



Fase3 - Cap 10

Cap 10 - Gatilhos mágicos

Exercendo a Automação

Integrantes

Elio Gonçalves de Lima - elio.designer@hotmail.com

Andrei Vedovato Alves Rodrigues - vedovato.andrei@gmail.com

Luis Carvalho dos Santos Junior - luiss225@hotmail.com

Vinicios Koji Yamamoto - viniciosyamamoto@gmail.com

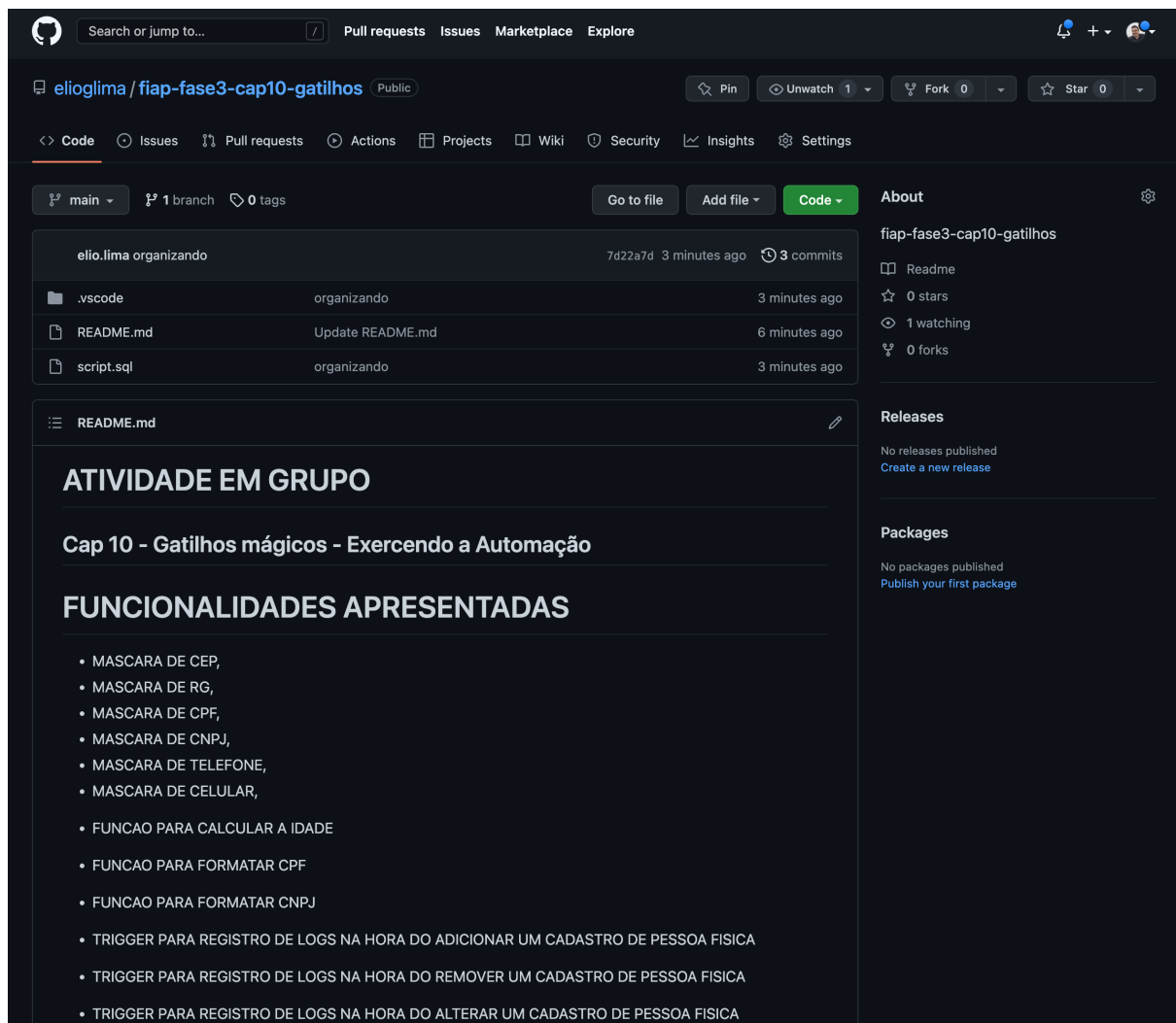
RAFAELA ROSSO PACHECO - rafaelarosso2011@hotmail.com

Github

<https://github.com/elioglima/fiap-fase3-cap10-gatilhos>

Download Script SQL

<https://github.com/elioglima/fiap-fase3-cap10-gatilhos/blob/main/script.sql>



OBJETIVO

Criar um conjunto de utilitário para automação do fluxo do sistema. Tem como visao efetuar o cadastro de pessoa e acionar o gatilho de uma trigger para gravar logs. Entre outras foi criado funções para manipulação de dados de documentos, telefones, e emails.

Abaixo segue passo a passo des da criacao dos scripts ate o a execucao da trigger.

Função genérica de formatação

```
CREATE OR REPLACE FUNCTION MASCARA
  (val IN VARCHAR2, mask IN VARCHAR2)
  RETURN VARCHAR2 AS masked VARCHAR(100) DEFAULT ' ';

  k INT DEFAULT 0;
  i INT DEFAULT 0;

  BEGIN WHILE i < LENGTH(mask) LOOP i: = i + 1;
    IF SUBSTR(mask, i, 1) = '#' THEN
      IF k < LENGTH(val) THEN k: = k + 1;
      masked: = CONCAT(masked, SUBSTR(val, k, 1));
```

```

        END IF;

    ELSE
        IF i < LENGTH(mask) THEN
            maskared:= CONCAT(maskared, SUBSTR(mask, i, 1));
        END IF;

        END IF;
    END LOOP;

    RETURN maskared;

END MASCARA;

```

FUNCAO GENERICA DE MASCARA

```

- MASK("12345123", "#####-###") AS cep,
- MASK("121231231", "##.###.###-#") AS rg,
- MASK("1212312312", "##.###.###-##") AS cpf,
- MASK("12123123123412", "##.###.###/####-##") AS cnpj,
- MASK("1212341234", "(##) #####-####") AS telefone,
- MASK("12123451234", "(##) #####-####") AS celular

```

CEP: #####-###

Função que converte um texto de um numero de cep para um cep formatado.

RG: ##.###.###-#

Função que converte um texto de um numero de cep para um cep formatado.

CPF: ##.###.###-##

Função que converte um texto de um numero de cep para um cep formatado.

CNPJ: ##.###.###/####-##

Função que converte um texto de um numero de cep para um cep formatado.

TELEFONE: #####-####

Função que converte um texto de um numero de cep para um cep formatado.

CELULAR: #####-####

Função que converte um texto de um numero de cep para um cep formatado.

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Oracle Conexões' tree is visible, with 'fiap' selected. The 'Planilha' tab is active, displaying a SQL query in the 'Query Builder' window. The query is as follows:

```

SELECT
    FUN_CALCULAIDADE(TO_DATE('1979-11-04', 'YYYY-MM-DD')) AS IDADE,
    MASCARA('12345123', '#####-###') AS cep,
    MASCARA('121231231', '##.###.###-#') AS rg,
    MASCARA('1212312312', '##.###.###-##') AS cpf,
    MASCARA('12123123123412', '##.###.###/####-##') AS cnpj,
    MASCARA('1212341234', '(##) #####-####') AS telefone,
    MASCARA('12123451234', '(##) #####-####') AS celular
FROM dual;

```

Below the query, the 'Resultado da Consulta' window shows the output of the query. It displays a single row of data with the following values:

IDADE	CEP	RG	CPF	CNPJ	TELEFONE	CELULAR
1	42 12345-123	12.123.123-1	12.123.123-12	12.123.123/1234-12	(12) 1234-1234	(12) 12345-1234

Criação das tabelas

```
# TABELA DE CADASTRO DE PESSOA FISICA

CREATE TABLE PESSOAFISICA (
    ID NUMBER(*, 0) GENERATED ALWAYS AS IDENTITY INCREMENT BY 1 MAXVALUE 999999 MINVALUE 1 CACHE 20 NOT NULL,
    NOME VARCHAR2(100 BYTE) NOT NULL,
    CPF VARCHAR2(20 BYTE) NOT NULL,
    RG VARCHAR2(20 BYTE) NOT NULL,
    TELEFONE VARCHAR2(15 BYTE) NOT NULL,
    CELULAR VARCHAR2(15 BYTE) NOT NULL,
    DATA_NASCIMENTO DATE NOT NULL,
    CONSTRAINT PESSOAFISICA_PK PRIMARY KEY (ID)
    USING INDEX (
        CREATE UNIQUE INDEX PESSOAFISICA_PK ON PESSOAFISICA (ID ASC)
        LOGGING TABLESPACE TBSPC_ALUNOS
        PCTFREE 10
        INITRANS 2
        STORAGE (INITIAL 65536 NEXT 1048576
            MINEXTENTS 1
            MAXEXTENTS UNLIMITED BUFFER_POOL DEFAULT
        ) NOPARALLEL
    ) ENABLE
) LOGGING TABLESPACE TBSPC_ALUNOS
    PCTFREE 10
    INITRANS 1
    STORAGE (INITIAL 65536 NEXT 1048576
        MINEXTENTS 1
        MAXEXTENTS UNLIMITED BUFFER_POOL DEFAULT
    ) NOCOMPRESS NO INMEMORY NOPARALLEL;

# TABELA DE LOGS

CREATE TABLE LOGS (
    ID NUMBER(*, 0) GENERATED ALWAYS AS IDENTITY INCREMENT BY 1 MAXVALUE 999999 MINVALUE 1 CACHE 20 NOT NULL,
    DESCRICAO VARCHAR2(1000 BYTE) NOT NULL,
    FUNCIONALIDADE VARCHAR2(45 BYTE) NOT NULL,
    DATA TIMESTAMP(6) DEFAULT CURRENT_TIMESTAMP NOT NULL,
    ID_REFERENCIA NUMBER(*, 0)
) LOGGING TABLESPACE TBSPC_ALUNOS PCTFREE 10 INITRANS 1 STORAGE (
    INITIAL 65536 NEXT 1048576
    MINEXTENTS 1 MAXEXTENTS UNLIMITED BUFFER_POOL DEFAULT
) NOCOMPRESS NO INMEMORY NOPARALLEL;
```

Criação de Trigger`s

```
# TRIGGER DE REGISTRO DE LOGS ADICIONAR
CREATE OR REPLACE TRIGGER TRG_PESSOAFISICA_LOGS_ADICIONA
    AFTER INSERT ON PESSOAFISICA FOR EACH ROW BEGIN

    INSERT INTO
        LOGS (FUNCIONALIDADE, ID_REFERENCIA, DESCRICAO, DATA)
    VALUES (
        'NOVO CADASTRO DE PESSOA FISICA',
        :new.ID,
        :new.NOME,
        CURRENT_TIMESTAMP
    );

END;

# TRIGGER DE REGISTRO DE LOGS REMOVE
CREATE OR REPLACE TRIGGER TRG_PESSOAFISICA_LOGS_REMOVE
    AFTER DELETE ON PESSOAFISICA FOR EACH ROW BEGIN

    INSERT INTO
        LOGS (FUNCIONALIDADE, ID_REFERENCIA, DESCRICAO, DATA)
    VALUES
        (
            'REMOVE CADASTRO DE PESSOA FISICA',
```

```

        :old.ID,
        :old.NOME,
        CURRENT_TIMESTAMP
    );
END;

# TRIGGER DE REGISTRO DE LOGS ATUALIZA
CREATE OR REPLACE TRIGGER TRG_PESSOA FISICA_LOGS_ATUALIZA
    AFTER UPDATE ON PESSOA FISICA FOR EACH ROW BEGIN

    INSERT INTO
        LOGS (FUNCIONALIDADE, ID_REFERENCIA, DESCRICAO, DATA)
    VALUES (
        'NOVO CADASTRO DE PESSOA FISICA',
        :new.ID,
        :new.NOME,
        CURRENT_TIMESTAMP
    );
END;

```

```

INSERT INTO
    pessoafisica (nome, cpf, rg, telefone, celular, data_nascimento)
values
    (
        'Joao Almenda Brandao',
        '64443404031',
        '451089923',
        '1120823541',
        '11952550331',
        TO_DATE('1979-11-04', 'YYYY-MM-DD')
    );

INSERT INTO
    pessoafisica (nome, cpf, rg, telefone, celular, data_nascimento)
values
    (
        'Elisa Gomes da Silva',
        '98771181067',
        '228949609',
        '1120823541',
        '11952550331',
        TO_DATE('1986-03-11', 'YYYY-MM-DD')
    );

INSERT INTO
    pessoafisica (nome, cpf, rg, telefone, celular, data_nascimento)
values
    (
        'Pedro Alencar dos Santos',
        '05955141030',
        '358771663',
        '1120823541',
        '11952550331',
        TO_DATE('1955-09-07', 'YYYY-MM-DD')
    );

INSERT INTO
    pessoafisica (nome, cpf, rg, telefone, celular, data_nascimento)
values
    (
        'Abigail da Costa',
        '78766609089',
        '355373415',
        '1120823541',
        '11952550331',
        TO_DATE('1965-02-01', 'YYYY-MM-DD')
    );

```

Consultas

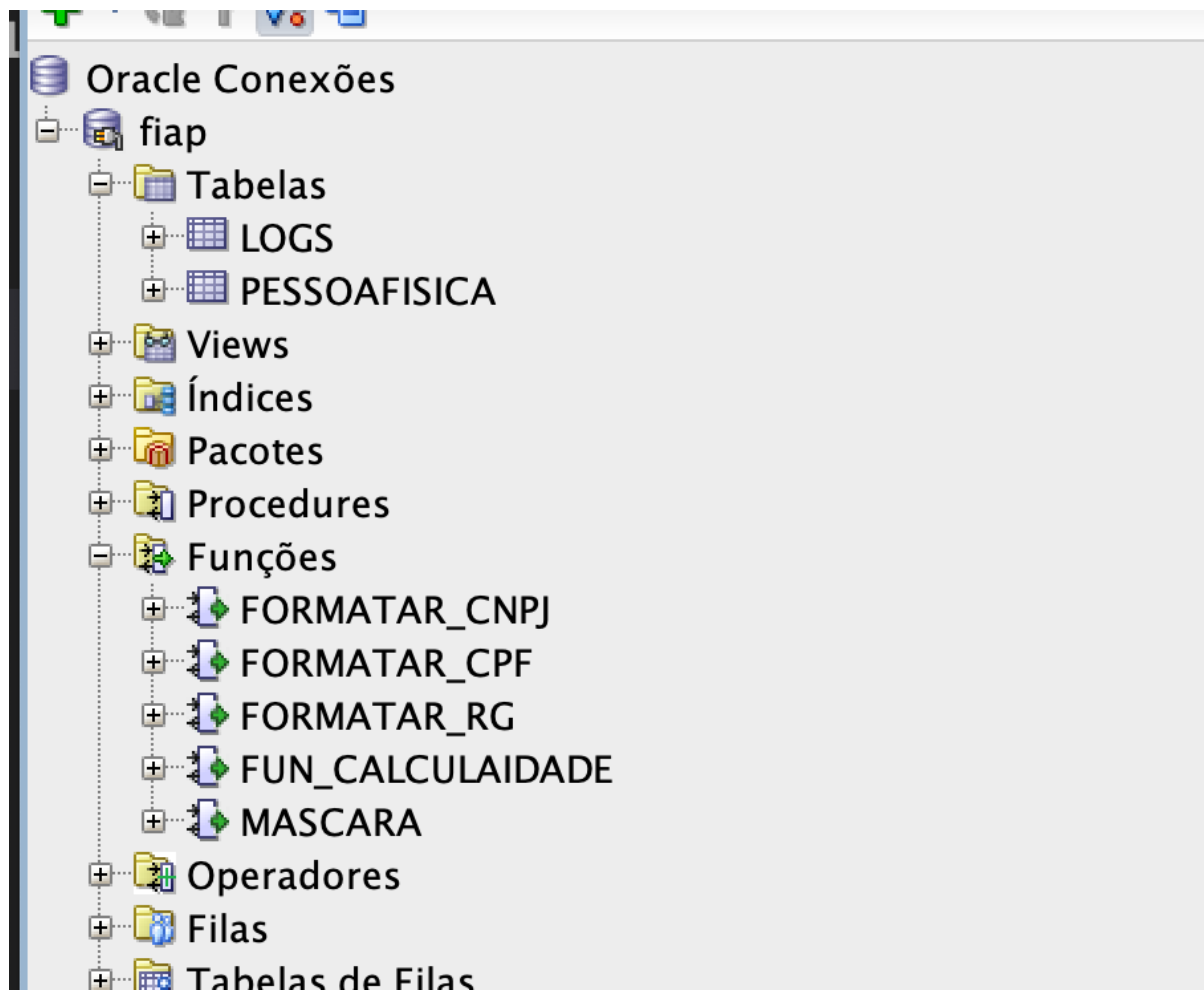
```
# VISUALIZA OS LOGS

SELECT l.* FROM logs l;

# LISTA PESSOA FISICA

SELECT
    PF.*,
    FUN_CALCULAIDADE(PF.DATA_NASCIMENTO) AS IDADE
FROM PESSOAFISICA PF;
```

Evidências e testes



Oracle Conexões

- fiap
 - Tabelas
 - LOGS
 - PESSOAFISICA
 - Views
 - Índices
 - Pacotes
 - Procedures
 - Funções
 - FORMATAR_CNPJ
 - FORMATAR_CPF
 - FORMATAR_RG
 - FUN_CALCULAIDADE
 - MASCARA
 - Operadores
 - Filas
 - Tabelas de Filas
 - Triggers
 - TRG_PESSOAFISICA_LOGS
 - Tipos
 - Sequências
 - Views Materializadas
 - Logs de Views Materializadas
 - Sinônimos
 - Sinônimos Públicos

Relatórios

Todos os Relatórios

Planilha Query Builder

```
SELECT
  PF.ID,
  PF.NOME,
  FORMATAR_CPF(PF.CPF) AS CPF,
  FUN_CALCULAIDADE(PF.DATA_NASCIMENTO) AS IDADE
FROM PESSOAFISICA PF;
```

Saída do Script x Resultado da Consulta x

SQL Todas as Linhas Extraídas: 4 em 0,069 segundos

ID	NOME	CPF	IDADE
1	1 Joao Almenda Brandao	644.434.040-31	42
2	2 Elisa Gomes da Silva	987.711.810-67	36
3	3 Pedro Alencar dos Santos	059.551.410-30	67
4	5 Abigail da Costa	787.666.090-89	57

Oracle Conexões

- fiap
 - Tabelas
 - LOGS
 - PESSOAFISICA
 - Views
 - Índices
 - Pacotes
 - Procedures
 - Funções
 - FORMATAR_CNPJ
 - FORMATAR_CPF
 - FORMATAR_RG
 - FUN_CALCULAIDADE
 - MASCARA
 - Operadores
 - Filas
 - Tabelas de Filas
 - Triggers
 - TRG_PESSOAFISICA_LOGS
 - Tipos
 - Sequências
 - Views Materializadas
 - Logs de Views Materializadas

Planilha Query Builder

```
SELECT
  FUN_CALCULAIDADE(TO_DATE('1979-11-04', 'YYYY-MM-DD')) AS IDADE,
  MASCARA('12345123', '#####-###') AS cep,
  MASCARA('121231231', '##.###-###-#') AS rg,
  MASCARA('1212312312', '##.###-###-##') AS cpf,
  MASCARA('12123123123412', '##.###-###/#####-##') AS cnpj,
  MASCARA('1212341234', '(##) #####-####') AS telefone,
  MASCARA('12123451234', '(##) #####-#####') AS celular
FROM dual;
```

Saída do Script x Resultado da Consulta x

SQL Todas as Linhas Extraídas: 1 em 0,087 segundos

IDADE	CEP	RG	CPF	CNPJ	TELEFONE	CELULAR
42	12345-123	12.123.123-1	12.123.123-12	12.123.123/1234-12	(12) 1234-1234	(12) 12345-1234