

Nome:	Turma:
Professor:	Data: 15/02/2019

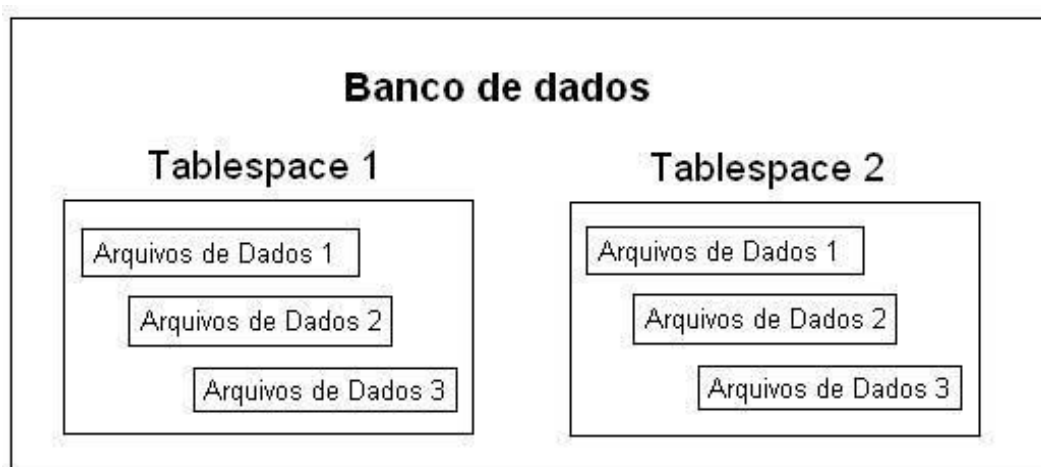
Prática01**Instruções:**

- ✓ A prática poderá ser realizada em dupla;
- ✓ Entregar um arquivo doc ou txt com os comandos de criação das tabelas.
- ✓ Entregar até 18/02/2019

Introdução ao SGBD Oracle**Primeiros passos****1. Introdução: Estrutura de armazenamento do Oracle**

O Oracle aloca espaço físico para o Banco de Dados, utilizando as seguintes definições, hierarquicamente organizadas:

- **Banco de Dados:** Uma coleção lógica de dados compartilhados armazenados em *tablespaces*.
- **Tablespace:** repositório lógico para dados fisicamente agrupados.
- **Arquivo de Dados:** arquivo de dados físico pertencendo a uma única *tablespace*.

**2. Criando uma tablespace no Oracle**

Para criar tablespaces no Oracle é necessário logar no SQL-Plus (ou outra ferramenta de gerenciamento do oracle) com um usuário administrador, por exemplo, o usuário SYSTEM ou o usuário SYS. As senhas desses usuários foram definidas durante a instalação do Oracle.

OBS: Você deve criar a pasta que será utilizada para armazenar o datafile.

Logado com o usuário SYSTEM ou SYS, executar o seguinte comando, para criar uma **tablespace**:

```
CREATE TABLESPACE TS_BD  
DATAFILE 'C:\BD\ts_bd.dbf' SIZE 1M  
AUTOEXTEND ON;
```

3.

Criando um usuário DBA e dando controle a uma determinada *tablespace*

Logado com o usuário SYSTEM ou SYS, executar os comandos abaixo:

Comando 1) Criar o usuário

```
CREATE USER USER_BD  
IDENTIFIED BY ALUNO  
DEFAULT TABLESPACE TS_BD  
TEMPORARY TABLESPACE TEMP  
QUOTA UNLIMITED ON TS_BD;
```

Comando 2) Conceder privilégio

```
GRANT DBA TO USER_BD WITH ADMIN OPTION;
```

4. Listando os objetos criados.

- Para listar as tabelas criadas pelo usuário logado, basta rodar o comando abaixo:

```
SELECT OBJECT_NAME  
FROM USER_OBJECTS
```

- Para listar a estrutura de uma tabela, utilize o comando *DESC*, como abaixo:

```
DESC NOME_DA_TABELA;
```

- Para listar as restrições criadas para cada tabela, utilize o comando abaixo:

```
SELECT CONSTRAINT_NAME, CONSTRAINT_TYPE, STATUS, SEARCH_CONDITION  
FROM USER_CONSTRAINTS  
WHERE TABLE_NAME = 'NOME_DA_TABELA';
```

5. Boas práticas de desenvolvimento

- Tabelas e colunas:

- Nomes das tabelas e colunas sempre em CAIXA ALTA
- Nomes sempre no singular (ALUNO, CURSO, etc)
- Colunas com chaves primárias sempre com um prefixo. Exemplos:
 - COD_ALUNO, CD_ALUNO, ID_ALUNO, NUM_ALUNO, etc.

- Constraints:

- Ao definir um nome para uma *constraint*, procure definir o nome de acordo com um padrão;
- Abaixo segue um exemplo de padrão para nomenclatura de *constraints*:
 - Sempre em CAIXA ALTA
 - SIGLA_ + NOME_TABELA + NOME_DA_COLUNA.
 - Exemplos:

UNIQUE KEY → UK_ALUNO_CPF

PRIMARY KEY → PK_ALUNO_COD_ALUNO ou apenas PK_ALUNO

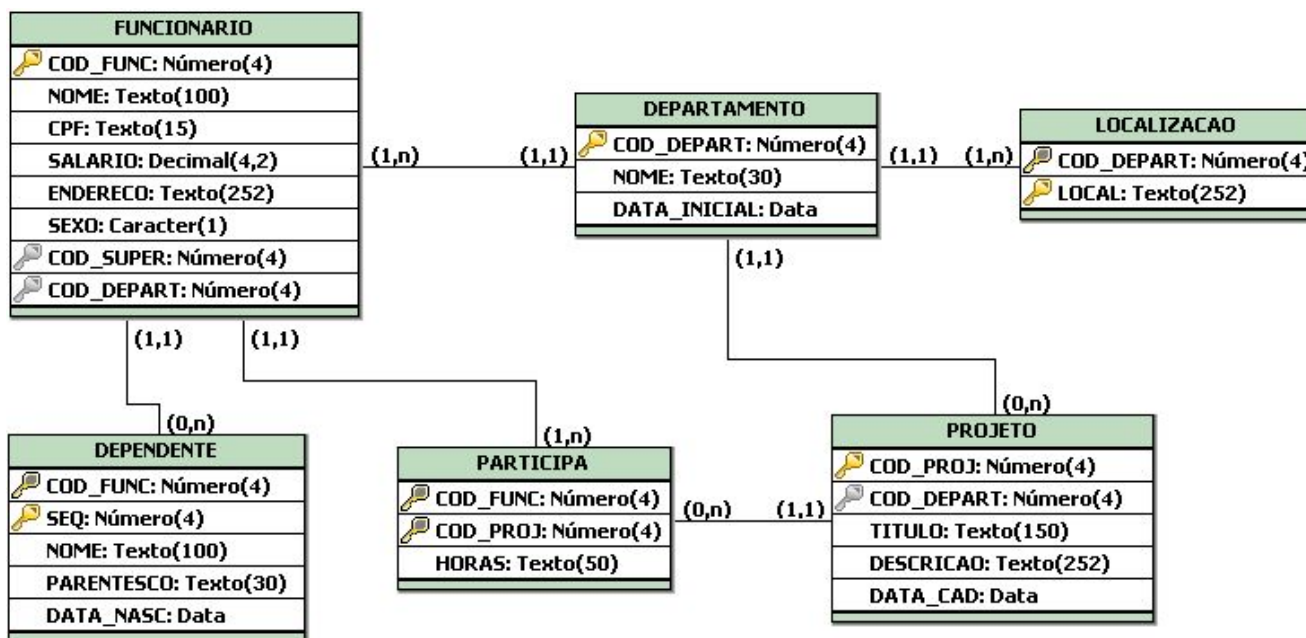
FOREIGN KEY → FK_ALUNO_CURSO

CHECK → CK_ALUNO_SEXO

6. Utilizando os conceitos apresentados em aulas, criar o esquema físico para o modelo lógico apresentado abaixo.

Observações:

- Deverão ser criadas as tabelas e *constraints*
- Logar no Oracle com o usuário criado no item 3: **USER_BD**



Regras de negócio:

FUNCIONARIO.NOME = Preenchimento obrigatório

FUNCIONARIO.CPF = Não pode haver duas instâncias com mesmo valor no campo CPF.

FUNCIONARIO.SALARIO = O salário dos funcionários não pode ser menor que 1000,00.

FUNCIONARIO.SEXO = Só aceitar os caracteres "M" ou "F".

FUNCIONARIO.COD_SUPER = coluna usada para definição do auto-relacionamento

DEPARTAMENTO.NOME = Preenchimento obrigatório

DEPENDENTE.NOME = Preenchimento obrigatório

DEPENDENTE.PARENTESCO = Preenchimento obrigatório

DEPENDENTE.PARENTESCO = Só aceitar valores dentro do conjunto ("PAI", "MÃE", "IRMAOS", "FILHO")

PROJETO.TITULO = Preenchimento obrigatório

PROJETO.DATA_CAD = Data de cadastro dos projetos com valor padrão igual à data do dia. A função que retorna a data do dia no oracle é **Sysdate**.