



SHIFT



FIAP





MASTERING C#

CODE EVERYTHING

SHIFT  FIAP

BANCO **DE DADOS**



BANCO DE DADOS

1

AULA 1

CONTEÚDO – Introdução ao Contexto de Dados e a Linguagem SQL

2

AULA 2

CONTEÚDO – Criação de Objetos (DDL)

3

AULA 3

CONTEÚDO – Alteração e Exclusão de Objetos (DDL)

4

AULA 4

CONTEÚDO – Inserção, Atualização e Exclusão de Dados (DML)



BANCO DE DADOS

5

AULA 5

CONTEÚDO – Seleção e Manipulação de Dados (DML)

6

AULA 6

CONTEÚDO – Seleção e Junção de Dados (JOIN)

7

AULA 7

CONTEÚDO – Funções e Cálculos

8

AULA 8

CONTEÚDO – Funções e Agrupamentos



AULA 7

FUNÇÕES **E CÁLCULOS**



Introdução a Funções



Funções Numéricas



Funções de Caracteres



Funções de Conversão



Funções de Data



Exercícios



INTRODUÇÃO **A FUNÇÕES**



INTRODUÇÃO

A FUNÇÕES

- As funções SQL, assim como em outras linguagens, **recebem e processam argumentos, retornando um determinado resultado** ao ambiente de chamada, de acordo com a solicitação.
- As funções de linha são utilizadas para manipular itens de dados, ou seja, aceitam um ou mais argumentos e retornam um valor para cada linha apresentada pela consulta.
- **Manipulam caracteres, números, datas, além de converterem dados**, podendo ser utilizadas na lista de colunas ou em condições para recuperação de linhas.

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



INTRODUÇÃO

A FUNÇÕES

- Embora o SQL seja uma linguagem padronizada, consulte o material da aula 01 para mais detalhes, suas **funções podem variar de acordo com** cada **SGBD**.
- Algumas funções no Oracle, por exemplo, são diferentes do SQL Server ou do MySQL.
- As funções **podem possuir parâmetros** que, normalmente, são identificados por estarem dentro de **parênteses**.
- O retorno de uma função pode ser numérico, caracter (string), data ou NULO.

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



TIPOS DE FUNÇÕES

NATIVAS DO ORACLE

- **Funções Numéricas:** recebem, em sua maioria, um número, como parâmetro, e retornam um outro, como resultado. Por exemplo, a função ROUND que arredonda um número decimal.
- **Funções de Caracteres:** aceitam somente caracteres como parâmetros de entrada e retornam tanto caracteres como números. Por exemplo, a função LENGTH, que retorna a quantidade de caracteres de um campo.

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



TIPOS DE FUNÇÕES

NATIVAS DO ORACLE

- **Funções de Data:** recebem parâmetros do tipo data e retornam tanto resultados em formato de data como em formato numérico. Por exemplo, a função `ADD_MONTHS`, que adiciona ou subtrai meses de um campo do tipo data.
- **Funções de Conversão:** utilizadas quando é necessário converter um valor de entrada em outro. Por exemplo, a função `TO_CHAR`, que converte um campo em texto.

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



FUNÇÕES NUMÉRICAS



FUNÇÕES NUMÉRICAS

ALUNO(A)	RM
ROUND (X, [Y])	Retorna o resultado do arredondamento de X com Y casas decimais (opcional). Se Y for omitido, arredonda em 0 casa decimal. Se Y for negativo, X será arredondado à esquerda do ponto decimal.
MOD (X, Y)	Retorna o resto, quando X é dividido por Y.
TRUNC (X, [Y])	Retorna o resultado do truncamento de X com Y casas decimais opcionais. Se Y for omitido, trunca em 0 casa decimal. Se Y for negativo, X será truncado à esquerda do ponto decimal.
ABS (X)	Retorna o valor absoluto de X. Ignora o sinal (+ ou -).
SQRT (X)	Retorna a raiz quadrada de X.
FLOOR (X)	Função para arredondamento de números decimais para baixo.
CEIL (X)	Função para arredondamento de números decimais para cima.

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



FUNÇÕES NUMÉRICAS

- As funções numéricas aceitam entrada numérica e retornam valores numéricos.

	FUNC_ROUND	FUNC_MOD	FUNC_TRUNC	FUNC_ABS	FUNC_SQRT	FUNC_FLOOR	FUNC_CEIL
1	1234,56	0,56	1234	1234,56	35,13630600	1234	1235
2	2345	1	2345	2345	48,42520005	2345	2345
3	7654,99	0,99	7654	7654,99	87,49279970	7654	7655
4	3452,12	0,12	3452	3452,12	58,75474448	3452	3453
5	980	0	980	980	31,30495168	980	980
6	1780	0	1780	1780	42,19004621	1780	1780

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



FUNÇÕES NUMÉRICAS

```
SELECT
    LAST_NAME,
    SALARY,
    ROUND (SALARY, 4)      FUNC_ROUND,
    MOD (SALARY, 2)        FUNC_MOD,
    TRUNC (SALARY, 0)      FUNC_TRUNC,
    ABS (SALARY * (-1))    FUNC_ABS,
    SQRT (SALARY)          FUNC_SQRT,
    FLOOR (SALARY)         FUNC_FLOOR,
    CEIL (SALARY)          FUNC_CEIL
FROM EMPLOYEES;
```

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



FUNÇÕES **DE CARACTERES**



FUNÇÕES DE CARACTERES

- Funções para retornar caracteres em maiúsculas ou minúsculas.

FUNÇÃO	DESCRIÇÃO
LOWER (X)	Converte as letras de X para minúsculas e retorna a nova string.
UPPER (X)	Converte as letras de X para maiúsculas e retorna a nova string.
INITCAP (X)	Converte a letra inicial de cada palavra da string X em maiúsculas e retorna a nova string.

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



EXEMPLO:

FUNÇÕES DE CARACTERES

```
SELECT
    LOWER(LAST_NAME)    FUNC_LOWER,
    UPPER(LAST_NAME)    FUNC_UPPER,
    INITCAP(LAST_NAME)  FUNC_INITCAP
FROM EMPLOYEES;
```

	FUNC_LOWER	FUNC_UPPER	FUNC_INITCAP
1	oconnell	CONNELL	Oconnell
2	grant	GRANT	Grant
3	whalen	WHALEN	Whalen
4	hartstein	HARTSTEIN	Hartstein
5	fay	FAY	Fay
6	mavris	MAVRIS	Mavris
7	baer	BAER	Baer
8	higgins	HIGGINS	Higgins

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



FUNÇÕES DE CARACTERES

FUNÇÃO	DESCRIÇÃO
CONCAT (X, Y)	Concatena o conteúdo de Y com o de X, retornando um resultado único.
SUBSTR (X, Início, Comprimento)	Retorna uma substring de X que começa na posição especificada por início. O comprimento pode ser fornecido opcionalmente.
LENGTH (X)	Retorna a quantidade de caracteres de X.
INSTR (X, Localiza String, Início, Ocorrência)	Procura em X uma string específica e retorna a posição em que ela ocorre. Pode ser informada a posição início (opcional) para iniciar a busca. Pode ser fornecida a ocorrência (opcional), que indica qual ocorrência de "Localiza String" deve ser retornada.
LPAD/ RPAD (X, Largura, Caractere de Preenchimento)	Preenche X com espaços, a esquerda (LPAD) ou direita (RPAD), para que o comprimento total da string tenha até n caracteres de largura. Pode ser fornecida uma string para o preenchimento a esquerda ou a direita de X. Quando não fornecida a string de preenchimento é utilizado, como padrão, os espaços em branco.
LTRIM / RTRIM (X, String Corte) TRIM (String Corte FROM X)	Corta caracteres a direita (RTRIM), a esquerda (LTRIM) ou em ambos (TRIM) de X. String Corte (opcional) especifica os caracteres a serem cortados. Se nenhuma string for fornecida, por padrão são cortados os espaços em branco.
REPLACE (X, String Busca, String Substituição)	Procura em X um conjunto de string e a substitui.

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>

EXEMPLO: FUNÇÕES DE CARACTERES (CONCAT, SUBSTR E LENGTH)

```
SELECT
    CONCAT(LAST_NAME, LAST_NAME) FUNC_CONCAT_EX01,
    CONCAT(LAST_NAME, '_ABC')    FUNC_CONCAT_EX02,
    SUBSTR(LAST_NAME, 1, 3)      FUNC_SUBSTR_EX01,
    SUBSTR(LAST_NAME, 3)         FUNC_SUBSTR_EX02,
    LENGTH(LAST_NAME)            FUNC_LENGTH
FROM EMPLOYEES;
```

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



EXEMPLO: FUNÇÕES DE CARACTERES

(CONCAT, SUBSTR E LENGTH)

FUNC_CONCAT_EX01	FUNC_CONCAT_EX02	FUNC_SUBSTR_EX01	FUNC_SUBSTR_EX02	FUNC_LENGTH
1 OConnellOConnell	OConnell ABC	OC	onnell	8
2 GrantGrant	Grant ABC	Gra	ant	5
3 WhalenWhalen	Whalen ABC	Wha	alen	6
4 HartsteinHartstein	Hartstein ABC	Har	rtstein	9
5 FayFay	Fay ABC	Fay	y	3
6 MavrisMavris	Mavris ABC	Mav	vris	6
7 BaerBaer	Baer ABC	Bae	er	4
8 HigginsHiggins	Higgins ABC	Hig	ggins	7

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>

EXEMPLO: FUNÇÕES DE CARACTERES (INSTR)

```
SELECT
    LAST_NAME,
    INSTR(LAST_NAME, 'a')          FUNC_INSTR_EX01,
    INSTR(LAST_NAME, 'a', 3)       FUNC_INSTR_EX02,
    INSTR(LAST_NAME, 'a', 3, 2)    FUNC_INSTR_EX03
FROM EMPLOYEES
```

LAST_NAME	FUNC_INSTR_EX01	FUNC_INSTR_EX02	FUNC_INSTR_EX03
1 OConnell	0	0	0
2 Grant	3	3	0
3 Whalen	3	3	0
4 Hartstein	2	0	0
5 Fay	2	0	0
6 Mavris	2	0	0
7 Baer	2	0	0
8 Higgins	0	0	0
9 Gietz	0	0	0
10 King	0	0	0
11 Kochhar	6	6	0
12 De Haan	5	5	6
13 Hunold	0	0	0

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



EXEMPLO: FUNÇÕES DE CARACTERES (RPAD E LPAD)

```
SELECT
    LAST_NAME,
    RPAD(LAST_NAME, 20)          FUNC_RPAD_EX01,
    RPAD(LAST_NAME, 20, '.')    FUNC_RPAD_EX02,
    LPAD (LAST_NAME, 20, '*')   FUNC_LPAD_EX01
FROM EMPLOYEES;
```

	LAST_NAME	FUNC_RPAD_EX01	FUNC_RPAD_EX02	FUNC_LPAD_EX01
1	OConnell	OConnell	OConnell.....	*****OConnell
2	Grant	Grant	Grant.....	*****Grant
3	Whalen	Whalen	Whalen.....	*****Whalen
4	Hartstein	Hartstein	Hartstein.....	*****Hartstein
5	Fay	Fay	Fay.....	*****Fay
6	Mavris	Mavris	Mavris.....	*****Mavris
7	Baer	Baer	Baer.....	*****Baer

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>

EXEMPLO: FUNÇÕES DE CARACTERES

(LTRIM, RTRIM E TRIM)

```
SELECT
    LAST_NAME,
    LTRIM(LAST_NAME, 'K')          FUNC_LTRIM,
    RTRIM(LAST_NAME, 'g')          FUNC_RTRIM,
    TRIM('D' FROM LAST_NAME)       FUNC_TRIM
FROM EMPLOYEES
WHERE EMPLOYEE_ID IN (100,101,102);
```

	LAST_NAME	FUNC_LTRIM	FUNC_RTRIM	FUNC_TRIM
1	King	ing	Kin	King
2	Kochhar	ochhar	Kochhar	Kochhar
3	De Haan	De Haan	De Haan	e Haan



EXEMPLO: FUNÇÕES DE CARACTERES (REPLACE)

```
SELECT  
    LAST_NAME,  
    REPLACE (LAST_NAME, 'a' , '*')  
    FUNC_NM_NOME  
FROM EMPLOYEES;
```

	LAST_NAME	FUNC_NM_NOME
1	OConnell	OConnell
2	Grant	Gr*nt
3	Whalen	Wh*len
4	Hartstein	H*rtstein

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>

FUNÇÕES **DE CONVERSÃO**



FUNÇÕES DE CONVERSÃO

FUNÇÃO	DESCRIÇÃO
TO_CHAR (X, Formato)	Converte X em uma string (VARCHAR2). Pode ser fornecido um formato (opcional), indicando o formato de X.
TO_NUMBER (X, Formato)	Converte X em um valor numérico (NUMBER).
TO_DATE (X, Formato)	Converte X em um valor de data (DATE).

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



FUNÇÃO TO_CHAR

- A função de linha **TO_CHAR** (argumento1, argumento2) **converte o valor** (data ou número) representado em argumento1, **para** uma cadeia de caracteres com um **formato predeterminado**, representado por argumento2.
- A conversão de números em caracteres é utilizada, especialmente, para manipulações que requerem a apresentação de números em determinados formatos.

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



FUNÇÃO TO_CHAR

ELEMENTO	DESCRIÇÃO	FORMATO	RESULTADO
\$	Exibe o cifrão do lado esquerdo do valor.	\$99999	\$1234
,	Separador de milhar.	999,999	1,234
.	Separador de decimal.	9999.99	1234.00
0	Insere zeros à esquerda do número para completar o valor que contiver menos caracteres que o especificado.	099999	01234
9	Insere “espaços em branco” à esquerda do número que contiver menos caracteres que o especificado.	99999	1234
B	Se a parte inteira de um número de ponto fixo é zero, retorna espaços para os zeros.	B099999	001234.00
C	Retorna o símbolo de moeda ISO na posição especificada. O símbolo vem do parâmetro de banco de dados NLS_ISO_CURRENCY definido pelo DBA.	C99999	BRL1234

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



EXEMPLO:

FUNÇÃO TO_CHAR

ENTRADA	TO_CHAR_EX01	TO_CHAR_EX02	TO_CHAR_EX03
1234	\$1234	1,234	1234.00

TO_CHAR_EX04	TO_CHAR_EX05	TO_CHAR_EX06	TO_CHAR_EX07
001234	1234	001234	BRL1234

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



EXEMPLO:

FUNÇÃO TO_CHAR

SELECT

1234 ENTRADA,	
TO_CHAR(1234, '\$99999')	TO_CHAR_EX01,
TO_CHAR(1234, '999,999')	TO_CHAR_EX02,
TO_CHAR(1234, '9999.99')	TO_CHAR_EX03,
TO_CHAR(1234, '099999')	TO_CHAR_EX04,
TO_CHAR(1234, '99999')	TO_CHAR_EX05,
TO_CHAR(1234, 'B099999')	TO_CHAR_EX06,
TO_CHAR(1234, 'C99999')	TO_CHAR_EX07

FROM DUAL;

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



FUNÇÃO TO_CHAR

- A função **TO_CHAR** também **possibilita converter** datas em caracteres e, assim como com números, torna-se interessante para algum tipo de manipulação que requeira a apresentação **de datas em formatos específicos**.

ELEMENTO	DESCRIÇÃO	RESULTADO
DD	Dia do mês.	18
DY	Abreviação do nome do dia da semana com 3 letras.	SÁB
DAY	Nome do dia da semana.	SÁBADO
DDSP	Nome do dia no mês por extenso.	EIGHTEEN
MM	Número do mês.	08
MON	Abreviação do nome do mês com três letras.	AGO
MONTH	Nome do mês por extenso.	AGOSTO
YY	Ano com dois dígitos.	18
RR	Ano com dois dígitos. Criado para solucionar o problema do bug do milênio, que aconteceu na virada para o ano de 2000. Datas entre 00 a 49 são consideradas 2000 a 2049. Datas entre 50 a 99 são consideradas 1950 a 1999.	18
YYYY	Ano com quatro dígitos.	2018
RRRR	Ano com quatro dígitos, atendendo também aos requisitos do bug do milênio.	2018
HH:MI:SS	Hora, minuto e segundo.	08:52:53
HH24	Hora, contemplando números de 0 a 23.	20

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



EXEMPLO:

FUNÇÃO TO_CHAR

ENTRADA	DD	DY	DDAY	DDSP	MM	MMON
18/08/18	18	SÁB	SÁBADO	EIGHTEEN	08	AGO

MMONTH	YY	RR	YYYY	RRRR	HHMISS	HH24
AGOSTO	18	18	2018	2018	09:06:32	21

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



EXEMPLO:

FUNÇÃO TO_CHAR

SELECT

CURRENT_DATE	ENTRADA,	TO_CHAR(CURRENT_DATE, 'YY')	YY,
TO_CHAR(CURRENT_DATE, 'DD')	DD,	TO_CHAR(CURRENT_DATE, 'RR')	RR,
TO_CHAR(CURRENT_DATE, 'DY')	DY,	TO_CHAR(CURRENT_DATE, 'YYYY')	YYYY,
TO_CHAR(CURRENT_DATE, 'DAY')	DDAY,	TO_CHAR(CURRENT_DATE, 'RRRR')	RRRR,
TO_CHAR(CURRENT_DATE, 'DDSP')	DDSP,	TO_CHAR(CURRENT_DATE, 'HH:MI:SS')	HHMISS,
TO_CHAR(CURRENT_DATE, 'MM')	MM,	TO_CHAR(CURRENT_DATE, 'HH24')	HH24
TO_CHAR(CURRENT_DATE, 'MON')	MMON,	FROM DUAL;	
TO_CHAR(CURRENT_DATE, 'MONTH')	MMONTH,		



VISUALIZAÇÃO DE DATA NO ORACLE

- O padrão de exibição, de um campo do tipo data, no Oracle é:
 - dia, mês e ano (com dois dígitos).
- Para alterar o formato de exibição é utilizado a sintaxe:
 - `ALTER SESSION SET NLS_DATE_FORMAT = formato_data.`
 - `ALTER SESSION SET NLS_DATE_FORMAT = 'DD/MM/YYYY'.`

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



VISUALIZAÇÃO DE DATA NO ORACLE

⚡ COD	⚡ UPPER(NOME)	⚡ DT_NASC	⚡ SEXO	⚡ DT_OBTO
1010	STEPHEN HAWKING	08/01/42	M	14/03/18
1011	EINSTEIN	14/03/79	M	18/04/55
1012	NEWTON	04/01/43	M	31/03/27
1013	GALILEU GALILEI	15/02/64	M	08/01/42
1014	NICOLAU COPERNICO	19/02/73	M	21/05/43

⚡ COD	⚡ UPPER(NOME)	⚡ DT_NASC	⚡ SEXO	⚡ DT_OBTO
1010	STEPHEN HAWKING	08/01/1942	M	14/03/2018
1011	EINSTEIN	14/03/1879	M	18/04/1955
1012	NEWTON	04/01/1643	M	31/03/1727
1013	GALILEU GALILEI	15/02/1564	M	08/01/1642
1014	NICOLAU COPERNICO	19/02/1473	M	21/05/1543

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/Functions.html#>



FORMATO DE IDIOMA NO ORACLE

- Para alterar o idioma de exibição da data é utilizado a sintaxe:
 - ALTER SESSION SET NLS_DATE_LANGUAGE = idioma.
 - ALTER SESSION SET NLS_DATE_LANGUAGE = 'PORTUGUESE'.

SELECT

SYSDATE,

TO_CHAR(SYSDATE, 'MON') MESABV,

TO_CHAR(SYSDATE, 'MONTH') MES

FROM DUAL;



FORMATO DE IDIOMA NO ORACLE

DIAMESANO	MESABV	MES
28/01/18	JAN	JANEIRO

DIAMESANO	MESABV	MES
28/01/18	JAN	JANUARY

DIAMESANO	MESABV	MES
28/01/18	ENE	ENERO



FUNÇÃO

TO_NUMBER

- A função de linha **TO_NUMBER** (argumento1) converte uma cadeia de caracteres numéricos em um valor numérico.

ENTRADA	TO_NUMBER_EX01
1234	2468

SELECT

'1234' ENTRADA,

TO_NUMBER('1234') + 1234 TO_NUMBER_EX01

FROM DUAL;



FUNÇÃO

TO_DATE

- A função de linha **TO_DATE** (argumento1, argumento2) converte uma cadeia de caracteres (argumento1), para o formato de data (argumento2).

ENTRADA	TO_DATE_EX01
03/01/2018	03/01/18

```
SELECT
    '03/01/2018' ENTRADA,
    TO_DATE('03/01/2018', 'DD/MM/YYYY')
    TO_DATE_EX01
FROM DUAL;
```



FUNÇÕES **DE DATA**



FUNÇÕES DE DATA

- Existem algumas funções para operar campos do tipo data e facilitar sua manipulação.

FUNÇÃO	DESCRIÇÃO
SYSDATE /CURRENT_DATE	Retorna a data local do banco de dados.
SYSTIMESTAMP / CURRENT_TIMESTAMP	Retorna a data, hora e fuso local do banco de dados.
ADD_MONTHS (X, Y)	Adiciona em X a quantidade Y de mês(es).
MONTHS_BETWEEN (Data Final, Data Inicial)	Subtrai da Data Final a Data Inicial.



EXEMPLO:

FUNÇÃO DE DATA

SELECT

CURRENT_DATE	ENTRADA,
SYSDATE	ENTRADA,
SYSTIMESTAMP	ENTRADA,
CURRENT_TIMESTAMP	ENTRADA,
ADD_MONTHS (CURRENT_DATE, 2)	ADD_MONTHS_EX01,
ADD_MONTHS (CURRENT_DATE, -2)	ADD_MONTHS_EX02

FROM DUAL;

FONTE: https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/CURRENT_DATE.html#

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/SYSDATE.html#>

FONTE: <https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/SYSTIMESTAMP.html#>

FONTE: https://docs.oracle.com/en/database/oracle/oracle-database/19/sqlrf/ADD_MONTHS.html#



EXEMPLO:

FUNÇÃO DE DATA

ENTRADA	ENTRADA_1	ENTRADA_2
19/08/18	19/08/18	19/08/18 12:34:42,625000000 -03:00

ENTRADA_3
19/08/18 12:34:42,625000000 AMERICA/SAO_PAULO

ADD_MONTHS_EX01	ADD_MONTHS_EX02
19/10/18	19/06/18



EXEMPLO:

FUNÇÃO DE DATA

SELECT

```
MONTHS_BETWEEN(CURRENT_DATE, '01/01/2018') MONTHS_BETWEEN_EX01,  
MONTHS_BETWEEN('05/05/2018', '01/01/2018')  
MONTHS_BETWEEN_EX02,  
MONTHS_BETWEEN('05/05/2017', '01/01/2018')  
MONTHS_BETWEEN_EX03
```

FROM DUAL;

MONTHS_BETWEEN_EX01
7,70818324372759856630824372759856630824

MONTHS_BETWEEN_EX02
4,12903225806451612903225806451612903226

MONTHS_BETWEEN_EX03
-7,87096774193548387096774193548387096774



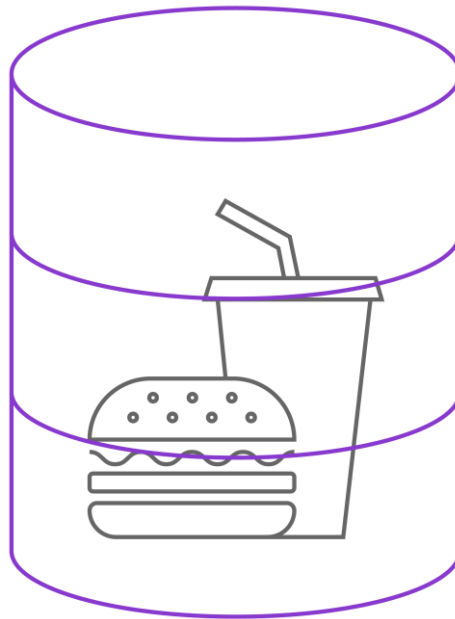
EXERCÍCIOS

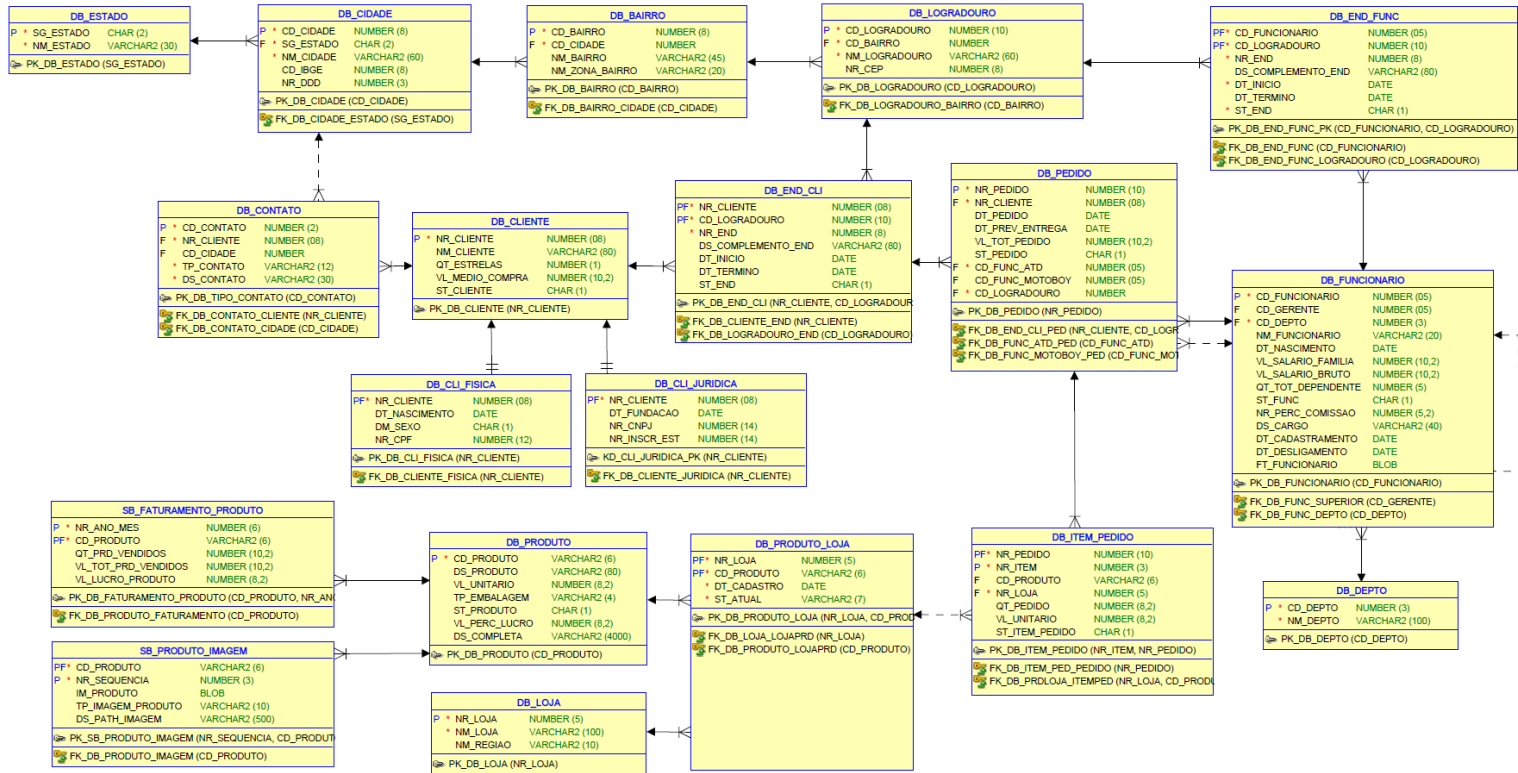


PROJETO: DBURGER

Instruções para iniciar o projeto:

1. Execute o bloco DDL para a criação da estrutura do Banco.
2. Execute o bloco DML para popular as tabelas criadas.
3. O MER (modelo entidade relacionamento) do projeto está no slide seguinte (32).





CONSULTA

UTILIZANDO FUNÇÕES

1. Trazer o nome do funcionário com, somente, as iniciais em maiúsculo.
2. Trazer o nome do funcionário, a data de nascimento e calcular a idade.
3. Informar se o funcionário possui uma idade par ou ímpar.
4. Trazer o nome do funcionário e o salário, criando uma regra de arredondamento salarial para cima.



DESAFIO



DESAFIO

1. Supondo que, com 55 anos de idade, uma pessoa se aposente. Quais são os profissionais que estarão aposentados em 2025?
2. Quais funcionários não estarão aposentados em 2025 e quantos anos faltarão para se aposentarem?
3. Supondo que, após se aposentar, o funcionário passe a receber 70% do valor de seu salário atual, quanto ele receberá?



BIBLIOGRAFIA BÁSICA

- **TAYLOR, A. SQL para Leigos.** São Paulo: Alta Books, 2016.
- **PUGA, S.; FRANÇA, E; GOYA, M.** Banco de Dados: Implementação em SQL, PL/SQL e Oracle 11g. New Jersey: Pearson Universities, 2013.
- **ORACLE LIVE SQL.** Learn and share SQL: Running on Oracle Database 19c. [s.d.]. Disponível em: <<https://livesql.oracle.com/apex/f?p=590:1000>>. Acesso em: 02/12/2020.
- **ORACLE HELP CENTER.** Oracle Database 19c Get Started. [s.d.]. Disponível em: <<https://docs.oracle.com/en/database/oracle/oracle-database/19/index.html>>. Acesso em: 02/12/2020.



OBRIGADO



in /alexandrebarcelos

FIAP

Copyright © 2021 | Professor MSc. Alexandre Barcelos

Todos os direitos reservados. Reprodução ou divulgação total ou parcial deste documento, é expressamente proibido sem consentimento formal, por escrito, do professor/autor.



SHIFT



FIAP

