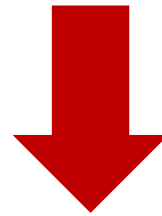


Prática em Banco de Dados

Carlos Baltazar & Henry Ungaro
Banco de Dados
UNIP

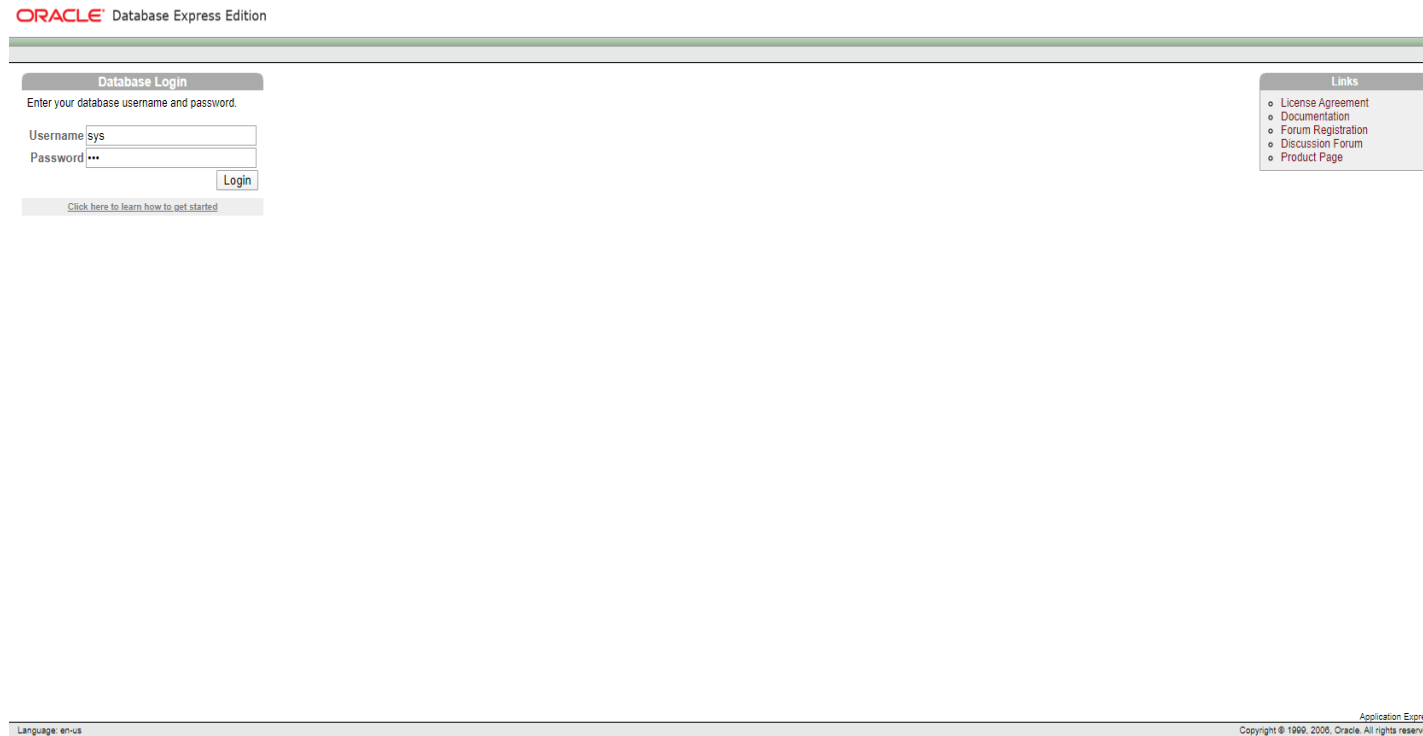
ORACLE 10g



<https://1drv.ms/u/s!Ai-HKwTL27hchMkOBIcc3P9Q6F88Ng?e=ftoFAd>

- Criando o usuário no ORACLE10g:

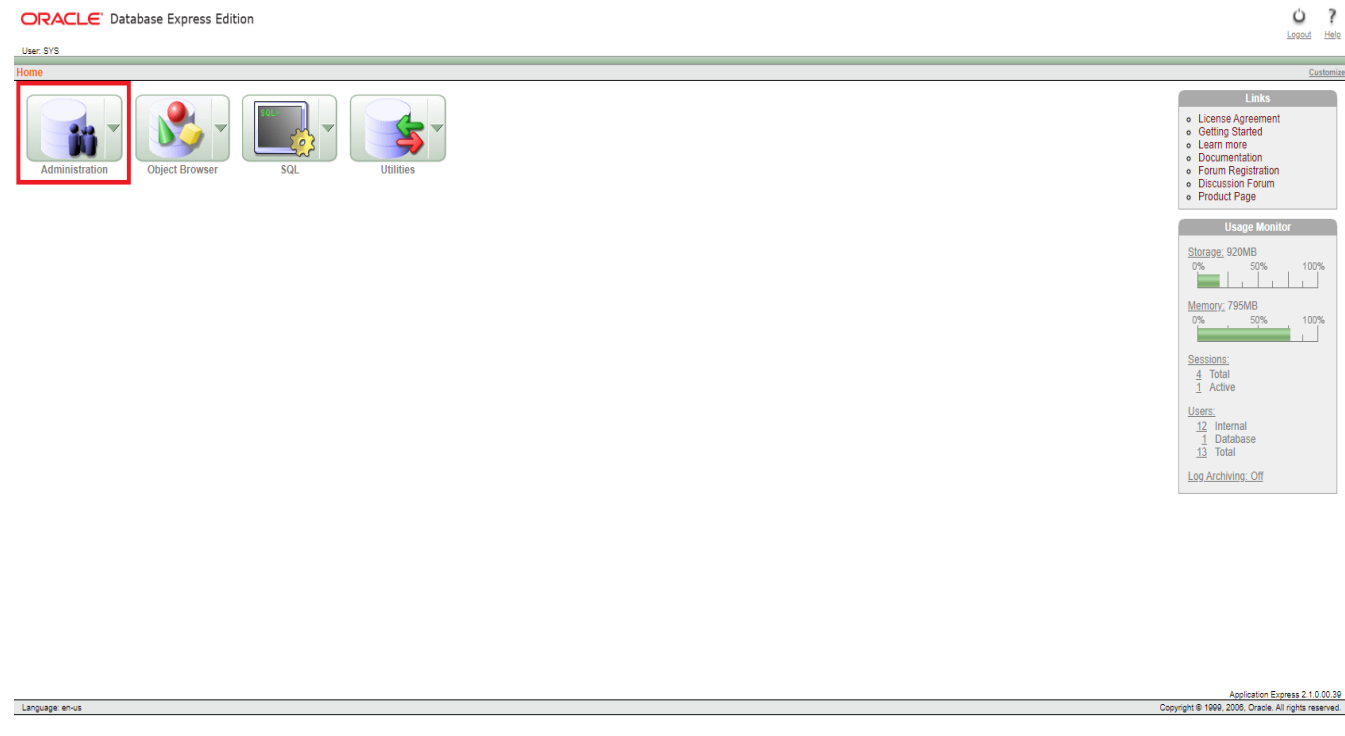
Acesse a homepage do SGBD e entre com as credenciais do usuário sys definidas no momento da instalação:



The screenshot shows the Oracle Database Express Edition login interface. At the top, it says "ORACLE Database Express Edition". Below this is a "Database Login" section with the instruction "Enter your database username and password." There are two input fields: "Username" with "sys" entered, and "Password" with masked characters. A "Login" button is to the right of the password field. Below the login fields is a link that says "Click here to learn how to get started". To the right of the login section is a "Links" box containing a list of links: "License Agreement", "Documentation", "Forum Registration", "Discussion Forum", and "Product Page". At the bottom of the page, there is a footer with "Language: en-us" on the left and "Application Express Copyright © 1999, 2006, Oracle. All rights reserved." on the right.

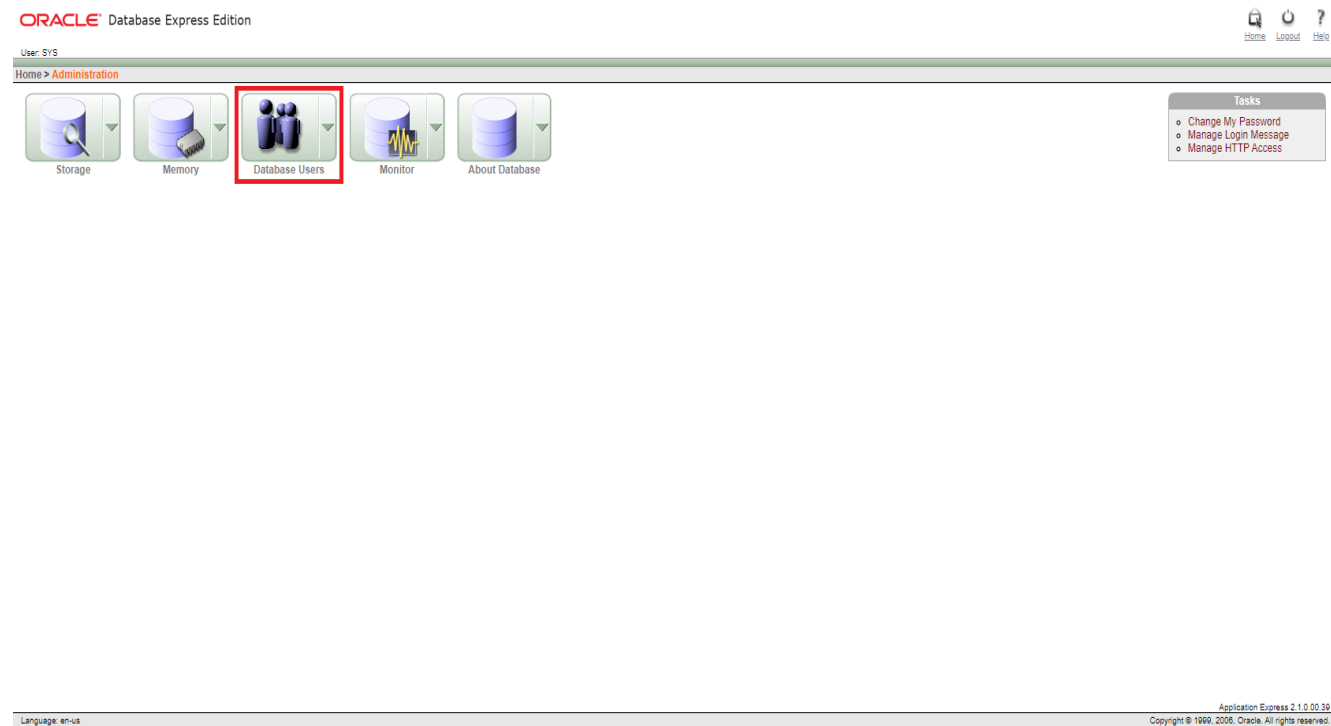
- Criando o usuário no ORACLE10g:

Acesse o painel de controle e selecione o painel de administração:



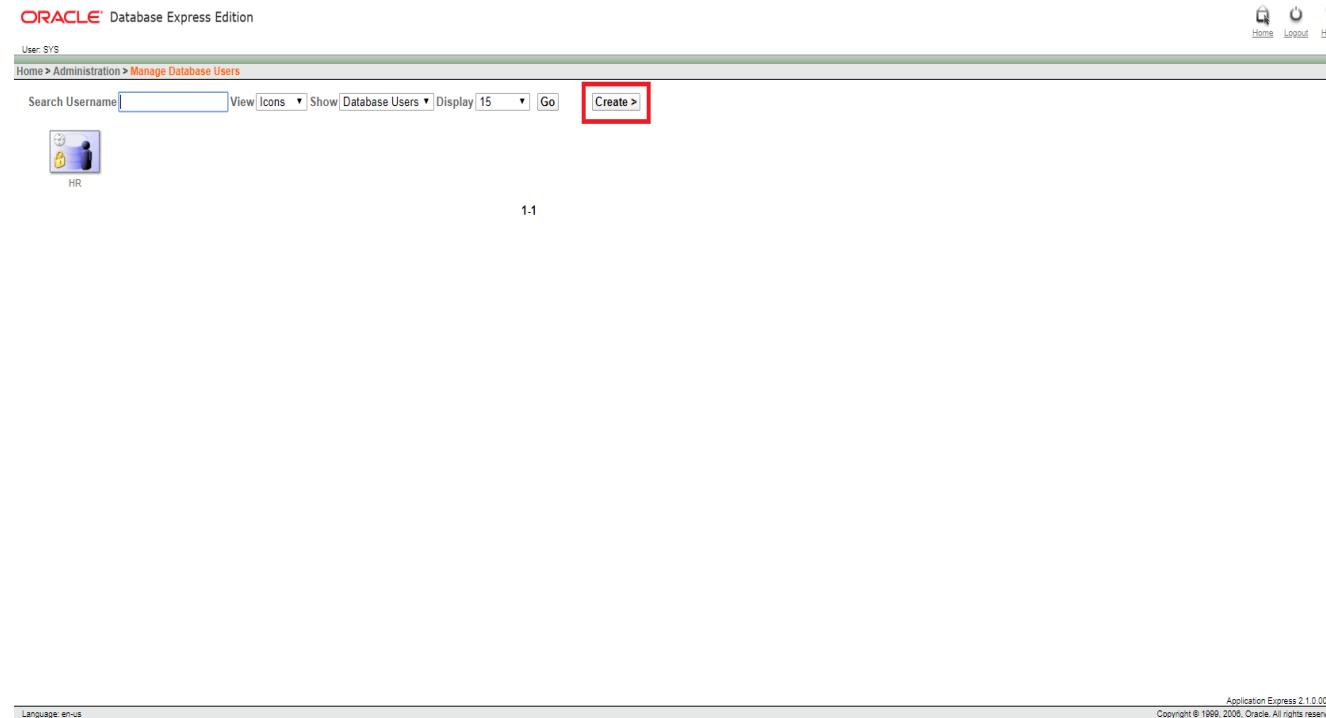
- Criando o usuário no ORACLE10g:

Acesse o painel de usuários do banco de dados:



- Criando o usuário no ORACLE10g:

Clique no botão criar:

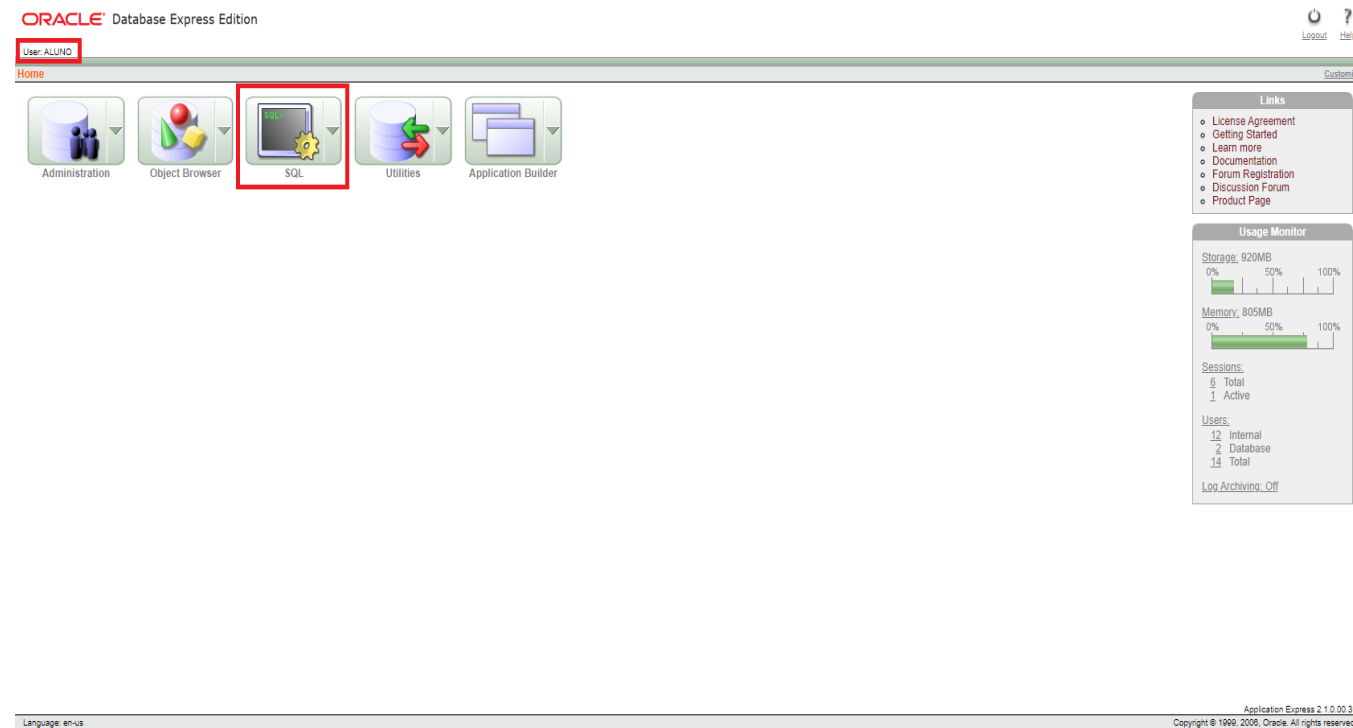


- Criando o usuário no ORACLE10g:

Crie um usuário aluno, com senha padrão 10g, selecione o status da conta como desbloqueado, marque a opção DBA e marque também todas as opções abaixo. Em seguida clique no botão criar:

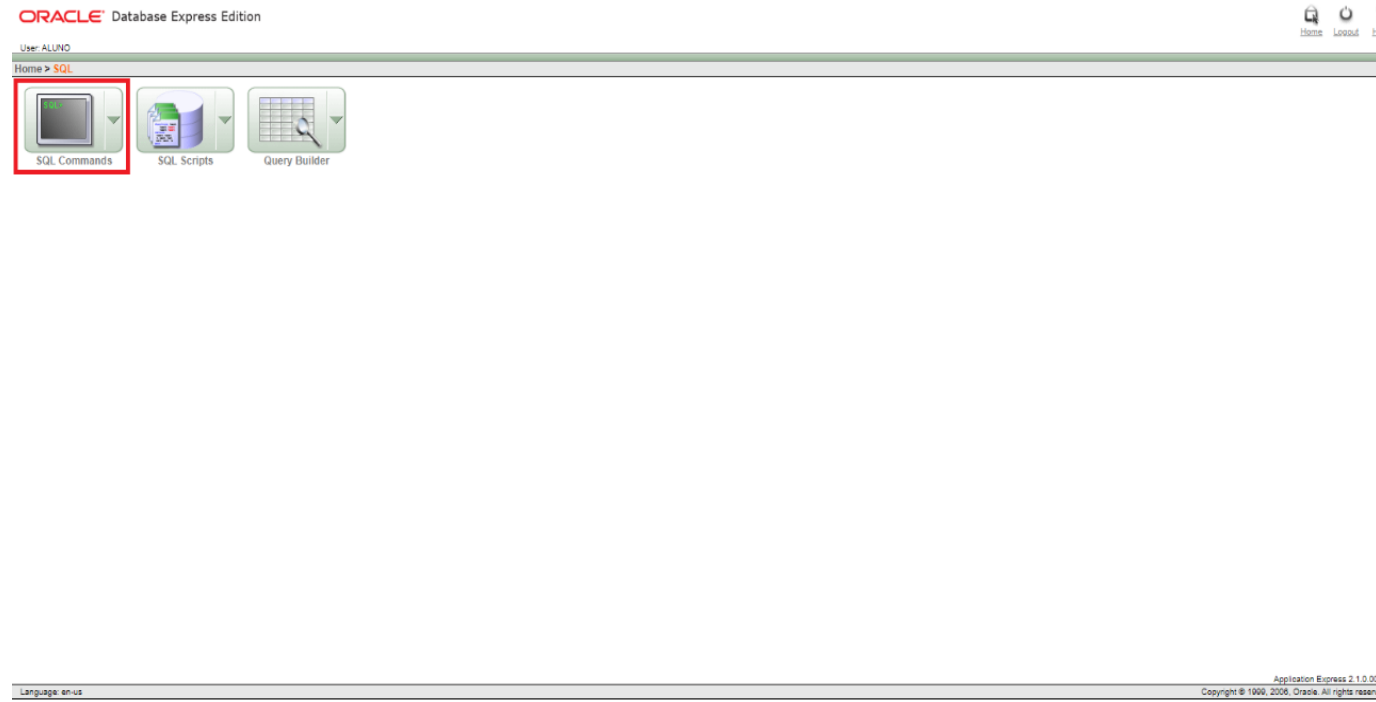
- Criando o usuário no ORACLE10g:

Em seguida saia da conta sys e faça login com a conta aluno, voltando para o painel inicial. Então acesse o painel SQL:



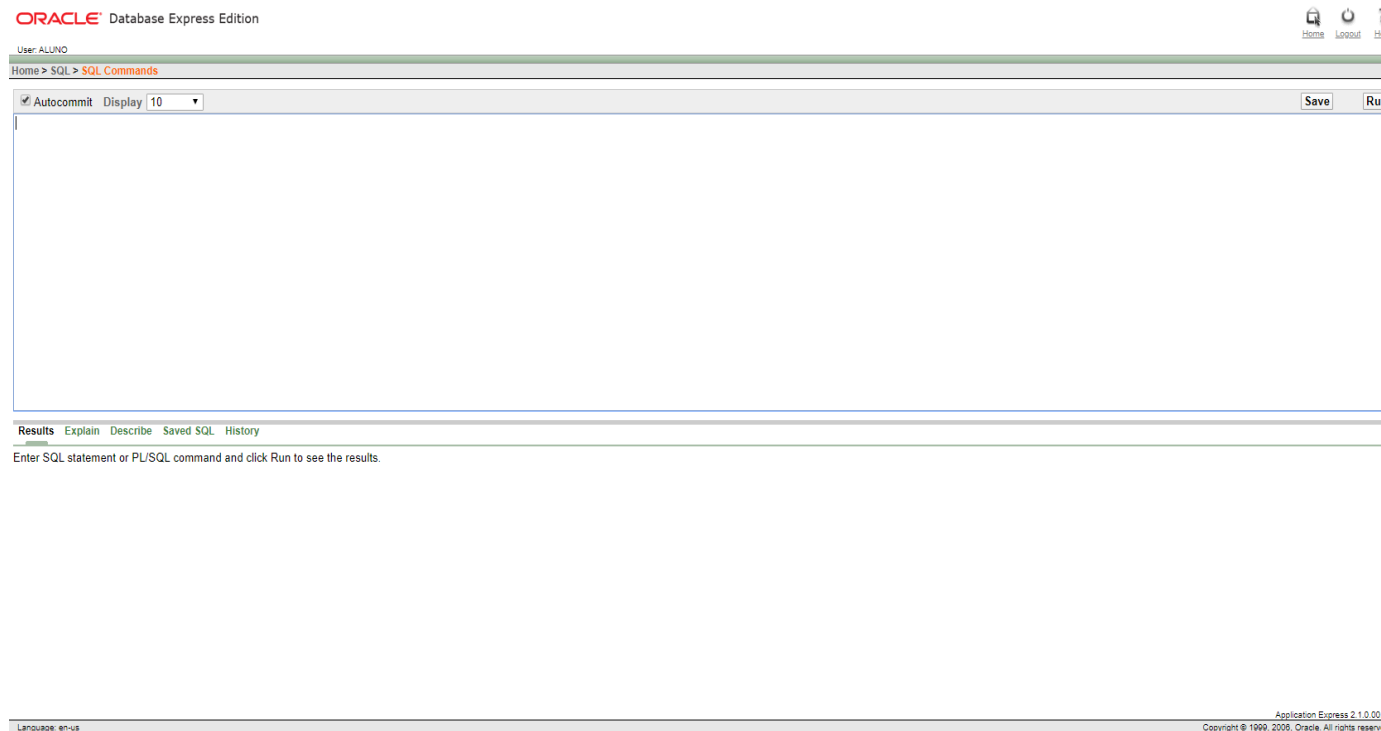
- Criando o usuário no ORACLE10g:

No painel SQL, acesse o painel Comandos SQL para acessar a IDE de programação em SQL no ORACLE:



- Criando o usuário no ORACLE10g:

Então iremos visualizar a IDE de programação em SQL, onde na parte superior escreveremos os comandos de no painel inferior serão exibidos os resultados:



Structured Query Language, ou Linguagem de Consulta Estruturada ou SQL, é a linguagem de pesquisa declarativa padrão para banco de dados relacional (base de dados relacional). Muitas das características originais do SQL foram inspiradas na álgebra relacional.

A linguagem é um grande padrão de banco de dados. Isto decorre da sua simplicidade e facilidade de uso. Ela se diferencia de outras linguagens de consulta a banco de dados no sentido em que uma consulta SQL especifica a forma do resultado e não o caminho para chegar a ele. Ela é uma linguagem declarativa em oposição a outras linguagens procedurais. Isto reduz o ciclo de aprendizado daqueles que se iniciam na linguagem.

Um mercado local decidiu informatizar sua lista de clientes e sua lista de estoque. Na lista de clientes deve ser capaz de armazenar dados como: nome; sobrenome; endereço; telefone; RG e CPF. Além disso, a lista de produtos deve ser capaz de armazenar: a descrição do produto; preço unitário e a quantidade em estoque.

Porém, antes de partimos para a implementação, é importante ressaltar que por mais que os vários SGBDs sejam relacionais e rodem SQL, a sintaxe pode acabar variando entre eles. Portanto, vamos nos acostumar com a sintaxe SQL do ORACLE antes de prosseguir.

Para começar, vamos definir os domínios que podemos utilizar traçando um paralelo com o que já estamos acostumados nas linguagens de programação:

Tipo (domínio)	Linguagem de programação	Banco de dados	Observações
Inteiro	Int	Number(n)	n: Algarismos inteiros
Real	Float/double	Number(n,m)	n: Algarismos inteiros m: Algarismos decimais
caractere	Char	Char(n)	n: quantidade de caracteres
Vetor de caracteres	Char[n]		
Texto	String	Varchar(n)	n: quantidade de caracteres
		Varchar2(n)	n: quantidade de caracteres Libera espaço não utilizado.

Então, para implementar tal banco de dados é necessário ir até a IDE do ORACLE10g e executar comandos de criação de tabela com seguinte estrutura:

```
CREATE TABLE NOME_DA_TABELA
```

```
(
```

```
    ATRIBUTO
```

```
    DOMÍNIO
```

```
    RESTRIÇÃO
```

```
);
```

Conforme aplicado nos exemplos abaixo:

```
CREATE TABLE CLIENTE_01
```

```
(  
    A01_NOME                VARCHAR2(50)  NOT NULL,  
    A01_SOBRENOME           VARCHAR2(50)  NOT NULL,  
    A01_ENDERECO             VARCHAR2(100) NOT NULL,  
    A01_TELEFONE             CHAR(15),  
    A01_RG                   CHAR(9)       NOT NULL,  
    A01_CPF                  CHAR(11)      NOT NULL UNIQUE  
);
```

Conforme aplicado nos exemplos abaixo:

```
CREATE TABLE PRODUTO_02
```

```
(
```

```
    A02_DESCRICAO                VARCHAR2(100) NOT NULL,
```

```
    A02_PRECO_UNITARIO           NUMBER(5,2)   NOT NULL,
```

```
    A02_ESTOQUE                  NUMBER(5)      NOT NULL
```

```
);
```


Caso desejemos inserir dados na tabela CLIENTE_01 utilizaremos o comando de inserção com a seguinte estrutura:

```
INSERT INTO NOME_DA_TABELA (ATRIBUTO1 ... ATRIBUTON) VALUES (VALOR1 ... VALORN);
```

Conforme os exemplos abaixo:

```
INSERT INTO CLIENTE_01 (A01_NOME, A01_SOBRENOME, A01_ENDERECO, A01_TELEFONE,  
A01_RG, A01_CPF) VALUES ("José", "Silva", "Rua São João", '+5511956326730', '167355819',  
'07167355819');
```

```
INSERT INTO PRODUTO_02 (A02_DESCRICAO, A02_PRECO_UNITARIO, A02_ESTOQUE) VALUES  
(‘Sabão em pó’, 24.99, 158);
```

Agora insira pelo menos mais quatro registros em cada tabela.

Caso desejemos consultar dados das tabelas criadas utilizaremos o comando de seleção com a seguinte estrutura:

```
SELECT * FROM nome_tabela;
```

Conforme os exemplos abaixo:

```
SELECT * FROM CLIENTE_01;
```

```
SELECT * FROM PRODUTO_02;
```

OBRIGADO