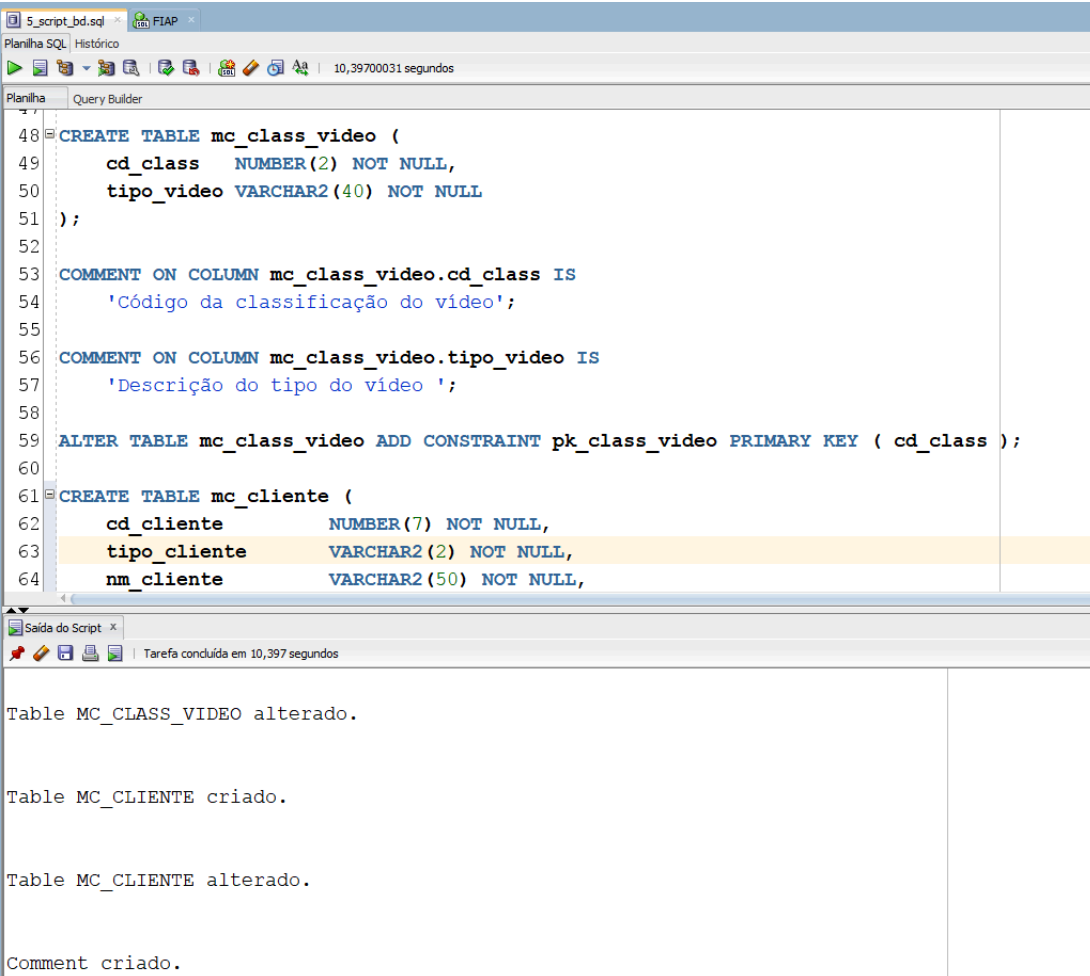


EVIDÊNCIAS DE IMPLANTAÇÃO



The screenshot shows a SQL script execution window with the following content:

```
5_script_bd.sql | FIAP
Planilha SQL | Histórico
10,39700031 segundos

Query Builder

48 CREATE TABLE mc_class_video (
49     cd_class    NUMBER(2) NOT NULL,
50     tipo_video  VARCHAR2(40) NOT NULL
51 );
52
53 COMMENT ON COLUMN mc_class_video.cd_class IS
54     'Código da classificação do vídeo';
55
56 COMMENT ON COLUMN mc_class_video.tipo_video IS
57     'Descrição do tipo do vídeo ';
58
59 ALTER TABLE mc_class_video ADD CONSTRAINT pk_class_video PRIMARY KEY ( cd_class );
60
61 CREATE TABLE mc_cliente (
62     cd_cliente    NUMBER(7) NOT NULL,
63     tipo_cliente  VARCHAR2(2) NOT NULL,
64     nm_cliente    VARCHAR2(50) NOT NULL,
```

Salida do Script x

Tarefa concluída em 10,397 segundos

Table MC_CLASS_VIDEO alterado.

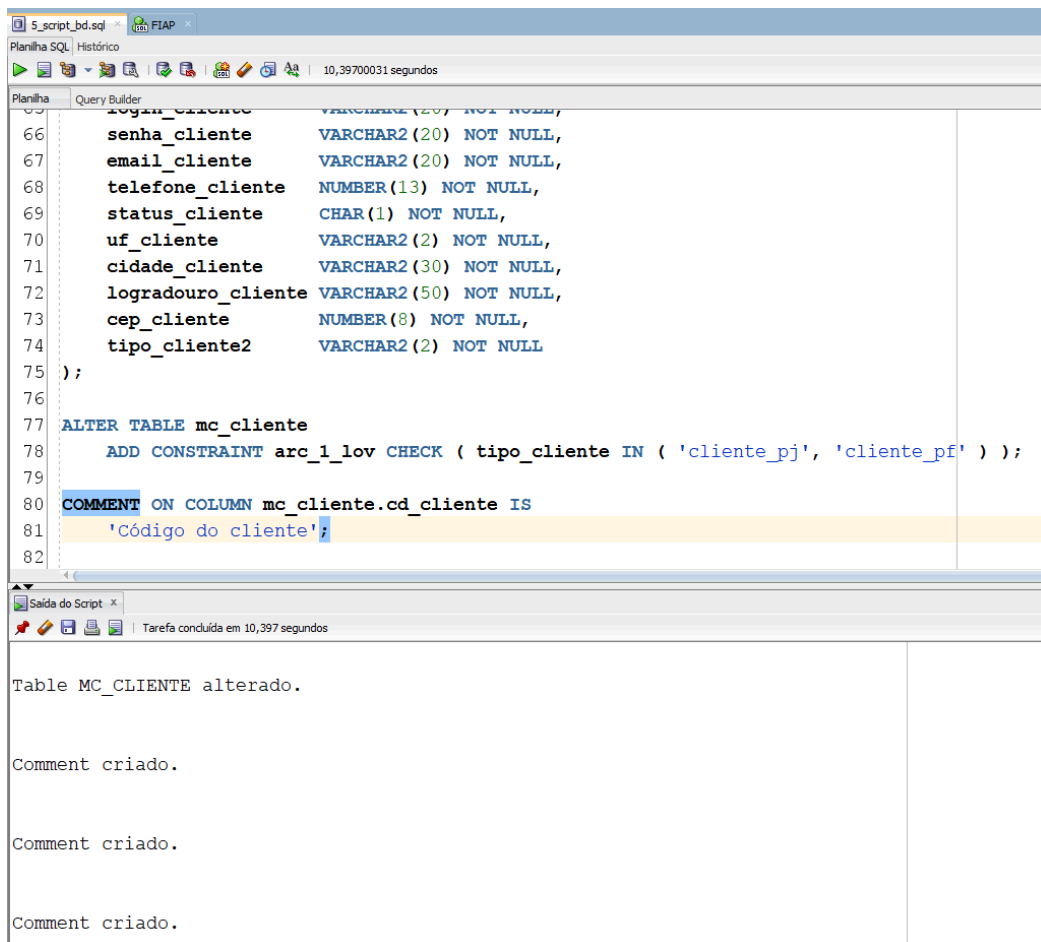
Table MC_CLIENTE criado.

Table MC_CLIENTE alterado.

Comment criado.

Nosso código se inicia com a tabela de classificação de vídeo que determina a quantidade de tempo em que um vídeo foi reproduzido.

Os comentários necessários para descrever cada coluna foram realizados em cada atributo nesta tabela assim como em todas as tabelas adiantes.



```
5_script_bd.sql x FIAP
Planilha SQL Histórico
10,39700031 segundos

Planilha Query Builder
66 login_cliente VARCHAR2(20) NOT NULL,
67 senha_cliente VARCHAR2(20) NOT NULL,
68 email_cliente VARCHAR2(20) NOT NULL,
69 telefone_cliente NUMBER(13) NOT NULL,
70 status_cliente CHAR(1) NOT NULL,
71 uf_cliente VARCHAR2(2) NOT NULL,
72 cidade_cliente VARCHAR2(30) NOT NULL,
73 logradouro_cliente VARCHAR2(50) NOT NULL,
74 cep_cliente NUMBER(8) NOT NULL,
75 tipo_cliente2 VARCHAR2(2) NOT NULL
76 );
77
78 ALTER TABLE mc_cliente
79 ADD CONSTRAINT arc_1_lov CHECK ( tipo_cliente IN ( 'cliente_pj', 'cliente_pf' ) );
80
81 COMMENT ON COLUMN mc_cliente.cd_cliente IS
82 'Código do cliente';

Saída do Script x
Tarefa concluída em 10,397 segundos

Table MC_CLIENTE alterado.

Comment criado.

Comment criado.

Comment criado.
```

Aqui uma das principais tabelas, onde ficam guardadas a maioria das informações dos clientes da Melhores Compras, onde guarda informações importantes como nome, login, senha, endereço, status e etc.

The screenshot shows the SQL Developer interface with a script named '5_script_bd.sql'. The script contains five SQL comments for columns in the 'mc_cliente' table. The comments are as follows:

- Line 83: `COMMENT ON COLUMN mc_cliente.tipo_cliente IS`
- Line 84: `'Define se o cliente é pessoa física (PF) ou jurídica (PJ)';`
- Line 86: `COMMENT ON COLUMN mc_cliente.nm_cliente IS`
- Line 87: `'Nome completo do cliente';`
- Line 89: `COMMENT ON COLUMN mc_cliente.login_cliente IS`
- Line 90: `'E-mail pessoal do cliente que permite o acesso à plataforma';`
- Line 92: `COMMENT ON COLUMN mc_cliente.senha_cliente IS`
- Line 93: `'Senha pessoal do cliente que permite o acesso à plataforma';`
- Line 95: `COMMENT ON COLUMN mc_cliente.email_cliente IS`
- Line 96: `'E-mail pessoal do cliente';`
- Line 98: `COMMENT ON COLUMN mc_cliente.telefone_cliente IS`
- Line 99: `'Telefone pessoal do cliente';`

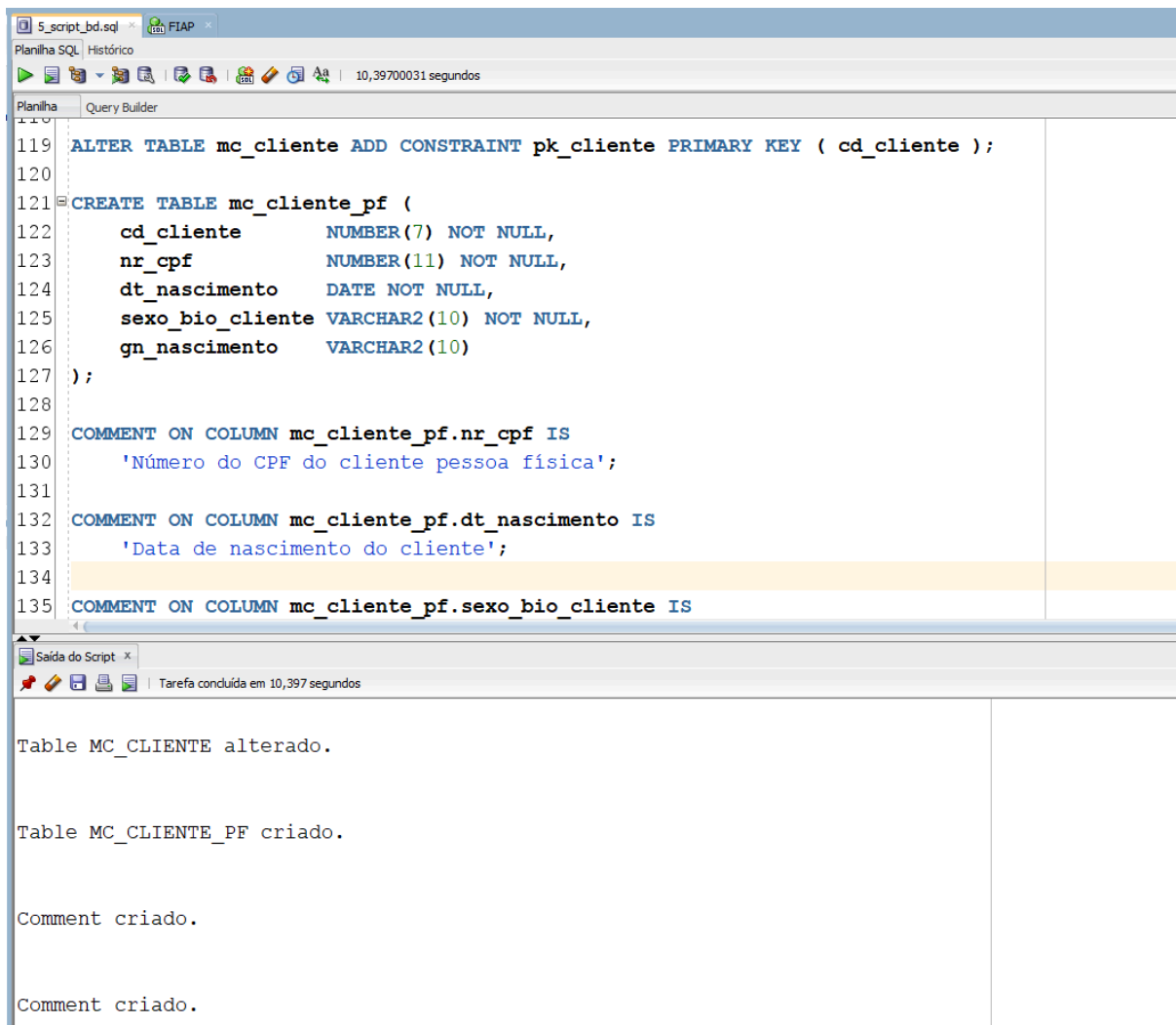
The output window at the bottom shows the result of the script execution: 'Comment criado.' repeated four times, corresponding to the first four comments in the script.

The screenshot shows the SQL Developer interface with a script named '5_script_bd.sql'. The script contains five SQL comments for columns in the 'mc_cliente' table. The comments are as follows:

- Line 101: `COMMENT ON COLUMN mc_cliente.status_cliente IS`
- Line 102: `'Status do cliente, podendo ser ATIVO ou INATIVO';`
- Line 104: `COMMENT ON COLUMN mc_cliente.uf_cliente IS`
- Line 105: `'Sigla do UF do Cliente ';`
- Line 107: `COMMENT ON COLUMN mc_cliente.cidade_cliente IS`
- Line 108: `'Nome da cidade do Cliente';`
- Line 110: `COMMENT ON COLUMN mc_cliente.logradouro_cliente IS`
- Line 111: `'Logradouro do Cliente';`
- Line 113: `COMMENT ON COLUMN mc_cliente.cep_cliente IS`
- Line 114: `'Número de CEP do cliente';`
- Line 116: `COMMENT ON COLUMN mc_cliente.tipo_cliente2 IS`
- Line 117: `'Define se é pessoa física (PF) ou jurídica (PJ)';`

The output window at the bottom shows the result of the script execution: 'Comment criado.' repeated four times, corresponding to the first four comments in the script.

Comentários feitos para descrever cada coluna da tabela MC_CLIENTE.



The screenshot shows a SQL script editor with a toolbar at the top. The script contains the following SQL commands:

```
119 ALTER TABLE mc_cliente ADD CONSTRAINT pk_cliente PRIMARY KEY ( cd_cliente );
120
121 CREATE TABLE mc_cliente_pf (
122     cd_cliente      NUMBER(7) NOT NULL,
123     nr_cpf          NUMBER(11) NOT NULL,
124     dt_nascimento   DATE NOT NULL,
125     sexo_bio_cliente VARCHAR2(10) NOT NULL,
126     gn_nascimento   VARCHAR2(10)
127 );
128
129 COMMENT ON COLUMN mc_cliente_pf.nr_cpf IS
130     'Número do CPF do cliente pessoa física';
131
132 COMMENT ON COLUMN mc_cliente_pf.dt_nascimento IS
133     'Data de nascimento do cliente';
134
135 COMMENT ON COLUMN mc_cliente_pf.sexo_bio_cliente IS
```

Below the script, the output of the execution is displayed:

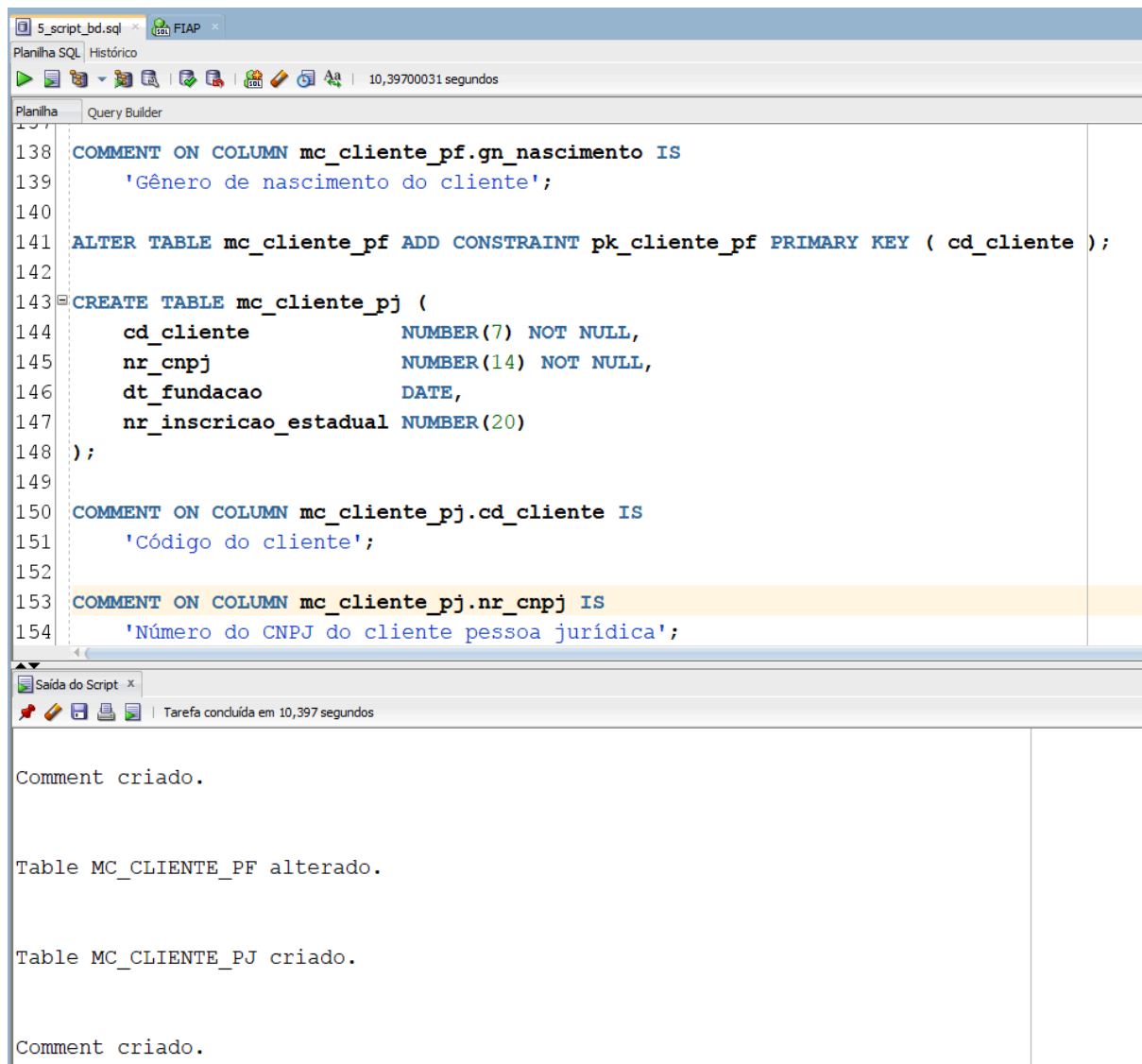
```
Table MC_CLIENTE alterado.

Table MC_CLIENTE_PF criado.

Comment criado.

Comment criado.
```

Criação da tabela responsável por guardar as informações dos clientes que são pessoa física e necessitam de uma tabela a parte que faz herança com a MC_CLIENTE.



The screenshot shows a SQL script editor with a toolbar at the top. The script contains several SQL commands for creating and modifying tables and comments. The output window at the bottom shows the results of these commands.

```
138 COMMENT ON COLUMN mc_cliente_pf.gn_nascimento IS
139     'Gênero de nascimento do cliente';
140
141 ALTER TABLE mc_cliente_pf ADD CONSTRAINT pk_cliente_pf PRIMARY KEY ( cd_cliente );
142
143 CREATE TABLE mc_cliente_pj (
144     cd_cliente          NUMBER(7) NOT NULL,
145     nr_cnpj              NUMBER(14) NOT NULL,
146     dt_fundacao          DATE,
147     nr_inscricao_estadual NUMBER(20)
148 );
149
150 COMMENT ON COLUMN mc_cliente_pj.cd_cliente IS
151     'Código do cliente';
152
153 COMMENT ON COLUMN mc_cliente_pj.nr_cnpj IS
154     'Número do CNPJ do cliente pessoa jurídica';
```

Output:

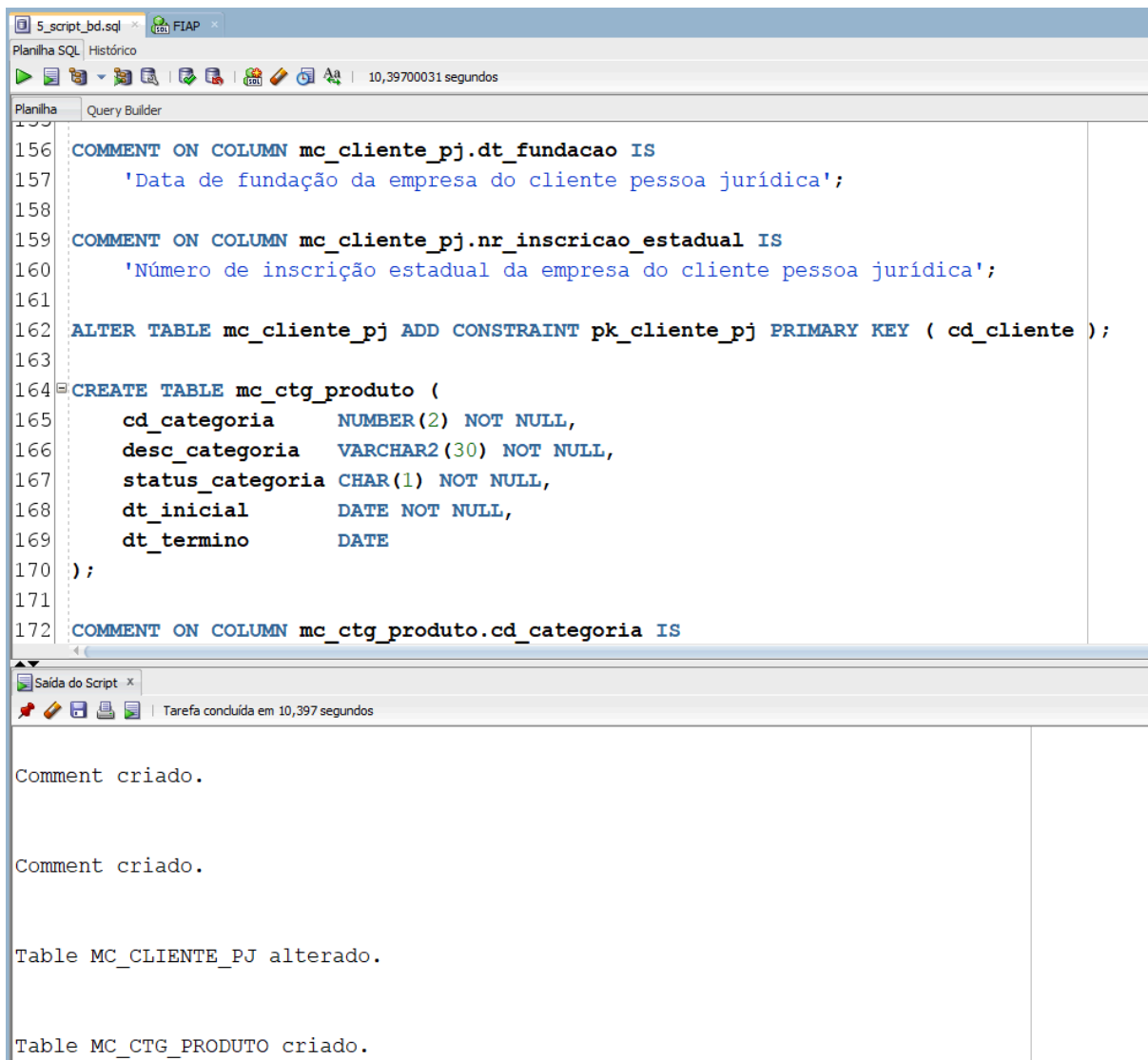
```
Comment criado.

Table MC_CLIENTE_PF alterado.

Table MC_CLIENTE_PJ criado.

Comment criado.
```

Clientes que possuem CNPJ também têm uma tabela exclusiva que faz herança com a tabela de clientes da Melhores Compras.



The screenshot displays the SQL Developer interface. The top pane, titled 'Query Builder', contains a SQL script with the following lines:

```
156 COMMENT ON COLUMN mc_cliente_pj.dt_fundacao IS
157     'Data de fundação da empresa do cliente pessoa jurídica';
158
159 COMMENT ON COLUMN mc_cliente_pj.nr_inscricao_estadual IS
160     'Número de inscrição estadual da empresa do cliente pessoa jurídica';
161
162 ALTER TABLE mc_cliente_pj ADD CONSTRAINT pk_cliente_pj PRIMARY KEY ( cd_cliente );
163
164 CREATE TABLE mc_ctg_produto (
165     cd_categoria      NUMBER(2) NOT NULL,
166     desc_categoria    VARCHAR2(30) NOT NULL,
167     status_categoria  CHAR(1) NOT NULL,
168     dt_inicial        DATE NOT NULL,
169     dt_termino        DATE
170 );
171
172 COMMENT ON COLUMN mc_ctg_produto.cd_categoria IS
```

The bottom pane, titled 'Saída do Script', shows the execution results:

```
Comment criado.

Comment criado.

Table MC_CLIENTE_PJ alterado.

Table MC_CTG_PRODUTO criado.
```

A tabela categoria do produto é de suma importância, pois guarda detalhes sobre os produtos do e-commerce, a descrição da categoria dos produtos foram definidas como UNIQUE respeitando as regras de negócios.

The image shows a screenshot of a SQL script editor and its output window. The editor window, titled "5_script_bd.sql", contains the following SQL code:

```
174  
175 COMMENT ON COLUMN mc_ctg_produto.desc_categoria IS  
176     'Descrição da categoria';  
177  
178 COMMENT ON COLUMN mc_ctg_produto.status_categoria IS  
179     'Status da categoria';  
180  
181 COMMENT ON COLUMN mc_ctg_produto.dt_inicial IS  
182     'Data do início';  
183  
184 COMMENT ON COLUMN mc_ctg_produto.dt_termino IS  
185     'Data do término';  
186  
187 ALTER TABLE mc_ctg_produto  
188     ADD CONSTRAINT ck_mc_ctg_produto_status CHECK ( status_categoria = 'A'  
189                                                     OR dt_termino IS NOT NULL );  
190
```

The output window, titled "Saída do Script", displays the following messages:

```
Comment criado.  
  
Comment criado.  
  
Table MC_CLIENTE_PJ alterado.  
  
Table MC_CTG_PRODUTO criado.
```

Comentários detalhados sobre a tabela, que são de extrema importância para o entendimento dos atributos relacionados ao produto.

The screenshot shows a SQL script editor with a tab titled '5_script_bd.sql'. The editor contains the following SQL commands:

```
187 ALTER TABLE mc_ctg_produto
188     ADD CONSTRAINT ck_mc_ctg_produto_status CHECK ( status_categoria = 'A'
189                                                    OR dt_termino IS NOT NULL );
190
191 ALTER TABLE mc_ctg_produto ADD CONSTRAINT pk_ctg_produto PRIMARY KEY ( cd_categoria );
192
193 ALTER TABLE mc_ctg_produto ADD CONSTRAINT un_mc_ctg_produto_desc UNIQUE ( desc_categoria );
194
195 CREATE SEQUENCE sequence_ctg_produto
196     START WITH 1
197     INCREMENT BY 1
198     NOCACHE
199     ORDER;
200
201 CREATE TABLE mc_funcionario (
202     cd_func      NUMBER(4) NOT NULL,
203     nm_func      VARCHAR2(20) NOT NULL,
```

Below the script, the output window shows the results of the execution:

```
Table MC_CTG_PRODUTO criado.

Comment criado.

Comment criado.

Comment criado.
```

Grande destaque para nossas constraints CHECK, validando o status e a data de término para identificar se o produto atende os requisitos, visualizando vemos também a sequência da regra de negócio 5 que foi criada.

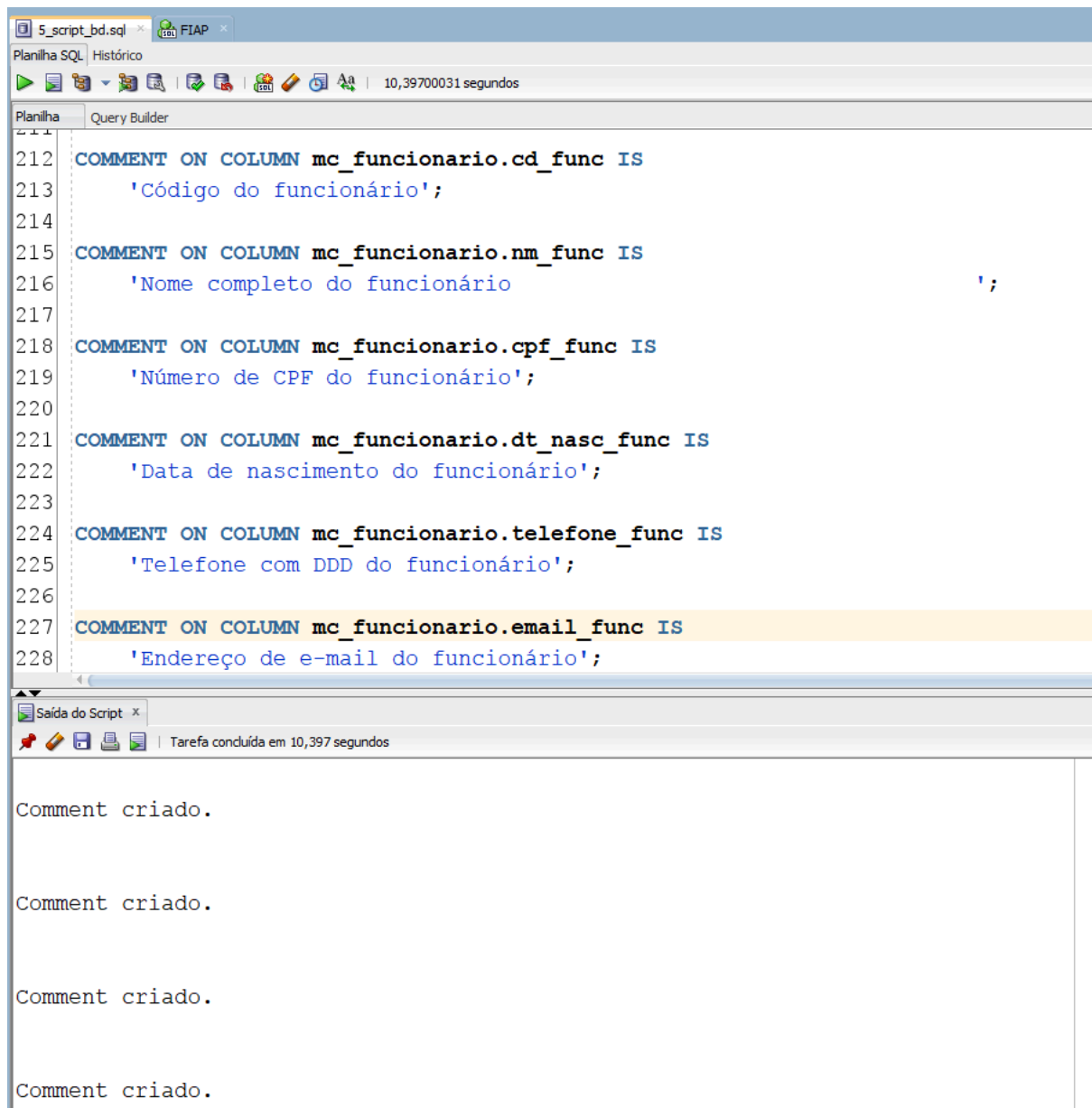
The screenshot shows a SQL script execution window with two panes. The top pane, titled 'Query Builder', contains the following SQL code:

```
201 CREATE TABLE mc_funcionario (  
202     cd_func      NUMBER(4) NOT NULL,  
203     nm_func      VARCHAR2(20) NOT NULL,  
204     cpf_func     NUMBER(11) NOT NULL,  
205     dt_nasc_func DATE NOT NULL,  
206     telefone_func NUMBER(13) NOT NULL,  
207     email_func   VARCHAR2(20) NOT NULL,  
208     cargo_func   VARCHAR2(24) NOT NULL,  
209     depto_func   VARCHAR2(40) NOT NULL  
210 );  
211  
212 COMMENT ON COLUMN mc_funcionario.cd_func IS  
213     'Código do funcionário';  
214  
215 COMMENT ON COLUMN mc_funcionario.nm_func IS  
216     'Nome completo do funcionário';  
217
```

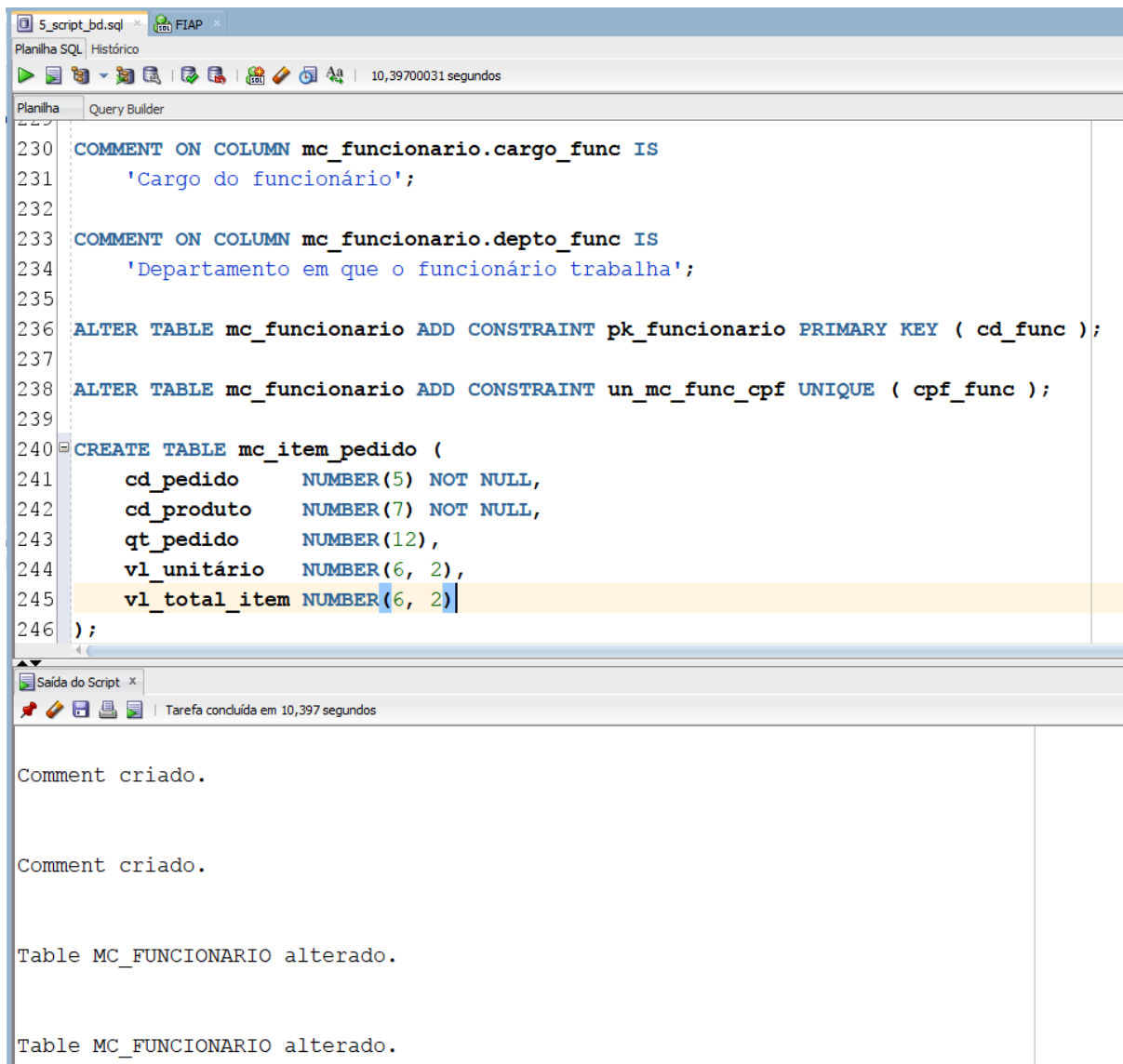
The bottom pane, titled 'Saída do Script', shows the output of the script execution:

```
Table MC_FUNCIONARIO criado.  
  
Comment criado.  
  
Comment criado.  
  
Comment criado.
```

No print acima estão os atributos dos funcionários que têm um papel forte no SAC como uma FK para identificação a qual funcionário pertence o chamado.



Descrição dos atributos dos funcionários.



The screenshot shows a SQL script execution window with two panes. The top pane, titled 'Query Builder', contains the following SQL script:

```
230 COMMENT ON COLUMN mc_funcionario.cargo_func IS
231     'Cargo do funcionário';
232
233 COMMENT ON COLUMN mc_funcionario.depto_func IS
234     'Departamento em que o funcionário trabalha';
235
236 ALTER TABLE mc_funcionario ADD CONSTRAINT pk_funcionario PRIMARY KEY ( cd_func );
237
238 ALTER TABLE mc_funcionario ADD CONSTRAINT un_mc_func_cpf UNIQUE ( cpf_func );
239
240 CREATE TABLE mc_item_pedido (
241     cd_pedido     NUMBER(5) NOT NULL,
242     cd_produto    NUMBER(7) NOT NULL,
243     qt_pedido     NUMBER(12),
244     vl_unitário   NUMBER(6, 2),
245     vl_total_item NUMBER(6, 2)
246 );
```

The bottom pane, titled 'Saída do Script', shows the output of the script execution:

```
Comment criado.

Comment criado.

Table MC_FUNCIONARIO alterado.

Table MC_FUNCIONARIO alterado.
```

Uma entidade associativa criada para melhorar o relacionamento entre produto e cliente, onde são identificados o item que está no pedido com uma PF do código do produto e do código do pedido.

The screenshot shows a SQL script editor with a toolbar at the top. The script contains several SQL statements to add comments to columns and to add a primary key constraint to the MC_ITEM_PEDIDO table. The script is as follows:

```
248 COMMENT ON COLUMN mc_item_pedido.cd_pedido IS
249     'Código do pedido';
250
251 COMMENT ON COLUMN mc_item_pedido.cd_produto IS
252     'Código do produto';
253
254 COMMENT ON COLUMN mc_item_pedido.qt_pedido IS
255     'quantidade de pedidos';
256
257 COMMENT ON COLUMN mc_item_pedido.vl_unitário IS
258     'valor unitário';
259
260 COMMENT ON COLUMN mc_item_pedido.vl_total_item IS
261     'valor total do item';
262
263 ALTER TABLE mc_item_pedido ADD CONSTRAINT pk_item_pedido PRIMARY KEY ( cd_pedido,
264                                                                                               cd_produto );
```

Below the script, the output of the script execution is displayed. It shows four messages indicating that comments were created for the specified columns and that the table MC_ITEM_PEDIDO was altered with the primary key constraint.

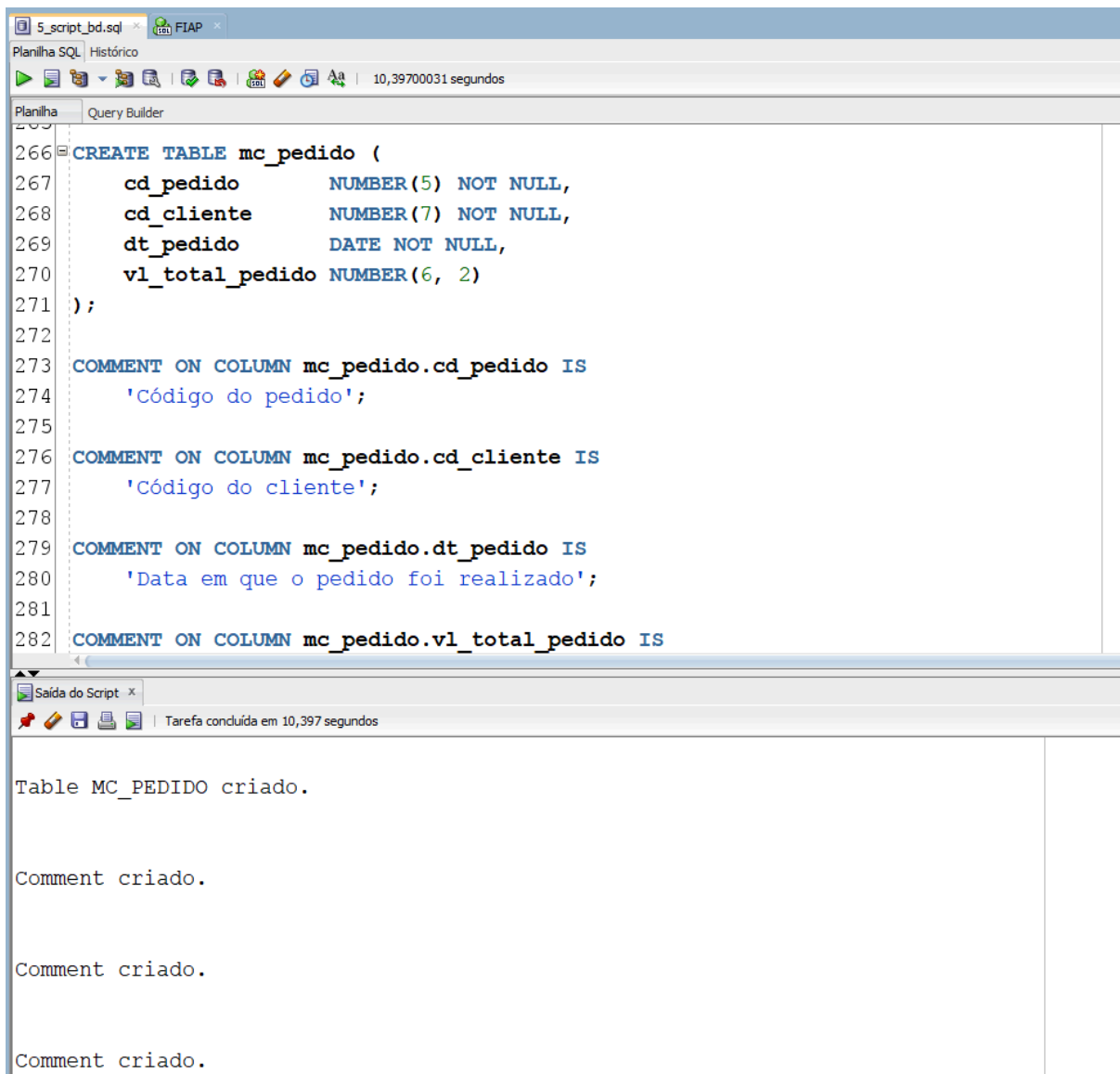
```
Comment criado.

Comment criado.

Comment criado.

Table MC_ITEM_PEDIDO alterado.
```

Detalhes dos atributos do MC_ITEM_PEDIDO.



The screenshot shows a SQL script editor with a file named '5_script_bd.sql' and a connection to 'FIAP'. The script defines a table 'mc_pedido' with four columns: 'cd_pedido' (NUMBER(5) NOT NULL), 'cd_cliente' (NUMBER(7) NOT NULL), 'dt_pedido' (DATE NOT NULL), and 'vl_total_pedido' (NUMBER(6, 2)). It also includes four comments for each column. The execution log at the bottom shows the successful creation of the table and each comment.

```
266 CREATE TABLE mc_pedido (  
267     cd_pedido      NUMBER(5) NOT NULL,  
268     cd_cliente      NUMBER(7) NOT NULL,  
269     dt_pedido       DATE NOT NULL,  
270     vl_total_pedido NUMBER(6, 2)  
271 );  
272  
273 COMMENT ON COLUMN mc_pedido.cd_pedido IS  
274     'Código do pedido';  
275  
276 COMMENT ON COLUMN mc_pedido.cd_cliente IS  
277     'Código do cliente';  
278  
279 COMMENT ON COLUMN mc_pedido.dt_pedido IS  
280     'Data em que o pedido foi realizado';  
281  
282 COMMENT ON COLUMN mc_pedido.vl_total_pedido IS
```

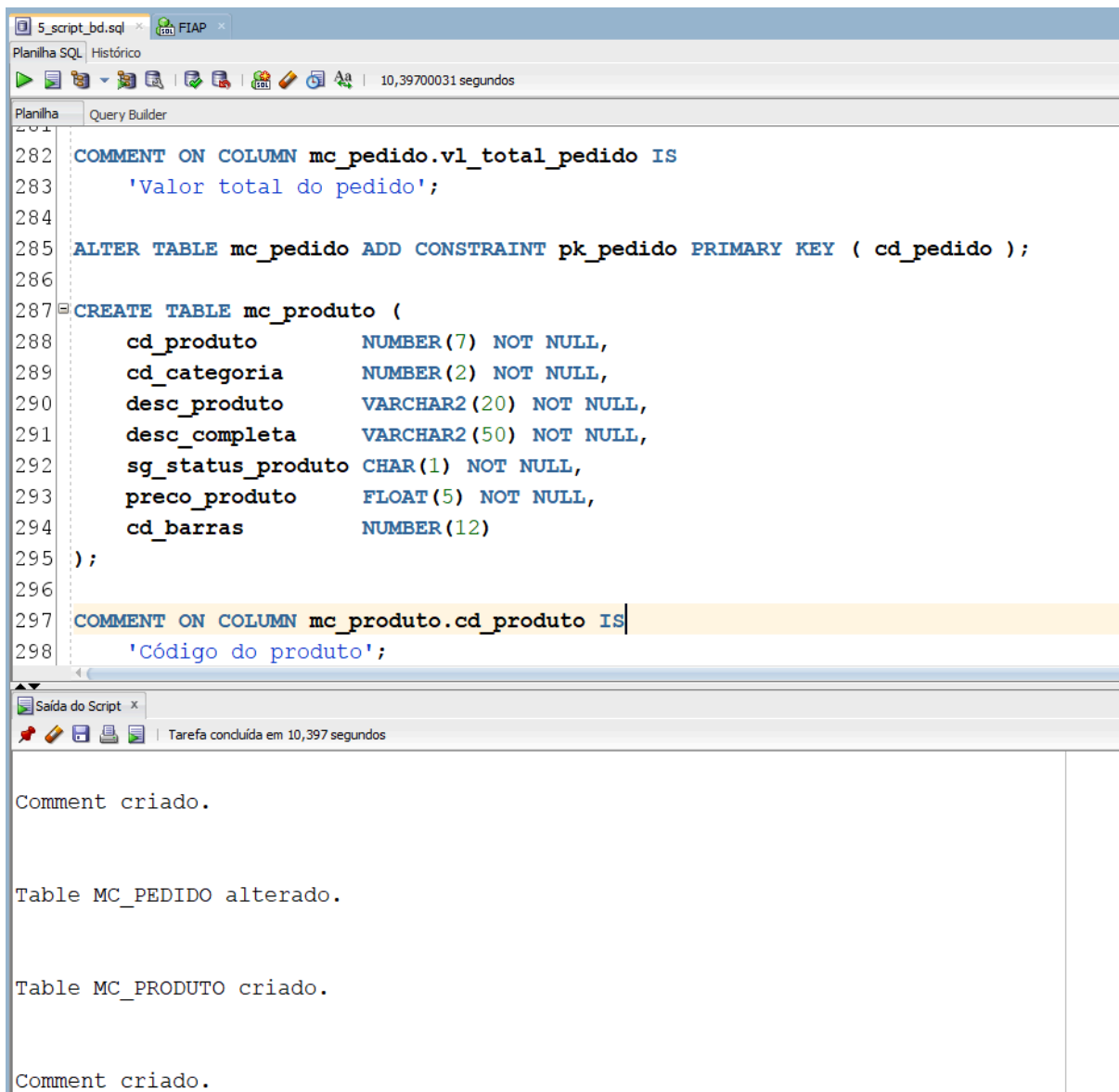
Table MC_PEDIDO criado.

Comment criado.

Comment criado.

Comment criado.

A tabela MC_PEDIDO foi criada como uma entidade associativa entre as tabelas MC_CLIENTE e MC_PRODUTO.



The screenshot displays a SQL script editor window titled "5_script_bd.sql" with a "Planilha SQL" and "Histórico" tab. The script contains the following SQL commands:

```
282 COMMENT ON COLUMN mc_pedido.vl_total_pedido IS
283     'Valor total do pedido';
284
285 ALTER TABLE mc_pedido ADD CONSTRAINT pk_pedido PRIMARY KEY ( cd_pedido );
286
287 CREATE TABLE mc_produto (
288     cd_produto          NUMBER(7) NOT NULL,
289     cd_categoria        NUMBER(2) NOT NULL,
290     desc_produto        VARCHAR2(20) NOT NULL,
291     desc_completa       VARCHAR2(50) NOT NULL,
292     sg_status_produto   CHAR(1) NOT NULL,
293     preco_produto       FLOAT(5) NOT NULL,
294     cd_barras           NUMBER(12)
295 );
296
297 COMMENT ON COLUMN mc_produto.cd_produto IS
298     'Código do produto';
```

The output window, titled "Saída do Script", shows the results of the script execution:

```
Comment criado.

Table MC_PEDIDO alterado.

Table MC_PRODUTO criado.

Comment criado.
```

O produto tem grande função com ligações entre o pedido, categoria do produto e vídeo do produto, ele reúne suas principais informações que estão descritas abaixo:

The screenshot shows a SQL script execution window with the following content:

```
300 COMMENT ON COLUMN mc_produto.desc_produto IS
301     'Nome comum utilizado durante a comercialização do produto';
302
303 COMMENT ON COLUMN mc_produto.desc_completa IS
304     'Descrição com detalhamento do produto comercializado';
305
306 COMMENT ON COLUMN mc_produto.sg_status_produto IS
307     'Produto pode estar ativo (A), inativo (I) ou em prospecção (P)';
308
309 COMMENT ON COLUMN mc_produto.preco_produto IS
310     'Preço unitário atual do produto';
311
312 COMMENT ON COLUMN mc_produto.cd_barras IS
313     'Código de barras do produto padrão EAN13';
314
315 ALTER TABLE mc_produto ADD CONSTRAINT pk_produto PRIMARY KEY ( cd_produto );
316
```

The output window shows the following messages:

```
Comment criado.

Comment criado.

Comment criado.

Table MC_PRODUTO alterado.
```

Comentários feitos para descrever cada coluna da tabela MC_PRODUTO e alteração realizada no atributo cd_produto como PK.

The screenshot shows a SQL script editor with a toolbar and a script titled '5_script_bd.sql'. The script contains the following SQL commands:

```
317 ALTER TABLE mc_produto ADD CONSTRAINT un_mc_produto_desc UNIQUE ( desc_produto );
318
319 CREATE SEQUENCE sequence_cd_produto
320 START WITH 1
321 INCREMENT BY 1
322 NOCACHE
323 ORDER;
324
325 CREATE TABLE mc_sac_chamado (
326     cd_chamado          NUMBER(10) NOT NULL,
327     cd_cliente          NUMBER(7) NOT NULL,
328     cd_func             NUMBER(4) NOT NULL,
329     dt_hora_chamado     DATE NOT NULL,
330     desc_chamado        VARCHAR2(4000) NOT NULL,
331     desc_retorno_chamado VARCHAR2(4000) NOT NULL,
332     cd_tipo_chamado     VARCHAR2(1) NOT NULL,
333     sg_status_chamado   VARCHAR2(1) NOT NULL,
```

Below the script, the 'Saída do Script' (Script Output) window shows the results of the execution:

```
Table MC_PRODUTO alterado.

Sequence SEQUENCE_CD_PRODUTO criado.

Table MC_SAC_CHAMADO criado.

Comment criado.
```

Acima no código ddl foi feita a criação da sequência do código do produto seguindo as regras de negócio.

É possível ver a criação de uma das mais complexas tabelas do modelo, MC_SAC_CHAMADO, que faz herança com os tipos de chamado, ajuda, sugestão e reclamação. Reunindo informações de cliente e funcionário em seus atributos para enriquecer as suas informações sobre o chamado.

The screenshot shows a SQL script editor window titled "5_script_bd.sql" with a "FIAP" logo. The script is being executed, as indicated by the "Tarefa concluída em 10,397 segundos" message at the bottom. The script contains the following SQL statements:

```
333 mc_sac_chamado VARCHAR(1) NOT NULL,  
334 indice_satisfacao NUMBER(3),  
335 dt_hr_atend DATE  
336 );  
337  
338 COMMENT ON COLUMN mc_sac_chamado.cd_chamado IS  
339 'Número de identificador único do SAC';  
340  
341 COMMENT ON COLUMN mc_sac_chamado.cd_cliente IS  
342 'Código do cliente';  
343  
344 COMMENT ON COLUMN mc_sac_chamado.cd_func IS  
345 'Código do funcionário';  
346  
347 COMMENT ON COLUMN mc_sac_chamado.dt_hora_chamado IS  
348 'Data e hora de abertura do chamado';  
349  
350 COMMENT ON COLUMN mc_sac_chamado.desc_chamado IS
```

The output of the script execution is displayed in the "Saída do Script" window, showing four lines of "Comment criado." messages, indicating that the comments were successfully created for each column.

Comentários feitos para descrever cada coluna da tabela MC_SAC_CHAMADO.

The screenshot shows a SQL script editor window titled '5_script_bd.sql' with a 'Planilha SQL' tab. The script contains the following SQL commands:

```
353 COMMENT ON COLUMN mc_sac_chamado.desc_retorno_chamado IS
354     'Descrição detalhada do retorno do chamado pelo funcionário';
355
356 COMMENT ON COLUMN mc_sac_chamado.cd_tipo_chamado IS
357     'Código do tipo de chamado, indicando sua natureza: Ajuda (A), Reclamação (R) ou Sugestão (S)';
358
359 COMMENT ON COLUMN mc_sac_chamado.sg_status_chamado IS
360     'Status do chamado: ativo (A), inativo (I) ou pendente (P)';
361
362 COMMENT ON COLUMN mc_sac_chamado.indice_satisfacao IS
363     'Índice de satisfação atribuído pelo cliente';
364
365 COMMENT ON COLUMN mc_sac_chamado.dt_hr_atend IS
366     'Data e hora do atendimento do chamado';
367
368 ALTER TABLE mc_sac_chamado
369     ADD CONSTRAINT ck_mc_sac_chamado_status CHECK ( sg_status_chamado = 'A'
```

Below the script, the 'Saída do Script' (Script Output) window shows the results of the execution:

```
Comment criado.

Comment criado.

Comment criado.

Table MC_SAC_CHAMADO alterado.
```

Comentários feitos para descrever cada coluna da tabela MC_SAC_CHAMADO.

5_script_bd.sql FIAP

Planilha SQL | Histórico

10,39700031 segundos

Planilha Query Builder

```
369 ALTER TABLE mc_sac_chamado
370 ADD CONSTRAINT ck_mc_sac_chamado_status CHECK ( sg_status_chamado = 'A'
371 OR sg_status_chamado = 'E'
372 OR sg_status_chamado = 'C'
373 OR sg_status_chamado = 'F'
374 OR sg_status_chamado = 'X' );
375
376 ALTER TABLE mc_sac_chamado
377 ADD CONSTRAINT ck_mc_sac_chamado_idc_stf CHECK ( indice_satisfacao IS NULL
378 OR indice_satisfacao BETWEEN 1 AND 10 );
379
380 ALTER TABLE mc_sac_chamado
381 ADD CONSTRAINT ck_mc_sac_chamado_class_tipo CHECK ( cd_tipo_chamado = 'A'
382 OR cd_tipo_chamado = 'R'
383 OR cd_tipo_chamado = 'S' );
384
385 ALTER TABLE mc_sac_chamado ADD CONSTRAINT pk_sac PRIMARY KEY ( cd_chamado );
```

Salida do Script

Tarefa concluída em 10,397 segundos

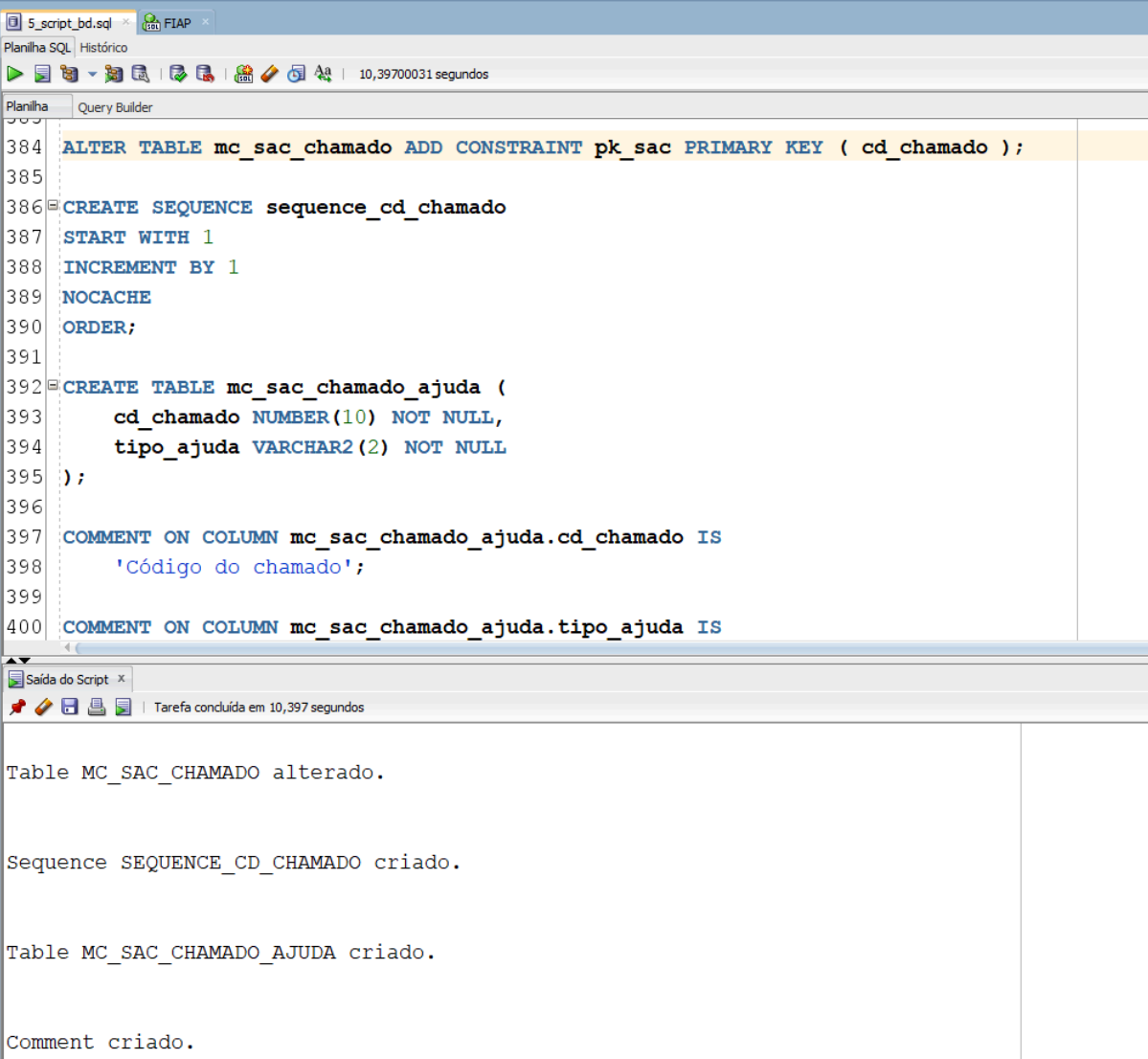
Table MC_SAC_CHAMADO alterado.

Table MC_SAC_CHAMADO alterado.

Table MC_SAC_CHAMADO alterado.

Table MC_SAC_CHAMADO alterado.

Alterações realizadas na tabela MC_SAC_CHAMADO para criação de CHECKs, que são fundamentais para cumprir a RN024 que foi criada pelo grupo.



The screenshot shows a SQL IDE window titled "5_script_bd.sql" with a toolbar and a "Planilha SQL" tab. The script contains the following SQL commands:

```
384 ALTER TABLE mc_sac_chamado ADD CONSTRAINT pk_sac PRIMARY KEY ( cd_chamado );
385
386 CREATE SEQUENCE sequence_cd_chamado
387 START WITH 1
388 INCREMENT BY 1
389 NOCACHE
390 ORDER;
391
392 CREATE TABLE mc_sac_chamado_ajuda (
393     cd_chamado NUMBER(10) NOT NULL,
394     tipo_ajuda VARCHAR2(2) NOT NULL
395 );
396
397 COMMENT ON COLUMN mc_sac_chamado_ajuda.cd_chamado IS
398     'Código do chamado';
399
400 COMMENT ON COLUMN mc_sac_chamado_ajuda.tipo_ajuda IS
```

Below the script, the "Saída do Script" tab shows the execution results:

```
Table MC_SAC_CHAMADO alterado.

Sequence SEQUENCE_CD_CHAMADO criado.

Table MC_SAC_CHAMADO_AJUDA criado.

Comment criado.
```

Sequência incrementada a partir da RN012.

A MC_SAC_CHAMADO_AJUDA entidade que descreve o tipo de ajuda e faz herança com o SAC.

```
S_script_bd.sql FIAP
Planilha SQL Histórico
10,39700031 segundos

Planilha Query Builder
399
400 COMMENT ON COLUMN mc_sac_chamado_ajuda.tipo_ajuda IS
401 'Classificação do chamado Ajuda, podendo ser: Alteração de dados de compras (AD), Atualização cadastral (AC) ou Financeiro (FN)';
402
403 CREATE UNIQUE INDEX mc_sac_chamado_ajuda__idx ON
404 mc_sac_chamado_ajuda (
405 cd_chamado
406 ASC );
407
408 ALTER TABLE mc_sac_chamado_ajuda ADD CONSTRAINT pk_class_ajuda PRIMARY KEY ( cd_chamado,
409 tipo_ajuda );
410
411 CREATE TABLE mc_sac_chamado_reclamacao (
412 cd_chamado NUMBER(10) NOT NULL,
413 tipo_reclamacao VARCHAR2(2) NOT NULL,
414 prioridade_reclamacao VARCHAR2(8) NOT NULL
415 );

Saída do Script x
Tarefa concluída em 10,397 segundos

Comment criado.

INDEX MC_SAC_CHAMADO_AJUDA__IDX criado.

Table MC_SAC_CHAMADO_AJUDA alterado.

Table MC_SAC_CHAMADO_RECLAMACAO criado.
```

Constraint UNIQUE criada para associar o atributo cd_chamado da tabela MC_SAC_CHAMADO como atributo primário e único da tabela MC_SAC_CHAMADO_AJUDA.

```
S_script_bd.sql FIAP
Planilha SQL Histórico
10,39700031 segundos

Planilha Query Builder
417 COMMENT ON COLUMN mc_sac_chamado_reclamacao.cd_chamado IS
418 'Código do chamado';
419
420 COMMENT ON COLUMN mc_sac_chamado_reclamacao.tipo_reclamacao IS
421 'Categoria dos chamados definidos como Reclamação: Suporte técnico (N1) ou Suporte especializado (N2)';
422
423 COMMENT ON COLUMN mc_sac_chamado_reclamacao.prioridade_reclamacao IS
424 'Podendo ser considerado: Baixa (B), Média (M), Alta (A) ou Urgente (U)';
425
426 CREATE UNIQUE INDEX mc_sac_chamado_reclamacao__idx ON
427 mc_sac_chamado_reclamacao (
428 cd_chamado
429 ASC );
430
431 ALTER TABLE mc_sac_chamado_reclamacao ADD CONSTRAINT pk_class_reclamacao PRIMARY KEY ( cd_chamado,
432 tipo_reclamacao );
433

Saída do Script x
Tarefa concluída em 10,397 segundos

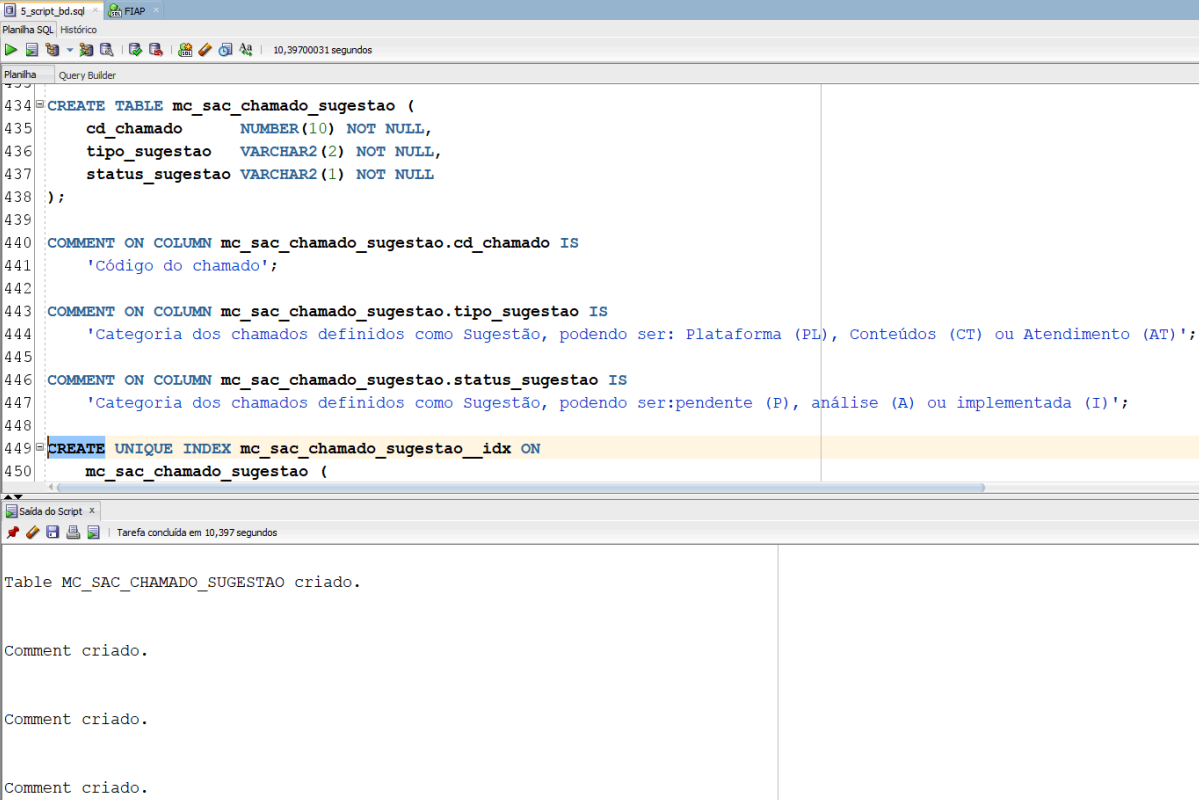
Comment criado.

Comment criado.

INDEX MC_SAC_CHAMADO_RECLAMACAO__IDX criado.

Table MC_SAC_CHAMADO_RECLAMACAO alterado.
```

Comentários feitos para descrever cada coluna da tabela MC_SAC_CHAMADO_RECLAMACAO e criação da constrain UNIQUE associado ao atributo cd_chamado.



The screenshot displays the Oracle SQL Developer environment. The top pane, titled 'Query Builder', contains a SQL script for creating a table and adding comments. The script is as follows:

```
434 CREATE TABLE mc_sac_chamado_sugestao (  
435     cd_chamado      NUMBER(10) NOT NULL,  
436     tipo_sugestao   VARCHAR2(2) NOT NULL,  
437     status_sugestao VARCHAR2(1) NOT NULL  
438 );  
439  
440 COMMENT ON COLUMN mc_sac_chamado_sugestao.cd_chamado IS  
441     'Código do chamado';  
442  
443 COMMENT ON COLUMN mc_sac_chamado_sugestao.tipo_sugestao IS  
444     'Categoria dos chamados definidos como Sugestão, podendo ser: Plataforma (PL), Conteúdos (CT) ou Atendimento (AT)';  
445  
446 COMMENT ON COLUMN mc_sac_chamado_sugestao.status_sugestao IS  
447     'Categoria dos chamados definidos como Sugestão, podendo ser:pendente (P), análise (A) ou implementada (I)';  
448  
449 CREATE UNIQUE INDEX mc_sac_chamado_sugestao_idx ON  
450     mc_sac_chamado_sugestao (  
451     cd_chamado
```

The bottom pane, titled 'Saída do Script', shows the output of the script execution:

```
Table MC_SAC_CHAMADO_SUGESTAO criado.  
  
Comment criado.  
  
Comment criado.  
  
Comment criado.
```

Criação da tabela MC_SAC_CHAMADO_SUGESTAO e os comentários realizados para definir cada um de seus atributos, além da constraint UNIQUE criada para associar com o cd_chamado da tabela MC_SAC_CHAMADO.

The screenshot shows a SQL script editor with a toolbar at the top. The script contains the following SQL commands:

```
452 ASC );
453
454 ALTER TABLE mc_sac_chamado_sugestao ADD CONSTRAINT pk_class_sugestao PRIMARY KEY ( cd_chamado,
455 tipo_sugestao );
456
457 CREATE TABLE mc_video_produto (
458     cd_video          NUMBER(3) NOT NULL,
459     cd_class           NUMBER(2) NOT NULL,
460     cd_produto         NUMBER(7) NOT NULL,
461     dt_cadastro_video  DATE NOT NULL,
462     status_video       CHAR(1) NOT NULL
463 );
464
465 COMMENT ON COLUMN mc_video_produto.cd_video IS
466 'Código do vídeo';
467
468 COMMENT ON COLUMN mc_video_produto.cd_class IS
```

Below the script editor, a 'Saída do Script' (Script Output) window shows the execution results:

```
INDEX MC_SAC_CHAMADO_SUGESTAO__IDX criado.

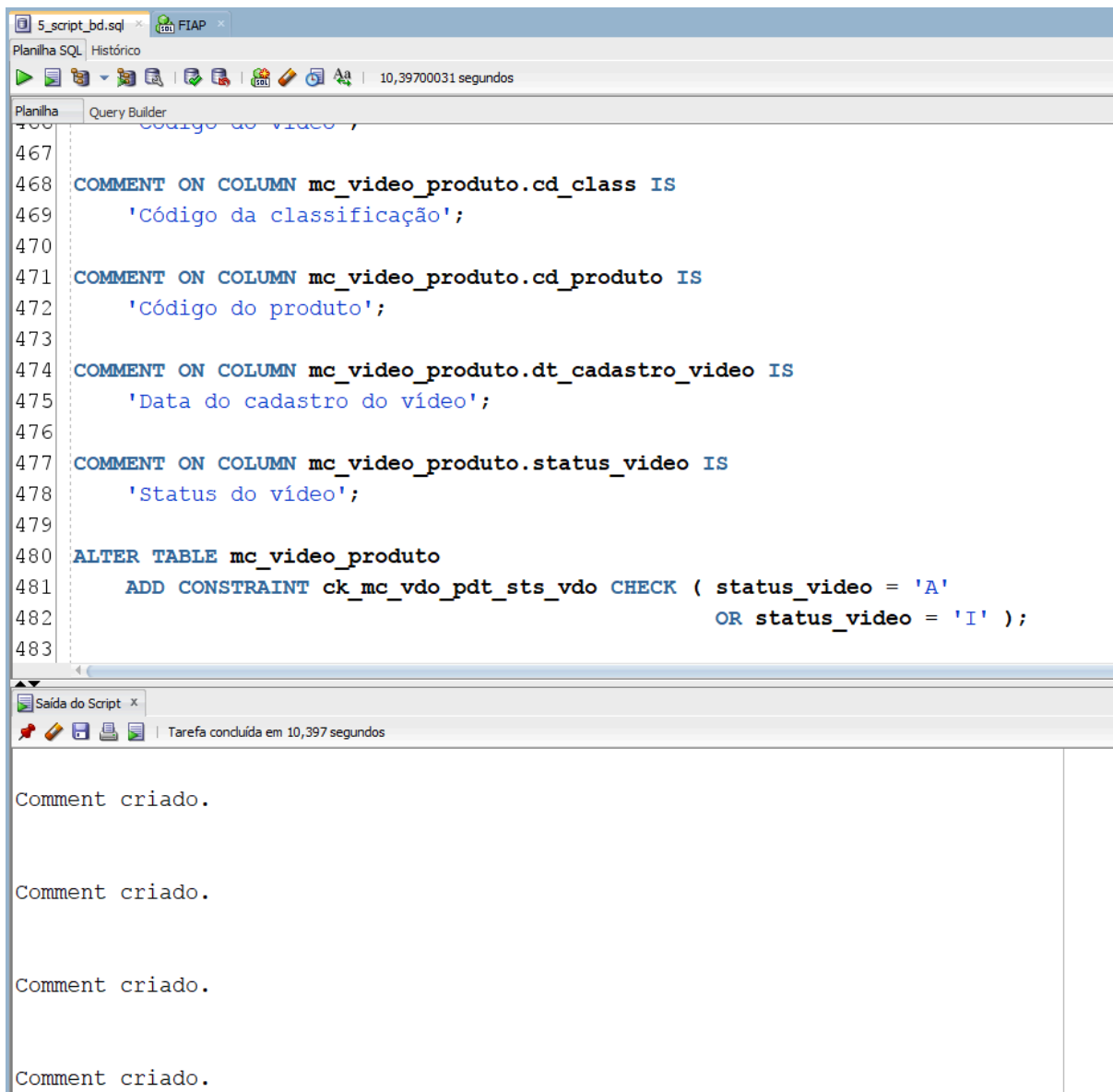
Table MC_SAC_CHAMADO_SUGESTAO alterado.

Table MC_VIDEO_PRODUTO criado.

Comment criado.
```

Acima sendo definido as duas chaves primárias da tabela
MC_SAC_CHAMADO_SUGESTAO.

Uma entidade essencial a MC_VIDEO_PRODUTO também é criada para guardar os vídeos demonstrativos dos produtos e se ligam a visualização e também a classificação dos vídeos que são outras tabelas.



The image shows a screenshot of a SQL script editor window titled "5_script_bd.sql" with a "FIAP" icon. The editor displays a SQL script with line numbers 467 through 483. The script includes comments for columns and an ALTER TABLE statement. Below the editor is an output window titled "Saída do Script" showing the execution results.

```
467
468 COMMENT ON COLUMN mc_video_produto.cd_class IS
469     'Código da classificação';
470
471 COMMENT ON COLUMN mc_video_produto.cd_produto IS
472     'Código do produto';
473
474 COMMENT ON COLUMN mc_video_produto.dt_cadastro_video IS
475     'Data do cadastro do vídeo';
476
477 COMMENT ON COLUMN mc_video_produto.status_video IS
478     'Status do vídeo';
479
480 ALTER TABLE mc_video_produto
481     ADD CONSTRAINT ck_mc_vdo_pdt_sts_vdo CHECK ( status_video = 'A'
482                                                OR status_video = 'I' );
483
```

The output window shows the following results:

```
Comment criado.

Comment criado.

Comment criado.

Comment criado.
```

Detalhes dos atributos da MC_VIDEO_PRODUTO.

The screenshot shows a SQL script editor with a toolbar at the top. The script contains the following SQL statements:

```
484 ALTER TABLE mc_video_produto ADD CONSTRAINT pk_video_produto PRIMARY KEY ( cd_video );
485
486 CREATE TABLE mc_visualizacao_video (
487     cd_visualizacao NUMBER(7) NOT NULL,
488     cd_video          NUMBER(3) NOT NULL,
489     dt_visualizacao  DATE NOT NULL
490 );
491
492 COMMENT ON COLUMN mc_visualizacao_video.cd_visualizacao IS
493     'Código da visualização';
494
495 COMMENT ON COLUMN mc_visualizacao_video.cd_video IS
496     'Código do vídeo';
497
498 COMMENT ON COLUMN mc_visualizacao_video.dt_visualizacao IS
499     'Data de visualização';
500
```

Below the script, the execution output is displayed:

```
Table MC_VIDEO_PRODUTO alterado.

Table MC_VIDEO_PRODUTO alterado.

Table MC_VISUALIZACAO_VIDEO criado.

Comment criado.
```

No código da imagem acima a visualização do produto ligada aos vídeos dos produtos contam visualizações de usuários cadastrados e guardam nesta identidade, é possível ver detalhes sobre nos comentários de suas colunas.

The screenshot shows a SQL script editor with a toolbar at the top. The script contains several SQL statements to alter tables and add constraints. The output window at the bottom shows the successful execution of these statements.

```
501 ALTER TABLE mc_visualizacao_video ADD CONSTRAINT pk_vsz_video PRIMARY KEY ( cd_visualizacao );
502
503 ALTER TABLE mc_sac_chamado
504     ADD CONSTRAINT abrir FOREIGN KEY ( cd_cliente )
505     REFERENCES mc_cliente ( cd_cliente );
506
507 ALTER TABLE mc_sac_chamado
508     ADD CONSTRAINT atender FOREIGN KEY ( cd_func )
509     REFERENCES mc_funcionario ( cd_func );
510
511 ALTER TABLE mc_produto
512     ADD CONSTRAINT categorizar FOREIGN KEY ( cd_categoria )
513     REFERENCES mc_ctg_produto ( cd_categoria );
514
515 ALTER TABLE mc_sac_chamado ajuda
516     ADD CONSTRAINT classificar FOREIGN KEY ( cd_chamado )
517     REFERENCES mc_sac_chamado ( cd_chamado );
```

Table MC_VISUALIZACAO_VIDEO alterado.

Table MC_SAC_CHAMADO alterado.

Table MC_SAC_CHAMADO alterado.

Table MC_PRODUTO alterado.

Definimos com nomes únicos cada constraint FK feita.

```
515 ALTER TABLE mc_sac_chamado_ajuda
516     ADD CONSTRAINT classificar FOREIGN KEY ( cd_chamado )
517     REFERENCES mc_sac_chamado ( cd_chamado );
518
519 ALTER TABLE mc_video_produto
520     ADD CONSTRAINT classificarv1 FOREIGN KEY ( cd_class )
521     REFERENCES mc_class_video ( cd_class );
522
523 ALTER TABLE mc_sac_chamado_sugestao
524     ADD CONSTRAINT classificarv3 FOREIGN KEY ( cd_chamado )
525     REFERENCES mc_sac_chamado ( cd_chamado );
526
527 ALTER TABLE mc_sac_chamado_reclamacao
528     ADD CONSTRAINT classificarv4 FOREIGN KEY ( cd_chamado )
529     REFERENCES mc_sac_chamado ( cd_chamado );
530
531 ALTER TABLE mc_item_pedido
```

Saída do Script x

Tarefa concluída em 10,397 segundos

Table MC_SAC_CHAMADO_AJUDA alterado.

Table MC_VIDEO_PRODUTO alterado.

Table MC_SAC_CHAMADO_SUGESTAO alterado.

Table MC_SAC_CHAMADO_RECLAMACAO alterado.

Alterações realizadas nas tabelas MC_SAC_CHAMADO_AJUDA, MC_VIDEO_PRODUTO, MC_SAC_CHAMADO_SUGESTAO e MC_SAC_CHAMADO_RECLAMACAO para adicionar as respectivas FK necessárias.

The screenshot shows a SQL script editor window titled 'S_script_bd.sql' with a 'Query Builder' tab. The script contains several SQL statements for altering tables and adding foreign key constraints. The output window below shows the execution results, indicating that four tables were successfully altered: MC_ITEM_PEDIDO, MC_VIDEO_PRODUTO, MC_PEDIDO, and MC_VISUALIZACAO_VIDEO.

```
529 REFERENCES mc_sac_chamado ( cd_chamado );
530
531 ALTER TABLE mc_item_pedido
532     ADD CONSTRAINT pertencer FOREIGN KEY ( cd_pedido )
533     REFERENCES mc_pedido ( cd_pedido );
534
535 ALTER TABLE mc_cliente_pf
536     ADD CONSTRAINT pode_ser FOREIGN KEY ( cd_cliente )
537     REFERENCES mc_cliente ( cd_cliente );
538
539 ALTER TABLE mc_cliente_pj
540     ADD CONSTRAINT pode_serv2 FOREIGN KEY ( cd_cliente )
541     REFERENCES mc_cliente ( cd_cliente );
542
543 ALTER TABLE mc_item_pedido
544     ADD CONSTRAINT "pode ter" FOREIGN KEY ( cd_produto )
545     REFERENCES mc_produto ( cd_produto );
546
```

Saida do Script x
Tarefa concluída em 10,397 segundos

Table MC_ITEM_PEDIDO alterado.

Table MC_VIDEO_PRODUTO alterado.

Table MC_PEDIDO alterado.

Table MC_VISUALIZACAO_VIDEO alterado.

Cada constraint FK feita tem seu nome único de acordo com as entidades.

The screenshot shows a SQL script execution window with the following content:

```
547 ALTER TABLE mc_video_produto
548     ADD CONSTRAINT possuir FOREIGN KEY ( cd_produto )
549     REFERENCES mc_produto ( cd_produto );
550
551 ALTER TABLE mc_pedido
552     ADD CONSTRAINT realizar FOREIGN KEY ( cd_cliente )
553     REFERENCES mc_cliente ( cd_cliente );
554
555 ALTER TABLE mc_visualizacao_video
556     ADD CONSTRAINT visualizar FOREIGN KEY ( cd_video )
557     REFERENCES mc_video_produto ( cd_video );
558
559 CREATE OR REPLACE TRIGGER arc_arc_1_mc_cliente_pj BEFORE
560     INSERT OR UPDATE OF cd_cliente ON mc_cliente_pj
561     FOR EACH ROW
562 DECLARE
563     d VARCHAR2 (2) ;
```

The output window shows the following messages:

```
Table MC_VIDEO_PRODUTO alterado.

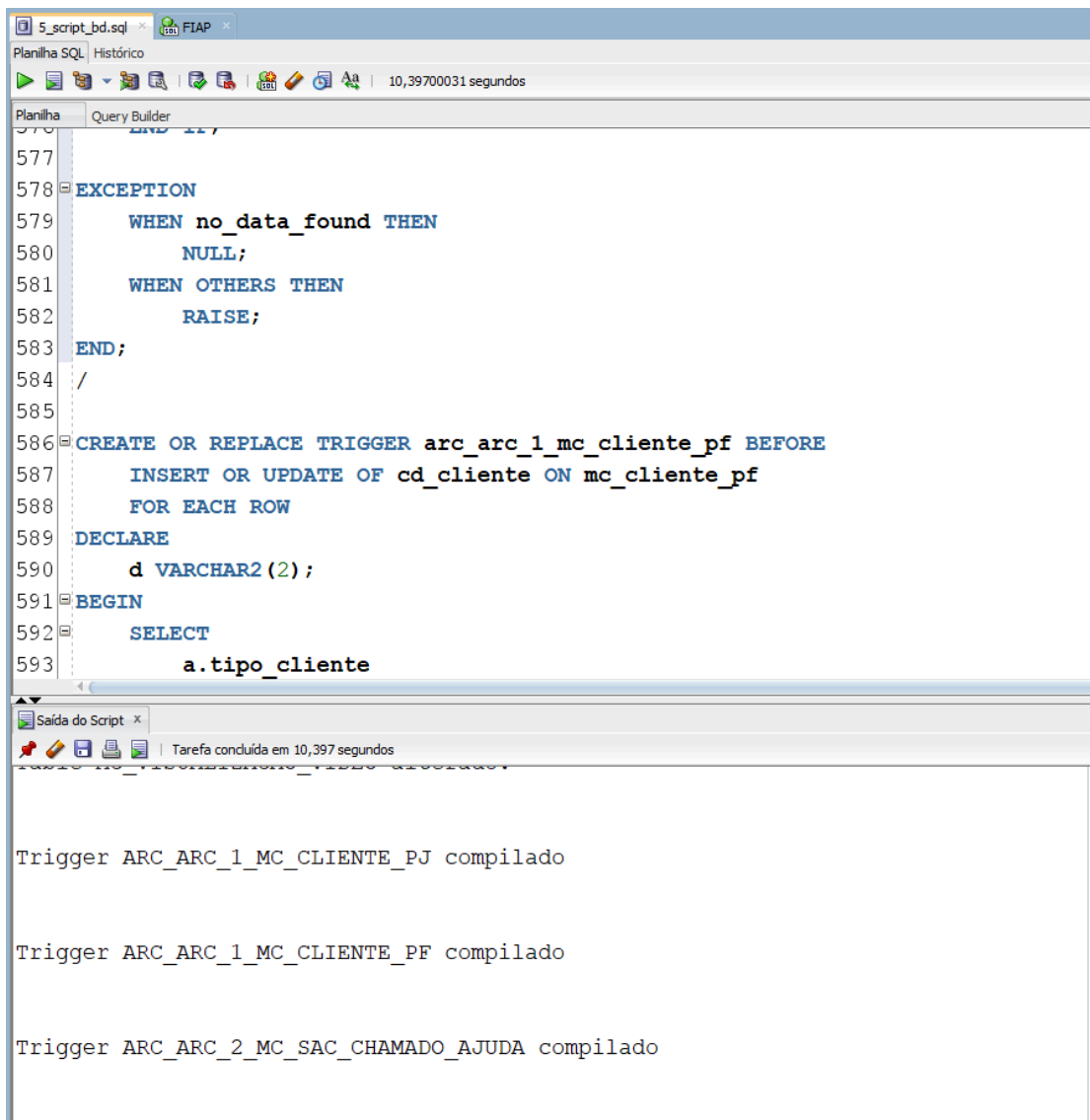
Table MC_PEDIDO alterado.

Table MC_VISUALIZACAO_VIDEO alterado.

Trigger ARC_ARC_1_MC_CLIENTE_PJ compilado
```

Alterações realizadas nas tabelas MC_VIDEO_PRODUTO, MC_PEDIDO e MC_VISUALIZACAO_VIDEO a fim de adicionar as constraints FK correspondente a cada tabela.

Criação do TRIGGER relacionado ao arco de herança entre a tabela MC_CLIENTE e as tabelas herdeiras MC_CLIENTE_PF e MC_CLIENTE_PJ.

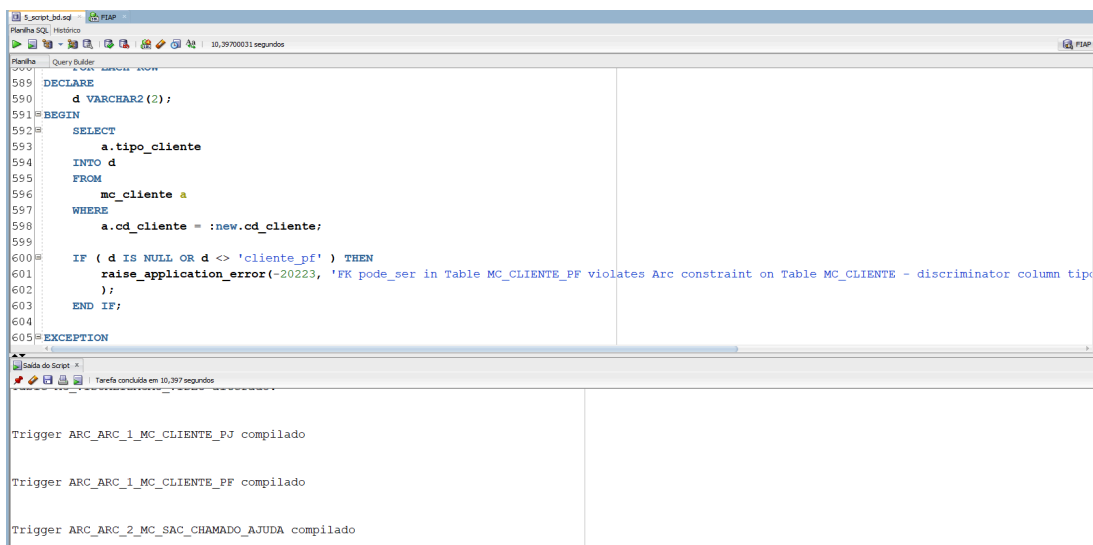


```
577
578 EXCEPTION
579     WHEN no_data_found THEN
580         NULL;
581     WHEN OTHERS THEN
582         RAISE;
583 END;
584 /
585
586 CREATE OR REPLACE TRIGGER arc_arc_1_mc_cliente_pf BEFORE
587     INSERT OR UPDATE OF cd_cliente ON mc_cliente_pf
588     FOR EACH ROW
589 DECLARE
590     d VARCHAR2(2);
591 BEGIN
592     SELECT
593         a.tipo_cliente
```

Trigger ARC_ARC_1_MC_CLIENTE_PJ compilado

Trigger ARC_ARC_1_MC_CLIENTE_PF compilado

Trigger ARC_ARC_2_MC_SAC_CHAMADO_AJUDA compilado



```
589 DECLARE
590     d VARCHAR2(2);
591 BEGIN
592     SELECT
593         a.tipo_cliente
594     INTO d
595     FROM
596         mc_cliente a
597     WHERE
598         a.cd_cliente = :new.cd_cliente;
599
600     IF ( d IS NULL OR d <> 'cliente_pf' ) THEN
601         raise_application_error(-20223, 'FK pode_ser in Table MC_CLIENTE_PF violates Arc constraint on Table MC_CLIENTE - discriminator column tipo_cliente');
602     END IF;
603
604 EXCEPTION
```

Trigger ARC_ARC_1_MC_CLIENTE_PJ compilado

Trigger ARC_ARC_1_MC_CLIENTE_PF compilado

Trigger ARC_ARC_2_MC_SAC_CHAMADO_AJUDA compilado

Continuação do TRIGGER relacionado ao arco de herança entre a tabela MC_CLIENTE e as tabelas herdeiras MC_CLIENTE_PF e MC_CLIENTE_PJ.

The screenshot shows a SQL script editor with a tab labeled '5_script_bd.sql'. The script contains the following SQL code:

```
611 /
612
613 CREATE OR REPLACE TRIGGER arc_arc_2_mc_sac_chamado_ajuda BEFORE
614     INSERT OR UPDATE OF cd_chamado ON mc_sac_chamado_ajuda
615     FOR EACH ROW
616 DECLARE
617     d VARCHAR2(1);
618 BEGIN
619     SELECT
620         a.cd_tipo_chamado
621     INTO d
622     FROM
623         mc_sac_chamado a
624     WHERE
625         a.cd_chamado = :new.cd_chamado;
626
627     IF ( d IS NULL OR d <> 'tipo_ajuda' ) THEN
```

Below the script editor, there is a 'Saída do Script' (Script Output) window showing the execution results:

```
Trigger ARC_ARC_2_MC_SAC_CHAMADO_AJUDA compilado

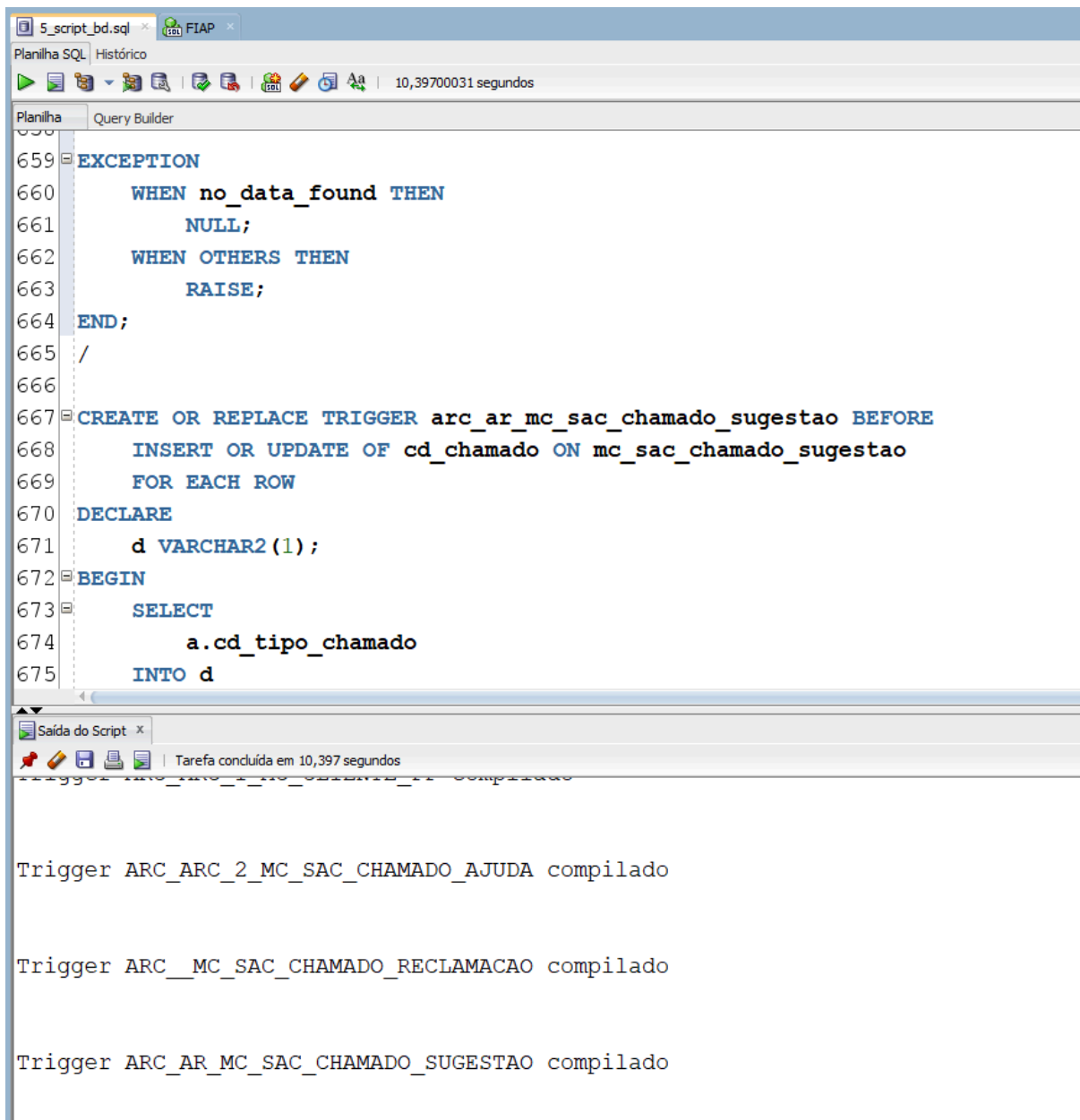
Trigger ARC_MC_SAC_CHAMADO_RECLAMACAO compilado

Trigger ARC_AR_MC_SAC_CHAMADO_SUGESTAO compilado
```

Criação de parte do código que verificará se o valor do atributo `cd_tipo_chamado` na tabela `MC_SAC_CHAMADO` é igual ao atributo `tipo_ajuda` para um novo registro na tabela `MC_SAC_CHAMADO_AJUDA`.

```
5_script_bnd.ed | FIAP
Planilha SQL - Histórico
10,39700031 segundos
Planilha | Query Builder
raise_application_error(-20223, 'FK classificar in Table MC_SAC_CHAMADO_AJUDA violates Arc constraint on Table MC_SAC_CHAMADO - discriminat
);
END IF;
EXCEPTION
WHEN no_data_found THEN
NULL;
WHEN OTHERS THEN
RAISE;
END;
/
CREATE OR REPLACE TRIGGER arc_mc_sac_chamado_reclamacao BEFORE
INSERT OR UPDATE OF cd_chamado ON mc_sac_chamado_reclamacao
FOR EACH ROW
DECLARE
d VARCHAR2(1);
Trigger ARC_ARC_2_MC_SAC_CHAMADO_AJUDA compilado
Trigger ARC_MC_SAC_CHAMADO_RECLAMACAO compilado
Trigger ARC_AR_MC_SAC_CHAMADO_SUGESTAO compilado
```

```
5_script_bnd.ed | FIAP
Planilha SQL - Histórico
10,39700031 segundos
Planilha | Query Builder
DECLARE
d VARCHAR2(1);
BEGIN
SELECT
a.cd_tipo_chamado
INTO d
FROM
mc_sac_chamado a
WHERE
a.cd_chamado = :new.cd_chamado;
IF ( d IS NULL OR d <> 'tipo_reclamacao' ) THEN
raise_application_error(-20223, 'FK classificarv4 in Table MC_SAC_CHAMADO_RECLAMACAO violates Arc constraint on Table MC_SAC_CHAMADO - disc
);
END IF;
EXCEPTION
Trigger ARC_ARC_2_MC_SAC_CHAMADO_AJUDA compilado
Trigger ARC_MC_SAC_CHAMADO_RECLAMACAO compilado
Trigger ARC_AR_MC_SAC_CHAMADO_SUGESTAO compilado
```

The image shows a screenshot of a SQL script editor window titled '5_script_bd.sql' and 'FIAP'. The editor displays a SQL script with line numbers 659 to 675. The script includes an exception block and the beginning of a trigger definition. Below the editor, there is an output window titled 'Saída do Script' showing the execution results of the script.

```
659 EXCEPTION
660     WHEN no_data_found THEN
661         NULL;
662     WHEN OTHERS THEN
663         RAISE;
664 END;
665 /
666
667 CREATE OR REPLACE TRIGGER arc_ar_mc_sac_chamado_sugestao BEFORE
668     INSERT OR UPDATE OF cd_chamado ON mc_sac_chamado_sugestao
669     FOR EACH ROW
670 DECLARE
671     d VARCHAR2(1);
672 BEGIN
673     SELECT
674         a.cd_tipo_chamado
675     INTO d
```

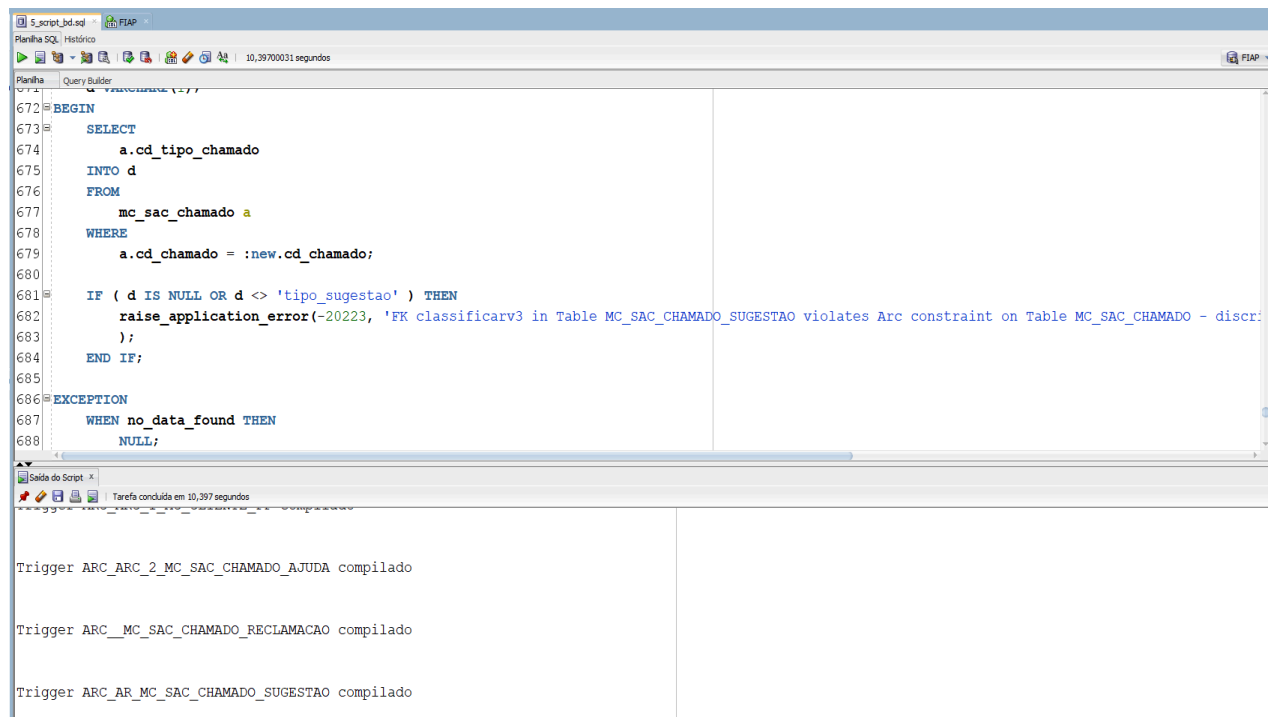
The output window shows the following messages:

```
Trigger ARC_ARC_2_MC_SAC_CHAMADO_AJUDA compilado

Trigger ARC_MC_SAC_CHAMADO_RECLAMACAO compilado

Trigger ARC_AR_MC_SAC_CHAMADO_SUGESTAO compilado
```

Continuações do código descrito anteriormente.



The screenshot displays a PL/SQL script editor window with a toolbar at the top. The script is a trigger definition for a table named MC_SAC_CHAMADO. The trigger is named ARC_ARC_2_MC_SAC_CHAMADO_AJUDA and is triggered on INSERT and UPDATE events. The trigger body is a PL/SQL block that begins with a SELECT statement to retrieve the cd_tipo_chamado from the MC_SAC_CHAMADO table. It then checks if the retrieved value is NULL or does not match the value of the attribute tipo_sugestao. If this condition is met, it raises an application error with the message 'FK classificav3 in Table MC_SAC_CHAMADO_SUGESTAO violates Arc constraint on Table MC_SAC_CHAMADO - discr:'. The script also includes an EXCEPTION block to handle the no_data_found exception by returning NULL. Below the script editor, the output window shows the successful compilation of the trigger.

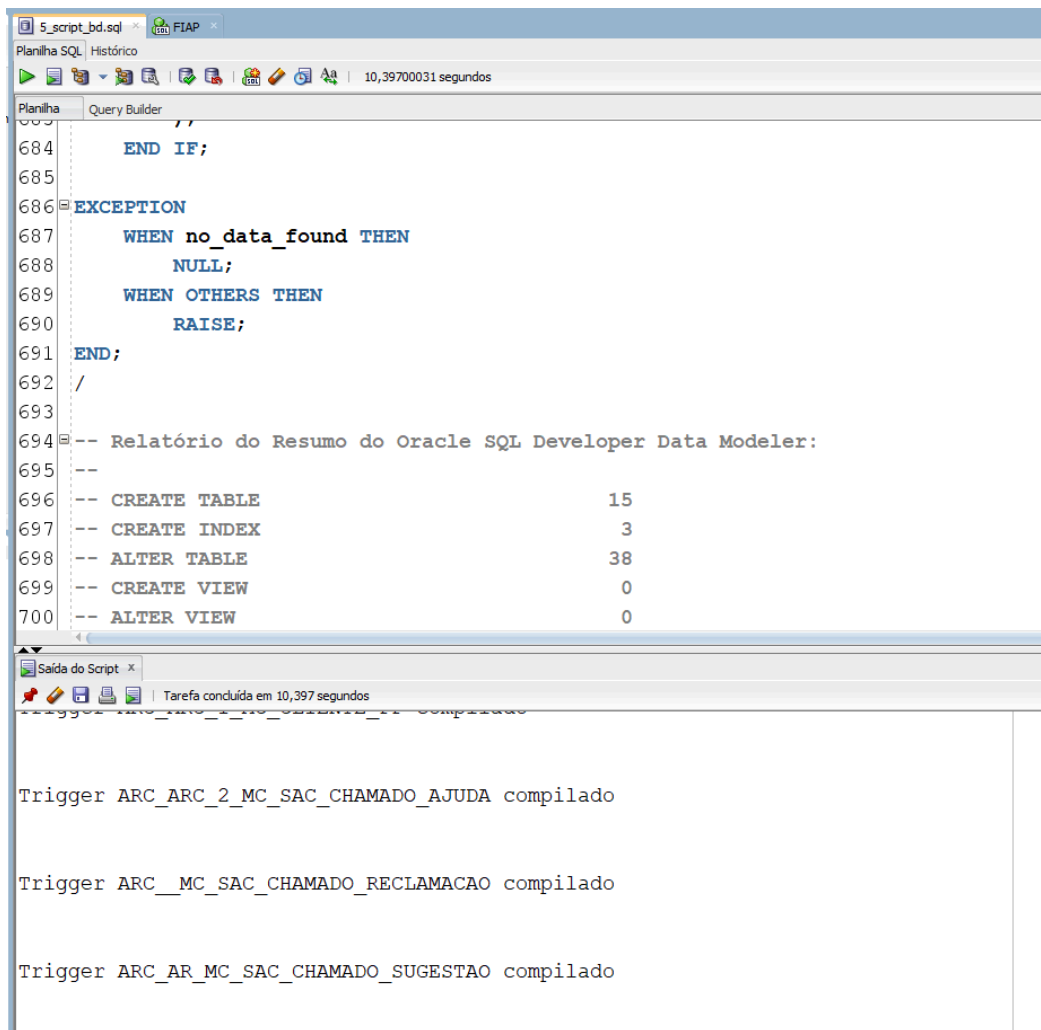
```
672 BEGIN
673 SELECT
674     a.cd_tipo_chamado
675 INTO d
676 FROM
677     mc_sac_chamado a
678 WHERE
679     a.cd_chamado = :new.cd_chamado;
680
681 IF ( d IS NULL OR d <> 'tipo_sugestao' ) THEN
682     raise_application_error(-20223, 'FK classificav3 in Table MC_SAC_CHAMADO_SUGESTAO violates Arc constraint on Table MC_SAC_CHAMADO - discr:
683 );
684 END IF;
685
686 EXCEPTION
687 WHEN no_data_found THEN
688     NULL;
```

Trigger ARC_ARC_2_MC_SAC_CHAMADO_AJUDA compilado

Trigger ARC_MC_SAC_CHAMADO_RECLAMACAO compilado

Trigger ARC_AR_MC_SAC_CHAMADO_SUGESTAO compilado

Parte do código que verifica se o valor do atributo `cd_tipo_chamado` corresponde ao atributo `tipo_sugestao` para, assim, garantir a integridade entre as tabelas `MC_SAC_CHAMADO` e `MC_SAC_CHAMADO_SUGESTAO`.



The screenshot displays the Oracle SQL Developer interface. The top window, titled '5_script_bd.sql', shows a SQL script with the following content:

```
684     END IF;
685
686 EXCEPTION
687     WHEN no_data_found THEN
688         NULL;
689     WHEN OTHERS THEN
690         RAISE;
691 END;
692 /
693
694 -- Relatório do Resumo do Oracle SQL Developer Data Modeler:
695 --
696 -- CREATE TABLE                15
697 -- CREATE INDEX                 3
698 -- ALTER TABLE                38
699 -- CREATE VIEW                  0
700 -- ALTER VIEW                   0
```

The bottom window, titled 'Saída do Script', shows the output of the script execution:

```
Trigger ARC_ARC_2_MC_SAC_CHAMADO_AJUDA compilado

Trigger ARC__MC_SAC_CHAMADO_RECLAMACAO compilado

Trigger ARC_AR_MC_SAC_CHAMADO_SUGESTAO compilado
```

Finalização do código referente aos ajustes na relação de herança, tanto referente à tabela Clientes quanto Chamados.

Nota: Todas as SEQUENCES criadas foram feitas via Oracle SQL Developer para cumprir com as regras de negócios impostas pela atividade.