

CONCURSO PÚBLICO

Maio - 2009



Tecnólogo/Desenvolvedor de Software

Leia estas instruções:

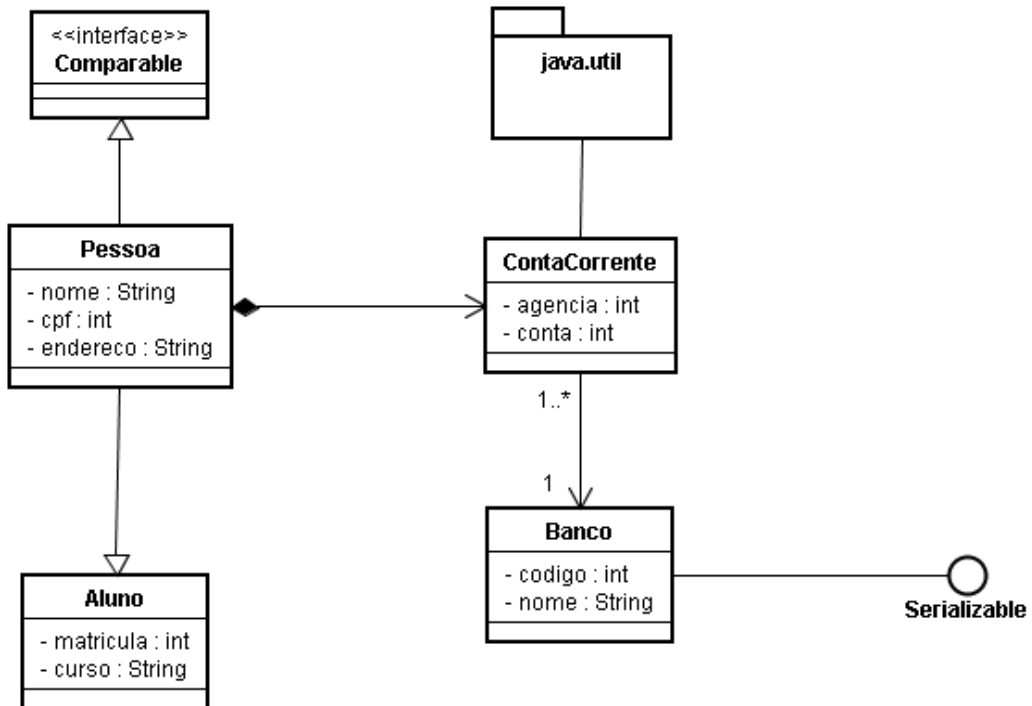
1	Confira se os dados contidos na parte inferior desta capa estão corretos e, em seguida, assine no espaço reservado para isso. Caso se identifique em qualquer outro local deste Caderno, você será eliminado do Concurso.
2	Este Caderno contém 34 questões de Conhecimentos Específicos: quatro discursivas e trinta de múltipla escolha , respectivamente.
3	Quando o Fiscal autorizar, confira se este Caderno está completo e se não apresenta imperfeição gráfica que impeça a leitura. Se você verificar algum problema, comunique-o imediatamente ao Fiscal.
4	Nas questões discursivas, será avaliado, exclusivamente , o que você escrever dentro do espaço destinado a cada resposta.
5	Escreva de modo legível. Dúvida gerada por grafia ou rasura implicará redução de pontos.
6	Cada questão de múltipla escolha apresenta apenas uma resposta correta.
7	Interpretar as questões faz parte da avaliação; portanto, não adianta pedir esclarecimentos aos Fiscais.
8	Utilize, para rascunhos, qualquer espaço em branco deste Caderno e não destaque nenhuma folha.
9	Os rascunhos e as marcações que você fizer neste Caderno não serão considerados para efeito de avaliação.
10	Você dispõe de quatro horas, no máximo, para responder às questões discursivas e de múltipla escolha e preencher a Folha de Respostas.
11	O preenchimento da Folha de Respostas é de sua inteira responsabilidade.
12	Antes de retirar-se definitivamente da sala, devolva ao Fiscal a Folha de Respostas e este Caderno.

Assinatura do Candidato: _____

Questões Discursivas

Questão 1

Observe o diagrama UML abaixo, implementado fisicamente na linguagem Java.



Com base nas análises da semântica e da sintaxe da linguagem UML, explicita quatro erros existentes nesse diagrama.

Espaço destinado à Resposta

Fim do espaço destinado à Resposta

Questão 2.

A plataforma *Java Enterprise Edition* possui a tecnologia EJB (*Enterprise JavaBeans*), que é estratégica.

Considerando essa tecnologia,

- A) quais os tipos de beans, as características e os serviços fornecidos?
- B) qual o relacionamento do EJB com as camadas WEB e EIS – *Enterprise Information System*?
- C) cite quatro diferenças básicas entre a versão 2.1 e a 3.0.

Espaço destinado à Resposta

Fim do espaço destinado à Resposta

Questão 3

A tecnologia *JavaServer Faces* é utilizada para o desenvolvimento WEB orientado a componentes, no contexto da plataforma JEE.

Considerando essa tecnologia,

- A) qual a principal função da fase Restore View e Apply Request Values?
- B) qual a principal função da *Component Tree*?
- C) mencione o nome das demais fases do ciclo de vida de uma requisição JSF e descreva, resumidamente, a função de cada uma delas.

Espaço destinado à Resposta

Fim do espaço destinado à Resposta

Questão 4

Observe a seguinte estrutura de tabelas de um banco de dados.

PESSOA				DISCIPLINA			
CODPESSOA	INTEGER	PK		CODDISCIPLINA	INTEGER	PK	
NOME	VARCHAR(200)	NOT NULL		NOME	VARCHAR(100)	NOT NULL	
CPF	BIGINT	NOT NULL		CODIGO	CHAR(4)	NOT NULL	
ENDERECO	VARCHAR(300)						
ALUNO				TURMA			
CODALUNO	INTEGER	PK		CODTURMA	INTEGER	PK	
MATRICULA	CHAR(12)	NOT NULL		ANO	INTEGER	NOT NULL	
NOMECURSO	VARCHAR(200)	NOT NULL		SEMESTRE	BYTE	NOT NULL	
CODPESSOA	INTEGER	FK NOT NULL		CODDISCIPLINA	INTEGER	FK	
				MATRICULA			
				CODMATRICULA	INTEGER	PK	
				CODTURMA	INTEGER	FK	NOT NULL
				CODALUNO	INTEGER	FK	NOT NULL
				NOTA	DECIMAL(5,2)	NOT NULL	
				STATUS	CHAR(2)	NOT NULL	

Considerando a extração de dados por meio de consultas SQL às tabelas acima, explicita

- A) o total de alunos, por turma, com notas acima de cinco;
- B) o total de matrículas, por ano/semestre, com STATUS AP, RN (AP = Aprovado e RN = Reprovado por Nota);
- C) a lista contendo a matrícula e o nome dos alunos com média superior a sete, em todas as turmas.

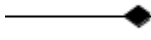

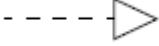
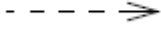
Espaço destinado à Resposta

Mais Espaço na Folha seguinte

Continuação do Espaço destinado à Resposta da Questão 4

Fim do espaço destinado à Resposta

06. Sobre os diagramas de classe na linguagem UML, pode-se afirmar:

- A) A composição indica uma generalização forte entre o todo-parte e é representada pela notação  .
- B) A associação é utilizada para indicar que uma determinada classe importa ou usa outra classe e é representada pela notação  .
- C) A realização é utilizada para denotar a implementação de uma interface por uma classe e é representada pela notação  .
- D) A dependência é um relacionamento bidirecional que indica se uma classe depende de outra e é representada pela notação  .

07. O polimorfismo é uma das técnicas mais importantes da orientação a objetos.

Uma definição correta dessa técnica é:

- A) tratamento de comportamentos diferentes de instâncias distintas, através de uma abstração.
- B) mecanismo que aumenta a modularidade do sistema, com o objetivo de diminuir acoplamentos.
- C) mecanismo que possibilita diversos comportamentos do mesmo método, através da sobrecarga (*overload*).
- D) tratamento de comportamentos diferentes de várias instâncias de uma mesma classe.

08. Considere as classes abaixo, definidas em arquivos-fontes diferentes:

```
import java.io.IOException;

1. public class Teste1 {
2.     public float metodo(float a, float b) throws IOException {
3.     }
4. }

1. public class Teste2 extends Teste1 {
2.
3. }
```

A instrução de código que, adicionada à linha 2 da classe Teste2, **não** ocasionaria erros de compilação é:

- A) `public float metodo (float a, float b) throws Exception { }`
- B) `float metodo(float a, float b) { }`
- C) `public int metodo (float a, float b) throws Exception { }`
- D) `public float metodo(float p, float q) { }`

09. Sobre o tratamento de exceções na linguagem Java, pode-se afirmar:

- A) A classe `java.lang.Exception` é considerada a base para todos os tipos de erros na linguagem Java.
- B) A exceção `java.lang.OutOfMemoryError`, emitida pela JVM, pode ser tratada por um *catch* em `java.lang.Error`.
- C) O *finally* é executado em qualquer situação, exceto se houver um *return* no bloco *try*.
- D) A cláusula *throws* pode ser indicada no método, mesmo se não existirem exceções disparadas no corpo do método.

10. Sobre a API *Java Collections* (pacote `java.util`), pode-se afirmar que
- A) a chamada do método `get(10)` em um objeto da classe `TreeSet` retorna o décimo primeiro elemento da lista.
 - B) o método `contains` utiliza o *equals* da classe do objeto comparado.
 - C) a classe `Vector` implementa a interface `java.util.Set`.
 - D) o uso de uma instância de `java.util.Collection` permite utilizar recursos comuns de todas as coleções.
11. Dentre os níveis de isolamento, no processamento de transações de banco de dados relacionais, aquele que **não** possibilita a ocorrência do problema da leitura fantasma (*phantom read*) e o que **não** possibilita o problema da leitura suja (*dirty read*) são, **respectivamente**,
- A) Read Committed e Read Uncommitted.
 - B) Serializable e Read Committed.
 - C) Repeatable read e Read Committed.
 - D) Repeatable read e Serializable.
12. Sobre o padrão de projeto **Iterator**, pode-se afirmar que sua principal motivação é
- A) fornecer uma maneira de se ter acesso, seqüencialmente, a elementos de um objeto agregado, sem se expor sua representação interna.
 - B) percorrer listas, permitindo que sejam excluídos elementos durante o percurso.
 - C) fornecer uma maneira de se percorrermos tabelas de banco de dados através de cursores.
 - D) possibilitar que sejam realizadas instruções iterativamente, independentemente da coleção de objetos em uso.
13. Considere a necessidade da criação de uma anotação para marcar classes que devem ser transferidas pela rede em um sistema distribuído. Por padrão, a anotação deve ter o nome de `@Network` e que deve receber como parâmetro a máquina destino da transferência (Ex: `@Network(host="10.6.7.98")`). Essa anotação deve ser capaz de ser lida em tempo de execução.

Com base nisso, a sintaxe correta para definição de tal anotação é:

	<pre>@Retention(RetentionPolicy.RUNTIME) @Target({ElementType.TYPE})</pre>
A)	<pre>public @interface Network { String host(); }</pre>
	<pre>@Target({ElementType.CLASS})</pre>
B)	<pre>public @interface Network { String host(); }</pre>
	<pre>public @interface Network { String host(); }</pre>
C)	<pre>public @interface Network { String host(); }</pre>
	<pre>@Retention(RetentionPolicy.EXECUTION)</pre>
D)	<pre>public @annotation Network { String host(); }</pre>

14. O *framework* Hibernate permite o mapeamento das associações de classe com o banco de dados relacional.
Sobre o mapeamento das associações utilizando as anotações do Hibernate, pode-se afirmar:
- A) O relacionamento @OneToMany possui *lazy loading* desabilitado por padrão.
 - B) O relacionamento @OneToMany exige a definição de uma coluna de junção na anotação.
 - C) As anotações @OneToOne e @Embedded possuem a mesma semântica.
 - D) A anotação @ManyToOne complementa-se com a definição da anotação @JoinColumn.
15. Suponha que você decida utilizar Hibernate na arquitetura de persistência de sua aplicação. Durante os testes de desempenho, o sistema apresenta consumo excessivo do processamento do banco de dados, devido ao excessivo número de consultas SQL, além de consultas com muitas junções e grande quantidade de campos.
Uma possível solução para esse problema será
- A) substituir a criação da sessão do Hibernate do SessionFactory e transferi-la para o SessionWeightFactory.
 - B) criar projeções em consultas e diminuir o uso de fetchs EAGER.
 - C) substituir as consultas Criteria por HQL, uma vez que elas utilizam os parâmetros de otimização definidos nas anotações.
 - D) refatorar as buscas dos dados para elas utilizarem DAO (*Data Access Object*) e DTO (*Data Transfer Object*).
16. Sobre o desenvolvimento WEB utilizando *Servlets*, pode-se afirmar:
- A) A recuperação dos parâmetros de inicialização definidos no web.xml é feita pelo método getInitParam em HttpServletResponse.
 - B) A manipulação do *content-type* dos dados gerados pela Servlet pode ser alterada pelo método setContentType da classe HttpServletRequest.
 - C) A recuperação dos parâmetros implícitos do *browser* pode ser feita pelo método getHeader da classe HttpServletRequest.
 - D) A tag url-pattern da definição da Servlet no web.xml define a URL que a Servlet irá redirecionar ao final da *request*.
17. Considere a sintaxe de operação de JavaServer Pages 2.X.
O fragmento de código que, dentro de um contexto de existência das variáveis, executaria sem erros de sintaxe é
- A) <% = 2 + 5; %>
 - B) \${param.num1 + sessionScope.mapNum[2]}
 - C) <%-- Comentário -->
 - D) #{request.usuario.nome}
18. Sobre a tecnologia JSP, pode-se afirmar:
- A) O objeto implícito *exception* é acessível caso a página seja indicada como errorPage="true" na diretiva <%@page.
 - B) A diretiva <%@include file inclui o arquivo indicado dinamicamente.
 - C) A diretiva <%@taglib utiliza o parâmetro *location* para indicar a localização da TLD.
 - D) Os parâmetros implícitos request, response e session são acessíveis caso a página seja indicada como implicitAttributes="true" na diretiva <%@page.

19. Ao serem criadas, as classes controladoras que tratam as requisições utilizando o Struts 1.X herdam a classe
- A) ActionServlet.
 - B) ActionController.
 - C) ActionForm.
 - D) Action.
20. O framework Struts possibilita que o desenvolvedor utilize *tags* predefinidas para criação de formulários.
- Considerando-se a criação de uma caixa de combinação (*combobox*) utilizando a *tag library* html, a opção que apresenta a sintaxe correta é:
- A)

```
<html:select property="tipoUsuario">
  <html:selectValues collection="tiposUsuario" property="id"
labelProperty="denominacao"/>
</html:select>
```
 - B)

```
<html:selectOneMenu property="tipoUsuario ">
  <html:options value=" ${tiposUsuario} " property="id" labelProperty="denominacao"/>
</html: selectOneMenu >
```
 - C)

```
<html:select property="tipoUsuario">
  <html:options collection="tiposUsuario" property="id"
labelProperty="denominacao"/>
</html:select>
```
 - D)

```
<html:select property="tipoUsuario">
  <html:options list="tiposUsuario" itemValue="id" itemLabel="denominacao"/>
</html:select>
```
21. Os componentes JSF possuem um atributo que conecta a variável do modelo usado ao *managed bean* e outro que decide se o componente será exibido na página. Esses atributos são, **respectivamente**:
- A) value e visible.
 - B) binding e visible.
 - C) value e rendered.
 - D) binding e rendered.
22. Os Session Beans são Enterprise JavaBeans que têm por finalidade o processamento da lógica de negócio. Na versão 2.1 dessa tecnologia, os
- A) session beans abortam a transação caso qualquer exceção seja disparada.
 - B) session beans implementam a interface Remota.
 - C) stateless session beans são instanciados a cada chamada remota.
 - D) stateful session beans fazem ativação e passivação em disco.

23. O padrão de projeto JEE (Java Enterprise Edition) cuja finalidade é encapsular as operações de *lookup* e complexas interfaces de operações de rede é o
- A) Front Controller.
 - B) Service Locator.
 - C) Transfer Object.
 - D) NetworkProxy.
24. Sobre o controle de transações em aplicações corporativas, mais especificamente na plataforma JEE, pode-se afirmar que
- A) uma transação entre *resource managers* diferentes necessita de um protocolo de DTP (*Distributed Transaction Processing*), como o X/Open XA.
 - B) as requisições aos eventos dos *backing beans* JSF são transacionalmente gerenciadas pelo *container*.
 - C) a utilização de transação no JEE só pode ser feita dentro de componentes EJB.
 - D) a API para uso do serviço transacional do JEE é a *UserTransaction*.
25. Sobre a configuração do banco de dados PostgreSQL, pode-se afirmar:
- A) O atributo *shared_buffers* indica o espaço utilizado para as operações de vacuum.
 - B) O atributo *listen_address* do *postgres.conf* indica a interface de rede cujo banco de dados irá aguardar conexões.
 - C) A configuração de quais redes e endereços IP podem conectar-se ao banco é feita no arquivo *pg_permission.conf*
 - D) O WRITE AHEAD LOG, por padrão, é desabilitado no banco de dados.
26. Sobre segurança de aplicações WEB, pode-se afirmar:
- A) O SSL utiliza criptografia assimétrica para os dados transmitidos na conexão.
 - B) O protocolo HTTP não é seguro ao realizar requisições através do método GET, mas, usando-se o método POST, os dados são enviados confidencialmente pela rede.
 - C) A utilização do SSL no protocolo HTTP exige um par de certificados: o do cliente e o do servidor.
 - D) As entidades certificadoras realizam a assinatura digital do certificado do *site* seguro.
27. O servidor WEB Apache pode servir como um balanceador via *software* para servidores Tomcat/JBoss.
- Sobre essa funcionalidade, pode-se afirmar:
- A) Suporta a linguagem Perl como elemento intermediador entre o Apache e Tomcat.
 - B) Utiliza o módulo do apache *mod_rewrite* para redirecionar ao Tomcat.
 - C) Utiliza o módulo do apache *mod_jk* suportado pelo Tomcat através do protocolo AJP13.
 - D) Suporta o balanceamento de carga através de memória compartilhada.
28. Sobre a metodologia de desenvolvimento ágil Scrum, pode-se afirmar que
- A) sua principal defesa são os testes unitários e a integração simplificada.
 - B) é uma metodologia iterativa e incremental semelhante à do processo unificado.
 - C) é uma metodologia de desenvolvimento iterativo baseada em Sprints e backlogs.
 - D) a reunião diária é a ocasião em que o gerente de projeto determina à equipe o que fazer.

29. De acordo com o RUP (*Rational Unified Process*), no escopo da iteração (ou projeto), o artefato responsável pela definição das metas e dos objetivos dos testes, dos itens-alvo, da abordagem adotada, dos recursos necessários e dos produtos que serão liberados denomina-se
- A) Plano de Testes.
 - B) Guia de Testes.
 - C) Arquitetura de Testes.
 - D) Caso de Testes.
30. O *JBoss Application Server* é um dos servidores de aplicações mais utilizados no mundo. Sobre esse servidor, pode-se afirmar:
- A) O cluster do JBoss utiliza o Smart-Proxy para soluções baseadas em RMI e HTTP.
 - B) Utiliza a arquitetura modular baseada em JavaGroups para integração entre o servidor WEB e EJB container.
 - C) Os *deployment descriptors* da aplicação WEB `jboss-web.xml` são portáveis para qualquer outro servidor JEE.
 - D) Utiliza a arquitetura JMX (*Java Management Extension*) para compor módulos flexíveis, que podem ser adicionados e retirados do servidor.