

Check Point-I 2º Semestre Sistemas de Informação - 3ºSIS

INSTRUÇÕES

- Essa avaliação é individual.
- A interpretação das questões faz parte dessa avaliação.

ATENÇÃO

- Só permitido 01 (um) preenchimento desse questionário.
- Vocês tem até o dia 27/08/2024 às 23h:59m para enviar as respostas.
- Não serão abertas exceções para mais de um preenchimento nem para extensão do prazo em nenhuma hipótese.
- Considere que TODAS as tabelas referenciadas pelos programas existem no Banco de Dados e TODAS estão populadas.
- Considere, obrigatoriamente, que todas as instruções serão executadas em um servidor de banco de dados relacional Oracle.
- Para todos os programas considere que a saída do PUT_LINE está habilitada.

Bom Trabalho!

Prof. Me. Alexandre Barcelos

Check Point-I 2º Semestre Sistemas de Informação - 3ºSIS

* 1. Informe o número do seu RM (**SOMENTE NÚMEROS**)

* 2. Informe o seu nome completo (**SEM ABREVIAR**)

Check Point-I 2º Semestre Sistemas de Informação - 3ºSIS

* 1. Assinale TODAS as alternativas corretas: O PL/SQL é:

- ☐ É uma extensão para a linguagem SQL com facilidades projetadas de linguagens de programação.
- ☐ Declarações de Manipulação de Dados e Consultas SQL são incluídas dentro de unidades de código procedural.
- ☐ É uma linguagem de programação que foi desenvolvida pela Oracle e é 100% compatível para desenvolver em banco de dados IBM DB2.
- ☐ Fornece uma estrutura de bloco para unidades de código executáveis. A manutenção do código é facilitada com uma estrutura bem definida.
- ☐ A linguagem PL/SQL é 100% compatível com a linguagem T-SQL (Transaction-SQL) da Microsoft.

* 2. Qual dos seguintes blocos PL/SQL é executado corretamente?

- ☐ BEGIN
END;
/
- ☐ DECLARE
amount INTEGER(10);
END;
/
- ☐ DECLARE
BEGIN
END;
/
- ☐ DECLARE
BEGIN
DBMS_OUTPUT.PUT_LINE(amount);
END;
/
- ☐ DECLARE
amount INTEGER(10);
BEGIN
DBMS_OUTPUT.PUT_LINE(amount);
END;
/

* 3. Assinale a opção que apresenta as sessões OPCIONAIS do bloco PL/SQL.

- ☐ DECLARE - EXCEPTION
- ☐ DECLARE - BEGIN - END
- ☐ BEGIN - END
- ☐ Todas as sessões são opcionais.
- ☐ Todas as sessões são obrigatórias.

* 4. Qual é o comando que habilita a saída do procedimento PUT_LINE?

- ☐ set serverout on
- ☐ set feed on
- ☐ set echo on
- ☐ set verify on
- ☐ set putline on

* 5. Considere a estrutura da tabela FUNCIONARIOS a seguir: Após a execução do bloco PL/SQL anterior o que irá acontecer?

Nome	Nulo?	Tipo
ID_FUNCIONARIO	NOT NULL	NUMBER (6)
PRIMEIRO_NOME		VARCHAR2 (20)
ULTIMO_NOME	NOT NULL	VARCHAR2 (25)
SALARIO		NUMBER (8,2)
PERCENTUAL_COMISSAO		NUMBER (2,2)

Analise o Bloco PL/SQL a seguir e escolha a alternativa correta.

```

DECLARE
    v_cod_func    NUMBER(6) = 100;
    v_ultimo_nome VARCHAR2(25);
    v_salario     NUMBER(8,2);
    v_taxa        NUMBER(2,1);
BEGIN
    dbms_output.put_line(v_taxa);
END;
/

```

- ☐ O bloco PL/SQL irá retornar um erro pois o operador de atribuição de valor para uma variável está incorreto.
- ☐ O bloco PL/SQL irá retornar um erro pois há três variáveis que não foram iniciadas.
- ☐ O bloco PL/SQL irá retornar um erro pois não existe tipo de dados VARCHAR2 pois o correto é VARCHAR.
- ☐ O bloco PL/SQL será executado com sucesso porém serão apresentados WARNINGS para as variáveis v_cod_func e v_salario pois elas deveriam ser obrigatórias a exemplo das colunas.
- ☐ O bloco PL/SQL será executado com sucesso.

* 6. Analise o Bloco PL/SQL a seguir e escolha a alternativa correta.

```

DECLARE
    v_cod_dept    INT NOT NULL;
    v_nome_depto  VARCHAR(50) := 'TI';
BEGIN
    dbms_output.put_line(v_nome_depto);
END;
/

```

Após a execução do bloco PL/SQL anterior o que irá acontecer?

- ☐ O bloco PL/SQL irá retornar um erro pois toda variável NOT NULL deve ser iniciada.
- ☐ O bloco PL/SQL irá retornar um erro pois a instrução put_line está incorreta. O correto é: dbms_output.put_line('v_nome_depto');
- ☐ O bloco PL/SQL irá retornar erro pois ele deve ter uma sessão para o tratamento de exceções.
- ☐ O bloco PL/SQL será executado com sucesso.
- ☐ O bloco PL/SQL será executado com sucesso mas será retornada uma mensagem de WARNING

* 7. Analise o Bloco PL/SQL a seguir e escolha a alternativa correta.

```
DECLARE
    v_nome VARCHAR(25) := 'Americas';
    v_count CONSTANT INT := 10;
BEGIN
    dbms_output.put_line(v_nome);
    v_count := v_count + 1;
    dbms_output.put_line(v_count);
END;
/
```

- ☐ O bloco PL/SQL irá retornar um erro pois a constante não pode ter o seu valor alterado.
- ☐ O bloco PL/SQL irá retornar um erro pois a variável v_nome deve ser not null.
- ☐ O bloco PL/SQL irá retornar um erro pois a variável v_count deveria iniciar com valor igual a 1 (UM) por se tratar de um contador.
- ☐ O bloco PL/SQL será executado com sucesso porém serão apresentados WARNINGS para a variável v_count.
- ☐ O bloco PL/SQL será executado com sucesso.

* 8. Analise o Bloco PL/SQL a seguir e escolha a alternativa correta.

```
DECLARE
    v_nome VARCHAR2(20) default 'Meg';
BEGIN
    v_nome := 'Peter';
    dbms_output.put_line('O nome é: ' || v_nome);
END;
/
```

- ☐ O bloco PL/SQL será executado com sucesso e irá exibir: Peter.
- ☐ O bloco PL/SQL irá retornar um erro pois a variável v_nome foi iniciada de maneira incorreta.
- ☐ O bloco PL/SQL irá retornar um erro pois a variável v_nome recebeu o valor 'Peter' na sessão de execução.
- ☐ O bloco PL/SQL será executado com sucesso e nada será exibido.
- ☐ O bloco PL/SQL será executado com sucesso e irá exibir: Meg

* 9. Considere a estrutura da tabela DEPARTMENTS a seguir:

Nome	Nulo	Tipo
DEPARTMENT_ID	NOT NULL	NUMBER(4)
DEPARTMENT_NAME	NOT NULL	VARCHAR(30)

Observando somente a sessão DECLARE, foi declarada a variável no bloco PL/SQL da seguinte forma:

```
DECLARE V_DEPARTMENT_NAME VARCHAR(100);
```

Refaça a declaração anterior de tal forma que se o tipo de dados da coluna DEPARTMENT_NAME da tabela DEPARTMENTS for alterado o tipo de dados da variável V_DEPARTMENT_NAME seja alterado dinamicamente.

- ☐ DECLARE
V_DEPARTMENT_NAME DEPARTMENTS.DEPARTMENT_NAME%TYPE;
- ☐ DECLARE
V_DEPARTMENT_NAME DEPARTMENTS.DEPARTMENT_NAME%DATATYPE;
- ☐ DECLARE
V_DEPARTMENT_NAME DEPARTMENTS.DEPARTMENT_NAME&TYPE;
- ☐ DECLARE
V_DEPARTMENT_NAME DEPARTMENTS.DEPARTMENT_NAME&DATATYPE;
- ☐ DECLARE
V_DEPARTMENT_NAME DEPARTMENTS.DEPARTMENT_NAME.DATATYPE;

* 10. Identifique o nome do identificador que é válido:

- ☐ number1to7
- ☐ date
- ☐ today's_date
- ☐ \$Isleap\$year
- ☐ #number

* 11. Qual o comando a seguir cria uma BIND VARIABLE: (1,0 ponto)

- ☐ VARIABLE b_bind NUMBER;
- ☐ DECLARE b_bind NUMBER;
- ☐ DECLARE VARIABLE b_bind NUMBER;
- ☐ VARIABLE b_bind NUMBER(5);
- ☐ VARIABLE b_bind NUMERIC;

* 12. Avalie o bloco PL/SQL a seguir e escolha a alternativa correta. Observação: Considere que o funcionário de código 100 (employee_id=100) existe na tabela de empregados (employees): Utilize a estrutura da tabela EMPLOYEES a seguir:

```

Nome           Nulo      Tipo
-----
EMPLOYEE_ID    NOT NULL  NUMBER(6)
LAST_NAME      NOT NULL  VARCHAR2(25)
JOB_ID          NOT NULL  VARCHAR2(10)
SALARY                    NUMBER(8,2)

SET SERVEROUTPUT ON
DECLARE
    V_LAST_NAME EMPLOYEES.LAST_NAME%TYPE;
    V_SALARY     EMPLOYEES.SALARY%TYPE;
BEGIN
    SELECT LAST_NAME, SALARY
    FROM   EMPLOYEES
    WHERE  EMPLOYEE_ID=100;

    DBMS_OUTPUT.PUT_LINE('Sobrenome:      ' || V_LAST_NAME || '      Salário:
' || V_SALARY);
END;
/

```

- ☐ O bloco não será executado pois há um erro de sintaxe na instrução SELECT.
- ☐ O bloco será executado com sucesso, porém não será exibido nada.
- ☐ O bloco será executado com sucesso, porém serão exibidos o nome e o salário do funcionário.
- ☐ O bloco não será executado pois a coluna LAST_NAME não existe na tabela EMPLOYEES.
- ☐ O bloco não será executado pois há um erro de lógica.

* 13. Para fazer um comentário de uma única linha para documentar o código devemos utilizar:

- ☐ Dois hifens(--)
- ☐ Asterisco Barra (*/)
- ☐ Dois asteriscos (**)
- ☐ Mais e hífen (+-)
- ☐ Duas Barras (//)

* 14. Um bloco PL/SQL pode ser composto por quais das divisões a seguir. Considere as divisões obrigatórias e opcionais.

- ☐ DECLARE, BEGIN, EXCEPTION e END
- ☐ BEGIN e EXCEPTION
- ☐ DECLARE e BEGIN
- ☐ DECLARE e EXCEPTION
- ☐ DECLARE, EXCEPTION e END

* 15. Considere a estrutura da tabela employees a seguir:

Nome	Nulo?	Tipo
-----	-----	-----
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)

Você necessita construir um bloco PL/SQL para fazer uma inserção de dados na tabela employees.

Analise o bloco PL/SQL a seguir e escolha a alternativa correta.

```
BEGIN
  INSERT INTO employees
    (employee_id, first_name, last_name, email, hire_date, job_id, salary)

    VALUES (employees_seq.NEXTVAL, 'Ruth', 'Cores', 'RCORES', sysdate, 'AD_AS
    ST', 4000);
  COMMIT;
END;
/
```

- ☐ A inserção será executada com sucesso.
- ☐ A instrução INSERT está com a sintaxe incorreta dentro do bloco PL/SQL.
- ☐ A instrução INSERT está com a sintaxe incorreta pois não foram informadas todas as colunas da tabela employees.
- ☐ Dentro de um bloco PL/SQL só é possível emitir comandos DQL.
- ☐ Dentro de um bloco PL/SQL não é permitido utilizar comandos DTL/TCL.

* 16. Analise o bloco PL/SQL a seguir e escolha a alternativa correta.

```
set serveroutput on
DECLARE
  valor INTEGER;
BEGIN
  IF valor < 11 THEN
    dbms_output.put_line(' Verdade');
  ELSE
    dbms_output.put_line(' Falsidade ');
  END IF;
END;
/
```

- ☐ Será exibida a mensagem: ' Falsidade'
- ☐ O bloco PL/SQL não irá executar porque a variável myage não foi iniciada.
- ☐ O bloco PL/SQL irá executar, mas não será exibido nada.
- ☐ Será exibida a mensagem: ' Verdade '
- ☐ Serão exibidas as duas mensagens: ' Verdade' e 'Falsidade '

* 17. Assinale a alternativa correta: Quando os cursores implícitos são declarados no Oracle RDBMS?

- ☐ São declarados para todas as instruções DML e PL/SQL SELECT.
- ☐ São declarados pelo programador.
- ☐ São declarados para todas as instruções DDL.
- ☐ São declarados para todas as instruções DML
- ☐ São declarados para todas as instruções DDL e DCL.

* 18. Escolha a alternativa correta: Os atributos de um cursor implícito no Oracle RDBMS são:

- ☐ %ISOPEN, %FOUND, %NOTFOUND e %ROWCOUNT
- ☐ &ISOPEN, &FOUND, &NOTFOUND e &ROWCOUNT
- ☐ \$ISOPEN, \$FOUND, \$NOTFOUND e \$ROWCOUNT
- ☐ _ISOPEN, _FOUND, _NOTFOUND e _ROWCOUNT
- ☐ @ISOPEN, @FOUND, @NOTFOUND e @ROWCOUNT

* 19. Avalie o bloco PL/SQL a seguir e escolha a alternativa correta:

```
DECLARE
    hire_date    employees.hire_date%TYPE;
    sysdate      hire_date%TYPE;
    employee_id   employees.employee_id%TYPE := 176;
BEGIN
    SELECT
        hire_date,
        sysdate
    INTO
        hire_date,
        sysdate
    FROM
        employees
    WHERE
        employee_id = employee_id;
END;
/
```

- ☐ O bloco PL/SQL não irá executar porque a variável employee_id tem o mesmo nome da coluna.
- ☐ O bloco PL/SQL irá executar com sucesso.
- ☐ O bloco PL/SQL não irá executar porque não foi utilizada a procedure PUT_LINE.
- ☐ O bloco PL/SQL não irá executar porque a variável não pode ser nomeada como sysdate.
- ☐ O bloco PL/SQL não irá executar porque será retornada mais de uma linha.

* 20. Considere a estrutura da tabela EMPLOYEES a seguir:

Nome	Nulo?	Tipo
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)

Analise o bloco PL/SQL a seguir e escolha a alternativa correta.

```
DECLARE
    employee_id NUMBER(6);
    sysdate     DATE;
BEGIN
    SELECT
        employee_id, sysdate
    INTO employee_id, sysdate
    FROM
        employees
    WHERE
        last_name = 'Kochhar';

    DBMS_OUTPUT.PUT_LINE (EMPLOYEE_ID);
END;
/
```

- ☐ O bloco PL/SQL será executado com sucesso.
- ☐ O bloco PL/SQL irá retornar um erro pois uma variável não pode ter o nome de sysdate.
- ☐ O bloco PL/SQL irá retornar um erro pois a variável não pode ter o nome de EMPLOYEE_ID que é o mesmo nome de coluna.
- ☐ O bloco PL/SQL irá retornar um erro pois uma variável EMPLOYEE_ID foi declarada em minúsculo e referenciada em maiúsculo.
- ☐ O bloco PL/SQL será executado com sucesso porém será exibida uma mensagem de WARNING.

* 21. Escolha TODAS as alternativas corretas sobre as variáveis BOOLEANAS.

Com o código PL/SQL você poderá comparar variáveis em instruções SQL e procedurais. Essas comparações, chamadas expressões booleanas, consistem em expressões simples ou complexas separadas por operadores relacionais.

- ☐ Somente os valores TRUE e FALSE podem ser atribuídos a uma variável booleana.
- ☐ As expressões condicionais usam os operadores lógicos AND, OR e operador unário NOT para verificar os valores das variáveis.
- ☐ As variáveis sempre produzem TRUE, FALSE, ou NULL.
- ☐ As expressões aritméticas, de caractere e de data podem ser usadas para retornar um valor booleano.

* 22. Escolha TODAS as alternativas corretas sobre as variáveis BIND. As variáveis Bind são:

- ☐ Criadas no ambiente.
- ☐ São também chamadas variáveis de host.
- ☐ Criadas com a palavra-chave VARIABLE.
- ☐ Utilizadas nas instruções SQL e blocos PL/SQL.
- ☐ Não podem ser acessadas após o bloco PL/SQL ser executado.
- ☐ São referenciadas com (: ponto sobre ponto) .

* 23. Analise o bloco PL/SQL a seguir e escolha a alternativa correta

```
BEGIN
  INSERT INTO employees
    (employee_id, last_name, email, hire_date, job_id, salary)
    VALUES (9232, 'Meg', 'Meg@xpto.com', sysdate, 'AD_ASST', 9000);
END;
/
```

- ☐ O bloco PL/SQL será executado com sucesso e será inserida uma linha na tabela EMPLOYEES.
- ☐ O bloco PL/SQL não será executado pois há um erro de sintaxe na instrução INSERT.
- ☐ O bloco PL/SQL não será executado pois não foi aberta a sessão DECLARE.
- ☐ O bloco PL/SQL não será executado pois não foi emitida a instrução COMMIT.
- ☐ O bloco PL/SQL será executado porém será exibida uma mensagem de WARNING.

* 24. Analise o bloco PL/SQL a seguir e escolha a alternativa correta.

```
DECLARE
  sal_increase  NUMBER(8,2) := 1500;
BEGIN
  UPDATE   employees
  SET      salary = salary + sal_increase
  WHERE    job_id = 'IT_PROG';
END;
/
```

- ☐ O bloco PL/SQL será executado com sucesso e todos os programadores irão receber um aumento de 1500.
- ☐ O bloco PL/SQL não será executado pois há um erro de sintaxe na instrução UPDATE.
- ☐ O bloco PL/SQL não será executado pois não foi declarada uma variável para a coluna SALARY.
- ☐ O bloco PL/SQL não será executado pois não foi emitida a instrução COMMIT.
- ☐ O bloco PL/SQL será executado porém será exibida uma mensagem de WARNING.

* 25. Considere a tabela "**PEDIDOS**" com as colunas "**id_pedido**," "**data_pedido**" e "**status_pedido**."

Você precisa escrever um bloco de código PL/SQL para excluir pedidos que estão no status "cancelado" e que foram feitos há mais de 30 dias.

Qual é o código PL/SQL correto para essa tarefa?

- ☐

```
DECLARE
    v_data_limite DATE := SYSDATE - 30;
BEGIN
    DELETE FROM pedidos
    WHERE status_pedido = 'cancelado' AND data_pedido <= v_data_limite;

    DBMS_OUTPUT.PUT_LINE('Pedidos cancelados com mais de 30 dias foram excluídos.');
```



```
END;
```
- ☐

```
DECLARE
    v_data_atual DATE := SYSDATE;
BEGIN
    DELETE FROM pedidos
    WHERE status_pedido = 'cancelado' AND data_pedido <= v_data_atual - 30;

    DBMS_OUTPUT.PUT_LINE('Pedidos cancelados com mais de 30 dias foram excluídos.');
```



```
END;
```
- ☐

```
DECLARE
    v_data_atual DATE := SYSDATE;
BEGIN
    DELETE FROM pedidos
    WHERE status_pedido = 'cancelado' AND data_pedido >= v_data_atual - 30;

    DBMS_OUTPUT.PUT_LINE('Pedidos cancelados com mais de 30 dias foram excluídos.');
```



```
END;
```
- ☐

```
DECLARE
    v_data_limite DATE := SYSDATE - 30;
BEGIN
    DELETE FROM pedidos
    WHERE status_pedido = 'cancelado' AND data_pedido >= v_data_limite;

    DBMS_OUTPUT.PUT_LINE('Pedidos cancelados com mais de 30 dias foram excluídos.');
```



```
END;
```
- ☐

```
DECLARE
    v_data_limite DATE := SYSDATE - 30;
BEGIN
    DELETE FROM pedidos
    WHERE status_pedido = 'cancelado' AND data_pedido < v_data_limite;

    DBMS_OUTPUT.PUT_LINE('Pedidos cancelados com mais de 30 dias foram excluídos.');
```



```
END;
```