

# MBA em IA e Big Data



## Curso 01 - Linguagens e Ferramentas para Inteligência Artificial e Big Data (Python e SQL)

SQL - Data Definition Language no Sistema Oracle

*Jose Fernando Rodrigues Junior*  
*ICMC-USP São Carlos*

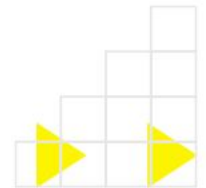
Objetivo: apresentar os comandos SQL usados para a definição da base de dados





# SQL - Structured Query Language

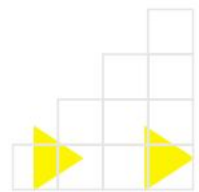
- **Linguagem declarativa** – expressa o que se quer, mas não como será a execução
- IBM - década de 70
- “Padrão” de mercado
  - Ansi/ISO
  - simplicidade
  - grande poder de consulta





# Data Definition Language (DDL)

Subconjunto do SQL para definição de esquemas: **CREATE, DROP, ALTER**





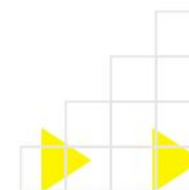
# DDL - Introdução

- CREATE, DROP, ALTER

- Elementos fundamentais da linguagem, aplicados a:

- DATABASE
- USER
- ROLE
- SCHEMA
- TABLESPACE
- TABLE
- INDEX
- FUNCTION
- SEQUENCE
- TRIGGER
- VIEW
- .....

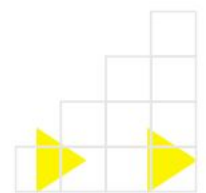
- Todos os elementos podem ser criados (**CREATE**), corrigidos (**ALTER**) e removidos (**DROP**)



# Comandos DDL

- **CREATE TABLE** - criar uma tabela, definir colunas e restrições

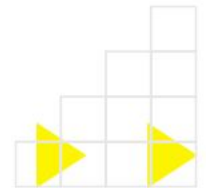
```
CREATE TABLE tabela (  
    atrib1 tipo [<restrições da coluna 1>],  
    atrib2 tipo [<restrições da coluna 2>],  
    ....  
    atribn tipo [<restrições da coluna n>],  
  
    <restrições da tabela>  
);
```



# CREATE TABLE

- Principais restrições de colunas
  - NOT NULL
  - DEFAULT *valor*

```
CREATE TABLE tabela (  
  atrib1 tipo [(tamanho)] [NOT NULL | DEFAULT valor],  
  atrib2 tipo [(tamanho)] [NOT NULL | DEFAULT valor],  
  ...
```



# CREATE TABLE

## ■ Restrições de tabela

- PRIMARY KEY ( *<atributos chave primária>* )
- UNIQUE ( *<atributos chave candidata>* )
- FOREIGN KEY ( *<atributos chave estrangeira>* REFERENCES *tabelaRef* [ *<chave primária>* ]

[*<ações>*]

### ■ *<ações>*

- ON DELETE | ON UPDATE
  - CASCADE | SET NULL | SET DEFAULT

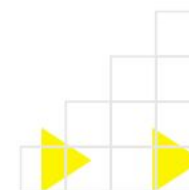
Em SQL padrão apenas; Oracle não suporta "ON UPDATE"





# SQL – Alguns tipos de dados

- **INTEGER | SMALLINT | DOUBLE | PRECISION | FLOAT | REAL**
- **DECIMAL [(precision, scale)]**
  - *precision* - número total de dígitos
  - *scale* - número de dígitos depois do ponto
- **NUMBER (precisão, escala)** ← tipo numérico ORACLE
- **CHAR (n)** - tamanho fixo - n caracteres
- **VARCHAR (n)** - tamanho variável ← ORACLE: **VARCHAR2**
  - máximo de n caracteres
- **BLOB** – *Binary Large Object*
- **CLOB** - *Character Large Object*
- **DATE | TIME | TIMESTAMP**







# SQL – Alguns tipos de dados do “padrão” SQL

	<b>int10</b>	<b>int6</b>	<b>int1</b>	<b>char(n)</b>	<b>blob</b>	<b>XML</b>
Oracle 11	NUMBER(10)	NUMBER(6)	NUMBER(1)	VARCHAR2(n)	BLOB	XMLType
MS SQL Server 2005	NUMERIC(10)	NUMERIC(6)	TINYINT	VARCHAR(n)	IMAGE	XML
Sybase system 10	NUMERIC(10)	NUMERIC(6)	NUMERIC(1)	VARCHAR(n)	IMAGE	
MS Access (Jet)	Long Int or Double	Single	Byte	TEXT(n)	LONGBINARY	
TERADATA	INTEGER	DECIMAL(6)	DECIMAL(1)	VARCHAR(n)	VARBYTE(20480)	
DB2	INTEGER	DECIMAL(6)	DECIMAL(1)	VARCHAR(n)	VARCHAR(255)	
RDB	INTEGER	DECIMAL(6)	DECIMAL(1)	VARCHAR(n)	LONG VARCHAR	
INFORMIX	INTEGER	DECIMAL(6)	DECIMAL(1)	VARCHAR(n)	BYTE	
RedBrick	integer	int	int	char(n)	char(1024)	
INGRES	INTEGER	INTEGER	INTEGER	VARCHAR(n)	VARCHAR(1500)	



## ■ CREATE TABLE

```
CREATE TABLE tabela (  
    atrib1 tipo [(tamanho)] [NOT NULL | DEFAULT valor],  
    atrib2 tipo [(tamanho)] [NOT NULL | DEFAULT valor],  
    ...  
    [CONSTRAINT nome da restrição] PRIMARY KEY (<atributos chave primária>),  
  
    [CONSTRAINT nome da restrição] UNIQUE (< atributos chave candidata>),  
  
    [CONSTRAINT nome da restrição] FOREIGN KEY (<atributos chave estrangeira>)  
    REFERENCES tabelaRef [(<chave primária>)] [ON DELETE CASCADE | SET NULL]  
    [ON UPDATE CASCADE | SET NULL]  
);
```

Aluno = { Nome, Nusp, Idade, DataNasc }

Professor = { Nome, NFunc, Idade, Titulação }

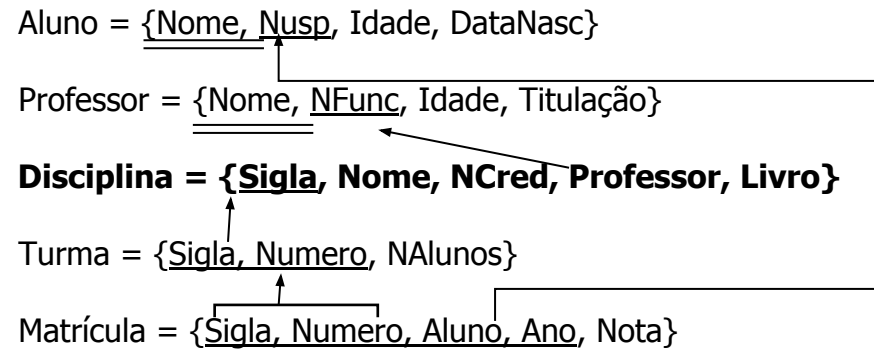
Disciplina = { Sigla, Nome, NCred, Professor, Livro }

Turma = { Sigla, Numero, NAlunos }

Matrícula = { Sigla, Numero, Aluno, Ano, Nota }

```
CREATE TABLE ALUNO (  
    NOME VARCHAR2(30) NOT NULL,  
    NUSP NUMBER NOT NULL,  
    IDADE NUMBER(3),  
    DATANASC DATE,  
  
    CONSTRAINT PK_ALUNO PRIMARY KEY (NUSP),  
    CONSTRAINT UN_NOME UNIQUE(NOME)  
  
);
```

```
CREATE TABLE PROFESSOR (  
    NOME VARCHAR2(30) NOT NULL UNIQUE,  
    NFUNC NUMBER NOT NULL PRIMARY KEY,  
    IDADE NUMBER(3),  
    TITULACAO CHAR(10) NOT NULL,  
  
    CONSTRAINT CH_TIT CHECK (TITULACAO IN ('MESTRE', 'DOUTOR', 'TITULAR'))  
  
);
```



```
CREATE TABLE DISCIPLINA (  
    SIGLA CHAR(6) NOT NULL,  
    NOME VARCHAR2(30) NOT NULL,  
    NCRED NUMBER NOT NULL,  
    PROFESSOR NUMBER ,  
    LIVRO VARCHAR2(30),  
    CONSTRAINT PK_DISCIPLINA PRIMARY KEY (SIGLA),  
    CONSTRAINT FK_DISCIPLINA FOREIGN KEY (PROFESSOR) REFERENCES  
PROFESSOR(NFUNC) ON DELETE SET NULL,  
    CONSTRAINT NCREDITOS CHECK (NCRED > 0)  
);
```

Aluno = {Nome, Nusp, Idade, DataNasc}

Professor = {Nome, NFunc, Idade, Titulação}

Disciplina = {Sigla, Nome, NCred, Professor, Livro}

**Turma = {Sigla, Numero, NAlunos}**

Matrícula = {Sigla, Numero, Aluno, Ano, Nota}

```
CREATE TABLE TURMA (  
    SIGLA CHAR(6) NOT NULL,  
    NUMERO NUMBER NOT NULL,  
    NALUNOS NUMBER NOT NULL  
    CHECK(NAlunos <= 70),  
    CONSTRAINT PK_TURMA PRIMARY KEY (SIGLA, NUMERO),  
    CONSTRAINT FK_TD FOREIGN KEY (SIGLA) REFERENCES DISCIPLINA(SIGLA)  
    ON DELETE CASCADE  
);
```

Aluno = {Nome, Nusp, Idade, DataNasc}

Professor = {Nome, NFunc, Idade, Titulação}

Disciplina = {Sigla, Nome, NCred, Professor, Livro}

Turma = {Sigla, Numero, NAlunos}

Matrícula = {Sigla, Numero, Aluno, Ano, Nota}

```
CREATE TABLE MATRICULA (  
  SIGLA CHAR(6) NOT NULL,  
  NUMERO NUMBER NOT NULL,  
  ALUNO NUMBER NOT NULL,  
  ANO NUMBER(4) NOT NULL,  
  NOTA FLOAT,  
  CONSTRAINT PK_MAT PRIMARY KEY (SIGLA, NUMERO, ALUNO, ANO),  
  CONSTRAINT FK_MT FOREIGN KEY (SIGLA, NUMERO)  
    REFERENCES TURMA(SIGLA, NUMERO)  
    ON DELETE CASCADE,  
  CONSTRAINT FK_MA FOREIGN KEY (ALUNO) REFERENCES ALUNO(NUSP)  
    ON DELETE CASCADE  
);
```

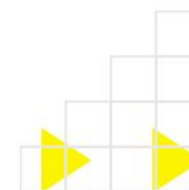


# Comandos DDL

- **ALTER TABLE** – incluir/alterar/remover definições de colunas e restrições  
**ALTER TABLE *tabela* <ação>;**

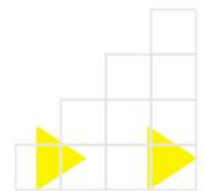
- **<ação>:**

- **ADD *novoAtrib* *tipo* [<restrições de coluna>]**
- **ADD [CONSTRAINT *nome*] <restrição de tabela>**
- **DROP COLUMN *atributo* [CASCADE | RESTRICT]**
- **DROP CONSTRAINT *nome***
- **ALTER *atributo* DROP DEFAULT;**
- **ALTER *atributo* SET DEFAULT <valor>;**

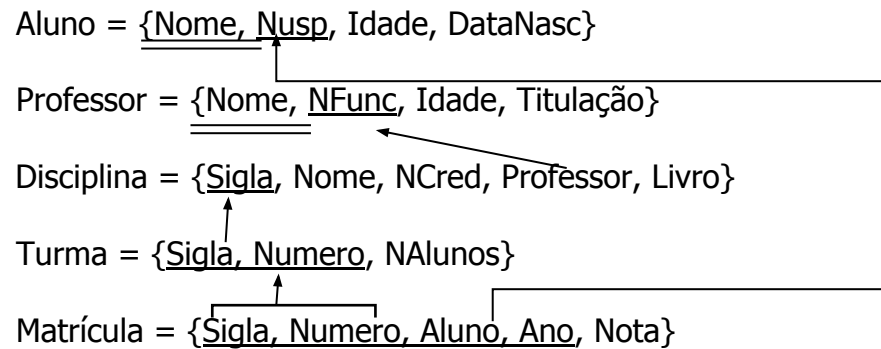


# ALTER TABLE

- **ADD** *novoStrib* *tipo* [*<restrições de coluna>*]
- **DROP COLUMN** *atributo* [**CASCADE** | **RESTRICT**]
  - **CASCADE** – todas as visões e restrições (*constraints*) que referenciam o atributo são removidas automaticamente (dados referenciados são mantidos)
  - **RESTRICT** – atributo só é removido se não houver nenhuma visão ou restrição que o referencie

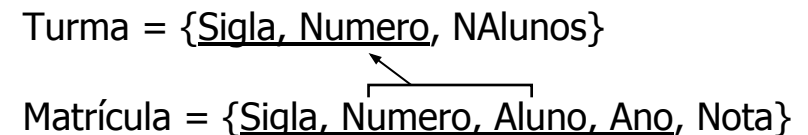




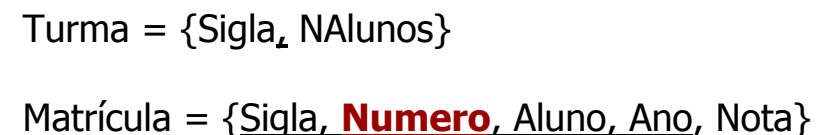


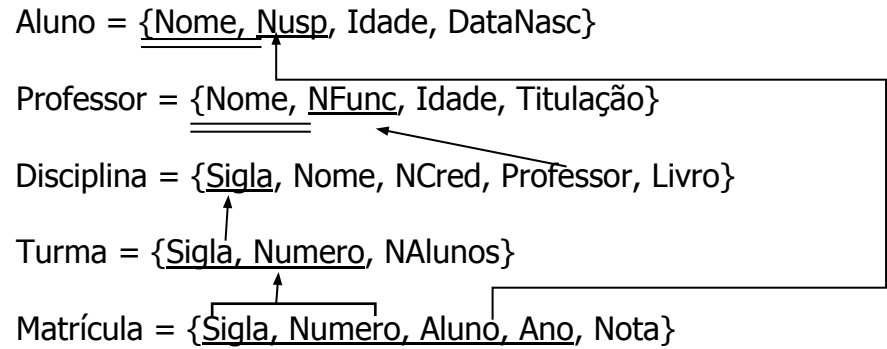
**alter table Aluno add CidadeOrigem varchar(30) default 'Sao Carlos';**

**alter table Turma drop COLUMN Numero; /\*restrict\*/**



**alter table Turma drop COLUMN Numero cascade constraints;**





**alter table Matricula add constraint nota check (nota > 0);**

**alter table Disciplina drop constraint NCREDITOS;**

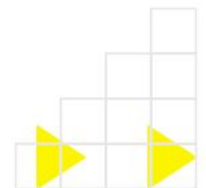
**alter table Aluno modify (CidadeOrigem default 'Sanca');**

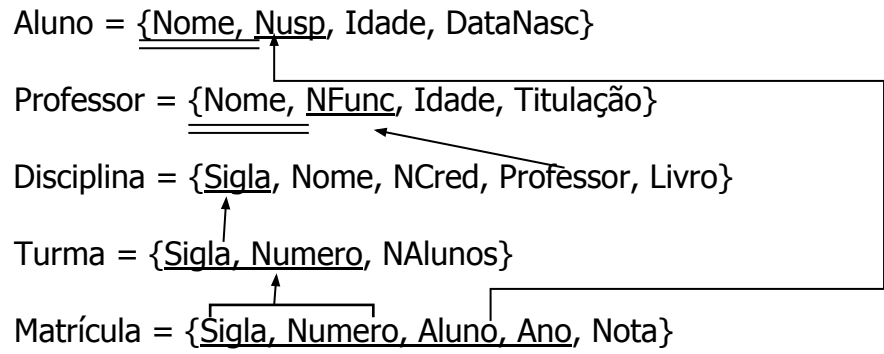
# Comandos DDL

- **DROP TABLE** - exclui uma tabela da base de dados

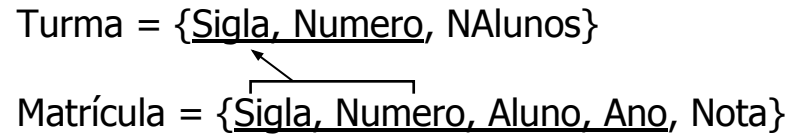
**DROP TABLE** *tabela* [**CASCADE** | **RESTRICT**];

- **CASCADE**: todas as visões e restrições que referenciam a tabela são removidas automaticamente (dados referenciados são mantidos)
- **RESTRICT**: a tabela é removida somente se não for referenciada em nenhuma restrição ou visão

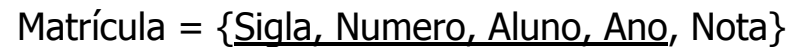




**drop table Turma; /\* restrict \*/**



**drop table Turma cascade constraints;**



# Usando o Sistema Oracle para praticar



## Opção 1

- + instalar o software SQLDeveloper
- + pedir instruções via e-mail ao tutor para conexão remota no servidor da USP

⇒ [Conexão via SQLDeveloper](#)

## Opção 2

- + baixar o software Oracle Express Edition e o SQLDeveloper
- + conectar no servidor localhost

⇒ [Oracle Express Edition](#)

## Opção 3

- + usar o Oracle via cloud

⇒ [Passo a passo para criar um database Oracle na nuvem](#)

## Opção 4

- + instalar o SGBD open-source PostgreSQL
- + adaptar o SQL visto para o SQL do PostgreSQL
  - > poucas alterações, na verdade

