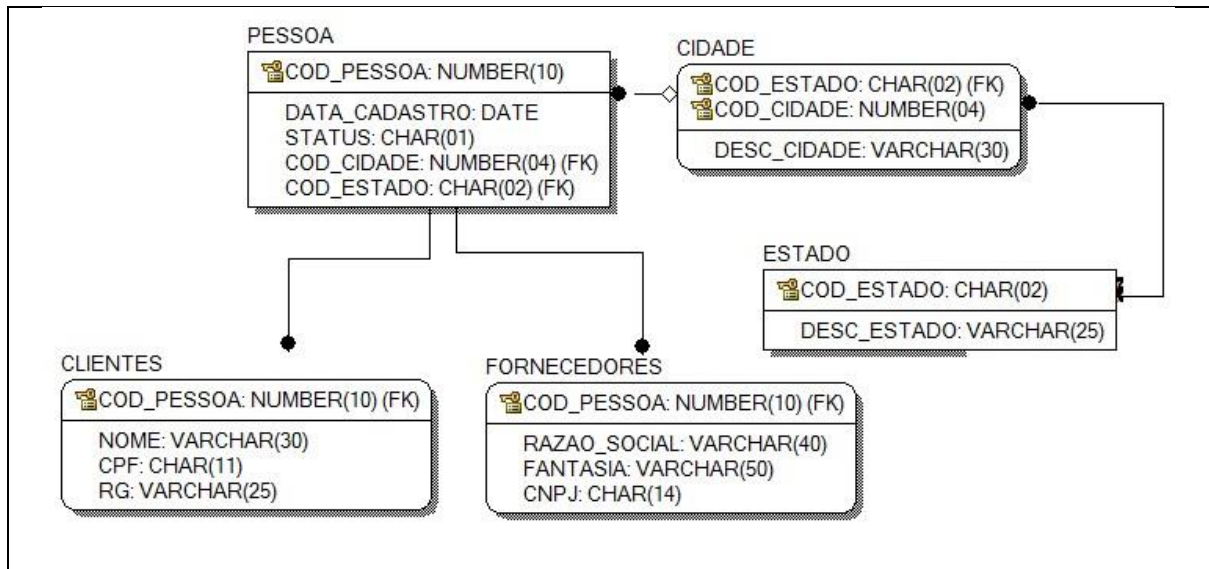


1. [BD-II] Referente a comandos DDL (Data Definition Language) utilizados em banco de dados relacionais, observe o seguinte diagrama:



Vinicius Antônio é DBA de desenvolvimento da empresa SOLUTION SYSTEM Ltda, empresa que atua no ramo de sistemas para varejo, para supermercados de pequeno e médio porte. De acordo com o diagrama desenvolvido por Vinicius, o mesmo precisa fazer a criação das tabelas conforme o diagrama. Vinicius está começando a codificar a criação no SqlDeveloper, porém na criação, a seguinte mensagem é mostrada:

```
Error report -
SQL Error: ORA-02260: table can have only one primary key
02260. 00000 - "table can have only one primary key"
*Cause:      Self-evident.
*Action:     Remove the extra primary key.
```

Qual(is) o(s) comando(s) que Vinicius digitou para que o erro acontecesse?

```

--I-----
CREATE TABLE ESTADO (
  COD_ESTADO CHAR(02) NOT NULL,
  DESC_ESTADO VARCHAR(25),
  CONSTRAINT PK_COD_ESTADO PRIMARY KEY(COD_ESTADO)
);

CREATE TABLE CIDADE (
  COD_ESTADO CHAR(02) NOT NULL,
  COD_CIDADE NUMBER(04),
  DESC_CIDADE VARCHAR(30),
  CONSTRAINT PK_CIDADE PRIMARY KEY(COD_ESTADO,COD_CIDADE),
  CONSTRAINT FK_CIDADE_COD_ESTADO
    FOREIGN KEY (COD_ESTADO) REFERENCES ESTADO(COD_ESTADO)
);

--II-----
CREATE TABLE ESTADO (
  COD_ESTADO CHAR(02) NOT NULL
  CONSTRAINT PK_COD_ESTADO PRIMARY KEY,
  DESC_ESTADO VARCHAR(25)
);

CREATE TABLE CIDADE (
  COD_ESTADO CHAR(02) NOT NULL CONSTRAINT
    FK_CIDADE_COD_ESTADO REFERENCES ESTADO(COD_ESTADO),
  COD_CIDADE NUMBER(04),
  DESC_CIDADE VARCHAR(30),
  CONSTRAINT PK_CIDADE PRIMARY KEY(COD_ESTADO,COD_CIDADE)
);

--III-----
CREATE TABLE ESTADO (
  COD_ESTADO CHAR(02) NOT NULL,
  DESC_ESTADO VARCHAR(25),
  CONSTRAINT PK_COD_ESTADO PRIMARY KEY(COD_ESTADO)
);

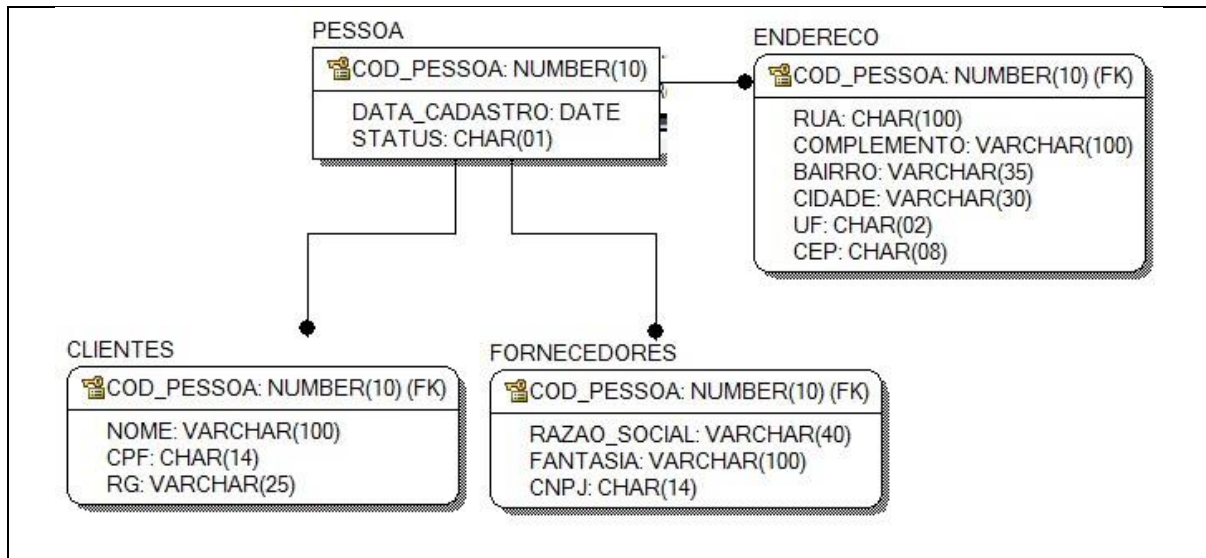
CREATE TABLE CIDADE (
  COD_ESTADO CHAR(02) NOT NULL PRIMARY KEY,
  COD_CIDADE NUMBER(04) NOT NULL PRIMARY KEY,
  DESC_CIDADE VARCHAR(30),
  CONSTRAINT PK_CIDADE PRIMARY KEY(COD_ESTADO,COD_CIDADE)
);

```

Está(ão) certo(s) apenas a(s) alternativa(s):

- (A) Somente II;
- (B) Somente I;
- (C) Somente III;
- (D) Os comandos I e II terão o mesmo efeito, apresentando a mensagem citada;
- (E) Nenhuma das alternativas estão corretas;

2. [BD-II] Referente a comandos DDL (Data Definition Language) utilizados em banco de dados relacionais, observe o seguinte diagrama:



Juarez Mendez é DBA de desenvolvimento da empresa PROMOTEC TECNOLOGIA EM VENDAS Ltda, empresa que atua no ramo de sistemas para automação comercial, para autopeças. De acordo com o diagrama desenvolvido pelo DBA anterior, o mesmo precisa fazer algumas manutenções no banco de dados, para que o mesmo melhore o desempenho, já que o mesmo já possui dados de média quantidade (em torno de 100 mil registros na tabela de pessoas).

Quais atitudes Juarez deverá fazer para melhorar a base de dados, considerando os itens abaixo?

I – Foi dado o comando “SELECT MAX(LENGTH(RUA)) FROM ENDERECO;” (sem aspas), e o mesmo retornou o valor 40. Isso quer dizer que o tamanho máximo de caracteres que tem esse campo são de 40 caracteres;

II – Os tipos de dados CHAR são considerados mais rápidos, porém tem a desvantagem de não ter o marcador de terminação da quantidade de campos;

III – Os tipos de dados VARCHAR, no caso do diagrama mostrado os que tem valores 100, deve ser avaliado os tamanhos máximos e ajustados os valores de mínimos possíveis;

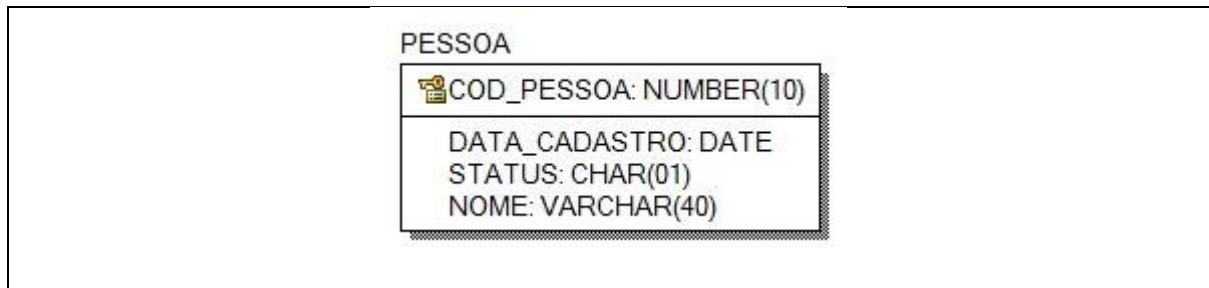
IV – O tipo de dado do campo CEP da tabela ENDERECO deverá ser alterada para o tipo VARCHAR(08), pois esse tipo de dado pode sofrer variação; {F}

Está(ão) certo(s) apenas a(s) alternativa(s):

- (A) II e III;
- (B) I, II e III;
- (C) III e IV;
- (D) Todos os itens estão corretos;
- (E) Nenhuma das alternativas estão corretas;

2016_BDII_008_COMANDOS_DDL

3. [BD-II] Referente a comandos DDL (Data Definition Language) utilizados em banco de dados relacionais, observe o seguinte diagrama:



Joana Siqueira é DBA de desenvolvimento da empresa POINTTEC PONTO EXATO Ltda, empresa que atua no ramo de sistemas para automação de ponto eletrônico, com reconhecimento de biométrica, retina, face a voz. Joana sabe que o servidor central está nos Estados Unidos e precisa desenvolver a tabela mostrada no diagrama e testar inserção de dados na tabela criada.

Qual(is) comando(s) está(ão) correto(s)?

```
--I
CREATE TABLE PESSOA (
  COD_PESSOA    NUMBER(10) NOT NULL,
  DATA_CADASTRO DATE,
  STATUS        CHAR(01),
  NOME          VARCHAR(40),
  CONSTRAINT PK_PESSOA PRIMARY KEY (COD_PESSOA)
);

--II
INSERT INTO PESSOA VALUES (4,'03/15/2000','A','JOAO MANUEL');
COMMIT;

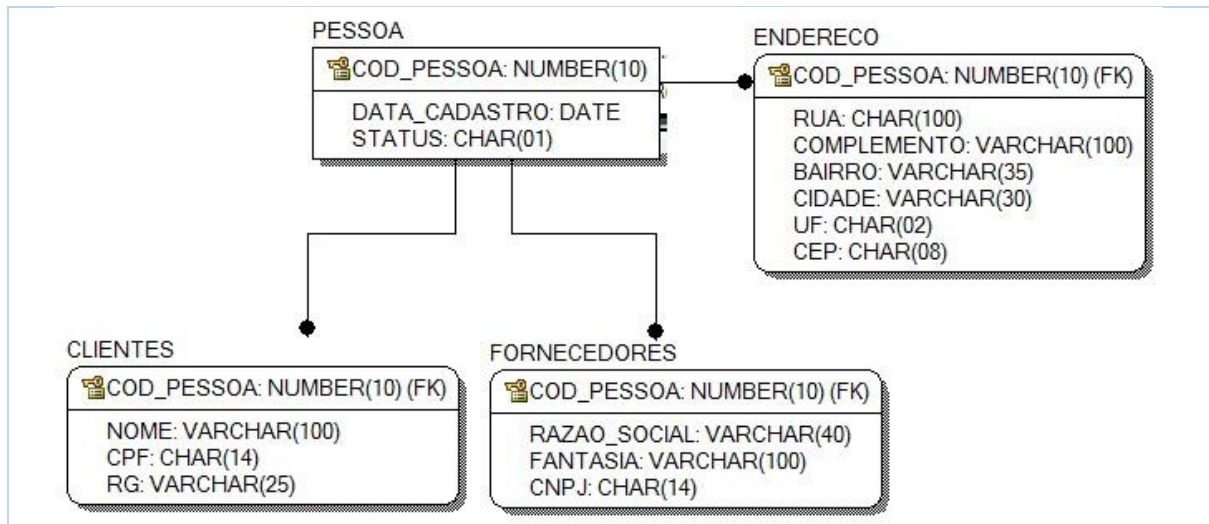
--III
INSERT INTO PESSOA VALUES (3,TO_DATE('05/03/2000',
                                     'MM/DD/YYYY'),'A','MANUEL DA SILVA');
COMMIT;

--IV
INSERT INTO PESSOA VALUES (5,TO_DATE('03/15/2000',
                                     'MM/DD/YYYY'),NULL,NULL);
COMMIT;
```

Está(ão) certo(s) apenas a(s) alternativa(s):

- (A) O item I (criação da tabela) está incorreto;
- (B) I, II e III;
- (C) I, III e IV;
- (D) Todos os itens estão corretos;
- (E) Nenhuma das alternativas estão corretas;

4. [BD-II] Referente a comandos DQL (Data Query Language) utilizados em banco de dados relacionais, observe o seguinte diagrama:



Giovani Souza é DBA de desenvolvimento da empresa AÇO E COMPANHIA LTDA, empresa que atua no ramo de distribuição de aço para construção civil. Giovani Precisa desenvolver um relatório que busque todos codigos, nome ou razao social, cpf ou cnpj dos clientes e fornecedores, para envio de comunicado interno da empresa. A SELECT que ela deveria ser feita seria (não importando a disposição dos dados):

Qual(is) comando(s) está(ão) correto(s)?

```
--I-----
SELECT P.COD_PESSOA, C.RAZAO, F.RAZAO_SOCIAL, F.CNPJ
FROM   PESSOA P
       INNER JOIN CLIENTES C
         ON P.COD_PESSOA = C.COD_PESSOA
       INNER JOIN FORNECEDORES F
         ON P.COD_PESSOA = F.COD_PESSOA
        AND C.COD_PESSOA = F.COD_PESSOA

--II-----
SELECT P.COD_PESSOA, C.RAZAO, F.RAZAO_SOCIAL, F.CNPJ
FROM   PESSOA P
       LEFT JOIN CLIENTES C
         ON P.COD_PESSOA = C.COD_PESSOA
       LEFT JOIN FORNECEDORES F
         ON P.COD_PESSOA = F.COD_PESSOA

--III-----
SELECT P.COD_PESSOA, C.RAZAO,
FROM   PESSOA P
       LEFT JOIN CLIENTES C
         ON P.COD_PESSOA = C.COD_PESSOA
UNION
SELECT F.RAZAO_SOCIAL, F.CNPJ
FROM   PESSOA P
       LEFT JOIN FORNECEDORES F
         ON P.COD_PESSOA = F.COD_PESSOA
```

Está(ão) certo(s) apenas a(s) alternativa(s):

(A) I, II e III;

- (B) II e III;
- (C) Somente I;
- (D) I e III;
- (E) Nenhuma das alternativas estão corretas;

5. [BD-II] Referente a LOOPS, utilizados em banco de dados relacionais Oracle, podemos afirmar que:

Baseado na base de dados padrão HR (imagem limpa, com os dados padrão), temos o seguinte código:

```
292 SET SERVEROUTPUT ON;
293
294 DECLARE
295     V_MIN_DEPTO NUMBER;
296     V_MAX_DEPTO NUMBER;
297     V_RESULTADO NUMBER(10,2);
298     V_TOTAL      NUMBER;
299     V_CONT       NUMBER := 1;
300
301 BEGIN
302     SELECT MIN(DEPARTMENT_ID)
303     INTO      V_MIN_DEPTO
304     FROM      DEPARTMENTS;
305
306     SELECT MAX(DEPARTMENT_ID)
307     INTO      V_MAX_DEPTO
308     FROM      DEPARTMENTS;
309
310     V_RESULTADO := V_MIN_DEPTO + V_MAX_DEPTO / 10;
311     V_RESULTADO := TRUNC(V_RESULTADO,0);
312
313     WHILE V_CONT < 5 LOOP
314         V_TOTAL := V_RESULTADO + V_CONT;
315         DBMS_OUTPUT.PUT_LINE('V_TOTAL É : ' || V_TOTAL);
316
317     END LOOP;
318 END;
```

Analise o código PL/SQL

Itens a serem analisados:

I – O código irá mostrar os resultados:

```
PL/SQL procedure successfully completed.
```

```
V_TOTAL É : 38
```

```
V_TOTAL É : 39
```

```
V_TOTAL É : 40
```

```
V_TOTAL É : 41
```

II – O código acima possui erro lógico, ou seja, o compilador não irá conseguir detectar o erro para o código ser executado;

III – O código irá mostrar os resultados:

```
PL/SQL procedure successfully completed.
```

```
V_TOTAL É : 29
```

```
V_TOTAL É : 30
```

```
V_TOTAL É : 31
```

```
V_TOTAL É : 32
```

IV – Na linha de código 311, “V_RESULTADO := TRUNC(V_RESULTADO,0)” está incorreto, pois V_REUSLTADO não pode receber o próprio valor;

Está(ão) certo(s) apenas a(s) alternativa(s):

- (A) Somente IV;
- (B) Somente I;
- (C) Somente III;
- (D) Somente II;
- (E) Nenhuma das alternativas estão corretas;

6. [BD-II] Referente a PROCEDURES, utilizados em banco de dados relacionais Oracle, podemos afirmar que as mesmas possuem 3 tipos de parâmetros, sendo elas a “IN”, “OUT” e a “IN OUT”. O tipo de parâmetro “IN” é o tipo mais comum (padrão caso não seja especificado). O tipo de parâmetro “OUT” é preciso especificar na declaração da função, onde valores podem ser de certa forma retornados pelo ambiente chamador, e o tipo de parâmetro “IN OUT” é um parâmetro do mesmo tipo que o “IN”, sendo a maior diferença é que a mesma é opcional o retorno de valores mesmo sendo declaradas;

(_V_)

(_F_)

7. [BD-II] Referente a PACKAGES, utilizados em banco de dados relacionais Oracle, podemos afirmar que as mesmas serem para substituir totalmente as procedures e funções em banco de dados (dentrod as packages, é possível declarar funções, procedures e variáveis públicas). Como são objetos que possuem grande flexibilidade

em código PL/SQL, após a utilização de PACKAGES, todas as procedures e funções podem ser substituídas por esse objeto.

(_V_)

(_F_)

8. [BD-II] Referente a CURSORES, utilizados em banco de dados relacionais Oracle, podemos afirmar que:

```
398 SET SERVEROUTPUT ON;
399 DECLARE
400     CURSOR EMPLOYEES_CURSOR IS
401         SELECT  EMPLOYEE_ID, LAST_NAME, SALARY, HIRE_DATE
402         FROM    EMPLOYEES
403         WHERE   DEPARTMENT_ID = &P_DEPARTMENT_ID;
404     V_EMPLOYEES_RECORD EMPLOYEES_CURSOR%ROWTYPE;
405     V_QTD_REGISTROS NUMBER;
406 BEGIN
407     SELECT  COUNT(*)
408     INTO    V_QTD_REGISTROS
409     FROM    EMPLOYEES
410     WHERE   DEPARTMENT_ID = &P_DEPARTMENT_ID;
411
412     OPEN EMPLOYEES_CURSOR;
413
414     IF V_QTD_REGISTROS < 4 THEN
415         DBMS_OUTPUT.PUT_LINE('NUMERO DE REGISTROS MENOR QUE 4');
416     ELSE
417         LOOP
418             FETCH EMPLOYEES_CURSOR INTO V_EMPLOYEES_RECORD;
419             IF EMPLOYEES_CURSOR%ROWCOUNT BETWEEN 4 AND 10 THEN
420                 DBMS_OUTPUT.PUT_LINE('RECORD NUMERO: ' || EMPLOYEES_CURSOR%ROWCOUNT);
421                 DBMS_OUTPUT.PUT_LINE('EMPLOYEE_ID...: ' || V_EMPLOYEES_RECORD.EMPLOYEE_ID);
422                 DBMS_OUTPUT.PUT_LINE('LAST_NAME.....: ' || V_EMPLOYEES_RECORD.LAST_NAME);
423                 DBMS_OUTPUT.PUT_LINE('SALARY.....: ' || V_EMPLOYEES_RECORD.SALARY);
424                 DBMS_OUTPUT.PUT_LINE('HIRE DATE.....: ' || V_EMPLOYEES_RECORD.HIRE_DATE);
425                 DBMS_OUTPUT.PUT_LINE('');
426             END IF;
427             EXIT WHEN EMPLOYEES_CURSOR%NOTFOUND;
428         END LOOP;
429     END IF;
430     CLOSE EMPLOYEES_CURSOR;
431 END;
432 /
```

Referente ao código PL/SQL podemos afirmar que:

Itens a serem analisados:

- I – O código PL/SQL é um código nomeado, ou seja, assim que compilado, será armazenado em um objeto no servidor Oracle;

II –É possível substituir o resultado do código PL/SQL descrito por uma SELECT COMPLEXA (considerando somente o resultado e não a disposição/layout dos dados apresentados);

III – O código apresentado não tratando resultados acima de 10 registros, ou seja, caso o retorno do select seja maior que 10, o código PL/SQL apresentará um erro;

IV – O código não apresenta problemas de sintaxe e será executado independente da quantidade de registros retornada;

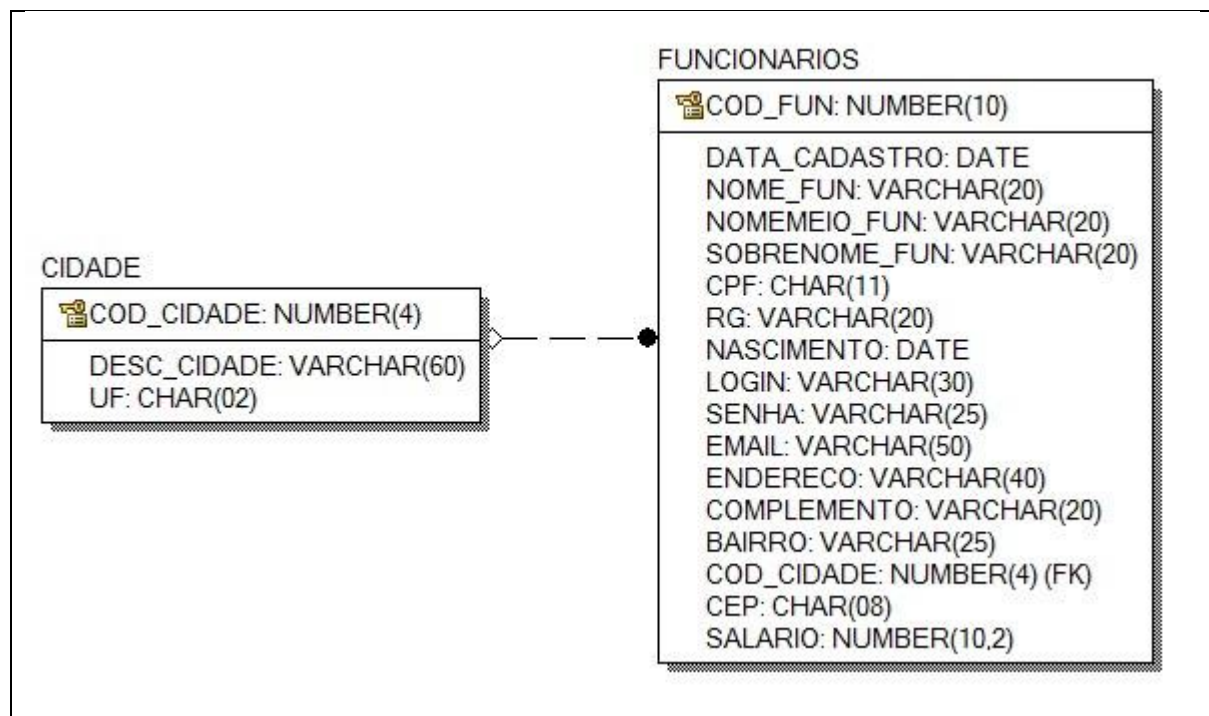
Está(ão) certo(s) apenas a(s) alternativa(s):

- (A) Somente IV;
- (B) I e III;
- (C) II e III;
- (D) Somente III;
- (E) Nenhuma das alternativas estão corretas;

9. [BD-II] Referente a objetos do banco de dados INDICES e CONSTRAINTS, utilizados em banco de dados relacionais Oracle, podemos afirmar que:

Cenário: Arun Karan é um novo DBA de infraestrutura de banco de dados da empresa HUMAM CENTER ATTENTION RESOURCES S/A, que trabalha contratação e recolocação de cargos entre empresas e pessoas em todo o mundo. Arun possui uma tabela de funcionários, a qual possui muitos campos, e seu gerente de TI solicitou que Arun fizesse algumas alterações no banco de dados:

Atualmente, a tabela com os campos principais encontram-se como o modelo:



Foi solicitado que, tivesse uma criação de um índice para cada campo da tabela funcionários, já que todos os dados seriam feitos pesquisas de funcionários.

Arun perguntou se havia uma grande quantidade de inserções, alterações e exclusões de dados de funcionário e seu gerente de TI disse que em média de inclusões e alterações eram de 100 mil registros hora (número considerado grande de inserções e alterações), considerando que sua base de funcionários havia mais de 10 milhões de registros. Também perguntou quais os campos fossem mais utilizados, para se ter uma alternativa em vez de criar um índice para cada campo. As principais considerações foram que o campo CPF era bem acessado para procurar registros para atualização do cadastro, havia um relatório que separava os funcionários por região (cidade) e que quando se havia uma procura por nome, o nome do meio geralmente era desprezado nas pesquisas.

Baseado no cenário citado, quais seriam algumas soluções que o DBA poderia tomar para ajudar a resolver os problemas mencionados?

Itens a serem analisados:

I – Arun deve criar para cada coluna um índice diferente na tabela (ou seja, todas as colunas devem ter um índice). Assim, a flexibilidade da aplicação seria muito maior;

II – Arun deve criar um índice único para cada campo mais utilizado, sendo um para CPF, um para a Cidade, e um pelo Nome;

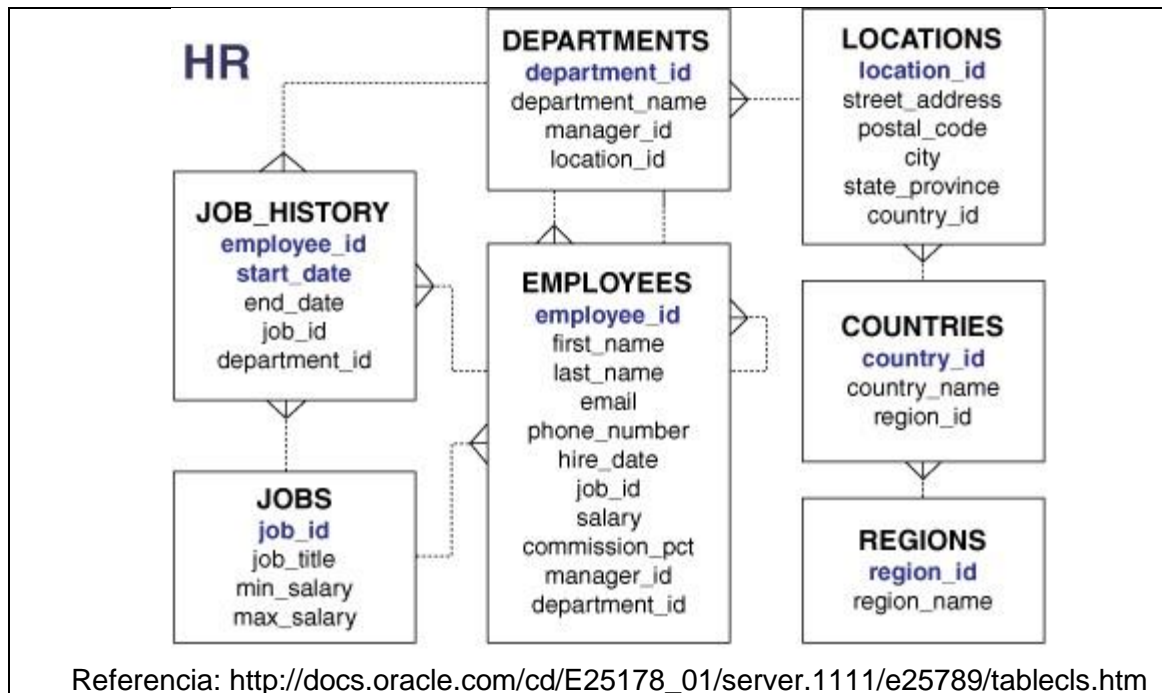
III – Arun deve criar um índice único para o campo CPF, NÃO deve criar o índice secundário para a Cidade (pois quando se é uma chave estrangeira já existe obrigatoriamente um índice associado), e deve criar um índice secundário pelo Nome e Sobrenome;

IV – Arun deve criar um índice único para o campo CPF, deve criar o índice secundário para a Cidade, e deve criar um índice secundário pelo Nome e Sobrenome;

Está(ão) certo(s) apenas a(s) alternativa(s):

- (A) Somente III;
- (B) Somente I;
- (C) Somente II;
- (D) Somente IV;
- (E) Nenhuma das alternativas estão corretas;

10. [BD-II] Referente a objetos do banco de dados VIEWS, utilizados em banco de dados relacionais Oracle, podemos afirmar que:



Narayana Suria é DBA de desenvolvimento da empresa FRITA O BOI CARNES MAGRAS LTDA, que trabalha com fornecimento de carnes para todo o estado do território nacional. Narayana precisa fazer com que somente o gerente do departamento tenha visão dos dados de seus próprios funcionários (que trabalham no determinado departamento). Para isso, criou algumas das seguintes views (uma para cada gerente com as devidas permissões).

22	DESC EMPLOYEES;
23	

Script Output x	Query Result x	Query Result 1 x
Task completed in 0.015 seconds		
Name	Null	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR2(20)
LAST_NAME	NOT NULL	VARCHAR2(25)
EMAIL	NOT NULL	VARCHAR2(25)
PHONE_NUMBER		VARCHAR2(20)
HIRE_DATE	NOT NULL	DATE
JOB_ID	NOT NULL	VARCHAR2(10)
SALARY		NUMBER(8,2)
COMMISSION_PCT		NUMBER(2,2)
MANAGER_ID		NUMBER(6)
DEPARTMENT_ID		NUMBER(4)

Algumas das seguintes VIEWS foram criadas com permissão de somente o gerente do departamento executar cada VIEW (ou seja, um gerente de outro departamento não conseguiu ver os funcionários que não fosse gerenciado por ele):

```
24  --Para o gerente de Marketing
25  CREATE OR REPLACE VIEW VW_FUNC_MARKETING AS
26      SELECT  E.EMPLOYEE_ID, E.LAST_NAME, E.FIRST_NAME,
27              E.EMAIL,          E.HIRE_DATE, E.JOB_ID,
28              E.DEPARTMENT_ID, E.MANAGER_ID
29      FROM EMPLOYEES E
30      WHERE E.MANAGER_ID = 201
31      WITH CHECK OPTION CONSTRAINT CK_FUNC_MKT_MNG_201;
32
33  --Para o gerente de TI
34  CREATE OR REPLACE VIEW VW_FUNC_IT AS
35      SELECT  E.EMPLOYEE_ID, E.LAST_NAME, E.FIRST_NAME,
36              E.EMAIL,          E.HIRE_DATE, E.JOB_ID,
37              E.DEPARTMENT_ID, E.MANAGER_ID
38      FROM EMPLOYEES E
39      WHERE E.MANAGER_ID = 103
40      WITH CHECK OPTION CONSTRAINT CK_FUNC_IT_MNG_103;
41
42  --Para o gerente de vendas
43  CREATE OR REPLACE VIEW VW_FUNC_SALES AS
44      SELECT  E.EMPLOYEE_ID, E.LAST_NAME, E.FIRST_NAME,
45              E.EMAIL,          E.HIRE_DATE, E.JOB_ID,
46              E.DEPARTMENT_ID, E.MANAGER_ID
47      FROM EMPLOYEES E
48      WHERE E.MANAGER_ID = 145
49      WITH CHECK OPTION CONSTRAINT CK_FUNC_SAL_MNG_145;
```

A solução atendia as demandas da empresa, até que certa vez, o gerente de vendas (Sales) pediu para o gerente de TI (IT) fazer uma inserção de emergência no banco de dados que continha o seguinte código. (Neste caso, o gerente de vendas era amigo do gerente de TI, que sabia que o gerente de TI teria acesso total ao banco de dados):

```
INSERT INTO VW_FUNC_IT VALUES (EMPLOYEES_SEQ.NEXTVAL, 'Maria', 'Ana',
                                'ANAMARIA', TO_DATE('06/04/16', 'MM/DD/YY'), 'SA_REP', 80, '145');
COMMIT;
```

O gerente de TI, por sua vez percebeu que o banco de dados NÃO efetivou a transação, pois:

Itens a serem analisados:

I – O DBA, quando desenvolveu a solução NÃO testou corretamente e criou SELECTS incorretos para cada gerente (erro lógico, a qual o banco de dados não acusa na compilação do objeto);

II- O parâmetro WITH CHECK OPTION deveria ser tirado do código que criou as views, para contemplar a solução do cenário como um todo (comprometendo a segurança criada de cada gerente encher somente seus próprios funcionários do departamento);

III – O gerente de vendas deveria utilizar sua view VW_FUNC_SALES para fazer a inserção dos dados desejados ;

IV – O gerente de TI não tinha acesso a tabela EMPLOYEES, sendo assim, o mesmo não poderia fazer inclusão de dados nessa tabela;

Está(ão) certo(s) apenas a(s) alternativa(s):

- (A) I e IV;
- (B) Somente III;
- (C) II e III;
- (D) Somente II;
- (E) Nenhuma das alternativas estão corretas;