 2 – 1 – 4 – 3
- c) 4 – 1 – 2 – 3
- d) 4 – 1 – 3 – 2

71 – Considere que, no banco de dados, existe a tabela “itens”, descrita a seguir:

Coluna	Tipo	
codigo_item	smallint	not null
nome_item	Character varying(80)	
preco_item	double precision	not null
cod_categoria	Character varying(4)	not null

Considere ainda que será utilizado o comando INSERT INTO abaixo, na linguagem SQL, que irá fazer a inserção de um produto na referida tabela:

```
INSERT INTO itens  
(codigo_item, nome_item, preco_item, cod_categoria)  
VALUES (05, 'camiseta branca', 49.90)
```

Em relação ao resultado produzido pelo comando citado na tabela, assinale a alternativa correta.

- a) Os valores do campo cod_categoria não podem ser declarados entre aspas simples, pois isso causará um erro na execução do comando.
- b) Os valores constantes dos campos declarados no comando acima ocuparão os campos, independente da ordem em que aparecem no comando.
- c) O campo código_item receberá o valor NULL.
- d) Ocorrerá um erro na execução do comando INSERT ao ser executado esse comando.

75 – Considere que os seguintes registros estão gravados na tabela abaixo, denominada “itens”:

codigo_item	nome_item	preco_item	cod_categoria
01	bola	10.20	AABB
02	DVD	2.00	BBZZ
03	camisa listrada	30.50	CCAA

Assinale a alternativa que traz o resultado correto da pesquisa realizada na tabela “itens” pelo seguinte comando:

```
SELECT codigo_item, nome_item, preco_item FROM itens  
WHERE preco_item>10.20;
```

a)

nome_item	preco_item	cod_categoria
bola	10.20	AABB
camisa listrada	30.50	CCAA

b)

codigo_item	nome_item	preco_item
01	bola	10.20
03	camisa listrada	30.50

c)

codigo_item	nome_item	preco_item
03	camisa listrada	30.50

d)

nome_item	preco_item	cod_categoria
camisa listrada	30.50	CCAA

76 – Considerando o comando a seguir, que utiliza a linguagem SQL, assinale a alternativa correta.

Obs.: os valores do campo “preço” são considerados em reais e todos os campos da tabela são de preenchimento obrigatório.

```
SELECT (codigo_produto, nome_produto, preco,  
cod_categoria FROM produtos)  
WHERE cod_categoria=3 AND preco<50.00;
```

- a) O campo codigo_produto contém um valor “NULL”.
- b) Os valores já existentes na tabela serão substituídos pelos valores declarados na cláusula WHERE, conforme a ordem definida.
- c) O comando retornará somente o valor constante do campo nome de todos os produtos cujo cod_categoria é “3” e que custam menos de R\$50.00.
- d) O comando retornará o valor constante dos campos: codigo_produto, nome_produto, preco, cod_categoria de todos os produtos cujo cod_categoria é “3” e que custam menos de R\$50.00.



78 – A seguir está um comando na linguagem SQL, que pode ser usado, por exemplo, no SGBD MySQL. Assinale a alternativa que contém a afirmativa correta.

`ALTER TABLE loja ADD empresa varchar(50);`

- a) Essa sintaxe do comando `ALTER TABLE` é usada para incluir ou excluir campos de uma tabela.
- b) O comando acima permite adicionar o campo `loja` à tabela `empresa`.
- c) O campo incluído na tabela pelo comando acima será incluído como o último campo da tabela.
- d) É possível afirmar que todos os registros da tabela referenciada no comando acima são do tipo `varchar(50)`.

