

INF321 - Projeto e Desenvolvimento de Sistemas para a Web

Roteiro de Aula Prática 01

Prof. Marcos Henrique Fonseca Ribeiro

15/09/2023

1 Informações iniciais

A aula prática de hoje visa exercitar conceitos vistos nas duas aulas anteriores da disciplina, e envolverá codificação de conteúdo em HTML e CSS.

Na pasta do Google Drive, juntamente com este documento, encontram-se três outros arquivos:

- `logodpi.jpg`: arquivo de imagem contendo a logo do Departamento de Informática. O arquivo contém uma imagem grande. **Não** redimensione ou edite a imagem, ela deve ser utilizada no trabalho do jeito que está.
- `logoufv.png`: arquivo de imagem contendo a logo da UFV. Assim como no caso anterior, o arquivo contém uma imagem grande. **Não** redimensione ou edite a imagem, ela deve ser utilizada no trabalho do jeito que está.
- `Conteúdo textual`: arquivo do Documentos Google que contém as informações que serão utilizadas para popular a página HTML que você desenvolverá durante a atividade. A ideia é que você não gaste tempo digitando conteúdo e apenas copie e cole deste arquivo.

Baixe os arquivos para **o computador local** no qual está trabalhando e siga as instruções das demais seções.

Observações:

- A consulta na Web é **liberada** e até incentivada na aula de hoje. Consulte o material disponível no Moodle à vontade, bem como os links recomendados pelo professor.
- Evite, apenas, consultar aos colegas, para tentarmos manter baixo o nível de ruído no laboratório. (Evitar não é proibir...)
- O envio da atividade deve ser feito via **Moodle**. Mais informações a respeito estão no final deste roteiro.
- A entrega via Moodle abrirá às 10:30h e se encerrará às 13:00h.

2 Preparação

Siga os passos a seguir, para fins de padronização e agilidade de correção do exercício:

1. Crie uma pasta vazia no armazenamento local da máquina na qual está trabalhando. O nome desta pasta não é importante, porém, recomenda-se que se use sua matrícula ou nome. Para fins de exemplo apenas, durante este roteiro esta suposta pasta será chamada de `p01`.
2. Dentro da pasta `p01`, crie duas subpastas, estas, sim, com nomes pré definidos:
 - `static`: mova ou copie as duas imagens que você baixou para dentro desta pasta
 - `style`: crie um arquivo vazio, dentro da mesma, chamado `geral.css`.
3. Novamente dentro de `p01`, crie agora um arquivo vazio chamado `index.html`

Após esta preparação inicial, sua estrutura de arquivos e pastas deve estar como a seguir:

```

p01
|
|--index.html
|
|--static
|   |
|   |--logodpi.jpg
|   |--logoufv.png
|   |
|--style
|   |
|   |--geral.css

```

Figura 1: Estrutura de arquivos e pastas

3 Codificando o HTML Básico

De início vamos preparar o código do arquivo `index.html` para ter o essencial. Edite seu conteúdo para que tenha as *tags* elementares:

- A declaração `<!DOCTYPE html>`
- O elemento `<html>`
- Os elementos `<head>` e `<body>`
- Dentro de `<head>`:
 - Use a tag `<meta>` para definir o conjunto de caracteres como `utf-8`
 - Defina `<title>` de acordo com o que está especificado no documento Google que você baixou no início da atividade. Em Conteúdo textual, procure por “*Título da aba do navegador*”.
 - Insira a referência ao arquivo `geral.css`. Lembre-se de usar o *path* relativo de maneira adequada e, caso não se recorde, consulte na Web como fazer a inclusão de folhas de estilo CSS a partir de um documento externo.

Para que possamos fazer logo um primeiro teste de visualização, inclua um elemento de cabeçalho de maior nível (`<h1>`) com o texto “*INF 493 - Tópicos Especiais III: Ciência de Dados*” dentro do corpo da página. Edite também o arquivo `geral.css` e inclua o conteúdo ilustrado na Figura 2.

```

1  /* Estilos por tipo de tag */
2  body {
3      font-family: Arial, Helvetica, sans-serif;
4      color: darkblue;
5  }
6
7  /* Estilos por classe */

```

Figura 2: Código CSS inicial

3.1 Testando o que foi feito

Para testar o seu código, abra o arquivo HTML no navegador. Por exemplo, assumindo que você esteja utilizando o Linux em uma máquina do laboratório, via usuário `alunos` e tenha criado a pasta `p01` diretamente na pasta do usuário, bastaria digitar, na barra de endereços do navegador, o seguinte endereço: `file:///home/alunos/p01/index.html`.

A visualização esperada do arquivo é algo como o que está na Figura 3, a seguir.



Figura 3: Visualização Esperada

Note que o endereço da barra de endereços pode ser diferente do que está na figura, a depender de onde a pasta e os arquivos estão no sistema de arquivos do Sistema Operacional do seu computador.

O importante, porém, é observar os textos presentes na aba do navegador e na tela principal. Estes têm de estar coerentes com o que foi pedido no enunciado. Além disso, note que o tipo de fonte e a cor do título grande em tela são diferentes daqueles que o navegador utiliza por padrão (*default*). Isso se deve ao uso de uma folha de estilos para formatar o conteúdo HTML. Veja, abaixo, como seria a visualização sem o uso dos estilos definidos em `geral.css`.

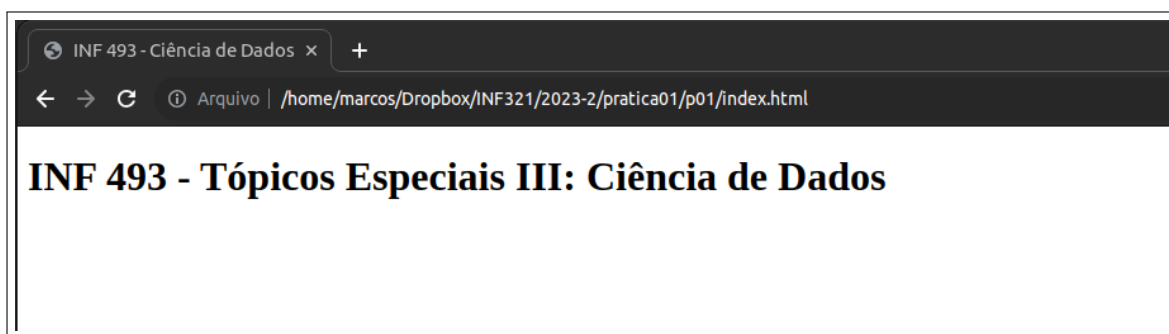


Figura 4: Visualização *Default* do Navegador

Se a sua visualização for como está na Figura 4, há algo errado com seu código HTML, ou seu código CSS ou, ainda, o seu navegador não conseguiu encontrar o arquivo de folha de estilo `geral.css`. Procure corrigir até que consiga uma visualização como a que está na Figura 3.

4 Desenvolvendo o Exercício

O primeiro passo do exercício é remover a instrução `color: darkblue;` do código do arquivo `geral.css` (ilustrado na Figura 2). O texto em azul foi apenas para ajudar a diferenciar melhor a saída em tela esperada daquela padrão do navegador. Não utilizaremos esta cor. Mas **manteremos a instrução das famílias de fonte** usadas.

O efeito que irá acontecer, ao aplicarmos o estilo do exemplo para a tag `body`, será que todo o conteúdo da páginas HTML terá sua fonte modificada para as famílias especificadas, a menos que alguma outra instrução de modificação de estilo seja aplicada a elementos específicos dentro do site.

Deste ponto do exercício em diante, trabalharemos com *screenshots* do resultado final esperado da atividade e as instruções de como fazer se darão a partir da descrição das capturas de tela.

Lembre-se, a consulta online está liberada...

4.1 O topo da página

Teremos dois elementos principais no topo da página que iremos construir:

- **Cabeçalho:** uma "barra superior", composta de dois logotipos (DPI e UFV) e um texto central, identificando a finalidade da página (trazer informações gerais de uma disciplina)
- **Título:** é o título do conteúdo, isto é, o nome da disciplina cujas informações serão mostradas ao longo da página. Este item é exatamente a tag `h1` que já adicionamos na Seção 3.

A visualização esperada de tais elementos se encontra na Figura 5, a seguir.

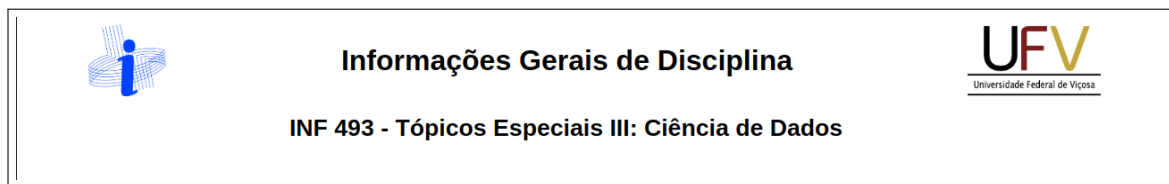


Figura 5: Visualização do Topo Esperado da Página

4.1.1 Descrição do HTML

A barra de cabeçalhos é uma subdivisão lógica da página, isto é, consiste em uma estrutura para organizar o conteúdo, agregando outros elementos, que podem possuir finalidades distintas. Para implementar este conceito deve-se criar um elemento HTML do tipo `<div>`, e vamos associá-lo a uma classe chamada `cabecalho`.

O conteúdo do cabeçalho será uma agregação da imagem do logo do DPI, associada à classe `logo`, um texto (*Informações Gerais de Disciplina*) que, para podermos aplicar estilo ao mesmo, vamos encapsular com uma tag do tipo `` e associar à classe `texto-cabecalho` e, por fim, outra imagem, também associada à classe `logo`, contendo o logo da UFV.

Não mexa em nenhum aspecto de visualização de nenhum destes elementos HTML. Faremos tudo por meio de folhas de estilo. Note que a barra de cabeçalhos deverá vir **antes** do título (`<h1>`) que já foi inserido anteriormente.

Em relação ao título previamente inserido, apenas o associe à classe `titulo`.

4.1.2 Descrição do CSS

Vamos, agora, especificar como a visualização dos elementos do topo da página deve ser feita, por meio de folhas de estilo CSS. Para isso, vamos editar o arquivo `geral.css`.

- Os parâmetros para a classe `cabecalho` são:
 - Alinhamento do conteúdo: centralizado
 - Margem do topo: 20 pixels
 - Largura, 100%. Lembre-se: este valor corresponde a 100% da largura do elemento que contém aquele que está sendo formatado. No nosso caso, o elemento `<body>` é quem é o *container* do elemento `<div>` que está sendo associado à classe `cabecalho`.
- Os parâmetros para a classe `logo` são:
 - Altura: 80 pixels
 - Alinhamento vertical: meio
- Os parâmetros para a classe `texto-cabecalho` são:
 - Tamanho da fonte: 24pt
 - Fonte em negrito
 - Deslocamento interno (*padding*) à esquerda: 15%
 - Deslocamento interno à direita: 15%
 - Altura: 100%
 - Alinhamento vertical: meio
- Por fim, os parâmetros para a classe `titulo` são:
 - Margem do topo: 10 pixels
 - Margem inferior: 10 pixels
 - Alinhamento do texto: centralizado
 - Tamanho de fonte: 20pt
 - Fonte em negrito
 - Alinhamento vertical: meio
 - Largura: 95%

4.2 Informações Gerais da Disciplina

Após o topo, segue-se uma lista contendo informações gerais sobre a disciplina descrita pela página. Mais uma vez, vamos agrupar diferentes elementos em um único "bloco lógico" de informações, por meio de um elemento do tipo `<div>`, dentro do qual teremos dois elementos principais.

- Título do bloco: auto descritivo
- Lista de informações: que é uma lista não ordenada, contendo as principais informações a respeito do funcionamento da disciplina.

Cada item de informação possui um nome (p.ex.: Carga horária total) e um valor (p.ex.: 60h), sendo que alguns destes valores possuem links para outros sites. Teremos, também formatações diferentes para cada um destes campos.

A visualização esperada de tais elementos se encontra na Figura 6, a seguir.

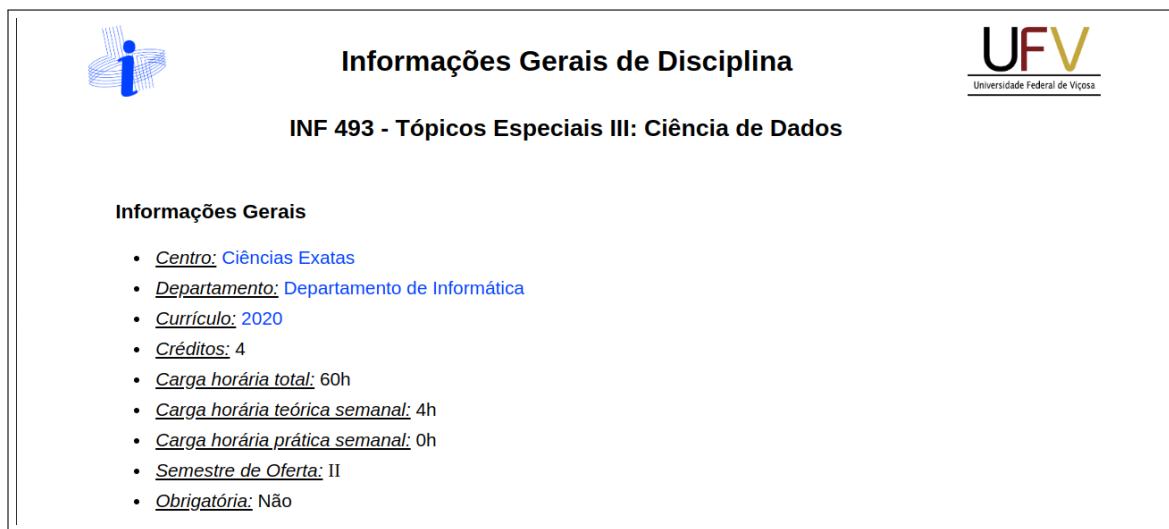


Figura 6: Visualização Esperada do Topo e Informações Gerais

Caso deseje poupar tempo de digitação (recomendo), use o arquivo `Conteúdo textual` que você baixou e apenas copie e cole os textos da lista.

4.2.1 Descrição do HTML

Para implementar o bloco de informações gerais, contendo título e lista, vamos criar um elemento HTML do tipo `<div>`, e vamos associá-lo a uma classe chamada `info-base`.

O conteúdo do bloco será uma agregação de outro elemento `<div>`, associado à classe `info-base-titulo`, contendo o texto *Informações Gerais* e uma lista não ordenada, que não necessita de associação a nenhuma classe CSS.

No entanto, dentro dos itens da lista, os elementos ``, teremos uma subdivisão, pois os nomes dos campos receberão uma formatação e os valores outra, ou outras, a depender do caso.

Encapsule o nome do campo com um elemento do tipo ``, associado à classe `campo` e não associe o valor a classe alguma. Porém, alguns dos valores também compõem links para sites institucionais. Nestes casos, apenas encapsule o valor do campo com a tag destinada à implementação de links (consulte online, se não se lembrar).

As URLs dos sites institucionais que devem ser utilizadas encontram-se dentro do arquivo de conteúdo textual. Por fim, acrescente o atributo HTML chamado `target` na tag do link, com o valor `"_blank"`. Este atributo controla o comportamento de onde o site endereçado pela URL irá abrir. No caso do valor `blank`, indica que o site será aberto em uma nova aba do navegador. Sempre é bom evitarmos de "tirar" o usuário completamente do nosso site.

Note que o atributo `target` não controla visual, mas comportamento. Portanto, não estamos indo contra um princípio básico desta atividade, que é separar estrutura de estilo visual.

Por fim, há um único caso especial. Espera-se o preenchimento com algarismos romanos no campo *Semestre de Oferta*. No entanto, a família de fontes utilizada em toda página, apesar de também poder atender ao propósito de representar numerais romanos, existem fontes que produzem um resultado esteticamente mais adequado, como fontes com serifa. Como se trata de um único caso, vamos marcar o valor II,

para o semestre de oferta, com uma tag `` e, ao invés de associá-la a uma classe, vamos associá-la ao id `romano`.

4.2.2 Descrição do CSS

Indo diretamente para as especificações de cada classe (e também o id) mencionada na Seção 4.2.1, temos:

- Os parâmetros para a classe `info-base` são:
 - Margens esquerda e direita: 8%
 - Margens inferior e superior: 5%
 - Alinhamento do texto: justificado
 - Tamanho de fonte: $1,3 \cdot$ tamanho corrente da fonte do elemento (consulte online, se não se lembrar da última aula)
- Os parâmetros para a classe `info-base-titulo` são:
 - Tamanho de fonte: $1,1 \cdot$ tamanho corrente da fonte do elemento
 - Fonte em negrito
- Os parâmetros para a classe `campo` são:
 - Texto sublinhado
 - Texto em itálico
- Por fim, o parâmetro para o id `romano` é:
 - Família de fonte: Times New Roman, Times, serif

Neste ponto, é importante destacar que **todos** elementos HTML do tipo "item de lista" devem possuir *padding* de 5 pixels (em todas direções) e que **todos** elementos do tipo "link" devem ter decoração de texto removida (use o valor `none`) e cor definida para os valores RGB (0, 65, 255), que corresponde ao tom de azul dos links presentes na Figura 6.

4.3 Blocos de Informação Textual

Na sequência, tem-se 4 blocos contendo informações mais longas a respeito da disciplina, em formato textual. Os quatro blocos seguem exatamente a mesma estrutura e vamos descrevê-la apenas uma vez aqui, portanto. Cada bloco é composto de duas partes, que receberão formatações distintas:

- Título do bloco: auto descritivo
- Texto do bloco auto descritivo.

A visualização esperada de tais elementos se encontra na Figura 7, a seguir.

Mais uma vez, caso deseje poupar tempo de digitação (recomendo), use o arquivo `Conteúdo textual` que você baixou e apenas copie e cole os textos da lista.

4.3.1 Descrição do HTML

Para implementar o bloco de informações textuais, contendo título e texto, vamos criar um elemento HTML do tipo `<div>`, e vamos associá-lo a uma classe chamada `info-textual`.

O conteúdo do bloco será uma agregação de outro elemento `<div>`, associado a **duas classes**: `info-textual-campo` e `info-textual-titulo`, e um parágrafo, também associado a duas classes: `info-textual-campo` e `info-textual-conteu`.

| <ul style="list-style-type: none"> • <u>Semestre de Oferta</u>: II • <u>Obrigatória</u>: Não | | |
|--|----------------|---|
| <table> <tr> <th>Objetivos</th></tr> <tr> <td>Prover ao estudante uma visão geral da área de Ciência de Dados (Data Science), com foco nas etapas de preparação de dados, análise de resultados e modelos e em aplicações reais ou realísticas. Relacionar a visão ampla do conceito de Ciência de Dados com as áreas correlatas de Aprendizado de Máquina e Mineração de Dados. Fornecer ao estudante um repertório de técnicas que o possibilite atuar em contextos envolvendo suporte à tomada de decisões, seja na área acadêmica ou não.</td></tr> </table> | Objetivos | Prover ao estudante uma visão geral da área de Ciência de Dados (Data Science), com foco nas etapas de preparação de dados, análise de resultados e modelos e em aplicações reais ou realísticas. Relacionar a visão ampla do conceito de Ciência de Dados com as áreas correlatas de Aprendizado de Máquina e Mineração de Dados. Fornecer ao estudante um repertório de técnicas que o possibilite atuar em contextos envolvendo suporte à tomada de decisões, seja na área acadêmica ou não. |
| Objetivos | | |
| Prover ao estudante uma visão geral da área de Ciência de Dados (Data Science), com foco nas etapas de preparação de dados, análise de resultados e modelos e em aplicações reais ou realísticas. Relacionar a visão ampla do conceito de Ciência de Dados com as áreas correlatas de Aprendizado de Máquina e Mineração de Dados. Fornecer ao estudante um repertório de técnicas que o possibilite atuar em contextos envolvendo suporte à tomada de decisões, seja na área acadêmica ou não. | | |
| <table> <tr> <th>Ementa</th></tr> <tr> <td>Introdução e terminologia básica. Fundamentos de análise de dados. Fundamentos de visualização de dados. Técnicas de pré-processamento e preparação de dados. Redução de dimensionalidade e seleção de atributos. Análise de Agrupamentos. Classificação. Regressão. Comitês. Explicabilidade de modelos preditivos. Outras técnicas relacionadas. Áreas de aplicação modernas.</td></tr> </table> | Ementa | Introdução e terminologia básica. Fundamentos de análise de dados. Fundamentos de visualização de dados. Técnicas de pré-processamento e preparação de dados. Redução de dimensionalidade e seleção de atributos. Análise de Agrupamentos. Classificação. Regressão. Comitês. Explicabilidade de modelos preditivos. Outras técnicas relacionadas. Áreas de aplicação modernas. |
| Ementa | | |
| Introdução e terminologia básica. Fundamentos de análise de dados. Fundamentos de visualização de dados. Técnicas de pré-processamento e preparação de dados. Redução de dimensionalidade e seleção de atributos. Análise de Agrupamentos. Classificação. Regressão. Comitês. Explicabilidade de modelos preditivos. Outras técnicas relacionadas. Áreas de aplicação modernas. | | |
| <table> <tr> <th>Pré-requisitos</th></tr> <tr> <td>INF 213 - Estruturas de Dados EST 105 - Iniciação à Estatística</td></tr> </table> | Pré-requisitos | INF 213 - Estruturas de Dados EST 105 - Iniciação à Estatística |
| Pré-requisitos | | |
| INF 213 - Estruturas de Dados EST 105 - Iniciação à Estatística | | |
| <table> <tr> <th>Co-requisitos</th></tr> <tr> <td>Não há.</td></tr> </table> | Co-requisitos | Não há. |
| Co-requisitos | | |
| Não há. | | |

Figura 7: Visualização Esperada dos Blocos de Texto

4.3.2 Descrição do CSS

Indo diretamente para as especificações de cada classe mencionada na Seção 4.3.1, temos:

- Os parâmetros para a classe `info-textual` são:
 - Margens inferior e superior: 20 pixels
 - Alinhamento do texto: centralizado
 - Tamanho de fonte: $1,1 \cdot$ tamanho corrente da fonte do elemento
- Os parâmetros para a classe `info-textual-campo` são:
 - Borda de 1 pixel de espessura, sólida e na cor preta
 - *Padding* de 10 pixels
 - Margens superior e inferior: 0 pixels
 - Margens direita e esquerda: 8%
 - Largura: 84%
- Os parâmetros para a classe `info-textual-titulo` são:
 - Texto centralizado
 - Texto em negrito
 - Cor de fundo: *darkgray*
- Por fim, o parâmetro para a classe `info-textual-conteudo` é:
 - Alinhamento do texto: justificado

4.4 Conteúdo Programático

Após os blocos de informação textual, vem a descrição do conteúdo programático da disciplina. Neste item, vamos mesclar classes CSS definidas anteriormente com novas classes, já que a estrutura básica deste trecho é similar à estrutura da parte relativa às informações gerais. No entanto, desta vez teremos o **aninhamento** de listas HTML.

Um pequeno exemplo de aninhamento de listas HTML encontra-se na Figura 8, cuja visualização encontra-se na Figura 9.

Note que, no exemplo, o uso *inline* do estilo dentro da tag `` foi só para exemplificar como os marcadores ou numerações de uma lista podem ser alterado para maior personalização.

```

<ul>
  <li>Refrigerantes</li>
    <ol>
      <li>Coca-cola</li>
      <li>Dolly</li>
      <li>Guaraná</li>
    </ol>
  <li>Cervejas</li>
    <ol style="list-style-type: lower-greek;">
      <li>Brahma</li>
      <li>Einsenbahn</li>
      <li>Kaiser</li>
    </ol>
</ul>

```

Figura 8: Exemplo de Listas HTML Aninhadas

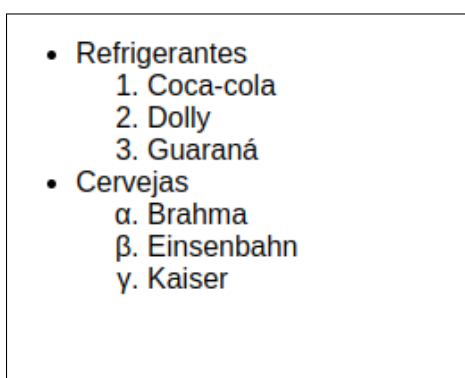


Figura 9: Visualização das Listas HTML Aninhadas

Retomando a descrição da estrutura HTML do que é esperado desta parte da atividade, o conteúdo programático da disciplina deve ser organizado dentro de uma `<div>`, da classe `info-base`, que já foi especificada na Seção 4.2.2.

Dentro desta `div`, teremos dois elementos:

- Título do bloco: auto descritivo. Associe-o à classe `info-base-titulo`, que já foi especificada na Seção 4.2.2.
- Lista de capítulos: uma lista HTML ordenada, que deve ser associada à classe `capitulo`.

A lista de capítulos, por sua vez, está dividida em itens e sub-listas. Seus itens deverão corresponder aos títulos dos capítulos do conteúdo programático e serem associados à classe `capitulo-li`.

Cada item de capítulo poderá ser seguido por uma sub-lista (dependendo do caso), também do tipo lista HTML ordenada, que deve estar associada à classe `topico`. Os itens desta sub-lista não precisam estar associados a nenhuma classe. Em um único caso, uma sub-lista terá uma "sub-sub-lista", que será uma lista HTML não ordenada comum, sem nenhuma associação a classes CSS.

Analise o documento *Conteúdo textual* para saber como organizar as listas aninhadas desta parte. Especialmente neste caso, recomendo que use o arquivo para copiar e colar os textos, poupando um grande tempo de digitação.

Desta vez, devido ao reaproveitamento de classes anteriormente desenvolvidas e à explicação de como funcionam listas aninhadas, não vamos especificar em detalhes a descrição do código HTML deste trecho.

Vamos, apenas, trazer a visualização, dividida em 3 partes, conforme ilustrado pelas figuras 10, 11 e 12.

| |
|---|
| Conteúdo Programático |
| 1. Introdução |
| a. O que é Ciência de Dados (<i>Data Science</i>) |
| b. Conceitos importantes: <i>Data Mining versus Machine Learning versus Data Science versus Inteligência Artificial</i> |
| c. Visão geral do curso e suas lições |
| 2. Fundamentos de análise de dados |
| a. Representação de Dados |
| b. Tipos de atributos |
| c. Fundamentos de Estatística |
| d. Visualização de dados |
| e. Análise Uni e Multivariada |
| f. Introdução aos testes de hipótese |
| 3. Pré-processamento e preparação de dados |
| a. Normalização (escalonamento e padronização) |
| b. Identificação e remoção de dados discrepantes |
| c. Tratamento de valores faltantes |
| d. Categorização de dados numéricos |
| e. Pré-processamento para dados categóricos |

Figura 10: Visualização Esperada do Conteúdo Programático: Parte 1

| |
|---|
| 4. Principais tarefas de mineração de dados e aprendizado de máquina |
| a. Agrupamentos (<i>clusters</i>) |
| b. Classificação |
| c. Regressão |
| d. Metodologias de treinamentos |
| e. Principais algoritmos |
| f. Métricas de avaliação |
| g. Comparação de modelos |
| h. Validação Cruzada |
| i. ANOVA |
| j. Casos especiais |
| ▪ <i>Overfitting e Underfitting</i> |
| ▪ Classes raras |
| ▪ Multirrótulo |
| ▪ Outros |
| 5. Seleção de atributos e redução de dimensionalidade |
| 6. Comitês |
| 7. Explicabilidade de modelos |
| 8. Outros tipos de aprendizado |

Figura 11: Visualização Esperada do Conteúdo Programático: Parte 2

| |
|---|
| 9. Algumas áreas de aplicação modernas |
| a. Processamento de linguagem natural |
| b. Visão computacional |
| c. Modelos generativos |
| d. <i>Federated Learning</i> |

Figura 12: Visualização Esperada do Conteúdo Programático: Parte 3

4.4.1 Descrição do CSS

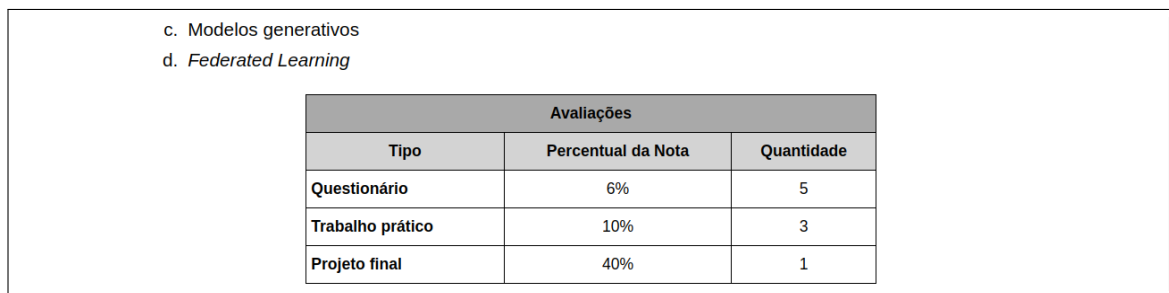
Indo diretamente para as especificações de cada classe nova mencionada na Seção 4.4, temos:

- O parâmetro para a classe `capitulo` é:
 - Tipo de marcador: numeração decimal comum
- O parâmetro para a classe `capitulo-li` é:
 - Texto em negrito
- O parâmetro para a classe `topico` é:
 - Tipo de marcador: ordem alfabética em letras minúsculas

4.5 Tabela de Avaliações

Para encerrar a página, há uma tabela contendo um sumário de informações sobre como seria a avaliação da disciplina. Desta vez, temos uma tabela HTML padrão, sem nenhum tipo de agregação de elementos distintos e sem uso de `<div>` e ``.

A visualização esperada da tabela, que é auto descritiva, se encontra na Figura 13, a seguir. Há também uma cópia das informações da mesma no arquivo `Conteúdo textual`.



| c. Modelos generativos | | |
|------------------------|--------------------|------------|
| d. Federated Learning | | |
| Avaliações | | |
| Tipo | Percentual da Nota | Quantidade |
| Questionário | 6% | 5 |
| Trabalho prático | 10% | 3 |
| Projeto final | 40% | 1 |

Figura 13: Visualização Esperada da Tabela de Avaliações

4.5.1 Descrição do HTML

A tabela HTML deve ser associada à classe `avaliacoes` e conterá uma seção de cabeçalho (`<thead>`) e uma seção de corpo (`<tbody>`). Estas seções não precisam estar associadas a nenhuma classe.

Ambas seções são subdivididas em linhas de tabela, que também não necessitam estar associadas a nenhuma classe. No entanto, as células da tabela (tanto do tipo `<th>`, quanto do tipo `<td>`) estarão **todas elas** associadas a uma classe chamada `celulas`. E, dependendo da função da célula, cada uma estará associada a uma segunda classe distinta.

A célula de título (com fundo cinza mais escuro) deverá estar associada também à classe `titulotabela` e deverá ter a extensão de 3 colunas. Este último aspecto se confunde entre estrutural e visual. Porém, a expansão de uma célula em várias linhas ou colunas diz mais respeito a estrutura do que ao visual, sendo este último uma consequência do aspecto estrutural. Portanto, especifique esta característica diretamente no código HTML.

As células de cabeçalho de cada coluna (com fundo cinza mais claro) deverão estar todas associadas também à classe `subcabeçalho`. Já as células de dados da coluna Tipo, com conteúdo textual, deverão receber a classe adicional `dadotxt`, enquanto as células de dados das demais colunas deverão adicionar a classe `dadonum` à sua lista de classes.

4.5.2 Descrição do CSS

Indo diretamente para as especificações de cada classe mencionada na Seção 4.5.1, temos:

- Os parâmetros para a classe `avaliacoes` são:
 - Bordas colapsadas e com espaçamento zero

- Margens inferior e superior: 0 pixels
- Margens esquerda e direita: 25%
- Largura: 50%
- Tamanho da fonte: 1.1em
- Alinhamento vertical: meio
- Os parâmetros para a classe `celulas` são:
 - Borda de 1 pixel de espessura, sólida e na cor preta
 - *Padding* de 10 e 5 pixels, respectivamente (consulte online a respeito da ordem)
 - Ajuste de texto (*word break*): normal
 - Ajuste de texto (*overflow*): oculto
- O parâmetro para a classe `titulotabela` é:
 - Cor de fundo: *darkgray*
- Os parâmetros para a classe `subcabecalho` são:
 - Cor de fundo: *lightgray*
 - Texto centralizado
 - Fonte em negrito
- Os parâmetros para a classe `dadotxt` são:
 - Texto alinhado à esquerda
 - Fonte em negrito
- O parâmetro para a classe `dadonum` é:
 - Texto centralizado

5 Entrega

1. Compacte **em formato .zip** (só será aceito nesse formato), **apenas** os arquivos e pastas representados na Figura 1.
 2. Não coloque arquivos adicionais, apenas os necessários para a renderização da página.
 3. Respeite a estrutura hierárquica das pastas e caminhos, conforme ilustrado na referida figura.
 4. Envie, via Moodle, o arquivo compactado contendo sua solução, como resposta à atividade.
- **Obs.:** não serão aceitos envios por e-mail.

Bons estudos.
Prof. Marcos