Por Morgana Maria



Exam 70-480: Programming in HTML5 with JavaScript and CSS3

**Guia de estudo para certificação**

**PLANO DE ESTUDO**

**Capítulo 1 – Implementar e manipular estruturas e objetos do documento**

**Objetivo 1.1: Criar a estrutura do documento usando HTML**

* + Estruturar a UI usando marcação semântica, incluindo marcações para mecanismos de pesquisa e leitores de tela, como Seção, Artigo, Nav, Cabeçalho, Rodapé e A parte; criar um contêiner de layout em HTML.
  + Criar a estrutura do documento;
  + Usar a semântica de mark-up do HTML5;
  + Otimização para buscadores;
  + Otimização para leitores;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 1.2: Escrever código que interaja com controles de UI**

* + Adicionar e modificar elementos HTML programaticamente; implementar controles de mídia; implementar tela HTML5 e elemento gráfico SVG.
  + Adicionar ou modificar elementos HTML;
  + Implementar controles de mídia;
  + Implementar gráficos com HTML5 <canvas> e SVG;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 1.3: Aplicar estilo a elementos HTML programaticamente**

* + Alterar a localização de um elemento; aplicar uma transformação; exibir e ocultar elementos.
  + Alterar a localização de um elemento;
  + Aplicar transformações;
  + Exibindo e escondendo elementos;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 1.4: Implementar APIs do HTML5**

* + Implementar APIs de armazenamento e API de Geolocation
  + Usar a Storage API;
  + Usar o AppCache API;
  + Usar o Geolocation API;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 1.5: Estabelecer o escopo de objetos e variáveis**

* + Definir variáveis do tempo de vida; manter objetos fora o namespace global; usar a palavra-chave “este” para fazer referência a um objeto que disparou um evento; estabelecer escopo de variáveis localmente e globalmente
  + Estabelecer o ciclo de vida de variáveis e escopos de variáveis;
  + Evitar usar namespace global;
  + Utilizando “this”;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 1.6: Criar e implementar objetos e métodos**

* + Implementar objetos nativos; criar objetos personalizados e propriedades personalizadas para objetos nativos usando protótipos e funções; herdar de um objeto; implementar métodos nativos e métodos personalizados
  + Implementar objetos nativos;
  + Criar objetos customizados e propriedades customizadas usando protótipos;
  + Implementar herança;
  + Implementar métodos nativos e criar métodos customizados;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Capítulo 2 - Implementar fluxo de programa**

**Objetivo 2.1: Implementar fluxo de programa**

* + Iterar em coleções e itens da matriz; gerenciar decisões do programa usando instruções switch, if/then e operadores; avaliar expressões
  + Analisar expressões;
  + Trabalhando com arrays;
  + Implementando tipos especiais de array;
  + Usando métodos avançados de array;
  + Implementação de controle de fluxo iterativo;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 2.2: Criar e gerir um evento**

* + Manipular eventos comuns expostos por DOM (OnBlur, OnFocus, OnClick); declarar e manipular eventos com bolhas; manipular um evento usando uma função anônima
  + Usar eventos, incluindo gerenciar o evento por uso de funções anônimas;
  + Gerenciar eventos DOM (Documento Object Model), ex: onBlur, onFocus, onClick e sua convenção de nomenclatura “on\_” + Nome do evento;
  + Criar eventos customizados;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 2.3: Implementar manipulação de exceções**

* + Definir e responder aos códigos de erro; lançar uma exceção; solicitar verificações de nulos; implementar blocos try-catch-finally
  + Implementar try-catch-finally, incluindo setar e responder a códigos e lançar exceções;
  + Checar valores nulos;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 2.4: Implementar callback**

* + Receber mensagens do API WebSocket do HTML5; usar o jQuery para fazer uma chamada AJAX; ligar um evento; implementar um retorno de chamada usando funções anônimas; manipular o ponteiro “this”
  + Implementar comunicação bi-direcional com WebSocket API;
  + Tornar páginas dinâmicas com JQuery e Ajax;
  + “Ligar” um evento com jQuery;
  + Implementar call-back com funções anônimas;
  + Usar o ponteiro this;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 2.5: Criar um processo de trabalho da Web**

* + Iniciar e parar um web worker; passar dados para um web worker; configurar tempos limite e intervalos em um web worker; registrar um ouvinte de eventos para o web worker; limitações de um web worker
  + Iniciando com WebWorker;
  + Criar um processo WebWorker com o WebWorker API;
  + Usar um WebWorker;
  + Entendendo as limitações do WebWorker;
  + Configurar timeouts e intervalos;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Capítulo 3 - Acessar e proteger dados**

**Objetivo 3.1: Validar entrada de usuário usando elementos do HTML5**

* + Escolher os controles apropriados com base nos requisitos; implementar tipos de entrada e atributos de conteúdo do HTML para coletar entrada do usuário
  + Escolher os controles HTML5 de entrada de dados de usuário;
  + Implementar atributos contente;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 3.2: Validar entrada de usuário usando JavaScript**

* + Avaliar uma expressão regular para validar o formato de entrada; validar que você está obtendo o tipo certo de tipo de dados usando funções internas; evitar injeção de código.
  + Avaliando expressões regulares;
  + Avaliando expressões regulares em JS;
  + Validando dados com funções nativas;
  + Prevenindo code injection;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 3.3: Dados de consumo**

* + Consumir dados JSON e XML; recuperar dados usando serviços Web; carregar dados ou obter dados de outras fontes usando XMLHTTPRequest
  + Consumir JSON e XML usando webservices;
  + Usar o objeto XMLHTTPRequest;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 3.4: Serializar, desserializar e transmitir dados**

* + Manusear dados binários; manusear dados de texto como JSON e XML; implementar o método de serialização jQuery, usar o Form.Submit; analisar dados; enviar dados usando XMLHTTPRequest; limpar a entrada usando URI/codificação de formulário
  + Enviar dados usando XMLHttpRequest;
  + Serializar e deserializar dados JSON;
  + Serializar e deserializar dados binários;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Capítulo 4 - Usar CSS3 em aplicativos**

**Objetivo 4.1: Definir estilo de propriedade de texto de HTML**

* + Aplicar estilos para a aparência do texto; aplicar estilos para a fonte do texto, incluindo WOFF, @font-face, tamanho e fontes coadjuvantes; aplicar estilos no alinhamento de texto, espaçamento e recuo; aplicar estilos para a hifenação de texto; aplicar estilos para uma sombra de texto.
  + Aplicar estilos à aparência do texto;
  + Aplicar estilos à fonte do texto;
  + Aplicando estilos ao alinhamento, espaçamento e recuo do texto;
  + Aplicando estilos à hifenização de texto;
  + Aplicando estilos drop shadow no texto;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 4.2: Definir estilo de propriedade de caixa de HTML**

* + Aplicar estilos para alterar atributos de aparência, incluindo tamanho, bordas, arredondamento dos cantos das bordas, contorno, preenchimento e margem; aplicar estilos para alterar efeitos gráficos, incluindo transparência, opacidade, imagem em segundo plano, gradientes, sombra e recorte; aplicar estilos para estabelecer e alterar a posição de um elemento.
  + Aplicando estilos para alterar atributos de aparência;
  + Aplicando estilos para alterar efeitos gráficos;
  + Aplique estilos para estabelecer e alterar a posição de um elemento;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 4.3: Criar um layout de conteúdo flexível**

* + Implementar um layout usando um modelo de caixa flexível; implementar um layout de colunas múltiplas usando várias colunas; implementar um layout usando flutuação e exclusões de posição; implementar um layout usando alinhamento de grade; implementar um layout usando regiões, agrupamento e aninhamento.
  + Implementar um layout usando um modelo de box model flexivel;
  + Implementar um layout usando multi-column;
  + Implementar um layout usando position, floating, and exclusions;
  + Implementar um layout usando grid alignment;
  + Implementar um layout usando regions, grouping, and nesting;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 4.4: Criar uma UI animada e adaptável**

* + Animar objetos aplicando transições CSS; aplicar transformações 2-D e 3-D; ajustar UI com base em consultas de mídia, incluindo adaptações de dispositivos para formatos de saída, visores e representações; ocultar ou desabilitar os controles.
  + Animando objetos aplicando transições CSS;
  + Aplicando transformações 3D e 2D;
  + Ajustando a interface do usuário com base em consultas de mídia;
  + Ocultando ou desativando controles;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 4.5: Localizar elementos usando seletores de CSS e JQuery**

* + Escolher o seletor correto para referenciar um elemento; definir seletores de elemento, estilo e atributos; encontrar elementos usando pseudoelementos e pseudoclasses.
  + Definindo seletores de elemento, estilo e atributo;
  + Escolhendo o seletor correto para referenciar um elemento;
  + Localizando Elementos Usando Pseudoelementos e pseudo-classes;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 4.6: Estruturar um arquivo CSS usando seletores CSS**

* + Referenciar elementos corretamente; implementar herança; substituir herança usando !important; estilo de um elemento baseado em pseudoelementos e pseudoclasses.
  + Implementando herança;
  + Substituindo herança usando !important;
  + Estilizando um elemento baseado em pseudo-elementos e pseudo-classes;
  + Sumário do objetivo;
  + Revisão do objetivo.

**ANOTAÇÕES**

**Capítulo 1 – Implementar e manipular estruturas e objetos do documento**

**Objetivo 1.1: Criar a estrutura do documento usando HTML**

* + Criar a estrutura do documento;
  + Usar a semântica de mark-up do HTML5;
  + Otimização para buscadores;
  + Otimização para leitores;
  + Sumário do objetivo;
  + Revisão do objetivo.

No HTML5 existe uma marcação semântica.

|  |  |
| --- | --- |
| **HTML5 element** | **Description** |
| <article> | Define áreas na página |
| <aside> | Define áreas de conteúdos menores fora do fluxo da página |
| <figcaption> | Define o título do elemento de figura |
| <figure> | Define um conteúdo que contém uma figura, imagem, gráfico |
| <footer> | Define o rodapé de uma seção ou página |
| <header> | Define o cabeçalho de uma seção ou página |
| <hgroup> | Define um grupo de headings (de h1 até h6) |
| <mark> | Define que um texto precisa ser destacado |
| <nav> | Define a navegação para outras páginas no site |
| <progress> | Define o progresso de uma tarefa |
| <section> | Define conteúdo distinto de um documento ou área |

**Objetivo 1.2: Escrever código que interaja com controles de UI**

* + Adicionar ou modificar elementos HTML;
  + Implementar controles de mídia;
  + Implementar gráficos com HTML5 <canvas> e SVG;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 1.3: Aplicar estilo a elementos HTML programaticamente**

* + Alterar a localização de um elemento;
  + Aplicar transformações;
  + Exibindo e escondendo elementos;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 1.4: Implementar APIs do HTML5**

* + Usar a Storage API;
  + Usar o AppCache API;
  + Usar o Geolocation API;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 1.5: Estabelecer o escopo de objetos e variáveis**

* + Estabelecer o ciclo de vida de variáveis e escopos de variáveis;
  + Evitar usar namespace global;
  + Utilizando “this”;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 1.6: Criar e implementar objetos e métodos**

* + Implementar objetos nativos;
  + Criar objetos customizados e propriedades customizadas usando protótipos;
  + Implementar herança;
  + Implementar métodos nativos e criar métodos customizados;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Capítulo 2 - Implementar fluxo de programa**

**Objetivo 2.1: Implementar fluxo de programa**

* + Analisar expressões;
  + Trabalhando com arrays;
  + Implementando tipos especiais de array;
  + Usando métodos avançados de array;
  + Implementação de controle de fluxo iterativo;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 2.2: Criar e gerir um evento**

* + Usar eventos, incluindo gerenciar o evento por uso de funções anônimas;
  + Gerenciar eventos DOM (Documento Object Model), ex: onBlur, onFocus, onClick e sua convenção de nomenclatura “on\_” + Nome do evento;
  + Criar eventos customizados;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 2.3: Implementar manipulação de exceções**

* + Implementar try-catch-finally, incluindo setar e responder a códigos e lançar exceções;
  + Checar valores nulos;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 2.4: Implementar callback**

* + Implementar comunicação bi-direcional com WebSocket API;
  + Tornar páginas dinâmicas com JQuery e Ajax;
  + “Ligar” um evento com jQuery;
  + Implementar call-back com funções anônimas;
  + Usar o ponteiro this;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 2.5: Criar um processo de trabalho da Web**

* + Iniciando com WebWorker;
  + Criar um processo WebWorker com o WebWorker API;
  + Usar um WebWorker;
  + Entendendo as limitações do WebWorker;
  + Configurar timeouts e intervalos;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Capítulo 3 - Acessar e proteger dados**

**Objetivo 3.1: Validar entrada de usuário usando elementos do HTML5**

* + Escolher os controles HTML5 de entrada de dados de usuário;
  + Implementar atributos contente;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 3.2: Validar entrada de usuário usando JavaScript**

* + Avaliando expressões regulares;
  + Avaliando expressões regulares em JS;
  + Validando dados com funções nativas;
  + Prevenindo code injection;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 3.3: Dados de consumo**

* + Consumir JSON e XML usando webservices;
  + Usar o objeto XMLHTTPRequest;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 3.4: Serializar, desserializar e transmitir dados**

* + Enviar dados usando XMLHttpRequest;
  + Serializar e deserializar dados JSON;
  + Serializar e deserializar dados binários;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Capítulo 4 - Usar CSS3 em aplicativos**

**Objetivo 4.1: Definir estilo de propriedade de texto de HTML**

* + Aplicar estilos à aparência do texto;
  + Aplicar estilos à fonte do texto;
  + Aplicando estilos ao alinhamento, espaçamento e recuo do texto;
  + Aplicando estilos à hifenização de texto;
  + Aplicando estilos drop shadow no texto;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 4.2: Definir estilo de propriedade de caixa de HTML**

* + Aplicando estilos para alterar atributos de aparência;
  + Aplicando estilos para alterar efeitos gráficos;
  + Aplique estilos para estabelecer e alterar a posição de um elemento;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 4.3: Criar um layout de conteúdo flexível**

* + Implementar um layout usando um modelo de box model flexivel;
  + Implementar um layout usando multi-column;
  + Implementar um layout usando position, floating, and exclusions;
  + Implementar um layout usando grid alignment;
  + Implementar um layout usando regions, grouping, and nesting;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 4.4: Criar uma UI animada e adaptável**

* + Animando objetos aplicando transições CSS;
  + Aplicando transformações 3D e 2D;
  + Ajustando a interface do usuário com base em consultas de mídia;
  + Ocultando ou desativando controles;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 4.5: Localizar elementos usando seletores de CSS e JQuery**

* + Definindo seletores de elemento, estilo e atributo;
  + Escolhendo o seletor correto para referenciar um elemento;
  + Localizando Elementos Usando Pseudoelementos e pseudo-classes;
  + Sumário do objetivo;
  + Revisão do objetivo.

**Objetivo 4.6: Estruturar um arquivo CSS usando seletores CSS**

* + Implementando herança;
  + Substituindo herança usando !important;
  + Estilizando um elemento baseado em pseudo-elementos e pseudo-classes;
  + Sumário do objetivo;
  + Revisão do objetivo.