



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE PELOTAS - CGIC

Concurso Público (Aplicação: 07/08/2011)
Cargo: Técnico de Tecnologia da Informação/Classe D-101

LEIA ATENTAMENTE AS SEGUINTE INSTRUÇÕES:

- Verifique, nos espaços devidos do CARTÃO-RESPOSTA, se o número de controle é o mesmo que está ao lado do seu nome na folha de chamada. Caso o número de controle não corresponda ao que está nessa folha, comunique imediatamente ao fiscal de prova. Não se esqueça de assinar seu nome no primeiro retângulo.
- Marque as respostas das questões no CARTÃO-RASCUNHO, a fim de transcrevê-las, com caneta esferográfica preta, de ponta grossa, posteriormente, no CARTÃO-RESPOSTA.
- Não pergunte nada ao fiscal, pois todas as instruções estão na prova. Lembre-se de que uma leitura competente é requisito essencial para a realização da prova.
- Não rasure, não amasse nem dobre o CARTÃO-RESPOSTA, para que ele não seja rejeitado pela leitora.

Responda às questões de 01 a 08 de acordo com o texto abaixo:

(...)

1 Os avanços advindos da tecnologia da informação nas últimas décadas aumentaram o nível de eficiência na
2 área da produção gráfica, tornando-a mais ágil e aumentando significativamente a qualidade obtida. No século XV, os
3 papéis de editor, tipógrafo, gravador e impressor podiam se exercidos por uma única pessoa, mas com o passar dos séculos
4 acabaram fragmentados em várias atividades.

5 Com as novas possibilidades tecnológicas, esse domínio do processo gráfico poderá voltar a ser controlado
6 em grande parte pelo designer, que conseguirá gerenciar todo o processo de uma forma mais prática e inteligente à medida
7 que esteja ciente da enorme gama de variáveis à sua disposição. Dentre essas variáveis possíveis estão os suportes; as
8 técnicas de produção, de impressão e de encadernação; e os acabamentos. Ou como o autor divide em seu livro: pré-
9 impressão, impressão e pós-impressão, destacando todas as suas possibilidades como também suas limitações.

Prefácio do livro “A arte das Artes Gráficas”, de Cleidson Gonçalves, pela Doutora e Mestre pela Faculdade de Arquitetura e Urbanismo da USP, Maria Helena Werneck Bomeny:

<http://www.aartedasartesgraficas.com.br/sobre-o-livro.htm>

1

O objetivo do texto é:

- (a) apresentar o livro a respeito do qual faz essas considerações.
- (b) comparar os avanços tecnológicos atuais com as técnicas utilizadas no século XV.
- (c) mostrar a evolução da tecnologia digital nas últimas décadas.
- (d) identificar as etapas que constituem o processo gráfico.
- (e) criticar o uso da tecnologia no campo das artes gráficas.

2

Na expressão “... tornando-a mais ágil...” (linha 02), o elemento sublinhado refere-se à

- (a) área da produção gráfica.
- (b) informação.
- (c) tecnologia da informação.
- (d) qualidade.
- (e) tecnologia.

3

Na sequência “... acabaram fragmentados em várias atividades”. (linha 04), a palavra sublinhada poderia ser substituída, no texto, sem alteração de sentido, por

- (a) reduzidos.
- (b) destruídos.
- (c) divididos.
- (d) fragilizados.
- (e) quebrados.

4

Em “No século XV, os papéis do editor, tipógrafo, gravador e impressor podiam ser exercidos por uma única pessoa, mas com o passar dos séculos acabaram fragmentados em várias atividades.” (linhas 02 a 04), o nexos sublinhado estabelece uma relação de

- (a) adversidade.
- (b) temporalidade.
- (c) causalidade.
- (d) conformidade.
- (e) proporcionalidade.

5

Das afirmações abaixo,

- I) Na linha 06, o nexu que se refere a designer.
- II) Há um equívoco quanto ao uso do ponto e vírgula na linha 07, portanto deveria ser substituído por uma vírgula.
- III) A expressão pós-impressão (linha 09) refere-se, no texto, a acabamentos (linha 08).

- (a) apenas a I e a II estão corretas.
- (b) apenas a I está correta.
- (c) apenas a II está correta.
- (d) apenas a III está correta.
- (e) apenas a I e a III estão corretas.

6

Nas linhas 06 e 07, o nexu à medida que estabelece uma relação de

- (a) finalidade.
- (b) causalidade.
- (c) proporcionalidade.
- (d) conformidade.
- (e) adversidade.

7

A única relação correta entre a palavra sublinhada e o elemento referido está na alternativa

- (a) seu livro (linha 08) – do livro
- (b) à sua disposição (linha 07) – do designer
- (c) suas possibilidades (linha 09) – da encadernação
- (d) suas limitações (linha 09) – do autor
- (e) fragmentados (linha 04) – o tipógrafo, o gravador e o impressor

8

Dos pares abaixo, o único em que os elementos estão numa relação de causa/consequência, segundo o texto, é

- (a) técnica de acabamento/gerenciamento pelo designer.
- (b) tecnologia da informação/produção gráfica.
- (c) aumento do nível de eficiência/possibilidade de impressão.
- (d) progresso tecnológico/maior agilidade e qualidade.
- (e) técnica de pré-impressão/técnica de produção.

CONHECIMENTOS ESPECÍFICOS

09

No SGBD PostgreSQL, existe um arquivo de configuração da autenticação dos clientes. Este arquivo permite configurar quais hosts têm permissão de conectar com o banco, como os clientes são autenticados, nomes dos usuários que podem usar e quais bancos eles podem acessar. **O nome deste arquivo é**

- (a) postmaster.opts.
- (b) postgres.conf.
- (c) pg_hba.conf.
- (d) pg_config.inc.
- (e) pg_sql.conf.

10

O conceito “Atributo ou conjunto de atributos que referenciam a chave primária em uma outra tabela (ou da mesma tabela em um autarelacionamento) e tem por objetivo manter a integridade referencial do banco” refere-se à

- (a) chave candidata.
- (b) chave de acesso.
- (c) chave integral.
- (d) chave estrangeira.
- (e) chave referencial.

11

Uma importante característica do SGBD PostgreSQL é o fato de se tratar de um software livre, regido pela licença

- (a) GPL.
- (b) LGPL.
- (c) Creative Commons.
- (d) BSD.
- (e) Pública Mozilla.

12

O SGBD PostgreSQL, por default, atende as requisições na porta

- (a) 3306.
- (b) 5432.
- (c) 8080.
- (d) 27017.
- (e) 28017.

13

Na técnica de gerenciamento de projetos conhecida por SCRUM, as pessoas podem assumir os papéis de

- (a) scrum líder equipe, scrum líder geral e scrum operacional.
- (b) scrum product master, scrum product sênior e scrum product júnior.
- (c) scrum analista, scrum líder e scrum desenvolvedor.
- (d) scrum master, product owner e scrum team.
- (e) scrum analista senior, scrum analista master e scrum analista.

14

No SCRUM, a lista contendo todas as funcionalidades desejadas para um produto é conhecida por

- (a) product list of items.
- (b) product items.
- (c) product backlog.
- (d) product demands.
- (e) product needs.

15

Planning Poker é uma técnica baseada no consenso de toda a equipe, usando um conjunto de cartas com valores específicos que são utilizados nas

- (a) estimativas de tempo das tarefas ou estórias.
- (b) escolhas de quais colaboradores serão sorteados para correções de codificação.
- (c) escolhas de quais tarefas serão prioritárias.
- (d) escolhas de quais sistemas (numerados) serão prioritários.
- (e) priorizações de quais módulos serão refactorados.

16

A programação extrema (do inglês eXtreme Programming), ou simplesmente XP, possui, como valores,

- (a) comunicação, simplicidade, feedback, coragem e respeito.
- (b) agilidade, corretude e refacturação.
- (c) transparência, auto-organização, integridade e agilidade.
- (d) uso de métodos, codificação ágil, documentação plena e honestidade.
- (e) agilidade, honestidade, transparência e integridade.

17

A equação de 2º grau $x^2 - 10x + 24$ pode ser resolvida, na linguagem PHP, com o comando

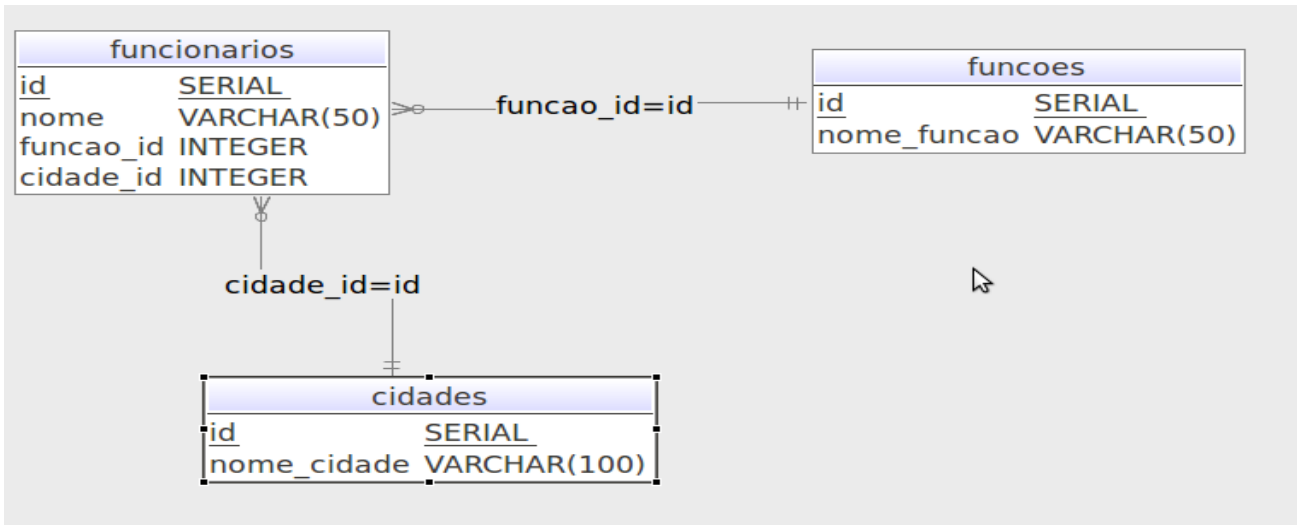
- (a) \$resultado = (\$x*\$x)-(10*\$x)+24;
- (b) %resultado = (%x^2)-(10*%x)+24;
- (c) \$resultado = (\$x elev 2)-(10 x \$x)+24;
- (d) resultado = (x & x)-(10 ** x) + 24;
- (e) \$resultado = (\$x2)-(10*\$x)+24;

18

Na linguagem PHP, permitem a execução em "loop", ou repetição de um bloco de código, os comandos

- (a) repeat, go/while e switch.
- (b) repeat, view e if.
- (c) replay, renew e repeat.
- (d) while, foreach e for.
- (e) perform, rewind e replay.

Considere o diagrama ER abaixo e o script SQL em seguida para geração do banco de dados.



```

CREATE TABLE funcoes (
    id SERIAL NOT NULL
    , nome_funcao VARCHAR(50)
    , PRIMARY KEY (id)
);
  
```

```

CREATE TABLE cidades (
    id SERIAL NOT NULL
    , nome_cidade VARCHAR(100)
    , PRIMARY KEY (id)
);
  
```

```

CREATE TABLE funcionarios (
    id SERIAL NOT NULL
    , nome VARCHAR(50)
    , funcao_id INTEGER
    , cidade_id INTEGER NOT NULL
    , PRIMARY KEY (id)
    , CONSTRAINT FK_funcionarios_1 FOREIGN KEY (funcao_id)
        REFERENCES funcoes (id)
    , CONSTRAINT FK_funcionarios_2 FOREIGN KEY (cidade_id)
        REFERENCES cidades (id)
);
  
```

Tendo como base as informações acima, a afirmativa correta é

- (a) para cada cidade, existe sempre um funcionário.
- (b) a relação entre a tabela “funcionarios” e a tabela “funcoes” é 1 para 1.
- (c) funcao_id é chave primária da tabela “funcionarios”.
- (d) um funcionário possui, obrigatoriamente, uma cidade vinculada.
- (e) toda função tem, pelo menos, um funcionário.

Considerando um relacionamento N:M (muitos para muitos) em um modelo Entidade-Relacionamento, é correto afirmar que, para efetivar em tabelas de um banco de dados relacional é necessário

- (a) criar, no mínimo, mais 2 tabelas, com as chaves primárias e estrangeiras envolvidas no relacionamento.
- (b) passar a chave primária do lado N para o lado M como chave estrangeira.
- (c) criar novas colunas (chaves estrangeiras) no lado M.
- (d) criar uma tabela auxiliar para cada uma das tabelas envolvidas no relacionamento.
- (e) criar uma nova tabela que possua como seus campos, pelo menos, as chaves primárias das tabelas envolvidas.

Considere a estrutura abaixo para responder às questões 21, 22, 23, 24 e 25.

Estrutura da tabela cursos

campo	tipo	observação
id	serial	este campo é chave primária da tabela cursos
nomecurso	text	

Estrutura da tabela alunos

campo	tipo	observação
id	serial	este campo é chave primária da tabela alunos
nome	text	
curso_id	integer	este campo é chave estrangeira e está relacionado a tabela/campo cursos/id

Conteúdo da tabela alunos

id	nome	curso_id
1	Ana	1
2	José	1
3	Maria	

Conteúdo da tabela cursos

id	nomecurso
1	Direito
2	Medicina
3	Informática

No psql, ambiente interativo do SGBD PostgreSQL, considerando que o usuário conectado tenha permissão para criar tabelas, os comandos que criam as tabelas “cursos” e “alunos” são, respectivamente,

- (a) CREATE TABLE cursos (id serial (pk), nomecurso text);
CREATE TABLE alunos (id serial (pk), nome text, curso_id integer (fk) (curso_id --> cursos(id));
- (b) CREATE TABLE cursos (id serial, primary key , nomecurso text);
CREATE TABLE alunos (id serial, primary key, nome text, curso_id integer, foreign key (curso_id)->cursos(id));
- (c) CREATE TABLE cursos (id serial (pk), nomecurso text);
CREATE TABLE alunos (id serial (pk), nome text, curso_id integer (foreign key) (curso_id -> cursos(id));
- (d) CREATE TABLE cursos (id serial chave primária, nomecurso text);
CREATE TABLE alunos (id serial chave primária, nome text, curso_id integer (chave estrangeira) (curso_id -> cursos(id));
- (e) CREATE TABLE cursos (id serial, nomecurso text, primary key (id));
CREATE TABLE alunos (id serial, nome text, curso_id integer, primary key (id), foreign key (curso_id) references cursos(id));

Considerando as tabelas “alunos” e “cursos” (estrutura e conteúdo), analise o comando select, descrito abaixo, e assinale o item que resultaria da execução deste comando no ambiente interativo do SGBD PostgreSQL chamado psql.

```
SELECT a.nome, c.nomecurso
FROM alunos AS a
LEFT JOIN cursos AS c
ON a.curso_id = c.id;
```

(a) nome | nomecurso
-----+-----
Ana | Direito
José | Direito
(2 linhas)

(d) nome | nomecurso
-----+-----
Maria |
| Medicina
| Informática
(3 linhas)

(b) nome | nomecurso
-----+-----
Ana | Direito
José | Direito
Maria |
| Medicina
| Informática
(5 linhas)

(e) nome | nomecurso
-----+-----
Ana | Direito
José | Direito
Maria | Medicina
Maria | Medicina
(5 linhas)

(c) nome | nomecurso
-----+-----
Ana | Direito
José | Direito
Maria |
(3 linhas)

Considerando as tabelas “alunos” e “cursos” (estrutura e conteúdo), analise o comando `select`, descrito abaixo, e assinale o item que resultaria da execução deste comando no ambiente interativo do SGBD PostgreSQL chamado `psql`.

```
SELECT a.nome, c.nomecurso
FROM alunos AS a
INNER JOIN cursos AS c
ON a.curso_id = c.id
```

(a) nome | nomecurso
-----+-----
Ana | Direito
José | Direito
(2 linhas)

(b) nome | nomecurso
-----+-----
Ana | Direito
José | Direito
Maria |
(3 linhas)

(c) nome | nomecurso
-----+-----
Ana | Direito
José | Direito
Maria |
| Medicina
| Informática
(5 linhas)

(d) nome | nomecurso
-----+-----
Maria |
| Medicina
| Informática
(3 linhas)

(e) nome | nomecurso
-----+-----
Ana | Direito
José | Direito
Maria | Medicina
Maria | Medicina
(5 linhas)

O comando que cria uma visão (*view*) chamada “`v_alunos_cursos`”, que apresenta os nomes dos alunos e seus respectivos cursos (somente os alunos que possuem cursos), é

- (a) `create view v_alunos_cursos as “select a.nome, c.nomecurso from alunos as a, cursos as c”;`
- (b) `create view (v_alunos_cursos) as select a.nome, c.nomecurso from alunos as a, cursos as c where a.id = c.id;`
- (c) `create view v_alunos_cursos as select a.nome, c.nomecurso from alunos as a, cursos as c where a.curso_id = c.id;`
- (d) `create v_alunos_cursos as view for select a.nome, c.nomecurso from alunos as a, cursos as c where a.curso_id = c.id;`
- (e) `create v_alunos_cursos for view as select a.nome, c.nomecurso from alunos as a, cursos as c where a.curso_id = c.id;`

O comando que conta o total de alunos por curso, ou seja, que apresenta o total de alunos e o nome do curso é

- (a) `select sum(“total de alunos”),c.nomecurso from cursos as c, alunos as a where a.curso_id=c.id group by c.nomecurso;`
- (b) `select sum(*),c.nomecurso from cursos as c, alunos as a where a.id=c_curso.id group by c.nomecurso;`
- (c) `select count(*),a.nomecurso from cursos as c, alunos as a where a.curso_id=c.curso_id group by c.nomecurso;`
- (d) `select count(*),c.nome from cursos as c, alunos as a where a.curso_id=c.id group by c.total_de_alunos;`
- (e) `select count(*),c.nomecurso from cursos as c, alunos as a where a.curso_id=c.id group by c.nomecurso;`

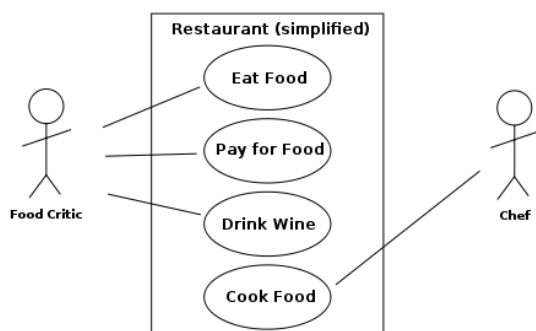


Fig. 1 - Diagrama UML

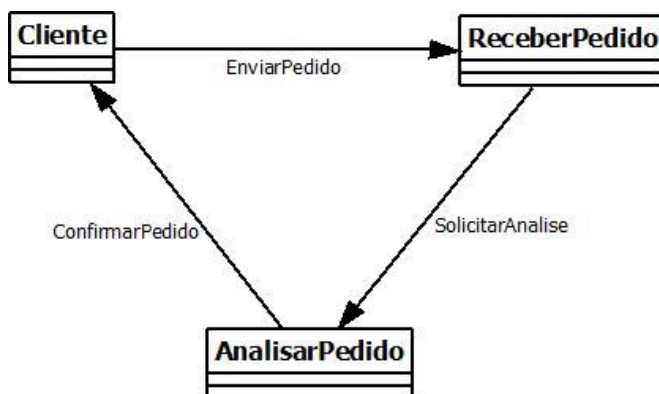


Fig. 2 - Diagrama UML

As duas figuras acima são diagramas da UML e representam, respectivamente,

- (a) diagrama de transição de estados e diagrama de classes.
- (b) diagrama de estrutura e diagrama de caso de uso.
- (c) diagrama de caso de uso e diagrama de colaboração ou comunicação.
- (d) diagrama de objetos e diagrama de transição de estados.
- (e) diagrama de caso de uso e diagrama de sequência.

Considere o seguinte trecho de código em PHP

```
<?php
$a = array('success' => true,
    'message' => 'registro gravado com sucesso',
    'result' => array(1,2,4));
print_r(json_encode($a));
?>
```

O resultado que será exibido ao executar o script acima será:

(a)

```
{["success":true,"message":"registro gravado com sucesso","result":[1,2,4]]}
```

(b)

```
{"success":true,"message":"registro gravado com sucesso","result":[1,2,4]}
```

(c)

```
[{"success":true},{"message":"registro gravado com sucesso"},{"result":[1,2,4]]}
```

(d)

```
["success":true,"message":"registro gravado com sucesso","result":{1,2,4}]
```

(e)

```
{"success":false,"message":"registro gravado com sucesso","result":[1,2,4]}
```

Considere a seguinte declaração de uma classe na linguagem **PHP5**:

```
<?php
class Pessoa {

    private $nome;

    public function __construct($nome) {
        $this->setNome($nome);
    }

    public function setNome($nome){
        $this->nome = $nome;
    }

    public function getNome(){
        return $this->nome;
    }

}
?>
```

Supondo que seja criada uma nova classe chamada *Professor* com o atributo *codigoProfessor*, que estenda a classe *Pessoa*, a alternativa que apresenta o construtor correto para a classe *Professor* é

- (a)

```
public function construct($nome, $codigoProfessor) {
    parent::construct($nome);
    $this->setCodigoProfessor($codigoProfessor);
}
```
- (b)

```
public function __construct($nome, $codigoProfessor) {
    super($nome);
    $this->setCodigoProfessor($codigoProfessor);
}
```
- (c)

```
public function __construct($nome, $codigoProfessor) {
    base($nome);
    $this->setCodigoProfessor($codigoProfessor);
}
```
- (d)

```
public function __construct($nome, $codigoProfessor) {
    parent::__construct($nome);
    $this->setCodigoProfessor($codigoProfessor);
}
```
- (e)

```
public function __construct($nome, $codigoProfessor) {
    __construct($nome);
    $this->setCodigoProfessor($codigoProfessor);
}
```

Em PHP, o trecho de código abaixo

```
<?
$var1 = "UNIVERSIDADE FEDERAL DE PELOTAS";
$var2 = "TÉCNICO EM TECNOLOGIA DA INFORMAÇÃO";
echo " UNIVERSIDADE: $var1 <br />". 'CARGO: $var2';
?>
```

exibe no navegador

- (a) UNIVERSIDADE: UNIVERSIDADE FEDERAL DE PELOTAS
CARGO: TÉCNICO EM TECNOLOGIA DA INFORMAÇÃO
- (b) UNIVERSIDADE: UNIVERSIDADE FEDERAL DE PELOTAS
CARGO: \$var2
- (c) UNIVERSIDADE: \$var1
CARGO: \$var2
- (d) UNIVERSIDADE: UNIVERSIDADE FEDERAL DE PELOTAS

CARGO:
- (e) UNIVERSIDADE: \$var1
CARGO: TÉCNICO EM TECNOLOGIA DA INFORMAÇÃO

Em PHP, o trecho de código abaixo

```
<?
$var = 1;
while(1) {
if($var == 20)
    break;
echo ++$var . "\n";
}
?>
```

exibe no navegador a sequência

- (a) 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
- (b) 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
- (c) 1
- (d) 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
- (e) 2 4 6 8 10 12 14 16 18 20

31

O elemento form da linguagem HTML contém alguns atributos. Os valores “GET” e “POST” são válidos para o atributo

- (a) action.
- (b) onsubmit.
- (c) href.
- (d) method.
- (e) style.

32

Considerando a programação Orientada a Objetos, a maneira recomendada de manipular um objeto é

- (a) por meio de seus atributos, sem a utilização de métodos.
- (b) por meio de suas constantes, que se alteram de acordo com a necessidade.
- (c) por meio de seus atributos privados, que pode ser acessado diretamente por qualquer classe filha.
- (d) por meio de variáveis estáticas.
- (e) por meio de seus métodos, uma vez que eles sabem como os dados dos atributos devem ser manipulados.

33

Considere o trecho de marcação HTML abaixo:

```
<div id="content" class="content">Ola Mundo</div>
```

Utilizando a biblioteca jQuery, a forma correta de exibir o conteúdo da DIV através do atributo ID do elemento é

- (a) \$('#content').html();
- (b) document.getElementById('content').value();
- (c) document.getElementById('#content').innerHTML();
- (d) \$('content').html();
- (e) \$('#content').val();

34

Em JavaScript, o operador lógico “&&” é utilizado para validar

- (a) se pelo menos uma das expressões for verdadeira.
- (b) se as duas expressões são diferentes.
- (c) se não existe expressão alfanumérica.
- (d) se uma expressão é numérica.
- (e) se uma expressão A e uma expressão B forem verdadeiras.

35

A principal linguagem utilizada na construção de aplicativos Ajax é

- (a) CSS.
- (b) JavaScript.
- (c) HTML.
- (d) Python.
- (e) PHP.

36

Em um sistema Web, utilizando o modelo MVC, a camada Model é responsável por

- (a) tudo que o usuário final visualiza.
- (b) realizar a persistência dos dados.
- (c) controlar o fluxo da aplicação.
- (d) interagir diretamente com a camada view.
- (e) aplicar o estilo no sistema.

37

São ferramentas de controle de versão (versionamento), EXCETO

- (a) subversion.
- (b) git.
- (c) rsync.
- (d) cvs.
- (e) sourceSafe.

38

Dado o código Python abaixo:

```
url = 'http://www.ufpel.edu.br'
print url[::-1]
```

Assinale o resultado que o código retornará:

- (a) r
- (b) http://www.ufpel.edu.br
- (c) www.ufpel.edu.br
- (d) h
- (e) rb.ude.lepfu.www//:ptth

39

Considerando a utilização do sistema de controle de versão subversion (SVN), o comando responsável por realizar o checkout de um projeto é

- (a) svn co.
- (b) svn up.
- (c) svn ci.
- (d) svn cl.
- (e) svn add.

40

Em Javascript, o trecho de código abaixo

```
<script type="text/javascript">
var alunos = [{nome:"Pedro"}, {nome:"Paulo"}, {nome:"Jose"}, {nome:"Ricardo"}];
for (i in alunos.reverse()){
    document.write('alunos[i].nome');
}
</script>
```

exibe no navegador

- (a) PedroPauloJoseRicardo
- (b) alunos[i].nomealunos[i].nomealunos[i].nomealunos[i].nome
- (c) RicardoJosePauloPedro
- (d) ordePoluaPesoJodraciR
- (e) nome.nome.nome.nome