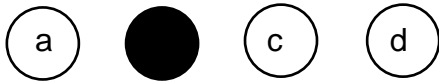




CAMPUS CHARQUEADAS

## INSTRUÇÕES GERAIS

- 1 - Este caderno de prova é constituído por 40 (quarenta) questões objetivas.
- 2 - A prova terá duração máxima de 04 (quatro) horas.
- 3 - Para cada questão são apresentadas 04 (quatro) alternativas (a – b – c – d). **APENAS UMA delas** constitui a resposta CORRETA.
- 4 - Após conferir os dados contidos no campo “Identificação do Candidato” no Cartão de Resposta, assine no espaço indicado.
- 5 - As alternativas assinaladas deverão ser transcritas para o Cartão de Resposta, que é o único documento válido para correção eletrônica.
- 6 - Marque o Cartão de Resposta conforme o exemplo abaixo, com caneta esferográfica azul ou preta, de ponta grossa:  


- 7 - **Em hipótese alguma haverá substituição do Cartão de Resposta.**
- 8 - Não deixe nenhuma questão sem resposta.
- 9 - O preenchimento do Cartão de Resposta deverá ser feito dentro do tempo previsto para esta prova, ou seja, 04 (quatro) horas.
- 10 - Serão anuladas as questões que tiverem mais de uma alternativa marcada, emendas e/ou rasuras.
- 11 - O candidato só poderá retirar-se da sala de prova após transcorrida 01 (uma) hora do seu início.
- 12 - Não é permitido o uso de calculadora.

**BOA PROVA!**



- 01.** Segundo C. J. Date, seja K um conjunto de atributos da RelVar R, então K é uma chave candidata para R, se e somente se, ela possui as seguintes propriedades
- unicidade e irredutibilidade.
  - unicidade e atomicidade.
  - irredutibilidade e atomicidade.
  - irredutibilidade e indivisibilidade.
- 02.** Segundo C. J. Date, uma RelVar está na terceira forma normal (3FN) se e somente se
- em todo o valor válido dessa RelVar, cada tupla contem exatamente um valor para cada atributo.
  - por todo o tempo, cada tupla consiste em um valor de uma chave primária que identifica alguma entidade, acompanhado por um conjunto de zero ou mais valores de atributos mutuamente independentes que descrevem essa entidade de alguma maneira.
  - todo atributo não chave é irredutivelmente dependente da chave primária.
  - por todo o tempo, o processo de normalização garantir que exista um conjunto de regras que possibilitem a não existência de inconsistência de dados, bem como o controle para que operações malsucedidas tenham plano de rollback.
- 03.** Considerando as definições de S. Silberschats, H. Korth e S. Sudarshan, **NÃO** é função de um administrador de banco de dados
- Estruturar o armazenamento e definição de método de acesso.
  - Conceder autorização para acesso a dados.
  - Revisar query SQL em nível de aplicação.
  - Definir esquema.
- 04.** Ilustrando a correspondência de padrões nas operações com string do comando SQL like, diz-se que:
- 'Perry%' localiza qualquer *string* que contenha "Perry".
  - '%IFSUL%' localiza qualquer *string* que contenha "IFSUL" como *substring*.
  - '\_ \_ \_' (leia-se três sublinhados, sem espaços em branco entre eles) localiza qualquer *string* com exatamente três caracteres.
- Está(ao) corretas a(s) afirmativa(s)
- I apenas.
  - I e II apenas.
  - II e III apenas.
  - I, II e III.

**05.** Considere a seguinte definição de dados descrita com o uso da linguagem SQL em um banco de dados relacional:

```
create table conta
(
    numero_conta      char(10),
    numero_agencia    char(15),
    saldo              numeric(12,2),
    primary key (numero_conta)
);
```

Qual das consultas abaixo apresenta o número da agência e o saldo médio, apenas das agências em que o saldo médio é maior que R\$ 1.200,00?

- a) SELECT numero\_agencia,avg(saldo) from conta group by numero\_agencia having sum(saldo) > 1200
- b) SELECT numero\_agencia,avg(saldo) from conta where avg(saldo) > 1200 group by numero\_agencia
- c) SELECT numero\_agencia,avg(saldo) from conta where saldo > 1200 group by numero\_agencia
- d) SELECT numero\_agencia,avg(saldo) from conta group by numero\_agencia having avg(saldo) > 1200

**06.** Considere a relação apresentada abaixo:

```
aluno(nome, id_aluno, nivel_grau)
```

Dentre as declarações SQL da relação aluno apresentadas abaixo, qual delas satisfaz a condição para que sejam permitidos apenas os valores 'Bacharelado', 'Mestrado' e 'Doutorado' no atributo nivel\_grau?

- a) create table aluno  
(  
nome char(15) not null,  
id\_aluno char (10),  
nivel\_grau char(15),  
primary key (id\_aluno),  
check (nivel\_grau in ('Bacharelado','Mestrado','Doutorado'))  
);
- b) create table aluno  
(  
nome char(15) not null,  
id\_aluno char (10),  
nivel\_grau char(15) check in ('Bacharelado','Mestrado','Doutorado'),  
primary key (id\_aluno),  
);
- c) create table aluno  
(  
nome char(15) not null,  
id\_aluno char (10),  
nivel\_grau char(15) just in ('Bacharelado','Mestrado','Doutorado'),  
primary key (id\_aluno),  
);
- d) create table aluno  
(  
nome char(15) not null,  
id\_aluno char (10),  
nivel\_grau char(15),  
primary key (id\_aluno),  
just (nivel\_grau in ('Bacharelado','Mestrado','Doutorado'))  
);

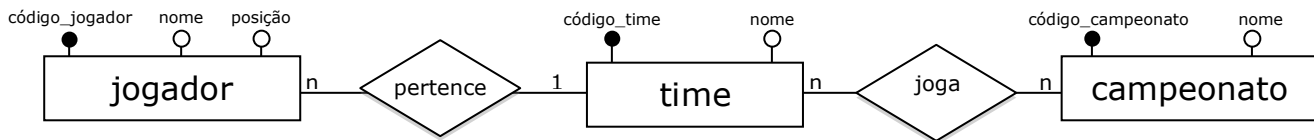
**07.** Suponha uma transação T que transfere R\$ 50,00 da conta A para a conta B. Essa transação pode ser definida como

```
T:  read(A);
     A := A -50;
     write(A);
     read(B);
     B := B + 50;
     write(B);
```

Qual das propriedades apresentadas abaixo garante que todas as ações da transação T sejam refletidas no banco de dados, ou nenhuma delas?

- Consistência.
- Atomicidade.
- Durabilidade.
- Isolamento.

**08.** Analise o diagrama entidade-relacionamento representado abaixo.



Qual o conjunto de relações que representam as tabelas estritamente necessárias para implementá-lo, considerando que as chaves primárias aparecem sublinhadas:

- jogador(código\_jogador, nome, posição)  
time(código\_time, nome)  
campeonato(código\_campeonato, nome)
- jogador(código\_jogador, nome, posição, código\_time)  
time(código\_time, nome)  
campeonato(código\_campeonato, nome)  
joga(código\_time, código\_campeonato)
- jogador(código\_jogador, nome, posição)  
time(código\_time, nome)  
campeonato(código\_campeonato, nome)  
joga(código\_time, código\_campeonato)
- jogador(código\_jogador, nome, posição, código\_time)  
time(código\_time, nome)  
campeonato(código\_campeonato, nome)

**09.** Segundo S. Silberschats, H. Korth e S. Sudarshan, o conceito de *view* é definido como

- um conjunto específico de relações disponibilizadas pelos sistemas gerenciadores de banco de dados, as quais permitem agrupar dados de interesse, a partir de informações contidas em outras relações.
- uma única relação disponibilizada pelo sistemas gerenciadores de bancos de dados, a qual permite agrupar dados de interesse, a partir de informações contidas em outras relações.
- qualquer relação pertencente ao modelo lógico, mas que se torna visível a um usuário como uma relação virtual.
- qualquer relação que não seja parte do modelo lógico, mas que se torna visível a um usuário como uma relação virtual.

**10.** Considere o código em linguagem PHP apresentado abaixo:

```
1. <?php
2. $x = 5;
3. $nome = "João";
4. echo _____;
5. ?>
```

Para complementar o comando *echo*, da linha 4, do código-fonte acima, de forma que o resultado apresentado utilizamos

João utilizou 5 caracteres especiais ~'\`'"

- a) '\$nome utilizou \$x caracteres especiais ~'\`'"
- b) \$nome.' utilizou '.\$x.' caracteres especiais ~'\`'"
- c) "\$nome utilizou \$x caracteres especiais ~'\`'"
- d) "\$nome utilizou \$x caracteres especiais ~'\`'"

**11.** Utilizando a linguagem PHP, precisamos elaborar um código-fonte que defina uma constante identificada por INSTITUICAO e com valor IFSUL. Imediatamente após a declaração da constante, precisamos fazer com o que o código imprima o texto "Bem-vindo ao IFSUL".

Considerando esses requisitos, qual dos códigos apresentados abaixo está correto?

- a) <?php  
const \$INSTITUICAO = "IFSUL";  
echo "Bem-vindo ao ".\$INSTITUICAO;  
?>
- b) <?php  
const INSTITUICAO "IFSUL";  
echo "Bem-vindo ao ".INSTITUICAO;  
?>
- c) <?php  
define ("INSTITUICAO","IFSUL");  
echo "Bem-vindo ao ".INSTITUICAO;  
?>
- d) <?php  
define ("IFSUL","INSTITUICAO");  
echo "Bem-vindo ao ".INSTITUICAO;  
?>

**12.** Considere o código em linguagem PHP apresentado abaixo:

```
1. <?php
2. $x = 50;
3. $y = 2.35;
4. $soma = _____;
5. echo $soma;
6. ?>
```

Ao complementar o código-fonte acima, na linha 4, o valor impresso pelo comando *echo*, da linha 5, qual **NÃO** é 52?

- a) (int) \$x + \$y
- b) (int) (\$x + \$y)
- c) (int) \$y + \$x
- d) (int) (\$y + \$x)

**13.** Considere o código em linguagem PHP apresentado abaixo:

```

1. <?php
2. $v1 = "1";
3. $v2 = "v3";
4. $v3 = 2;
5. $$v2+= $v1;
6. echo $$v2."-".$v3;
7. ?>

```

O que será impresso pelo comando *echo*, da linha 6, do código-fonte acima?

- a) v31-2
- b) 1-2
- c) 3-2
- d) 3-3

**14.** Considere o código em linguagem PHP apresentado abaixo:

```

1. <?php
2. $a = array("a"=>array(1,2,3),2=>array("d","e","f"),"c"=>array(4,5,6));
3. foreach($a as $b => $c)
4. {
5. echo $b." = ";
6. foreach($c as $d => $e)
7.     echo $d."-".$e." || ";
8. echo "\n";
9. }
10. ?>

```

O que será impresso por este código-fonte, considerando os comandos *echo* das linhas 5, 7 e 8?

- a) a = 0-1 || 1-2 || 2-3 ||  
2 = 0-d || 1-e || 2-f ||  
c = 0-4 || 1-5 || 2-6 ||
- b) a = 1-0 || 2-1 || 3-2 ||  
2 = d-0 || e-1 || f-2 ||  
c = 4-0 || 5-1 || 6-2 ||
- c) Array = 0-1 || 1-2 || 2-3 ||  
Array = 0-d || 1-e || 2-f ||  
Array = 0-4 || 1-5 || 2-6 ||
- d) Array = 1-0 || 2-1 || 3-2 ||  
Array = d-0 || e-1 || f-2 ||  
Array = 4-0 || 5-1 || 6-2 ||

**15.** Considere o código em linguagem PHP apresentado abaixo.

```

1. <?php
2. class c1
3.     {
4.         const nome = "Teste";
5.         function __construct()
6.             {
7.                 echo _____;
8.             }
9.     }
10. echo _____;
11. $ob = new c1();
12. ?>

```

Para que seja impresso o valor da constante nome, complementando os comandos *echo* das linhas 7 e 10, deverão ser utilizados, respectivamente, os comandos

- a) c1::nome e parent::nome
- b) self::nome e parent::nome
- c) c1::nome e self::nome
- d) self::nome e c1::nome

**16.** Considere o código em linguagem PHP apresentado abaixo.

```

1. <?php
2. class c1
3.     {
4.         public $a,$b,$c;
5.         function __construct($a,$b,$c)
6.             {
7.                 $this->a = $a;
8.                 $this->b = $b;
9.                 $this->c = $c;
10.            }
11.    }
12. $x = new c1(10,20,30);
13. $y = $x;
14. $y->a = $x->a + 100;
15. $y->b = $x->a + 110;
16. $y->c = $x->a + 120;
17. echo $x->a."-".$x->b."-".$x->c."\n";
18. echo $y->a."-".$y->b."-".$y->c."\n";
19. ?>

```

O que será impresso por este código-fonte, considerando os comandos *echo* das linhas 17 e 18?

- a) 110-120-130  
10-20-30
- b) 110-220-230  
110-220-230
- c) 10-20-30  
110-220-230
- d) 10-20-30  
110-120-130



**17.** Considere o código em linguagem PHP apresentado abaixo.

```
1. <?php
2. class c1
3. {
4.     public $a,$b;
5.     function __construct($a,$b)
6.     {
7.         $this->a = $a;
8.         $this->b = $b;
9.     }
10.    function __toString()
11.    {
12.        return $this->b;
13.    }
14. }
15. $x = new c1("IF","SUL");
16. echo $x.$x->a;
17. ?>
```

O que será impresso por este código-fonte, considerando o comando *echo* da linha 16?

- a) Object id #1SUL
- b) Object id #1IF
- c) SULIF
- d) IFSUL

**18.** Considere o código em linguagem PHP apresentado abaixo.

```
1. <?php
2. $x = 10;
3. $z = 20;
4. function f1(&$y)
5. {
6.     global $x;
7.     $x-=100;
8.     $y+= $x;
9. }
10. $x+= 100;
11. echo $x."\n";
12. f1($z);
13. echo $x."-".$z;
14. ?>
```

O que será impresso por este código-fonte, considerando os comandos *echo* das linhas 11 e 13?

- a) 110  
10-30
- b) 110  
10-20
- c) 110  
110-30
- d) 110  
110-20

**19.** Em HTML é considerado um elemento de nível de bloco

- a) iframe
- b) form
- c) map
- d) cite

**20.** Se tratando de Doctypes (DTD) em HTML, é considerado elemento válido para o DTD HTML 4.01 Strict

- a) <canvas>
- b) <center>
- c) <br>
- d) <font>

**21.** Na tag HTML <img> o atributo que especifica um texto alternativo para a imagem é

- a) desc.
- b) longdesc.
- c) title.
- d) alt.

**22.** Considerando o seguinte trecho de código HTML

1. <input type="radio" value="1"/>
2. <input type="radio" value="2"/>
3. <input type="radio" value="3"/>

O que deve ser feito de forma a garantir que só possa ser selecionado um dentre os três elementos input?

- a) Os três elementos input devem pertencer a um mesmo form.
- b) Os três elementos input devem ter mesmo valor no atributo id.
- c) Os três elementos input devem ter mesmo valor no atributo name.
- d) Os três elementos input devem ter mesmo valor no atributo class.

**23.** Com base no seguinte trecho de uma folha de estilos CSS.

```
h1#x
{
  background-color: #FF0000;
  color: #000000;
}
.y
{
  background-color: #00FF00;
  color: #000000;
}
p
{
  background-color: #0000FF;
  color: #000000;
}
```

Caso a página HTML associada a essa folha de estilos possua um elemento

```
<p class="y" id="x" >texto</p>
```

esse elemento terá cor de fundo igual a

- a) #FF0000
- b) #00FF00
- c) #0000FF
- d) #000000

**24.**Qual das regras CSS apresentadas abaixo, define que os parágrafos ficarão em negrito?

- a) `p {font-weight: bold;}`
- b) `p {text-style: bold;}`
- c) `p {font-decoration:bold;}`
- d) `p {text-decoration:bold;}`

**25.**Em CSS, a regra

```
img[width="100px"] {  
    border:1px solid #000000;  
}
```

- a) aplica uma borda e uma largura de 100 px a todas as imagens.
- b) aplica uma borda e uma altura de 100 px a todas as imagens.
- c) aplica uma borda a todas as imagens que possuem largura igual a 100 px.
- d) aplica uma borda a todas as imagens que possuem altura igual a 100 px.

**26.**Em CSS, caso exista a regra

```
p[lang|=pt] {  
    border:2px solid #FF0000;  
}
```

Dados os seguintes parágrafos:

- 1. `<p lang="pt-br">Parágrafo 1</p>`
- 2. `<p lang="pt">Parágrafo 2</p>`
- 3. `<p lang="en-gb">Parágrafo 3</p>`

terão borda os parágrafos apenas

- a) os parágrafos 1 e 2.
- b) os parágrafos 1 e 3.
- c) o parágrafo 2.
- d) o parágrafo 3.

**27.**No código HTML abaixo.

```
1. <html>  
2.   <head>  
3.     <title>TÍTULO</title>  
4.   </head>  
5.   <body>  
6.     <div>  
7.       <p>  
8.         Lorem ipsum dolor sit amet,  
9.         <span>consectetur adipiscing elit.</span>  
10.        Nulla vitae mauris lorem.  
11.       </p>  
12.       <span>Morbi eleifend facilisis erat, in laoreet dui facilisis vel.  
</span>  
13.     </div>  
14.   </body>  
15. </html>
```

Uma forma válida de usar seletores CSS de modo a aplicar uma formatação apenas ao elemento `<span>` da linha 12 é

- a) `div>span`
- b) `div span`
- c) `span`
- d) `div+span`

**28.** Considerando o código JavaScript apresentado abaixo.

```
1. <script type="text/javascript" language="JavaScript">
2.   var x;
3.   alert(x*5);
4. </script>
```

Qual será a informação apresentada pelo *alert* da linha 3?

- a) null
- b) undefined
- c) 0
- d) NaN

**29.** Considerando o código JavaScript apresentado abaixo.

```
1. <script type="text/javascript" language="JavaScript">
2.   alert(5+5+"5");
3. </script>
```

Qual será a informação apresentada pelo *alert* da linha 2?

- a) NaN
- b) 15
- c) 105
- d) 555

**30.** Considerando o código abaixo.

```
1. <html>
2.   <head>
3.     <script type="text/javascript" language="Javascript">
4.       var pessoa="Antonio";
5.       function f1(){
6.         pessoa="Beto";
7.       }
8.       function f2(){
9.         var pessoa="Carlos";
10.      }
11.    </script>
12.  </head>
13.  <body>
14.    <script type="text/javascript" language="Javascript">
15.      pessoa="Daniel";
16.      f1();
17.      f2();
18.      document.write("<p>"+pessoa+"</p>");
19.    </script>
20.  </body>
21. </html>
```

Qual será o código HTML gerado pelo script Javascript, especificamente na linha 18?

- a) <p>Antonio</p>
- b) <p>Beto</p>
- c) <p>Carlos</p>
- d) <p>Daniel</p>

**31.** Considerando o seguinte código Javascript.

```
1. <script type="text/javascript" language="JavaScript">
2.   var vetor=new Array(3);
3.   vetor[0]=5;
4.   vetor[2]=1;
5.   vetor[3]= 2;
6.   vetor[vetor[0]]=3;
7. </script>
```

O conteúdo de vetor[0], vetor[1] e vetor[vetor.length-1] são, respectivamente

- a) 5, null, 1
- b) 5, null, 3
- c) 5, undefined, 1
- d) 5, undefined, 3

**32.** Considerando o seguinte código Javascript.

```
1. <script type="text/javascript" language="JavaScript">
2.   var matriz=new Array();
3.   for (var i=0;i<=2;i++){
4.     var linha=new Array();
5.     matriz.push(linha);
6.     for(var j=1;j<=4;j++){
7.       linha.push(i*j);
8.     }
9.   }
10.  alert(matriz.length);
11. </script>
```

Qual será o resultado apresentado pelo alert da linha 10?

- a) 2
- b) 3
- c) 4
- d) 12

**33.** Suponha o seguinte trecho de código Javascript.

```
1. <script type="text/javascript" language="JavaScript">
2.   var vetor=new Array();
3.   vetor[0]=-1;
4.   vetor[1]=1;
5.   vetor[2]=5;
6.   vetor[3]=10;
7.   vetor[4]=20;
8.   vetor.sort();
9. </script>
```

O conteúdo de vetor após o comando vetor.sort() (linha 8) será

- a) {-1, 1, 10, 20, 5}
- b) {-1, 1, 5, 10, 20}
- c) {1, 10, 20, 5, -1}
- d) {20, 10, 5, 1, -1}

- 34.** Em JQuery, caso desejemos selecionar apenas os elementos *div* para os quais se tenha definido o atributo *id* com um valor terminado com "ato" (Exemplo: *div#pato*, *div#gato*, *div#rato*) devemos utilizar o comando
- a) `$ ('div[id!=ato]')`
  - b) `$ ('div[id^=ato]')`
  - c) `$ ('div[id*=ato]')`
  - d) `$ ('div[id$=ato]')`
- 35.** Na biblioteca JQuery, usamos a função `$.ajaxSetup` para configurar valores padrão para os diversos parâmetros de uma requisição AJAX. Dentre estes, qual parâmetro pode ser usado para chamar uma função que modifica o objeto XMLHttpRequest antes do seu envio?
- a) `editXMLHttpRequest`
  - b) `contentType`
  - c) `beforeSend`
  - d) `modifyObject`
- 36.** Em requisições AJAX com o uso de JQuery o evento que ocorre quando se inicia uma requisição, assumindo que nenhuma outra atividade está em curso na página, é chamado
- a) `ajax`.
  - b) `ajaxInit`.
  - c) `ajaxStart`.
  - d) `ajaxCreate`.
- 37.** O Ciclo de Vida de uma Atividade Android (Android Activity Lifecycle) define os estados ou eventos pelos quais uma atividade passa desde o momento em que ela é criada até o momento em que é finalizada.
- Nesse contexto, o primeiro método chamado quando na execução de uma atividade é
- a) `onInit`
  - b) `onResume`
  - c) `onStart`
  - d) `onCreate`
- 38.** No Android, em um TextView para que serve o atributo `autoText`?
- a) Quando o atributo `autoText` possui valor `true` alinha o texto no estilo justificado.
  - b) Quando o atributo `autoText` possui valor `true` redimensiona o texto para que melhor se adapte a resolução de tela.
  - c) Quando o atributo `autoText` possui valor `true` habilita-se a edição do texto.
  - d) Quando o atributo `autoText` possui valor `true` é executada a busca e correção de erros simples no texto.
- 39.** Segundo John W. Muchow, a verificação dos arquivos de classe não é uma operação trivial nos dispositivos móveis. Por este motivo, o processo de verificação é realizado em dois passos, respectivamente
- a) pré-verificação e verificação no dispositivo.
  - b) verificação no dispositivo e pré-verificação.
  - c) verificação local e verificação remota.
  - d) verificação estática e verificação dinâmica.
- 40.** Segundo John W. Muchow, o CLDC (Connected, Limited Device Configuration) possui dois objetivos, são eles
- a) definir uma especificação para a JVM; definir um conjunto de requisitos de hardware.
  - b) definir uma especificação para a JVM; definir um conjunto de classes Java (bibliotecas).
  - c) definir um conjunto de requisitos de hardware; definir um conjunto de classes Java (bibliotecas).
  - d) definir um conjunto de classes Java (bibliotecas); definir um protocolo para comunicação móvel.

