

Objetivos da Aula:

- SQL - (Strutured Query Language)
- DDL

Aula01

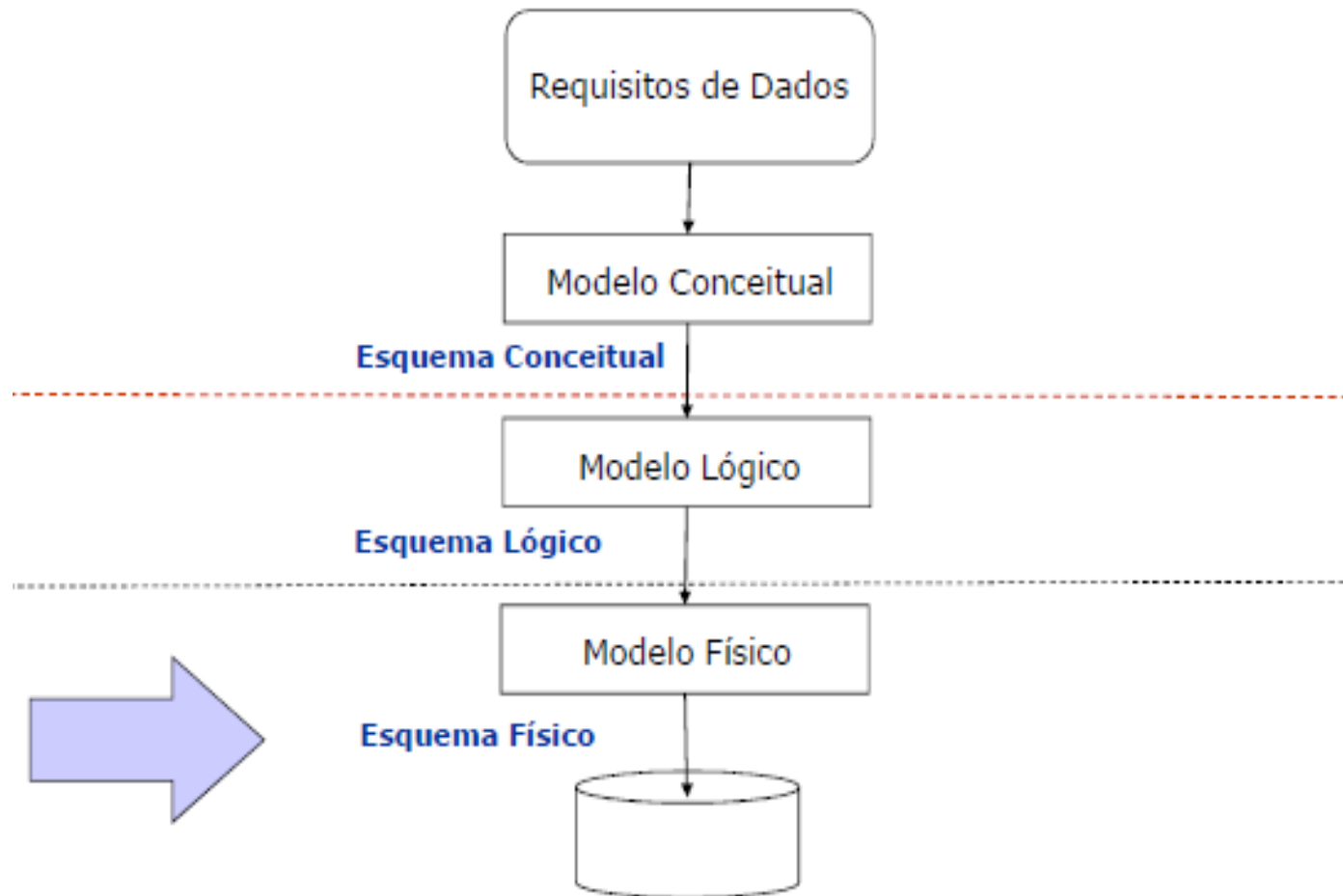
Professora: Juliana Forin Pasquini Martinez

08-02-2019

Tópicos

- Projeto de Banco de Dados
- Acessando o Oracle 11 G Express
- Descrição e Tipos de instruções SQL
 - Instruções DDL
 - Instruções DML
 - Instruções DCL
- Introdução aos comandos DDL: Parte 1

Projeto de Banco de Dados



Descrição da SQL

- SQL - *Structured Query Language*- *Linguagem Estruturada de Consulta*
- Linguagem para acesso a Sistemas de Bases de Dados Relacionais (padrão para diversos produtos comerciais).
- Uma base de dados é como uma coleção de tabelas (Modelo Relacional).
- ***Características da linguagem SQL:***
 - Independente do fabricante do SGBD
 - Portabilidade entre plataformas
 - Redução de custos com treinamento
 - Inglês estruturado de alto nível
 - Definição de múltiplas visões dos dados

Tipos de Instruções SQL

- **Comandos DDL (*Data Definition Language*):**
 - comandos destinados a manutenção do esquema do BD;
 - Manutenção de objetos (tabelas, índices, colunas, etc);
 - Especificação de restrições de integridade.

Principais comandos:

- **CREATE** ➡ Cria objetos no esquema
- **ALTER** ➡ Altera objetos do esquema
- **DROP** ➡ Exclui objetos do esquema

Tipos de Instruções SQL (cont...)

- ❑ **Comandos DML (*Data Manipulation Language*):**
comandos destinados a manipulação dos dados do banco de dados.

Principais comandos:

- SELECT Seleciona linhas de dados
- INSERT Inclusão de dados
- UPDATE Alteração de dados
- DELETE Exclusão de dados
- COMMIT Confirma operações com dados
- ROLLBACK Cancela operações com dados

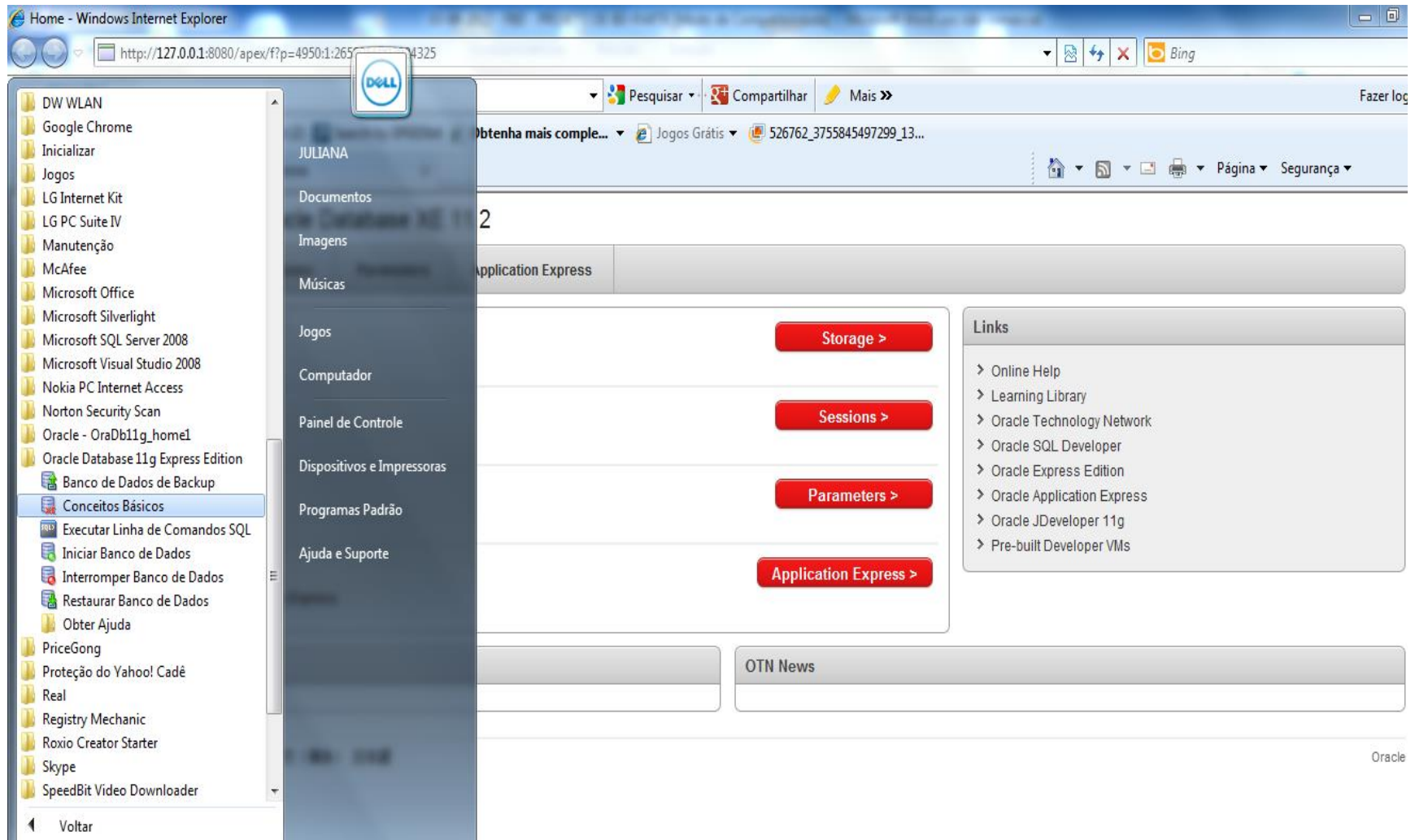
Tipos de Instruções SQL (cont...)

- ❑ **Comandos DCL (*Data Control Language*): comandos**
destinados ao controle de acesso aos dados, ou seja, definição dos privilégios dos usuários.

Principais comandos:

- GRANT → Concede privilégios aos usuários
- REVOKE → Revoga privilégios dos usuários

Acessando o Oracle 11



Home - Windows Internet Explorer

http://127.0.0.1:8080/apex/f?p=4950:1:2658956093024325

Google

Pesquisar

Compartilhar

Mais >>

Fazer login

Favoritos

Search by SPEEDbit (2)

Search by SPEEDbit

Obtenha mais comple...

Jogos Grátis

526762_3755845497299_13...

sql - linguagem estruturad...

Home

Página

Segurança

ORACLE

Oracle Database XE 11.2

Home

Storage

Sessions

Parameters

Application Express

Storage

View currently used storage.

Storage >

Sessions

View current database sessions.

Sessions >

Parameters

View initialization parameters.

Parameters >

Application Express

Get started with Oracle Application Express.

Application Express >

Links

> Online Help

> Learning Library

> Oracle Technology Network

> Oracle SQL Developer

> Oracle Express Edition

> Oracle Application Express

> Oracle JDeveloper 11g

> Pre-built Developer VMs

News

OTN News

Language: **English**, Português (Brasil), 中文 (简体), 日本語

Oracle

Internet | Modo Protegido: Desativado

100%

Application Express - Windows Internet Explorer

http://127.0.0.1:8080/apex/f?p=4950:7:2658956093024325::NO

Google Pesquisas Compartilhar Mais >> Fazer login

Favoritos Search by SPEEDbit (2) Search by SPEEDbit Obtenha mais comple... Jogos Grátis 526762_3755845497299_13...

sql - linguagem estruturad... Application Express

Página Segurança

ORACLE Oracle Database XE 11.2

Welcome: SYSTEM Logout

Home Storage Sessions Parameters **Application Express**

Home > Oracle Application Express

Create Application Express Workspace

Cancel Create Workspace

Database User ☒ Create New ☐ Use Existing

* Database Username USER_FBD

* Application Express Username system

* Password

* Confirm Password

Getting Started

Already have an account? Login Here

To get started with Oracle Application Express, create a workspace. You will need to specify:

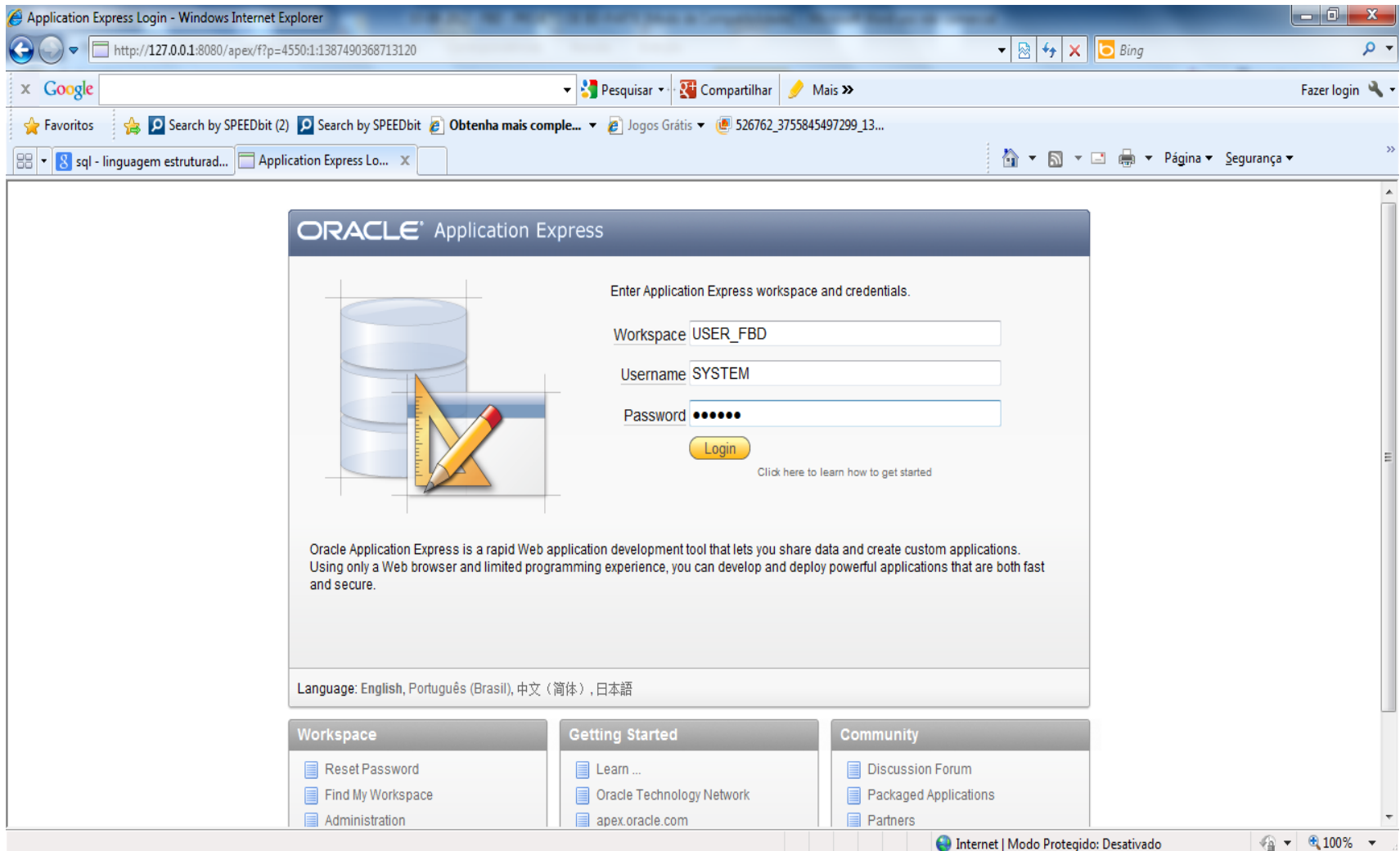
- Database Username - Name of the database user to be created
- Application Express Username - Your login name for the Application Express Workspace
- Password - Password of both your database user and Application Express user

Once created, you will be able to [login to your Application Express workspace](#) using these credentials

Oracle

Internet | Modo Protegido: Desativado 100%

Acessar o link: <http://127.0.0.1:8080/apex/>



Oracle Application Express - Windows Internet Explorer

http://127.0.0.1:8080/apex/f?p=4500:1000:3478077451446037

Google Pesquisas Compartilhar Mais >> Fazer login

Favoritos Search by SPEEDbit (2) Search by SPEEDbit Obtenha mais comple... Jogos Grátis 526762_3755845497299_13...

sql - linguagem estruturad... Oracle Application Expr... Página Segurança


ORACLE Application Express

Welcome SYSTEM (Logout)


Home Application Builder SQL Workshop Team Development Administration Search Workspace Help

Workspace USER_FBD


Learn more ...




Application Builder



SQL Workshop



Team Development



Administration

News

Top Applications

Top Users

Team Development

Show: All
Release: All Releases

Features	0
Todo's	0
Milestones	0
Bugs	0

Available Updates

Oracle APEX 4.1.1 available [More Information](#)

Language

- English
- Português (Brasil)
- 中文 (简体)
- 日本語

Concluído

Internet | Modo Protegido: Desativado 100%

SQL Commands - Windows Internet Explorer

http://127.0.0.1:8080/apex/f?p=4500:1003:3478077451446037::NO::

Google Pesquisas Compartilhar Mais >> Fazer login

Favoritos Search by SPEEDbit (2) Search by SPEEDbit Obtenha mais comple... Jogos Grátis 526762_3755845497299_13...

sql - linguagem estruturad... SQL Commands

Oracle Application Express Welcome SYSTEM (Logout)

Home Application Builder SQL Workshop Team Development Administration

Home > SQL Workshop > SQL Commands Schema USER_FBD Help

☒ Autocommit Rows 10 Save Run

Results Explain Describe Saved SQL History

Enter SQL statement or PL/SQL command and click Run to see the results.

Application Express 4.0.2.00.09

Concluído Internet | Modo Protegido: Desativado 100%



Comandos DDL

Parte 1

Create Table

□ Sintaxe:

CREATE TABLE TABELA

(

COLUNA TIPO [DEFAULT *exp*] [NULL | NOT NULL]

[CONSTRAINT da coluna],

...,

[CONSTRAINT da tabela]

);

Create Table (cont...)

```
CREATE TABLE TABELA  
(  
  COLUNA TIPO [DEFAULT exp]  
  [NULL / NOT NULL]  
  [CONSTRAINT da coluna],  
  ...,  
  [CONSTRAINT da tabela]  
);
```

□ Descrição da sintaxe:

- **TABELA:** é o nome da tabela
- **COLUNA:** é o nome da coluna
- **TIPO:** tipo de dados da coluna + tamanho
- **[DEFAULT exp]:** *especifica o valor que será utilizado quando um dado for omitido durante uma inclusão*
- **[NULL | NOT NULL]:** define se a coluna aceitará ou não valores nulos.
- **[CONSTRAINT]:** que especifica as restrições para uma coluna ou para a tabela.

Exemplo: Create Table

Create table Cliente

```
(  cod number(6) primary key,  
    nome varchar(30),  
    ativo number(2) DEFAULT 1,  
    data_nasc date);
```

Convenções de nomenclatura

- Regras para nomes das tabelas e colunas:
 - Devem começar com uma letra
 - Pode ter de 1 a 30 caracteres
 - Pode conter somente A-Z, a-z, 0-9, _, \$ e #
 - Os nomes devem ser únicos por usuário
 - Não podem ser utilizadas palavras reservadas

Tipos de Dados Básicos do Oracle

Tipo	Descrição
<code>char(n)</code>	Cadeia de caracteres com tamanho fixo n. O default é 1 e o máximo 2000.
<code>Varchar2(n)</code>	Cadeia contendo caracteres de tamanho variável com o máximo de n, em que n pode ter o valor até 4000.
<code>Clob</code>	Significa Character Long Object. Permite armazenar extensas cadeias contendo caracteres de tamanho variável com o máximo de 4 gigabytes.
<code>Long</code>	Permite armazenar longas cadeias de caracteres de tamanho variável com o máximo de 2 gigabytes.
<code>Number (p,e)</code>	Valores numéricos em que p indica a precisão (máximo 38) e indica a escala (número de casas decimais) que varia de -84 +127. Por exemplo, number (5,2) especifica um número na faixa ente -999,99 e +999,99
<code>Date</code>	Armazena data e hora, incluindo século, ano, mês, dia, hora, minuto e segundo. Campos deste tipo ocupam 7 bytes

Restrições (*Constraints*)

- As restrições impõem regras ao banco de dados.
- Devem ser definidas para evitar que dados incorretos (inválidos) sejam inseridos nas tabelas, garante a consistência dos dados.
- São válidos os seguintes tipos de restrições:
 - **NOT NULL** → Obrigatoriedade de valor no campo
 - **UNIQUE** → Unicidade de valor no campo
 - **PRIMARY KEY** → Definição de chave primária
 - **FOREIGN KEY** → Definição de chave estrangeira
 - **CHECK** → Validação de valor dentro de domínio

Constraint *UNIQUE KEY* (UK)

- Define que cada valor da coluna, ou conjunto de colunas, seja sempre único dentro da tabela.
- Uma coluna especificada com uma restrição *UNIQUE KEY* é chamada de chave exclusiva.
- Se a restrição *UNIQUE KEY* for formada por mais de uma coluna, o grupo de colunas é chamado de chave exclusiva composta.
- Pode ser definida tanto a nível de coluna quanto a nível de tabela.

Exemplo: UNIQUE KEY a nível de tabela

```
CREATE TABLE ALUNO
(
  COD_ALUNO number(4) primary key,
  MATRICULA number(4) NOT NULL,
  NOME varchar2(40),
  CPF varchar2(11),
  Constraint UK_CPF Unique(CPF)
);
```

Exemplo: UNIQUE KEY a nível de coluna

```
CREATE TABLE ALUNO  
(  
  COD_ALUNO number(4) NOT NULL,  
  MATRICULA number(4) NOT NULL,  
  NOME varchar2(40),  
  CPF number(11) Constraint UK_CPF Unique  
);
```

Constraint PRIMARY KEY (PK)

- Define uma ou mais colunas como chave primária da coluna.
- Portanto, essa restrição identifica exclusivamente cada linha de dados em uma tabela.
- Nenhuma coluna com definição de uma constraint PRIMARY KEY aceitará valores nulos.
- Uma constraint UNIQUE é automaticamente criada para uma coluna (ou conjunto de colunas) PRIMARY KEY.
- Pode ser definida a nível de coluna e tabela.

Exemplo: PRIMARY KEY a nível de tabela

```
CREATE TABLE ALUNO
(
  COD_ALUNO number(4),
  MATRICULA number(4) NOT NULL,
  NOME varchar2(40),
  CPF number(11),
  Constraint PK_COD_ALUNO Primary Key(COD_ALUNO)
);
```


Exemplo: PRIMARY KEY a nível de coluna

```
CREATE TABLE ALUNO  
(  
  COD_ALUNO number(4) Primary Key,  
  MATRICULA number(4) NOT NULL,  
  NOME varchar2(40),  
  CPF number(11)  
);
```

OBS: A definição de uma chave primária composta só pode ser realizada a nível de tabela!

Exemplo: PRIMARY KEY COMPOSTA

```
create table itempedido
(
  ped_cod number(6) constraint ip_ped_fk references pedido(ped_cod),
  pro_cod number(6) constraint ip_pro_fk references produto(prod_cod),
  ite_quantidade number(6),
  ite_valor number(7,2),
  constraint itempedido primary key (ped_cod,pro_cod));
```



```
CREATE TABLE PRODUTO
(
    prod_cod number(6) constraint pk_prod_cod primary key,
    pro_nome varchar2(30) not null,
    pro_quantidade number(4),
    pro_valor number(9,2)
);
```

Constraint FOREIGN KEY (FK)

- Define uma restrição de integridade referencial, designando uma ou mais colunas como chave estrangeira.
- Um valor de chave estrangeira deve corresponder a um valor existente na tabela referida ou valor NULL.
- Pode ser definida a nível de coluna e tabela

Exemplo: FOREIGN KEY a nível de tabela

```
CREATE TABLE ALUNO
(
  COD_ALUNO number(4),
  MATRICULA number(4) NOT NULL,
  COD_CURSO number(4),
  CPF varchar2(11),
  NOME varchar2(40),
  Constraint FK_COD_CURSO Foreign Key(COD_CURSO)
  References CURSO (COD_CURSO)
);
```

Exemplo: FOREIGN KEY a nível de coluna

```
CREATE TABLE ALUNO
(
  COD_ALUNO number(4) NOT NULL,
  MATRICULA number(4) NOT NULL,
  COD_CURSO number(4) Constraint FK_COD_CURSO
  References CURSO (COD_CURSO),
  CPF varchar2(11),
  NOME varchar2(40)
);
```

Constraint CHECK (CK)

- Define uma condição (regra) que cada valor da coluna de obedecer.

Exemplos:

- Restrição de valor mínimo para salário, ou seja, só aceitar valores maiores que 500,00.
 - Restrição de campo SEXO, permitindo apenas valores dentro do domínio ("F", "M")
- Pode ser definida a nível de coluna e tabela

Exemplo: CHECK a nível de tabela

```
CREATE TABLE ALUNO
```

```
(
```

```
  COD_ALUNO number(4) PRIMARY KEY,
```

```
  MATRICULA number(4) UNIQUE,
```

```
  CPF varchar2(11),
```

```
  NOME varchar2(40),
```

```
  SEXO char(1),
```

```
  Constraint CK_SEXO Check(SEXO in ('F', 'M'))
```

```
);
```

Exemplo: CHECK a nível de coluna

```
CREATE TABLE ALUNO
(
  COD_ALUNO number(4) NOT NULL,
  MATRICULA number(4) NOT NULL,
  NOME varchar2(40),
  CPF varchar2(20),
  SEXO char(1) Constraint CK_SEXO Check (SEXO in ('F', 'M'))
);
```

Dica: Como verificar as *constraints*

- Através de uma *VIEW*, chamada ***USER_CONSTRAINTS***

Sintaxe do comando:

Select CONSTRAINT_NAME, CONSTRAINT_TYPE,
STATUS, SEARCH_CONDITION

From USER_CONSTRAINTS

Where TABLE_NAME = 'ALUNO'

Dica: Como verificar as constraints (cont...)

- **CONSTRAINT_NAME** → nome da restrição
- **CONSTRAINT_TYPE** → tipo da restrição
 - C = Not Null
 - P = Primary Key
 - R = Foreign Key
 - U = Unique Key
- **STATUS** → indica se a restrição está ativa (ENABLED) ou não (DISABLE).

Boas práticas

- Ao definir um nome para uma *constraint*, procurar definir o nome de acordo com o padrão abaixo:
 - Sempre em CAIXA ALTA
- **SIGLA_** + NOME_TABELA + NOME_DA_COLUNA. Exemplos:
UNIQUE KEY UK_ALUNO_CPF
PRIMARY KEY PK_ALUNO_COD_ALUNO ou PK_ALUNO
FOREIGN KEY FK_ALUNO_CURSO
CHECK CK_ALUNO_SEXO
- OBS: Boas práticas = DICA, portanto não é uma regra!