

Introdução

Objetivos do Curso

- Após concluir este curso, você poderá:
 - Identificar os principais componentes estruturais do banco de dados Oracle
 - Consultar dados de tabelas através do comando SELECT
 - Criar relatórios classificando e restringindo dados
 - Utilizar funções SQL para gerar e recuperar dados
 - Executar instruções DML (data manipulation language) para atualizar dados em tabelas no banco de dados Oracle
 - Obter informações através de consultas às views de dicionário de dados

I-2

Objetivo do Curso

Este curso tem como objetivo principal apresentar conceitos básicos e avançados da linguagem de programação SQL para banco de dados Oracle, permitindo criar consultas e manipular dados em uma ou mais tabelas, e ainda criar objetos de banco de dados. Serão apresentados também conceitos de bancos de dados relacionais e da tecnologia de banco de dados Oracle.

O curso foi desenvolvido com a versão Oracle 10G Release 2.

Objetivos deste Capítulo

- Ao concluir este capítulo, você poderá:
 - Discutir os aspectos teóricos e físicos de um banco de dados relacional
 - Efetuar login, editar comandos SQL, formatar a saída de comandos e interagir com scripts no SQL*Plus
 - Listar as diferenças entre comandos SQL e comandos do SQL*Plus

I-3

Objetivos deste Capítulo

Neste capítulo, você compreenderá alguns componentes da arquitetura do banco de dados Oracle. Também serão apresentadas informações sobre:

- Instruções SQL específicas do Oracle
- O SQL*Plus, que é um ambiente usado para executar instruções SQL e para fins de formatação e geração de relatórios

Necessidade de Informação

- Os Sistemas de Informação são ferramentas necessárias para a tomada de decisão das empresas
- Os Bancos de Dados são responsáveis pelo armazenamento de dados e provêm:
 - Acessibilidade
 - Disponibilidade
 - Segurança
 - Ferramentas de Gerenciamento

I-4

Necessidade de Informação

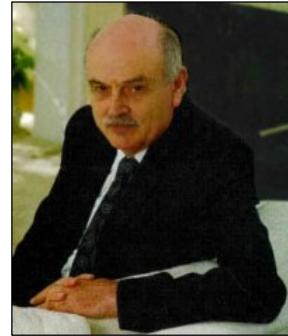
Todas as organizações possuem objetivos específicos e para decidir como atingir esses objetivos normalmente é necessária a manipulação de uma grande quantidade de informação. Porém, nem sempre essa informação está no formato correto quando ela é necessária.

Por isso, as empresas precisam dos Sistemas de Informação que são as ferramentas que modelam o mundo real.

Os Sistemas de Gerenciamento de Bancos de Dados são responsáveis pelo armazenamento das informações dos Sistemas de Informação que provêm acessibilidade, disponibilidade, segurança e ferramentas de gerenciamento.

Conceito de Banco de Dados Relacional

- O modelo relacional para sistemas de gerenciamento de banco de dados foi proposto pelo Dr. Edgar Frank Codd (1923 - 2003) em um artigo em 1970.
- Esse modelo é a base para o RDBMS
- O modelo relacional é composto de:
 - Uma coleção de tabelas relacionadas
 - Operadores que podem agir sobre as relações
 - Ferramentas de integridade de dados



I-5

Modelo Relacional

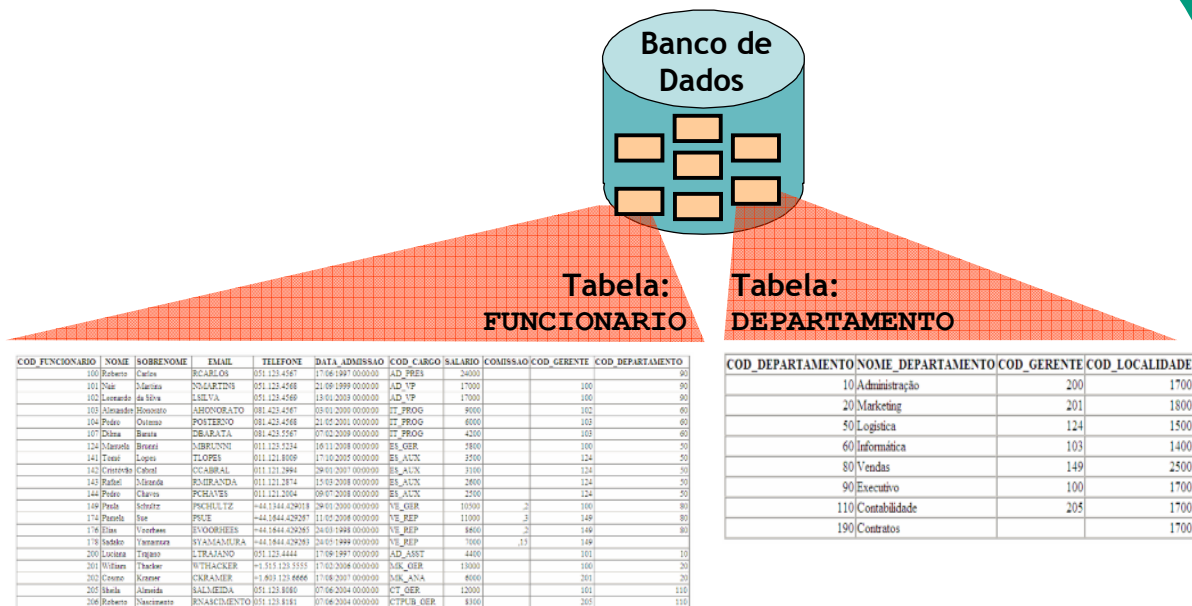
O conceito de um banco de dados relacional foi desenvolvido originalmente pelo Dr. Edgar Frank Codd. Ele descreveu a teoria dos bancos de dados relacionais em um trabalho chamado "A Relational Model of Data for Large Shared Data Banks" em junho de 1970. Um ano antes, ele havia publicado a mesma teoria em um documento interno na IBM. Porém, para seu desapontamento, a IBM só adotou o modelo relacional quando empresas rivais começaram a utilizá-la no desenvolvimento de produtos comerciais.

Os modelos comuns usados na época eram o hierárquico e o de rede, ou até mesmo estruturas de dados simples de flat files. O RDBMS (Relational Database Management System) tornou-se muito popular, especialmente pela facilidade de uso e flexibilidade em termos de estrutura. Além disso, vários fornecedores inovadores, como a Oracle, complementaram o RDBMS com um conjunto de produtos eficientes para usuários e desenvolvedores de aplicações, que compunham uma solução integral.

Os conceitos básicos do modelo relacional são fáceis de entender. Um banco de dados relacional é uma coleção de informações relacionadas que são organizadas em tabelas. Cada tabela armazena os dados na forma de registros ou linhas, e os dados são organizados em colunas. As tabelas são armazenadas em esquemas de banco de dados, que consistem de áreas onde os usuários guardam suas próprias tabelas. Um usuário pode dar permissão para outro acessar suas tabelas.

Definição de um Banco de Dados Relacional

- Um banco de dados relacional é um conjunto tabelas de duas dimensões relacionadas.



I-6

Definição de um Banco de Dados Relacional

Um banco de dados relacional usa tabelas de duas dimensões para armazenar informações, no jargão de banco de dados essas tabelas são comumente referidas como entidades.

Para cada entidade existe um conjunto de propriedades relevantes que são as colunas, normalmente chamadas de atributos.

Por exemplo, é possível armazenar informações sobre todos os produtos de uma loja. Em um banco de dados relacional, você cria diversas tabelas para armazenar diferentes informações sobre os produtos, como uma tabela de produtos, uma tabela de tipos de produtos e uma tabela de vendas.

Termos Usados em Banco de Dados Relacional

2

4

COD_FUNCIONARIO	NOME	SOBRENOME	COD_CARGO	SALARIO	COMISSAO	COD_GERENTE	COD_DEPARTAMENTO
100	Roberto	Carlos	AD_PRES	24000			90
101	Nair	Martins	AD_VP	17000		100	90
102	Leonardo	da Silva	AD_VP	17000		100	90
103	Alexandre	Herbert	IT_PROG	9000		102	60
104	Pedro	Ostrowski	IT_PROG	6000		103	60
107	Dilma	Barata	IT_PROG	4200		103	60
124	Manuela	Brunni	ES_GER	5800		100	50
141	Tomé	Lopes	ES_AUX	3500		124	50
142	Cristóvão	Cabral	ES_AUX	3100		124	50
143	Rafael	Miranda	ES_AUX	2600		124	50
144	Pedro	Chaves	ES_AUX	2500		124	50
149	Paula	Schultz	VE_GER	10500	.2	100	80
174	Pamela	Sue	VE_REP	11000	.3	149	80
176	Elias	Voorhees	VE_REP	8600	.2	149	80
178	Sadako	Yamamura	VE_REP	7000	.15	149	
200	Luciana	Trajano	AD_ASST	4400		101	10
201	William	Thacker	MK_GER	13000		100	20
202	Cosmo	Kramer	MK_ANA	6000		201	20
205	Sheila	Almeida	CT_GER	12000		101	110
206	Roberto	Nascimento	CTPUB_GER	8300		205	110

1

3

5

6

I-7

I-7

Termos Usados em Banco de Dados Relacional

Uma tabela é a estrutura básica de armazenamento de um Banco de Dados Relacional. Ela pode conter todos os dados necessários sobre algo relativo ao mundo real, como produtos, funcionários ou clientes.

Uma tabela é a estrutura básica de armazenamento de um Banco de Dados Relacional. Ela pode conter todos os dados necessários sobre algo relativo ao mundo real, como produtos, funcionários ou clientes.

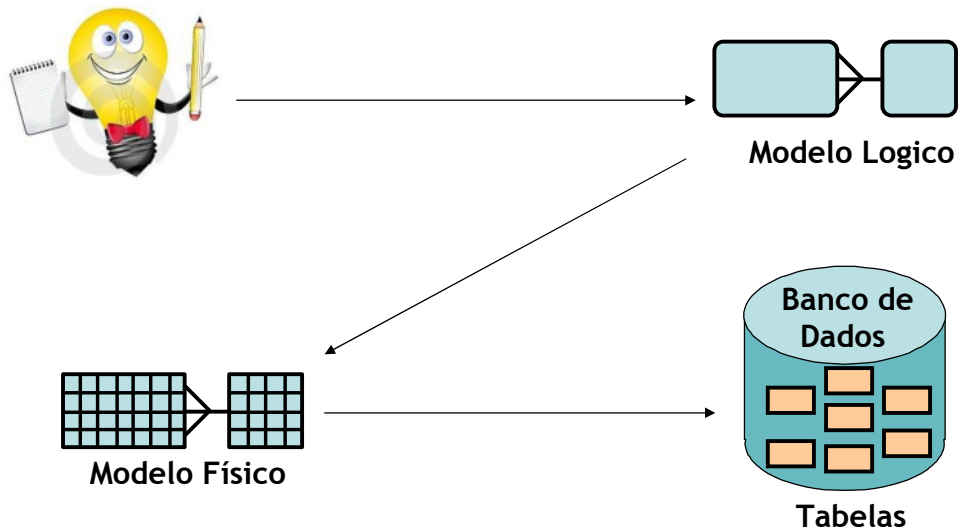
O slide mostra o conteúdo da tabela PRODUTO. Os números indicam:

1. Uma linha única (também chamada de tupla), que contém os dados representativos da entidade relacionada, no exemplo um produto. A ordem das linhas não é importante e o Oracle não garante a ordem de armazenamento dos dados para tabelas comuns. Quanto se deseja os dados numa ordem específica, essa ordem deve ser explicitada no comando de recuperação de dados.
2. A coluna, campo ou atributo que contém o código do produto. O código do produto identifica unicamente um produto na tabela PRODUTO. No exemplo, essa coluna é a chave primária da tabela. Por definição, chave primária é a coluna ou conjunto de colunas que devem sempre conter um valor e não podem se repetir. Por padrão deve-se ter uma chave primária em todas as tabelas de bancos de dados relacionais, pois a chave primária é a coluna que identifica o registro e impede que se tenham linhas duplicadas.

Termos Usados em Banco de Dados Relacional (continuação)

3. Uma coluna que não é um valor-chave. Uma coluna representa um tipo de dados em uma tabela; no exemplo, os dados representam os salários de todos os funcionários. A ordem das colunas não é importante durante o armazenamento de dados; especifique essa ordem quando os dados forem recuperados.
4. Uma coluna com o tipo do produto, que também é uma chave estrangeira. Uma chave estrangeira é uma coluna que define o relacionamento entre tabelas. Uma chave estrangeira faz referência a uma chave primária ou exclusiva na mesma tabela ou em outra. No exemplo, COD_TIPO_PRODUTO identifica com exclusividade um tipo de produto na tabela PRODUTO.
5. A localização de um campo é a interseção entre uma linha e uma coluna. Um campo só pode conter um valor.
6. Um campo pode não conter um valor. Nesse caso, seu valor é nulo, ou seja, não existe valor para esse campo

Modelagem de Dados



I-9

Modelagem de Dados

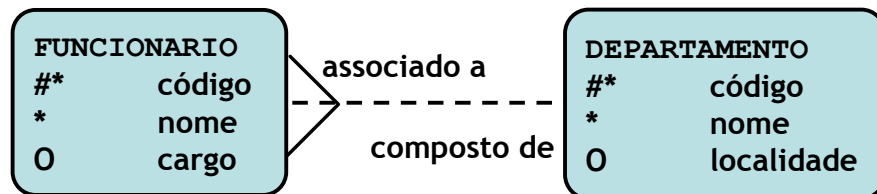
Modelagem de dados é a atividade de especificação das estruturas de dados e regras de negócio necessárias para o desenvolvimento de um sistema de informação que irá suportar uma determinada área de negócio da empresa.

Os modelos constituem a base do projeto. Os analistas de sistemas desenvolvem modelos para explorar idéias e compreender melhor o projeto do banco de dados.

Os modelos também são utilizados para ajudar na documentação dos sistemas de informação. É fundamental que se produza um modelo que possa ser compreendido por um usuário final e contenha detalhes suficientes para que um desenvolvedor crie um sistema de banco de dados.

Modelo Entidade-Relacionamento

- A Entidade é a representação de um objeto do mundo real no qual se deseja manter informações
- O Relacionamento é a representação de uma associação entre 2 ou mais entidades.



- Cenário
 - "... associe um ou mais funcionários a um departamento. . ."
 - "... alguns funcionários ainda não foram atribuídos a um departamento. . ."

I-10

Modelo Entidade-Relacionamento (ER)

É a técnica diagramática para representar um modelo conceitual. Serve para visualizar o relacionamento de tabelas de um banco de dados, no qual as relações são construídas através da associação de um ou mais atributos destas tabelas.

O modelo ER documenta as informações de maneira clara e de fácil leitura, possibilitando assim, que as informações sejam reutilizadas em diversos projetos.

Os principais componentes do modelo ER são:

- Entidade: representação de um objeto do mundo real no qual se deseja manter informações.
- Atributo: características das entidades que se deseja armazenar. Os atributos podem ser obrigatórios ou opcionais.
- Relacionamento: associação entre e ou mais entidades através de atributos específicos das entidades.

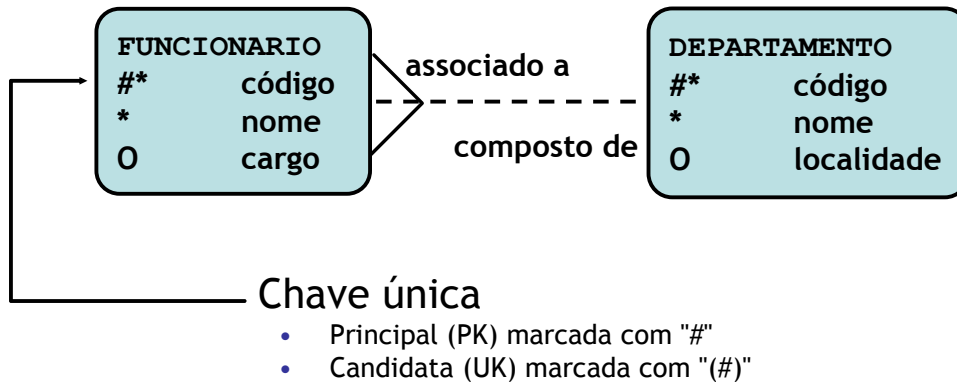
Convenções do Modelo ER

Entidade

- Nome único no esquema, singular
- Letras maiúsculas
- Sinônimo entre parênteses

Atributo

- Nome singular
- Letras minúsculas
- Obrigatório marcado com *
- Opcional marcado com "o"



I-11

Convenções da Modelagem ER

Entidades

Para representar uma entidade em um modelo, use as seguintes convenções:

- Nome de entidade único no esquema, singular
- Nome de entidade em maiúsculas
- Nomes de sinônimos opcionais em maiúsculas entre parênteses: ()

Atributos

Para representar um atributo em um modelo, use as seguintes convenções:

- Nome singular em minúsculas
- Tag de asterisco (*) para atributos obrigatórios
- Tag de letra "o" para atributos opcionais

Relacionamentos

Símbolo	Descrição
Linha tracejada	Elemento opcional que indica probabilidade
Linha sólida	Elemento necessário que indica obrigatoriedade
Pé-de-galinha	Elemento de grau que indica um ou mais
Linha única	Elemento de grau que indica apenas um

Relacionamento Entre Tabelas

- A chave primária (PK) identifica unicamente uma linha de uma tabela
- Os dados podem ser relacionados por meio de chaves estrangeiras (FKs).

Tabela: FUNCIONARIO

COD_FUNCIONARIO	NOME	SOBRENOME	COD_CARGO	SALARIO	COMISSAO	COD_GERENTE	COD_DEPARTAMENTO
100	Roberto	Carlos	AD_PRES	24000			90
101	Nair	Martins	AD_VP	17000		100	90
102	Leonardo	da Silva	AD_VP	17000		100	90
103	Alexandre	Honorato	IT_PROG	9000		102	60
104	Pedro	Osterno	IT_PROG	6000		103	60
107	Dina	Barata	IT_PROG	4200		103	60
124	Manuela	Brizzi	ES_GER	5800		100	50
141	Tomé	Lopes	ES_AUX	3500		124	50
142	Cristóvão	Cabral	ES_AUX	3100		124	50
143	Rafael	Miranda	ES_AUX	2600		124	50
144	Pedro	Chaves	ES_AUX	2500		124	50
149	Paula	Schultz	VE_GER	10500	.2	100	80
174	Pamela	Sue	VE_REP	11000	.3	149	80
176	Elias	Voorhees	VE_REP	8600	.2	149	80
178	Sadako	Yamamura	VE_REP	7000	.15	149	80
200	Luciana	Trajano	AD_ASST	4400		101	10
201	William	Thacker	MK_GER	13000		100	20
202	Cosmo	Kruser	MK_ANA	6000		201	20
205	Sheila	Almeida	CT_GER	12000		101	110
206	Roberto	Nascimento	CTPUB_GER	8300		205	110

Chave primária

Chave estrangeira

Tabela: DEPARTAMENTO

COD_DEPARTAMENTO	NOME_DEPARTAMENTO	COD_GERENTE	COD_LOCALIDADE
10	Administração	200	1700
20	Marketing	201	1800
50	Logística	124	1500
60	Informática	103	1400
80	Vendas	149	2500
90	Executivo	100	1700
110	Contabilidade	205	1700
190	Contratos		1700

Chave primária

I-12

Relacionamento Entre Tabelas

Muitas vezes é necessário combinar duas ou mais tabelas para responder a uma determinada pergunta. Por exemplo, talvez você queira saber qual nome do departamento de um determinado funcionário. Nesse caso, você precisará de informações da tabela FUNCIONARIO (que contém dados sobre os funcionários) e da tabela DEPARTAMENTO (que contém informações sobre os departamentos). Um Banco de Dados Relacional permite relacionar os dados de uma tabela aos de outra por meio das chaves estrangeiras. Uma chave estrangeira é uma coluna (ou um conjunto de colunas) que faz referência a uma chave primária ou candidata na mesma tabela ou em outra tabela.

Você pode usar a capacidade de relacionar dados de uma tabela a dados de outra para organizar informações em unidades separadas. É possível manter os dados sobre produtos logicamente separados dos dados sobre tipos de produtos armazenando estes últimos em outra tabela.

Visão Geral do SQL

- SQL é linguagem de consulta, manipulação e controle utilizada para interagir com o banco de dados
- SQL é a abreviação de (Structured Query Language) e pode ser pronunciada em inglês como na palavra SEQUEL
- A linguagem SQL pode ser usada de maneira interativa ou embutida
 - Interativa: Digitação de comandos para interpretação pelo SGBD
 - Embutida: Comandos presentes dentro das aplicações
- É dividida em 4 categorias:
 - Consulta
 - Manipulação de dados (DML - Data Manipulation Language)
 - Definição de dados (DDL - Data Definition Language)
 - Controle de dados (DCL - Data Control Language)

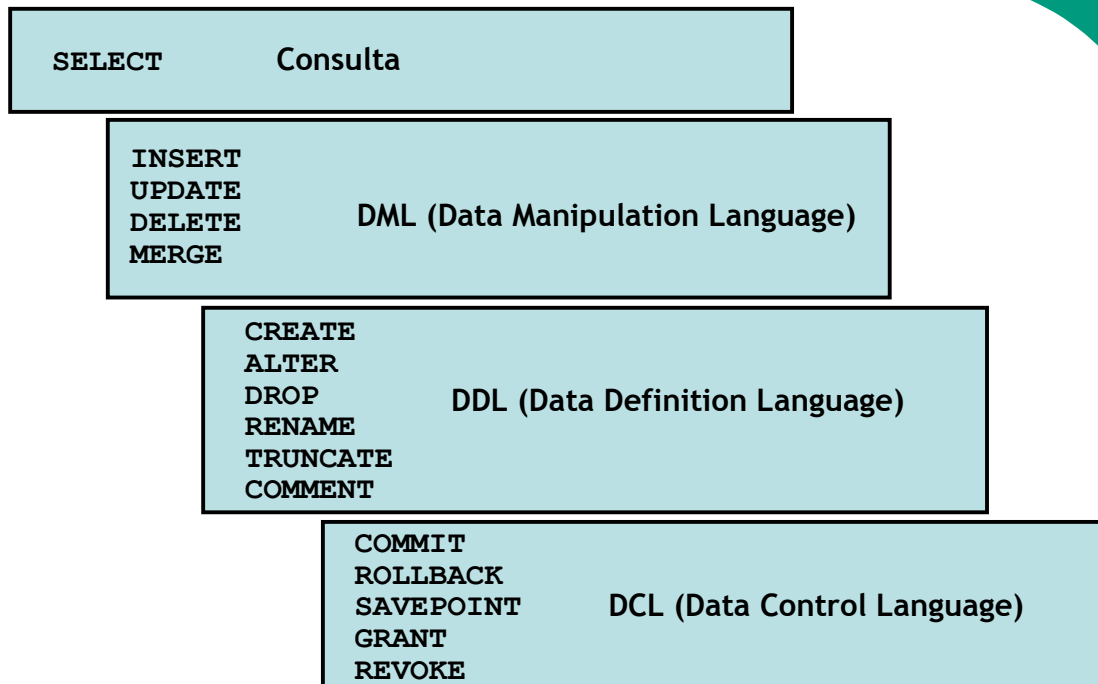
I-13

Visão Geral da Linguagem SQL

SQL é a abreviação de (Structured Query Language) e é uma linguagem que pode ser utilizada de duas maneiras: interativa ou embutida. Usar SQL de maneira interativa significa digitar comandos SQL ou executar scripts diretamente através de uma ferramenta cliente de banco de dados e receber os resultados na tela. Usar o SQL de maneira embutida envolve a incorporação dos comandos SQL em unidades de programas que podem ser criadas com qualquer linguagem de programação (por exemplo, Java ou Visual Basic). Neste curso trataremos apenas do uso de SQL de maneira interativa, porém todos os conceitos aqui descritos são válidos para o uso do SQL nas ferramentas de programação.

A linguagem SQL é eficiente, fácil de aprender e usar e possui toda a funcionalidade necessária para interagir com o banco de dados, pois permite definir, recuperar e manipular os dados das tabelas.

Tipos de Comandos SQL



I-14

Tipos de Comandos SQL

Existem 4 categorias de comandos SQL:

- Consulta: Comando SELECT.
- Manipulação de dados (DML - Data Manipulation Language): Comandos INSERT, UPDATE, DELETE, MERGE.
- Definição de dados (DDL - Data Definition Language) - Comandos CREATE, ALTER, DROP, RENAME, TRUNCATE, COMMENT.
- Controle de dados (DCL - Data Control Language) - Comandos COMMIT, ROLLBACK, SAVEPOINT, GRANT, REVOKE.

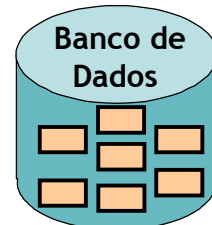
Comunicando-se de Maneira Interativa Através de SQL

Comando SQL é digitado.

```
SELECT nome_departamento  
FROM departamento;
```

NOME_DEPARTAMENTO
Administração
Marketing
Logística
Informática
Vendas
Executivo
Contabilidade
Contratos

Comando é enviado ao
Banco de Dados.



Comando é processado e
os dados são retornados
para o usuário.

I-15

Comunicando-se de Maneira Interativa Através de SQL

O procedimento para uso da linguagem SQL para obter dados do banco de dados é o seguinte:

1. Através de uma ferramenta cliente, como por exemplo o SQL*Plus, o usuário digita o comando SQL para obter os dados do banco de dados.
2. O comando é enviado para o banco de dados.
3. O Servidor de Banco de Dados processa o comando, obtém os dados das tabelas armazenadas no Banco de Dados e retorna o resultado para o usuário.
4. O resultado do comando é exibido na tela.

Tabelas Usadas neste Curso

FUNCIONARIO

COD_FUNCIONARIO	NOME	SOBRENOME	EMAIL	TELEFONE	DATA_ADMISSAO	COD_CARGO	SALARIO	COMISSAO	COD_GERENTE	COD_DEPARTAMENTO
100	Roberto	Carlos	RCARLOS	051.123.4567	17/06/1997 00:00:00	AD_PRES	24000			90
101	Nair	Martins	NMARTINS	051.123.4568	21/09/1999 00:00:00	AD_VP	17000		100	90
102	Leonardo	da Silva	LSILVA	051.123.4569	13/01/2003 00:00:00	AD_VP	17000		100	90
103	Alexandre	Honorato	AHONORATO	081.423.4567	03/01/2000 00:00:00	IT_PROG	9000		102	60
104	Pedro	Osterno	POSTERNO	081.423.4568	21/05/2001 00:00:00	IT_PROG	6000		103	60
107	Dilma	Barata	DBARATA	081.423.5567	07/02/2009 00:00:00	IT_PROG	4200		103	60
124	Manuela	Brunni	MBRUNNI	011.123.5234	16/11/2008 00:00:00	ES_GER	5800		100	50
141	Tomé	Lopes	TLOPES	011.121.8009	17/10/2005 00:00:00	ES_AUX	3500		124	50
142	Cristóvão	Cabral	CCABRAL	011.121.2994	29/01/2007 00:00:00	ES_AUX	3100		124	50
143	Rafael	Miranda	RMIRANDA	011.121.2874	15/03/2008 00:00:00	ES_AUX	2600		124	50
144	Pedro	Chaves	PCHAVES	011.121.2004	09/07/2008 00:00:00	ES_AUX	2500		124	50
149	Paula	Schultz	PSCHULTZ	+44.1344.429018	29/01/2000 00:00:00	VE_GER	10500	.2	100	80
174	Pamela	Sue	PSUE	+44.1644.429267	11/05/2006 00:00:00	VE_REP	11000	.3	149	80
176	Elias	Voorhees	EVOORHEES	+44.1644.429265	24/03/1998 00:00:00	VE_REP	8600	.2	149	80
178	Sadako	Yamamura	SYAMAMURA	+44.1644.429263	24/05/1999 00:00:00	VE_REP	7000	.15	149	80
200	Luciana	Trajan	LTRAJANO	051.123.4444	17/09/1997 00:00:00	AD_ASST	4400		101	10
201	William	Thacker	WTHACKER	+1.515.123.5555	17/02/2006 00:00:00	MK_GER	13000		100	20
202	Cosmo	Kramer	CKRAMER	+1.603.123.6666	17/08/2007 00:00:00	MK_ANA	6000		201	20
205	Sheila	Almeida	SALMEIDA	051.123.8080	07/06/2004 00:00:00	CT_GER	12000		101	110
206	Roberto	Nascimento	RNASCIMENTO	051.123.8181	07/06/2004 00:00:00	CTPUB_GER	8300		205	110

DEPARTAMENTO

COD_DEPARTAMENTO	NOME_DEPARTAMENTO	COD_GERENTE	COD_LOCALIDADE
10	Administração	200	1700
20	Marketing	201	1800
50	Logística	124	1500
60	Informática	103	1400
80	Vendas	149	2500
90	Executivo	100	1700
110	Contabilidade	205	1700
190	Contratos		1700

I-16

Tabelas Usadas neste Curso

Abaixo segue a descrição das tabelas usadas neste curso:

SQL> DESCRIBE funcionario

Nome	Nulo?	Tipo

COD_FUNCIONARIO	NOT NULL	NUMBER (6)
NOME		VARCHAR2 (20)
SOBRENOME	NOT NULL	VARCHAR2 (25)
EMAIL	NOT NULL	VARCHAR2 (25)
TELEFONE		VARCHAR2 (20)
DATA_ADMISSAO	NOT NULL	DATE
COD_CARGO	NOT NULL	VARCHAR2 (10)
SALARIO		NUMBER (8,2)
COMISSAO		NUMBER (2,2)
COD_GERENTE		NUMBER (6)
COD_DEPARTAMENTO		NUMBER (4)

SQL> DESCRIBE departamento

Nome	Nulo?	Tipo

COD_DEPARTAMENTO	NOT NULL	NUMBER (4)
NOME_DEPARTAMENTO	NOT NULL	VARCHAR2 (30)
COD_GERENTE		NUMBER (6)
COD_LOCALIDADE		NUMBER (4)

Tabelas Usadas neste Curso

LOCALIDADE

COD_LOCALIDADE	ENDERECO	CEP	CIDADE	ESTADO	COD_PAIS
1400	Rua da Aurora 174	50060-010	Recife	PE	BR
1500	Rua Haddock Lobo 1048	01414-000	São Paulo	SP	BR
1700	Rua Duque de Caxias 1187	90010-282	Porto Alegre	RS	BR
1800	460 Bloor St. W.	ON M5S 1X8	Toronto	Ontário	CA
2500	Magdalen Centre, The Oxford Science Park	OX9 9ZB	Oxford	Oxford	UK

PAIS

COD_PAIS	NOME_PAIS	COD_REGIAO
BR	Brasil	2
CA	Canadá	2
UK	Reino Unido	1

REGIAO

COD_REGIAO	NOME_REGIAO
1	Europa
2	Américas
3	Ásia
4	Oriente Médio e África

I-17

Tabelas Usadas neste Curso (continuação)

```
SQL> DESCRIBE localidade
```

```
Nome          Nulo?      Tipo
```

```
-----
```

```
COD_LOCALIDADE NOT NULL NUMBER(4)
ENDERECO        VARCHAR2(40)
CEP              VARCHAR2(12)
CIDADE          NOT NULL VARCHAR2(30)
ESTADO          VARCHAR2(25)
COD_PAIS        CHAR(2)
```

```
SQL> DESCRIBE pais
```

```
Nome          Nulo?      Tipo
```

```
-----
```

```
COD_PAIS      NOT NULL CHAR(2)
NOME_PAIS     VARCHAR2(40)
COD_REGIAO    NUMBER
```

```
SQL> DESCRIBE regiao
```

```
Nome          Nulo?      Tipo
```

```
-----
```

```
COD_REGIAO    NOT NULL NUMBER
NOME_REGIAO   VARCHAR2(25)
```

Tabelas Usadas neste Curso

CARGO

COD_CARGO	NOME_CARGO	MENOR_SALARIO	MAIOR_SALARIO
AD_PRES	Presidente	20000	40000
AD_VP	Vice Presidente Administrativo	15000	30000
AD_ASST	Assistente Administrativo	3000	6000
CT_GER	Gerente de Contas	8200	16000
CTPUB_GER	Gerente de Contas Publicas	4200	9000
VE_GER	Gerente de Vendas	10000	20000
VE_REP	Representante de Vendas	6000	12000
ES_GER	Gerente de Estoque	5500	8500
ES_AUX	Auxiliar de Estoque	2000	5000
LO_AUX	Auxiliar de Logistica	2500	5500
IT_PROG	Programador	4000	10000
MK_GER	Gerente de Marketing	9000	15000
MK_ANA	Analista de Marketing	4000	9000

FAIXA_SALARIO

COD_FAIXA	MENOR_SALARIO	MAIOR_SALARIO
A	1000	2999
B	3000	5999
C	6000	9999
D	10000	14999
E	15000	24999
F	25000	40000

HISTORICO_CARGO

COD_FUNCIONARIO	DATA_INICIO	DATA_FIM	COD_CARGO	COD_DEPARTAMENTO
102	13/01/2003 00:00:00	24/07/2008 00:00:00	IT_PROG	60
101	21/09/1999 00:00:00	27/10/2003 00:00:00	CTPUB_GER	110
101	28/10/2003 00:00:00	15/03/2007 00:00:00	CT_GER	110
201	17/02/2006 00:00:00	19/12/2008 00:00:00	MK_ANA	20
200	17/09/1997 00:00:00	17/06/2003 00:00:00	AD_ASST	90
176	24/03/1998 00:00:00	31/12/1998 00:00:00	VE_REP	80
176	01/01/1999 00:00:00	31/12/1999 00:00:00	VE_GER	80
200	01/07/2004 00:00:00	31/12/2008 00:00:00	CTPUB_GER	90

I-18

Tabelas Usadas neste Curso (continuação)

```
SQL> DESCRIBE cargo
```

```
Nome                Nulo?      Tipo
-----
COD_CARGO           NOT NULL  VARCHAR2(10)
NOME_CARGO          NOT NULL  VARCHAR2(35)
MENOR_SALARIO              NUMBER(6)
MAIOR_SALARIO              NUMBER(6)
```

```
SQL> DESCRIBE historico_cargo
```

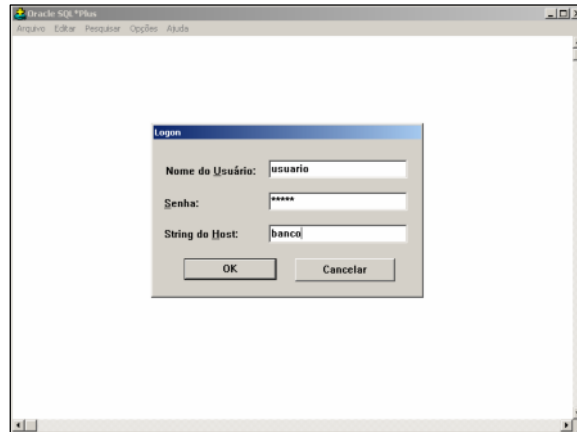
```
Nome                Nulo?      Tipo
-----
COD_FUNCIONARIO     NOT NULL  NUMBER(6)
DATA_INICIO          NOT NULL  DATE
DATA_FIM             NOT NULL  DATE
COD_CARGO            NOT NULL  VARCHAR2(10)
COD_DEPARTAMENTO              NUMBER(4)
```

```
SQL> DESCRIBE faixa_salario
```

```
Nome                Nulo?      Tipo
-----
COD_FAIXA              VARCHAR2(3)
MENOR_SALARIO              NUMBER
MAIOR_SALARIO              NUMBER
```

Visão Geral do SQL*Plus

- O SQL*Plus é a ferramenta cliente de linha de comando utilizada para interagir com o banco de dados Oracle.



```
sqlplus [usuario[/senha[@banco]]]
```

I-19

Visão Geral do SQL*Plus

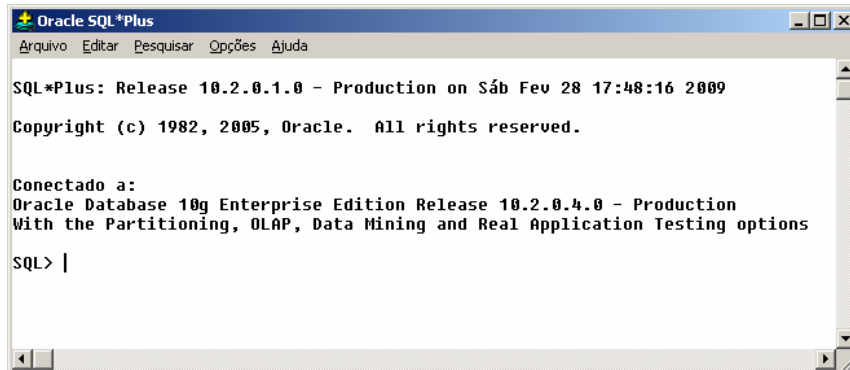
O SQL*Plus é a ferramenta cliente de linha de comando que é utilizada para interagir com o banco de dados Oracle. Apesar de não ser a ferramenta mais amigável, é a única que está presente em qualquer instalação do Oracle.

No sistema operacional Windows o SQL*Plus tem uma versão para o Prompt de Comando (plus80.exe ou sqlplus.exe) e uma versão gráfica (plus80w.exe ou sqlplusw.exe).

A figura do slide mostra a versão gráfica do SQL*Plus no momento em que é executado. Após o usuário entrar com o login, senha e string de conexão ao banco, ele irá se conectar ao banco de dados.

Visão Geral do SQL*Plus

- Os comandos SQL são guardados no buffer de instrução SQL.
- Os comandos específicos de SQL*Plus não são guardados no buffer e não exigem o sinal (;) ou (/) para serem executados



I-20

Visão Geral do SQL*Plus (continuação)

O SQL*Plus não apenas entende a linguagem SQL como também possui diversos comandos específicos da ferramenta.

Todo comando SQL que é digitado no SQL*Plus é colocado no buffer de instrução do SQL*Plus. A instrução só é removida do buffer de instrução SQL quando uma outra instrução SQL é digitada.

Os comandos específicos do SQL*Plus não são colocados em buffer. Esses comandos não são utilizados para alterar dados ou estruturas de tabelas do banco de dados.

Os comandos específicos do SQL*Plus podem ser abreviados e são informados uma linha por vez.

Listar a Estrutura de Tabelas

- O comando de SQL*Plus `DESCRIBE` exibe a estrutura de uma tabela:

```
DESC[RIBE] tabela
```

- Exemplo:

```
SQL> DESCRIBE faixa_salario
```

Nome	Nulo?	Tipo
-----	-----	-----
COD_FAIXA		VARCHAR2 (3)
MENOR_SALARIO		NUMBER
MAIOR_SALARIO		NUMBER

Listar a Estrutura de Tabelas

Para exibir a estrutura de uma tabela no SQL*Plus, você pode usar o comando `DESCRIBE`. Serão exibidos nomes de colunas e tipos de dados, bem como a indicação se a coluna pode conter valores nulos.

Na sintaxe do slide:

tabela Nome de um tabela, visão ou sinônimo existente e acessível ao usuário

Para ver a estrutura da tabela `DEPARTAMENTO`, faça o seguinte:

```
SQL> DESCRIBE departamento
Nome                Nulo?      Tipo
-----
COD_DEPARTAMENTO    NOT NULL   NUMBER(4)
NOME_DEPARTAMENTO    NOT NULL   VARCHAR2(30)
COD_GERENTE          NUMBER(6)
COD_LOCALIDADE       NUMBER(4)
```

Tipo de Dados	Descrição
NUMBER(<i>p</i> , <i>s</i>)	Numérico com um máximo de <i>p</i> dígitos e <i>s</i> dígitos à direita da vírgula decimal
VARCHAR2(<i>s</i>)	Caractere de tamanho variável com tamanho máximo igual a <i>s</i>
DATE	Data e hora entre 1º de janeiro de 4712 A.C. e 31 de dezembro de 9999 D.C.
CHAR(<i>s</i>)	Caractere de tamanho fixo com tamanho igual a <i>s</i>

Comandos de Edição do SQL*Plus

- **A[PPEND] *texto***
- **C[HANGE] / *antigo* / *novo***
- **C[HANGE] / *texto* /**
- **CL[EAR] BUFF[ER]**
- **DEL**
- **DEL *n***
- **DEL *m n***

Comandos de Edição do SQL*Plus

Abaixo a lista de comandos de edição do SQL*Plus:

Comando	Descrição
A[PPEND] <i>texto</i>	Adiciona <i>texto</i> ao final da linha atual
C[HANGE] / <i>antigo</i> / <i>novo</i>	Altera o texto <i>antigo</i> com o texto <i>novo</i> na linha atual
C[HANGE] / <i>texto</i> /	Deleta <i>texto</i> da linha atual
CL[EAR] BUFF[ER]	Deleta todas as linhas do buffer SQL
DEL	Deleta a linha atual
DEL <i>n</i>	Deleta a linha <i>n</i>
DEL <i>m n</i>	Deleta as linhas <i>m</i> a <i>n</i> inclusive

- Se você pressionar [Enter] antes de executar um comando, o SQL*Plus solicitará um número de linha.
- Para encerrar o buffer SQL, informe um dos caracteres finalizadores (ponto-e-vírgula ou barra) ou pressione [Enter] duas vezes. O prompt do SQL será exibido.

Comandos de Edição do SQL*Plus

- **I[NPUT]**
- **I[NPUT] *texto***
- **L[IST]**
- **L[IST] *n***
- **L[IST] *m n***
- **R[UN]**
- ***n***
- ***n texto***
- **0 *texto***

Comandos de Edição do SQL*Plus (continuação)

Comando	Descrição
I[NPUT]	Insere um número indefinido de linhas
I[NPUT] <i>texto</i>	Insere uma linha que consiste em <i>texto</i>
L[IST]	Lista todas as linhas no buffer SQL
L[IST] <i>n</i>	Lista uma linha (especificada por <i>n</i>)
L[IST] <i>m n</i>	Lista uma faixa de linhas (<i>m</i> a <i>n</i>) inclusive
R[UN]	Exibe e executa a instrução SQL atual no buffer
<i>N</i>	Especifica a linha que se tornará a atual
<i>n texto</i>	Substitui a linha <i>n</i> por <i>texto</i>
0 <i>texto</i>	Insere <i>texto</i> uma linha antes da linha 1

Observação: Para continuar um comando SQL*Plus na próxima linha, finalize a primeira linha com um hífen (-).

Exemplo de Uso do LIST, n e APPEND

```
SQL> LIST
```

```
1  SELECT nome  
2* FROM   funcionario
```

```
SQL> 1
```

```
1* SELECT nome
```

```
SQL> A , cod_cargo
```

```
1* SELECT nome, cod_cargo
```

```
SQL> L
```

```
1  SELECT nome, cod_cargo  
2* FROM   funcionario
```

Exemplo de Uso do LIST, n e APPEND

- Use o comando `L[IST]` para exibir o conteúdo do buffer SQL. O asterisco (*) ao lado da linha 2 no buffer indica que essa é a linha atual. Todas as edições feitas se aplicam à linha atual.
- Para alterar o número da linha atual, informe o número (n) da linha a ser editada. A nova linha atual será exibida.
- Use o comando `A[PPEND]` para adicionar texto à linha atual. A linha recém editada será exibida. Verifique o novo conteúdo do buffer com o comando `LIST`.

Exemplo de Uso do Comando CHANGE

```
SQL> L
```

```
1* SELECT * from funcionario
```

```
SQL> c/funcionario/departamento
```

```
1* SELECT * from departamento
```

```
SQL> L
```

```
1* SELECT * from departamento
```

Exemplo de Uso do Comando CHANGE

- Use `L[IST]` para exibir o conteúdo do buffer.
- Use o comando `C[HANGE]` para alterar o conteúdo da linha atual no buffer SQL. Neste caso, substitua a tabela `funcionario` pela tabela `cliente`. A nova linha atual será exibida.
- Use o comando `L[IST]` para exibir o novo conteúdo do buffer.

Comandos de Arquivo do SQL*Plus

- **SAVE** *nome_arquivo*
- **GET** *nome_arquivo*
- **START** *nome_arquivo*
- **@** *nome_arquivo*
- **EDIT** *nome_arquivo*
- **SPOOL** *nome_arquivo*
- **EXIT**

Comandos de Arquivo do SQL*Plus

Além dos comando de edição do SQL*Plus, existem os comandos de manipulação de arquivos. Você pode utilizar os comandos descritos nesta tabela:

Comando	Descrição
SAV[E] <i>nome_arquivo</i> [.ext] [REP[LACE]APP[END]]	Salva o conteúdo atual do buffer SQL em um arquivo. Use APPEND para adicionar o conteúdo a um arquivo existente e REPLACE para sobregravar um arquivo existente. A extensão default é .sql.
GET <i>nome_arquivo</i> [.ext]	Grava o conteúdo de um arquivo salvo anteriormente no buffer SQL. A extensão default do nome do arquivo é .sql.
STA[RT] <i>nome_arquivo</i> [.ext]	Executa um arquivo de comandos salvo anteriormente
@ <i>nome_arquivo</i>	Executa um arquivo de comandos salvo anteriormente (igual a START)
ED[IT]	Acessa o editor e salva o conteúdo do buffer no arquivo afiedt.buf
ED[IT] [<i>nome_arquivo</i> [.ext]]	Acessa o editor para editar o conteúdo de um arquivo salvo
SPO[OL] [<i>nome_arquivo</i> [.ext] [CRE[ATE] REP[LACE] APP[END]] OFF OUT]	Armazena os resultados da consulta em um arquivo. OFF fecha o arquivo de spool. OUT fecha o arquivo de spool e envia os resultados do arquivo para a impressora.
EXIT	Encerra o SQL*Plus

Exemplo de Uso dos Comandos SAVE e START

```
SQL> L
  1  SELECT nome, cod_gerente, cod_cargo
  2* FROM funcionario
SQL> SAVE c:\minha_consulta.sql
```

```
Criado file c:\minha_consulta.sql
```

```
SQL> START c:\minha_consulta.sql
```

NOME	COD_GERENTE	COD_CARGO
Roberto		AD_PRES
Nair	100	AD_VP
Leonardo	100	AD_VP
...		

20 linhas selecionadas.

SAVE

Use o comando `SAVE` para armazenar o conteúdo atual do buffer em um arquivo. Dessa maneira, você poderá armazenar os scripts usados com frequência para uso futuro.

START

Use o comando `START` para executar um script no SQL*Plus.

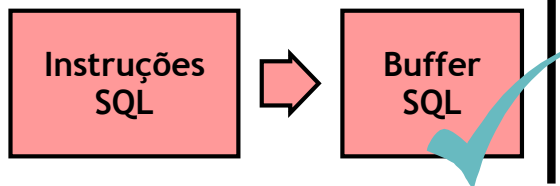
EDIT

Use o comando `EDIT` para editar um script existente. Esse procedimento abre um editor com o arquivo de script. Após fazer as alterações, encerre o editor e retorne à linha de comandos do SQL*Plus.

Instruções SQL vs Comandos SQL*Plus

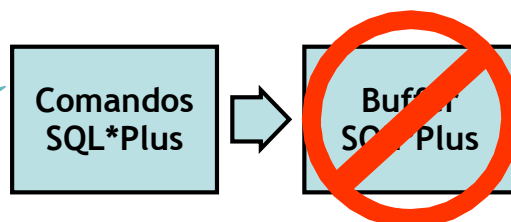
SQL

- Uma linguagem
- Padrão ANSI
- Não é possível abreviar palavras-chave.
- Instruções manipulam dados e definições de tabelas no banco.



SQL*Plus

- Uma ferramenta
- Propriedade Oracle
- É possível abreviar palavras-chave.
- Comandos não permitem a manipulação de valores no banco de dados.



Instruções SQL versus Comandos SQL*Plus

Esta tabela compara as instruções SQL e os comandos do SQL*Plus:

SQL	SQL*Plus
É uma linguagem para a comunicação com o servidor Oracle a fim de acessar os dados	Reconhece instruções SQL e as envia ao servidor
É baseada no padrão ANSI (American National Standards Institute) SQL	É a interface proprietária da Oracle para a execução de instruções SQL
Manipula dados e definições de tabelas no banco de dados	Não permite a manipulação de valores no banco de dados
É incluída no buffer SQL em uma ou mais linhas	É incluído em uma linha de cada vez; não é armazenado no buffer SQL
Não tem um caractere de continuação	Utiliza um traço (-) como caractere de continuação se o comando ultrapassa uma linha
Não pode ser abreviada	Pode ser abreviado
Utiliza um caractere de finalização para executar os comandos imediatamente	Não requer caracteres de finalização; executa os comandos imediatamente
Utiliza functions para aplicar formatação	Utiliza comandos para formatar dados

SQLTools

- Ferramenta leve e robusta para desenvolvimento com o banco de dados.
- Características:
 - Possui um editor poderoso, que suporta arrastar e colar, realce de sintaxe e edição em modo de coluna.
 - Conta com um Gerenciador de Objetos (Object Browser), que facilita a interação com os diversos tipos de objetos.
 - Tem um utilitário de extração de esquema, que grava todas as definições de objetos de um esquema em um arquivo local
- Software gratuito que pode ser baixado do site <http://www.sqltools.net>

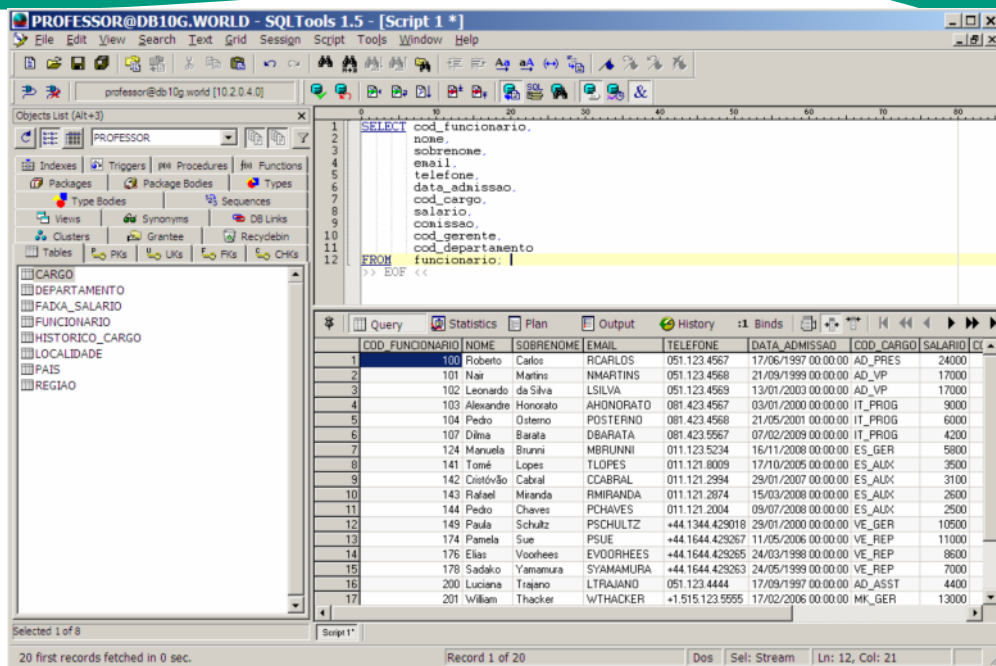
I-29

Ferramenta SQLTools

O SQLTools é uma ferramenta gratuita para desenvolvimento com o banco de dados. Ele é leve e robusto. O SQLTools possui um editor poderoso que pode realçar a sintaxe de diversos tipos de linguagem de programação (inclusive o SQL) e suporta o modo de edição em coluna.

Com o Gerenciador de Objetos (Object Browser) o usuário poderá interagir facilmente com os diversos tipos de objetos dos esquemas existentes no banco de dados. Um exemplo de interação que pode ser feita com o Object Browser é desabilitar várias constraints de banco de dados, selecionando todas elas, e escolhendo uma única opção de menu.

SQLTools



I-30

SQLTools

Outras ferramentas que concorrem com o SQLTools:

- SQL Developer
- TOra
- Dreamcoder for Oracle
- TOAD
- SQL Navigator
- PL/SQL Developer