

## Exercícios de Fixação

### 1 Ajax

1. Criar uma página que implemente uma enquete sobre a disciplina favorita dos alunos do 3º período. A página da enquete deve apresentar a lista de disciplinas onde o aluno deverá escolher uma única disciplina. O aluno poderá votar em uma disciplina ou ver o resultado parcial até o momento. Ao votar, uma nova página será aberta com o resultado parcial informando a quantidade total de votos, a quantidade de votos de cada disciplina recebida até o momento, o percentual de votos de cada disciplina e uma barra horizontal indicando graficamente o percentual de votos.
2. Criar uma página que realize o cadastramento das informações de funcionários de uma empresa. Cada funcionário será representado pelo nome, data de nascimento e salário. A tela principal deverá apresentar uma tabela com as informações de todos os funcionários cadastrados até o momento, apresentando uma mensagem caso não existam funcionários cadastrados. Um botão de inserção de novos funcionários deve aparecer ao lado do cabeçalho da tabela. Ao clicar neste botão, uma página será aberta com as três informações de um funcionário a serem digitadas. Considere as seguintes restrições aos dados dos funcionários:
  - a) O nome deve conter, no mínimo, três caracteres;
  - b) Nenhum funcionário pode ter menos que 18 anos ou mais que 70 anos;
  - c) Nenhum funcionário pode ganhar menos que um salário mínimo.Caso alguma das restrições não seja satisfeita, uma mensagem deve aparecer ao lado do campo com erro. Ao inserir um funcionário, deverá se retornar à página principal. Ao lado do nome de cada funcionário, deverão aparecer dois botões: um para alterar e outro para remover o respectivo funcionário.
3. O jogo Genius foi muito popular nos anos 80. Ele é um jogo de memória, onde quatro cores (amarelo, verde, vermelho e azul) são apresentadas ao jogador. A cada rodada o jogo informa uma sequência de cores que o usuário deve reproduzir na mesma ordem. Implementar o jogo Genius em Ajax, incluindo as regras de negócio no servidor, seguindo as seguintes regras:
  - a) Informar a cor modificando-a temporariamente para preto.
  - b) Enquanto o jogo estiver mostrando a sequência de cores, desabilitar a entrada de dados da sequência de cores.
  - c) Informar a rodada atual, começando em 1.
  - d) Informar, na rodada atual, quantas cores faltam para ele completar a rodada.
  - e) Caso o jogador erre a sequência, apresentar uma mensagem na tela. Desabilite as cores, apenas permitindo ao jogador começar novamente o jogo.
  - f) Associar a cada cor um som e, a cada cor mostrada pelo jogo ou pressionada pelo usuário, tocar o som correspondente.

## 2 Tecnologias XML, XSLT e XPath

1. Dado arquivo XML `cd_catalog.xml` presente na página do curso, obtê-lo através de Ajax e formatar as seguintes informações utilizando JavaScript:
  - a) Gerar uma tabela com os dados presentes no documento XML através de DOM onde cada coluna armazenará os dados dos elementos como artista, álbum, ano, etc.
  - b) Acrescentar ordenação das colunas nas células de cabeçalho da tabela através de DOM. Quando o usuário clicar no cabeçalho de uma coluna, ela deve ficar ordenada em ordem crescente e, clicando-se novamente nela, em ordem decrescente.
  - c) Acrescentar filtros às consultas. Pesquisar os álbuns que satisfazem a todas as condições informadas pelo usuário.
    - i. Permitir o usuário informar um intervalo de valores para o preço dos álbuns;
    - ii. Permitir o usuário informar um intervalo de valores para o ano dos álbuns;
    - iii. Apresentar em um campo de seleção os nomes das gravadoras.
2. Alterar as funcionalidades da questão anterior utilizando XSLT no lugar de JavaScript.
  - a) Gerar a tabela através de XSLT e acrescentar-a na página.
  - b) Realizar a ordenação das colunas utilizando XSLT, removendo a tabela anterior e acrescentando a nova com as colunas ordenadas.
3. Utilizando o arquivo XML acima, filtrar as seguintes informações utilizando XSLT, retornando o resultado como uma página HTML.
  - a) Imprimir os nomes dos artistas que começam com a letra 'A'.
  - b) Calcular a soma de todos os valores dos álbuns.
  - c) Retornar as informações sobre o álbum mais antigo.
  - d) Quantos álbuns tem seus nomes começando entre as letras 'F' e 'M'.
4. Baseado no arquivo `CB2014B.xml` presente na página do curso com os resultados do campeonato brasileiro da série B de 2014, implementar as seguintes funcionalidades:
  - a) Gerar uma tabela com a classificação no campeonato, gerando as colunas de pontos ganhos, número de vitórias, empates e derrotas, gols pró, contra e saldo de gols.
  - b) Gerar uma página com os resultados das partidas por rodada.
5. Baseado no arquivo XML da questão anterior, usar XPath para responder às seguintes questões:
  - a) Qual time fez mais gols em uma única partida e quantos gols ele fez?
  - b) Qual a maior goleada do campeonato, ou seja, cuja diferença de gols seja máxima?
  - c) Qual o melhor mandante e visitante do campeonato, ou seja, os times com melhor aproveitamento como mandante e visitante?
  - d) Qual time mais empatou no campeonato?
  - e) Qual rodada teve maior média de gols?

## 3 JSON

1. Alterar a funcionalidade da questão anterior para, ao invés de processar o arquivo XML em JavaScript, realizar a conversão do arquivo `cd_catalog.xml` para o formato JSON dentro do código Java, processando o arquivo XML, gerando objetos que representem os álbuns, e enviando-os para a página neste formato.

2. Outra forma de enviar os dados do arquivo XML no formato JSON é utilizar XSLT para realizar a conversão. Crie um arquivo XSLT que converta o arquivo XML `cd_catalog.xml` para o formato JSON. Aplique o documento XSLT ao documento XML utilizando Java, obtenha o resultado no formato JSON e envie-o de volta à página.

## 4 SVG

1. Baseado nos arquivos disponibilizados na página do curso contendo o mapa do Brasil em formato SVG e um documento XML contendo dados sobre o desemprego no Brasil no ano de 2010 (IBGE), implementar uma página com as seguintes características:
  - a) A partir do valor médio de desemprego em cada estado, alterar a cor de cada estado no mapa de acordo com os dados abaixo e criar uma tabela de legenda também a partir dos dados abaixo:
    - i. Até 5%: Azul
    - ii. Até 7,5%: Amarelo
    - iii. Até 10%: Verde
    - iv. Acima de 10%: Vermelho
  - b) Gerar uma tabela com os dados de desemprego por estado, com as seguintes colunas: nome do estado, percentual de municípios com pleno emprego, percentual de municípios com desemprego abaixo da média e percentual de municípios com desemprego acima da média. Ao lado de cada percentual, apresentar um círculo proporcional ao tamanho do respectivo desemprego por estado nas seguintes cores (cyan, lime e maroon). O raio máximo do círculo deve ser de 40px.
2. 12. Apresentar um relógio em uma página com o horário obtido do servidor, sendo atualizado automaticamente a cada segundo. Apresentar as horas de três formas:
  - a) Escrevendo diretamente na página no formato HH:mm:ss.
  - b) Gerando um relógio analógico com os ponteiros de hora, minuto e segundo se movendo no tempo.
  - c) Gerando um relógio digital.
  - d) Observações: As letras B e C podem ser resolvidas usando SVG, que permite criar gráficos vetoriais em XML e embutindo diretamente na página HTML.
3. A partir do arquivo SVG disponibilizado na página do curso, que representa o mapa do Brasil, e do arquivo XML, que possui informações sobre os estados brasileiros, criar uma página HTML, importar a figura e à medida que o usuário passar o mouse por cima dos estados, apresentar abaixo da figura as informações presentes no arquivo XML no formato de tabela: Nome do estado (título da tabela), população, PIB e capital. A tabela deve possuir cor de fundo igual às presentes no arquivo SVG do respectivo estado selecionado, para manter a consistência visual. Você deve obter o arquivo XML através de Ajax. Crie um arquivo XSLT que fará a transformação do arquivo XML e retornará a tabela a ser acrescentada na página. Aplique o XSLT ao XML em JavaScript e cole a tabela resultante na página. Os arquivos SVG e XML não podem ser alterados.

## 5 WebSockets

1. Implementar um sistema que apresente as informações sobre a quantidade de votos obtidos por cada partido em uma eleição, juntamente com a quantidade de cadeiras obtidas no pleito municipal através do cálculo dos quocientes eleitoral e partidário. A eleição é composta de seis partidos, e a quantidade de votos recebidos é gerado a partir das seguintes equações:
  - a) Partido A:  $1, 1^x$

- b) Partido B:  $\log_{1,0005} x$
- c) Partido C:  $70x$
- d) Partido D:  $6500 \sin \frac{x}{57,3}$
- e) Partido E:  $35x + 150$
- f) Partido F:  $5x \ln x$
- g) Brancos:  $8x$
- h) Nulos:  $28x$

A partir das fórmulas acima, gerar automaticamente a quantidade de votos obtida por cada partido político no intervalo de  $x \in [1, 100]$  e apresentar a cada 5 segundos um desses valores. Cada resultado parcial deverá ser obtido do servidor, através de WebSockets, e apresentado na tela, informando a quantidade de votos recebidos de cada partido, juntamente com a quantidade de cadeiras obtida, no formato de uma tabela. Para calcular os quocientes eleitoral e partidário, ler a página do TRE-PE com um exemplo [<http://www.tre-pe.jus.br/eleicoes/calculo-do-quociente-eleitoral>]. Ordenar a tabela de acordo com os partidos com mais cadeiras na câmara de vereadores, considerando que ela possui 21 cadeiras.