

Lista de Exercícios I - Prova I

1. O que você entende por Persistência de Dados?

Resposta: Persistência de dados refere-se à capacidade de manter informações ou dados de forma permanente, geralmente em um meio de armazenamento, para que possam ser recuperados e utilizados posteriormente, mesmo após o encerramento de um programa ou sistema. Isso envolve o armazenamento de dados em bancos de dados, arquivos, sistemas de arquivos, ou qualquer outro meio de armazenamento durável. A persistência de dados é fundamental em aplicativos e sistemas para preservar informações críticas e permitir que os dados sejam acessados e mantidos ao longo do tempo.

2. Quais as diferenças entre objetos transientes e objetos persistentes? Explique.

Resposta: Objetos transientes existem apenas temporariamente na memória durante a execução de um programa e não têm persistência. Objetos persistentes estão vinculados a um armazenamento permanente, como um banco de dados, e seus dados são duradouros, podendo ser recuperados e mantidos entre diferentes execuções do programa.

3. Qual a principal diferença entre Banco de Dados Relacional e Banco de Dados Orientado a Objetos?

Resposta:

Banco de Dados Relacional:

Organização: Os bancos de dados relacionais organizam os dados em tabelas com linhas e colunas. Cada tabela representa uma entidade e cada linha é um registro dessa entidade.

Modelo de Dados: Utilizam um modelo de dados relacional que se baseia na teoria de conjuntos e álgebra relacional. As relações entre as tabelas são estabelecidas por meio de chaves estrangeiras.

Linguagem de Consulta: Utilizam SQL (Structured Query Language) para consultar e manipular os dados.

Padrão: Exemplos incluem MySQL, PostgreSQL, Oracle, SQL Server.

Banco de Dados Orientado a Objetos:

Organização: Organizam os dados em estruturas semelhantes a objetos, incluindo classes e objetos, permitindo a representação de relacionamentos e comportamentos mais complexos.

Modelo de Dados: Baseiam-se no paradigma orientado a objetos, onde os dados são representados como objetos que têm atributos e métodos associados.

Linguagem de Consulta: Utilizam linguagens específicas de banco de dados orientado a objetos, como OQL (Object Query Language).

Padrão: Exemplos incluem ZODB (Zope Object Database) e db4o.

4. Qual o objetivo do mapeamento objeto-relacional (ORM)?

Resposta: O objetivo do mapeamento objeto-relacional (ORM) é fazer com que seja mais fácil para os programadores trabalharem com bancos de dados em seus aplicativos. Em vez de lidar diretamente com tabelas e consultas SQL, o ORM permite que eles usem objetos e classes (como em programação orientada a objetos) para realizar operações de banco de dados. Isso torna o desenvolvimento mais simples, o código mais fácil de entender e mantém os dados em sincronia entre o aplicativo e o banco de dados. Em resumo, o ORM é uma ferramenta que traduz entre a linguagem dos bancos de dados e a linguagem de programação.

5. Defina:

a) Dados Estruturados: São dados organizados em um formato fixo e previsível, geralmente em tabelas ou formatos tabulares. Cada valor de dado é atribuído a um campo específico, tornando-o facilmente pesquisável e acessível, como informações em um banco de dados relacional.

b) Dados Semiestruturados: São dados que não se encaixam perfeitamente em um formato rígido e fixo, mas ainda possuem alguma estrutura, como documentos XML, JSON ou até mesmo planilhas de Excel. Eles podem conter elementos repetidos ou opcionais e não seguem um esquema estrito de organização.

c) Dados Não Estruturados: São dados que não têm uma estrutura pré-definida e organização clara, como texto livre em documentos, áudio, vídeo, e-mails ou postagens em redes sociais. Eles são mais difíceis de processar e analisar automaticamente devido à falta de formatação consistente.

6. Qual o nome da biblioteca responsável pela extração/captura de dados disponíveis em arquivos HTML ou XML? Explique.

Resposta: A biblioteca comumente usada para extrair dados de arquivos HTML ou XML é o **JSoup**. Ela fornece uma interface fácil para analisar, navegar e manipular documentos web, permitindo a extração de informações específicas de forma programática. O JSoup é amplamente utilizado para tarefas de web scraping e análise de dados em documentos web.

7. Os arquivos do tipo XML (Extensible Markup Language) surgiram como forma de estruturação e troca de dados pela internet. Dentre suas principais características

preencha os seguintes questionamentos:

- a) Sintaxe inicial na primeira linha do arquivo.xml**
- b) Os dados são organizados em formato hierárquico ou tabular?**
- c) Quais são as formas de representação de um documento XML. Justifique.**

a) A sintaxe inicial em um arquivo XML geralmente inclui uma declaração XML na primeira linha, que é comumente escrita da seguinte forma:

```
<?xml version="1.0" encoding="UTF-8"?>
```

b) Os dados em um documento XML são organizados em formato hierárquico. A estrutura é baseada em elementos que podem conter outros elementos, criando uma árvore de dados. Essa hierarquia permite representar informações complexas de maneira organizada.

c) As formas de representação de um documento XML incluem:

Elementos: Os dados são representados por elementos que são delimitados por tags, como `<nome>` e `</nome>`. Os elementos podem conter texto e/ou outros elementos aninhados, formando uma hierarquia.

Atributos: Os elementos podem ter atributos que fornecem informações adicionais. Por exemplo, `< Pessoa nome="João" idade="30">`.

Comentários: Comentários podem ser inseridos no documento XML para fornecer explicações ou anotações, começando com `<!--` e terminando com `-->`.

Instruções de Processamento: São usadas para fornecer instruções específicas para o processamento do documento, começando com `<?` e terminando com `?>`.

Declaração XML: Como mencionado anteriormente, a declaração XML na primeira linha do documento indica a versão do XML e a codificação de caracteres utilizada.

8. Elabore um documento xml sobre produtos disponíveis para venda em empresas do comércio eletrônico/móveis/imóveis/roupas, a partir das seguintes condições:

- O produto deve possuir 5 características;

- Cada produto deve ter um nome de identificação;
- No documento deverá ter pelo menos dois produtos preenchidos.

Resposta:

```
<?xml version="1.0" encoding="UTF-8"?>
<produtos>
  <produto>
    <categoria>Móveis</categoria>
    <nome>Sofá de Couro</nome>
    <caracteristicas>
      <caracteristica>Cor: Marrom</caracteristica>
      <caracteristica>Material: Couro genuíno</caracteristica>
      <caracteristica>Estilo: Moderno</caracteristica>
      <caracteristica>Tamanho: 3 lugares</caracteristica>
      <caracteristica>Dimensões: 220cm x 90cm x 80cm</caracteristica>
    </caracteristicas>
  </produto>
  <produto>
    <categoria>Roupas</categoria>
    <nome>Vestido de Verão Floral</nome>
    <caracteristicas>
      <caracteristica>Cor: Azul e Amarelo</caracteristica>
      <caracteristica>Tamanho: M</caracteristica>
      <caracteristica>Material: Algodão</caracteristica>
      <caracteristica>Decote: V-neck</caracteristica>
      <caracteristica>Comprimento: Médio</caracteristica>
    </caracteristicas>
  </produto>
  <produto>
    <categoria>Eletrônicos</categoria>
    <nome>Smartphone Modelo X</nome>
    <caracteristicas>
      <caracteristica>Marca: ABC Electronics</caracteristica>
      <caracteristica>Sistema Operacional: Android</caracteristica>
      <caracteristica>Tela: 6 polegadas</caracteristica>
      <caracteristica>Armazenamento: 128GB</caracteristica>
      <caracteristica>Câmera: 16MP</caracteristica>
    </caracteristicas>
  </produto>
```

```
<produto>
  <categoria>Imóveis</categoria>
  <nome>Casa à Beira-Mar</nome>
  <caracteristicas>
    <caracteristica>Localização: Praia da Ilha</caracteristica>
    <caracteristica>Tipo: Casa de Praia</caracteristica>
    <caracteristica>Quartos: 4</caracteristica>
    <caracteristica>Banheiros: 3</caracteristica>
    <caracteristica>Área Total: 300m²</caracteristica>
  </caracteristicas>
</produto>
</produtos>
```

9. Defina o que é um documento JSON e quais suas principais características.

Resposta: Um documento JSON (JavaScript Object Notation) é um formato de dados leve e fácil de ler que é amplamente usado para representar e transmitir informações estruturadas entre um servidor e um cliente, ou entre diferentes partes de um sistema de software. Suas principais características são:

Sintaxe Simples: Usa pares chave-valor e estruturas aninhadas para representar dados de forma simples.

Facilidade de Leitura: Legível para humanos, o que facilita o entendimento.

Independência de Linguagem: Pode ser usado em várias linguagens de programação.

Tipagem Dinâmica: Suporta tipos de dados dinâmicos, como números, strings e objetos.

Amplamente Suportado: É amplamente suportado por muitas linguagens e é o formato padrão para muitas APIs web.

10. O que significa o processo de serialização (JSON.stringify) e desserialização (JSON.parse) de documentos do tipo JSON?

Resposta: Serialização com `JSON.stringify` significa transformar um objeto JavaScript em uma sequência de texto JSON.

Desserialização com `JSON.parse` significa converter uma sequência de texto JSON de volta em um objeto JavaScript que pode ser usado em código.

11. Faça um exemplo de documento JSON a partir de dados sobre serviços de vendas online.

• Utilize dados do tipo, string, inteiro, array e objetos.

Resposta:

```
{
  "servicos_de_vendas_online": [
    {
      "nome": "Entrega Expressa",
      "descricao": "Entrega rápida em até 24 horas",
      "taxa_entrega": 10.50,
      "disponivel_em_cidades": ["São Paulo", "Rio de Janeiro", "Belo Horizonte"],
      "opcoes_de_entrega": [
        {
          "tipo": "Entrega no mesmo dia",
          "custo_adicional": 5.00
        },
        {
          "tipo": "Entrega agendada",
          "custo_adicional": 3.00
        }
      ]
    },
    {
      "nome": "Pagamento com Cartão de Crédito",
      "descricao": "Aceitamos cartões Visa, MasterCard e American Express",
      "taxa_transacao": 1.50,
      "parcelamento_disponivel": true,
      "parcelas_maximas": 12
    },
    {
      "nome": "Entrega Padrão",
      "descricao": "Entrega em 3-5 dias úteis",
      "taxa_entrega": 5.00,
      "disponivel_em_cidades": ["Todas as Capitais"],
      "opcoes_de_entrega": [
        {
          "tipo": "Entrega econômica",
```

```
        "custo_adicional": 0.00
    }
]
}
]
```

12. Quais são as principais diferenças entre documentos do tipo JSON e XML.

Resposta: Sintaxe:

JSON: Usa uma sintaxe simples com pares chave-valor.

XML: Usa tags que envolvem os dados, tornando-o mais verboso.

Legibilidade:

JSON: Geralmente mais fácil de ler e escrever.

XML: Pode ser mais complexo e menos legível.

Estrutura:

JSON: Melhor para dados estruturados, como objetos e arrays.

XML: Mais flexível, usado para hierarquia de dados e documentos com marcação.

Tipagem:

JSON: Tipado, com suporte a diversos tipos de dados.

XML: Não tipado, dados podem ser interpretados de maneiras diferentes.

Uso Comum:

JSON: Amplamente usado em aplicativos web e APIs.

XML: Usado em documentos de marcação, configurações de serviços web e trocas de dados mais complexas.

13. Para que serve utilizar JDBC com Sistemas de Gerenciamento de Banco de Dados.

Resposta: Utilizar JDBC com Sistemas de Gerenciamento de Banco de Dados (SGDBs) permite que um programa em Java se conecte a um banco de dados e interaja com ele. Em termos simples, o JDBC serve para permitir que aplicativos em Java acessem, recuperem, atualizem e gerenciem dados armazenados em bancos de dados. É a ponte que permite que o código Java e o banco de dados se comuniquem.

14. Quais são os principais componentes durante a implementação do JDBC?

Explique.

Resposta:

Driver JDBC: O driver é responsável por conectar o aplicativo Java ao banco de dados.

Conexão com o Banco de Dados: A conexão é estabelecida usando uma URL que contém informações sobre o banco de dados.

Statement ou PreparedStatement: São usados para criar e executar consultas SQL no banco de dados.

ResultSet: Armazena os resultados das consultas SQL e permite o acesso aos dados recuperados.

Controle de Transações: Permite iniciar, confirmar ou reverter transações para garantir a consistência dos dados no banco de dados.

15. Cite restrições sobre a utilização do JDBC para sistemas atuais.

Resposta:

Segurança: É fundamental proteger contra injeções de SQL e outras vulnerabilidades de segurança ao usar JDBC. O código deve ser escrito de forma segura, usando instruções preparadas ou consultas parametrizadas.

Desempenho: O desempenho do JDBC depende da eficiência do código e da otimização das consultas. Consultas mal otimizadas e gerenciamento inadequado de conexões podem afetar o desempenho.

Gerenciamento de Conexões: O JDBC exige gerenciamento manual de conexões, o que pode levar a problemas de vazamento de recursos e escalabilidade. Usar um pool de conexões pode ajudar nesse aspecto.

Portabilidade: O JDBC é uma API Java, o que o torna mais adequado para aplicativos Java. Integrações com tecnologias não Java podem ser desafiadoras em termos de portabilidade e interoperabilidade.

Manutenção: Sistemas que utilizam JDBC podem exigir esforços significativos de manutenção ao longo do tempo, especialmente em ambientes grandes e complexos, à medida que os requisitos evoluem.

Lidar com essas restrições de maneira eficaz é essencial para garantir o bom funcionamento de sistemas que utilizam o JDBC.

