

Aula 20:

DIVs/CSS e Imagens

Karina S. Machado

Curso de Sistemas de Informação

Sistemas para Internet I

Data da prova

- 13/outubro
- Sala de Aula do Prédio 03
- Peso da nota da Prova no bimestre ➔ 6,0 (60%)

Trabalho do Bimestre

- Data de apresentação: 20/outubro
- Durante a aula
- Trabalhos em Duplas
- Peso na nota do bimestre: 4,0 (40%)

Trabalho do Bimestre

- Todos os trabalhos devem ser enviados no formato de um ZIP até o dia 15 de outubro no Moodle.
- Então eu organizarei todo o conteúdo para apresentação dia 20/10 (não tem tempo de haver troca de computador, notebook, etc)
- Assim todos os trabalhos já estarão no meu computador para apresentação
- **SEM POSSIBILIDADE DE ADIAMENTOS**
- **NÃO ENTREGA NA DATA (15/10) SIGNIFICA QUE NÃO PODE SER APRESENTADO DIA 20/10**

Trabalho do Bimestre

- Consiste na elaboração de uma página de **um Tutorial** sobre assuntos determinados em aula
- Quem ainda não escolheu assunto, escolher no WIKI no Moodle
- Não pode escolher tópicos já selecionados pelos colegas
- Veja as instruções nos slides da aula passada.

Aula Hoje

- Agrupamento de objetos – DIV
- Imagens
- Exercícios
- **ANOTE AS TAGS E COMANDOS
PARA REALIZAR OS EXERCÍCIOS E
FIXAR O CONTEÚDO**

Div - principais atributos

- O elemento <div> permite o agrupamento de objetos e a configuração desse grupo
- Class: atributo para especificar a classe de estilo a ser aplicada ao elemento <div>
- Id: especifica um identificador para o elemento DIV
- Align: direção do texto do bloco <div>
- Title: informação adicional ao DIV

DIV propriedades

- **Height**: permite configurar a altura de um elemento. Pode ser expresso em cm, px, porcentagem (que fica proporcional ao espaço disponível)
- **Width**: permite configurar a largura de um elemento.
 - Auto: o navegador calcula a largura atual
 - %: define a largura através de um percentual referente ao bloco no qual o elemento está contigo.
- **Position**: permite posicionar um elemento.

Position Static

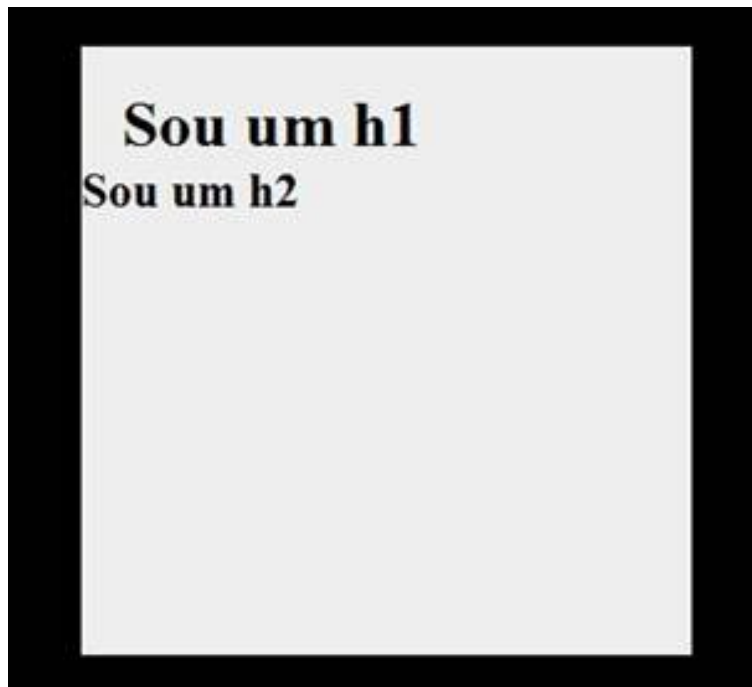
- É o padrão (se não for alterado é o que é atribuído)
- Segue o fluxo comum da página Web
- Não aplica as propriedades de bottom, top, left e right

Position Relative

- o elemento passa a aceitar as propriedades Top, Bottom, Left e Right. Com elas você pode alterar o posicionamento do elemento.
- Sempre a posição é em relação ao canto externo do próprio objeto

Exemplo

- `h1 { position: relative; top: 20px; left: 20px; }`
- `h2 { top: 20px; left: 20px; }`



Position Absolute

- O Position Absolute é um tanto diferente do Relative. Enquanto o elemento com Position Relative utiliza seu próprio canto para referenciar sua posição, o elemento com Position Absolute se utiliza do ponto superior esquerdo de outros elementos. Estes elementos são os parentes dele do elemento com position absolute. Mais especificamente o pai.

```
<div>


  <h1>

    <span></span>

  </h1>

  <p></p>
</div>
```

O elemento DIV é pai dos elementos H1, P e SPAN. Os elementos H1, P e SPAN são filhos do elemento DIV. Os elementos H1 e P são irmãos por estarem no mesmo nível. E o elemento SPAN é filho do elemento H1 e também do elemento DIV.



O elemento com position absolute se referencia pelo ponto superior esquerdo do elemento pai mais próximo que contenha também uma propriedade position relative ou absolute. Neste caso, a posição deste elemento se referenciaria pelo div com borda azul.

Exemplo

```
* { margin: 0; padding: 0;} /*Reset*/
body { background: #000; }
div { background: #eee; margin: 50px; width: 300px;
      height: 300px; position: relative; }
h1 { position: absolute; top: -20px;
     left: -20px; outline: 1px solid #444; background: #fff;
     }
```

```
<body>

  <div>
    <h1>Sou um h1</h1>
  </div>

</body>
```

Exemplo

Sou um h1



Position Fixed

- O position: fixed; irá fixar a posição do elemento na coordenada que você definir. A medida que a página é rolada, o elemento continua fixo na posição que você definiu e o conteúdo da página rola normalmente.
- Geralmente é usado para fixar elementos como cabeçalhos ou sidebars.
- O position fixed se comporta de maneira semelhante ao absolute, deixando de fazer parte do fluxo comum da página.
- Porém a grande diferença dele é que ele passa a se referenciar ao window do seu navegador, ou seja, a área que aparece para o usuário independente de barra de rolagem

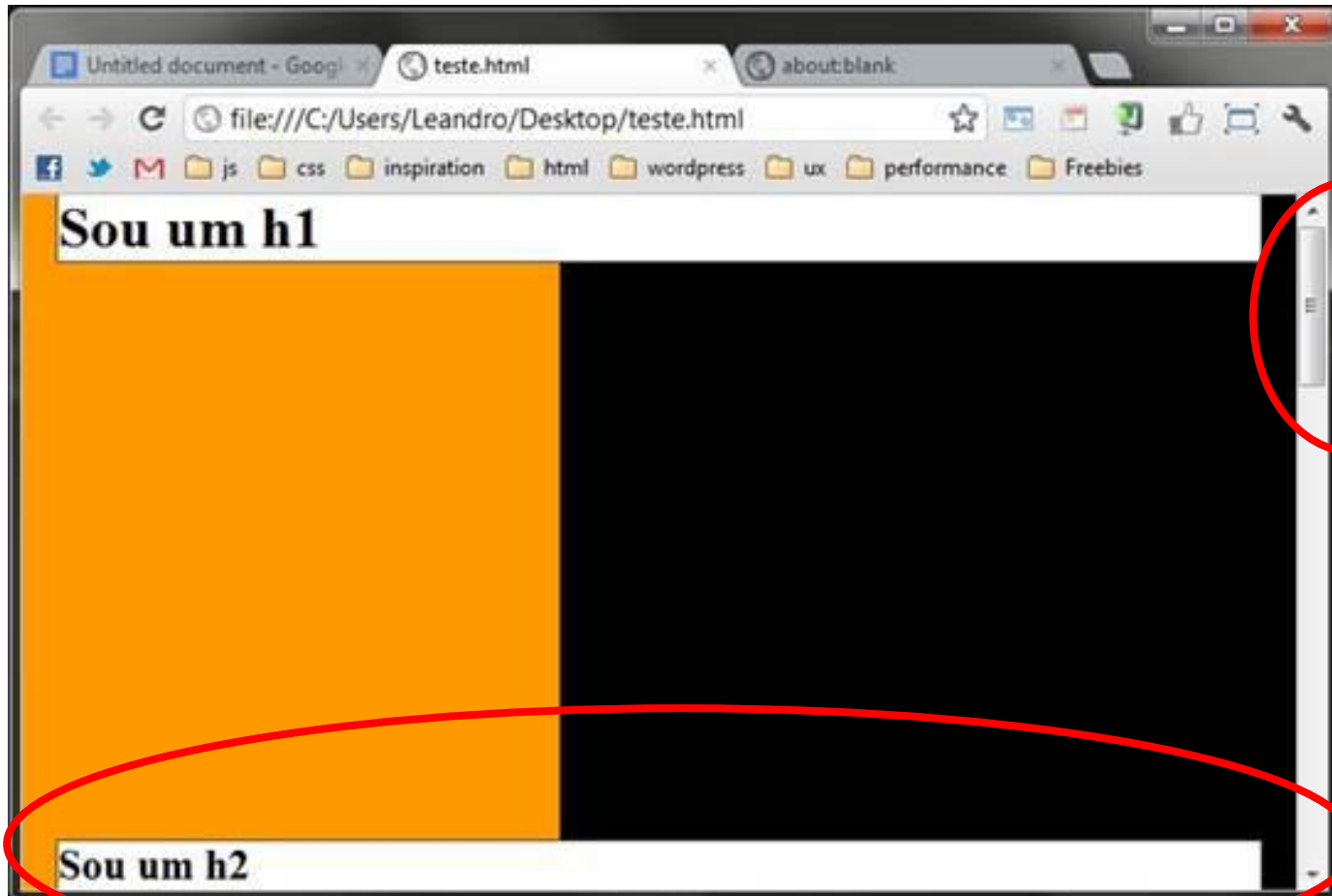
Exemplo

```
* { margin: 0; padding: 0;} /*Reset*/
body { background: #000; }
div { background: #f90; width: 300px; height: 1500px; position: relative; }
h1 { position: fixed; top: 0; left: 20px; right: 20px;
    outline: 1px solid #444; background: #fff; }
h2 { position: fixed; bottom: 0; left: 20px; right: 20px;
    outline: 1px solid #444; background: #fff;}
```

```
<body>
  <div>
    <h1>Sou um h1</h1>
    <h2>Sou um h2</h2>
  </div>
</body>
```

Exemplo

Se fosse o absolute, o H2 não apareceria, estaria relacionado com seu pai, o DIV e apareceria ao final, só rolando a barra de rolagem



Floating

- Com o CSS Float um elemento pode ser reposicionado a esquerda ou direita
- É muito utilizado para imagens, mas também é útil para layouts
- Elementos podem flutuar horizontalmente (left e right)
- O elemento então vai se mover para a esquerda ou direita o Maximo que ele conseguir -> os objetos após o elemento floating vão se mover ao redor dele, os de antes não são afetados
- Ex: se uma imagem é floated para a direita, o texto após ela será movido para a esquerda.

Floating - exemplo

```
<html>
<head>
<style>
img
{
float:right;
}
</style>
</head>

<body>
<p> No parágrafo abaixo a imagem foi adicionada com float:right o resultado é
que todo o texto após a imagem é deslocado a direito para preencher.</p>
<p>

Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto.
Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto.
Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto.
Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto. Texto.
Texto. Texto. Texto. Texto. Texto. Texto. Texto.
</p>
</body>

</html>
```


Float - Clear

- Depois de um elemento floating, os demais elementos vão se organizar em volta dele.
- Para evitar isso, pode-se utilizar a propriedade “clear”, atribuindo essa classe ao elemento onde quer se evitar a “flutuação”
- Exemplo:
- `.text_line`
`{`
`clear:both;`
`}`

Exercicio 01 galeria de fotos

- Utilizando Classe, float, Clear faça a página abaixo
 - Crie uma classe chamada “galeria” que será atribuída aos elementos img, especifique um tamanho padrão na classe

Galeria de fotos

Criando uma galeria de fotos utilizando o elemento float e Clear.

Galeria de fotos do Cassino



Galeria de fotos de Rio Grande



Exercicio 02 galeria de fotos

- Experimento variações:
 - Com e sem o uso do Float
 - No uso do Clear (experimente tirar o clear)
 - Modifique as configurações da classe
 - Tire a classe
- Teste e veja os resultados obtidos!!!!

Overflow

- A propriedade overflow especifica o que deve acontecer se o conteúdo de um elemento/grupo excede o tamanho de uma caixa.

Valor	Descrição
visible	Mostra o conteúdo todo, mesmo sobrepondo, é o default.
Hidden	O que não estiver dentro do limite da caixa não é exibido
Scroll	É incluído scroll na caixa para mostrar o conteúdo, caso o mesmo exceda o tamanho
Auto	Se o conteúdo excede o tamanho da caixa, automaticamente é incluído um scroll

Overflow

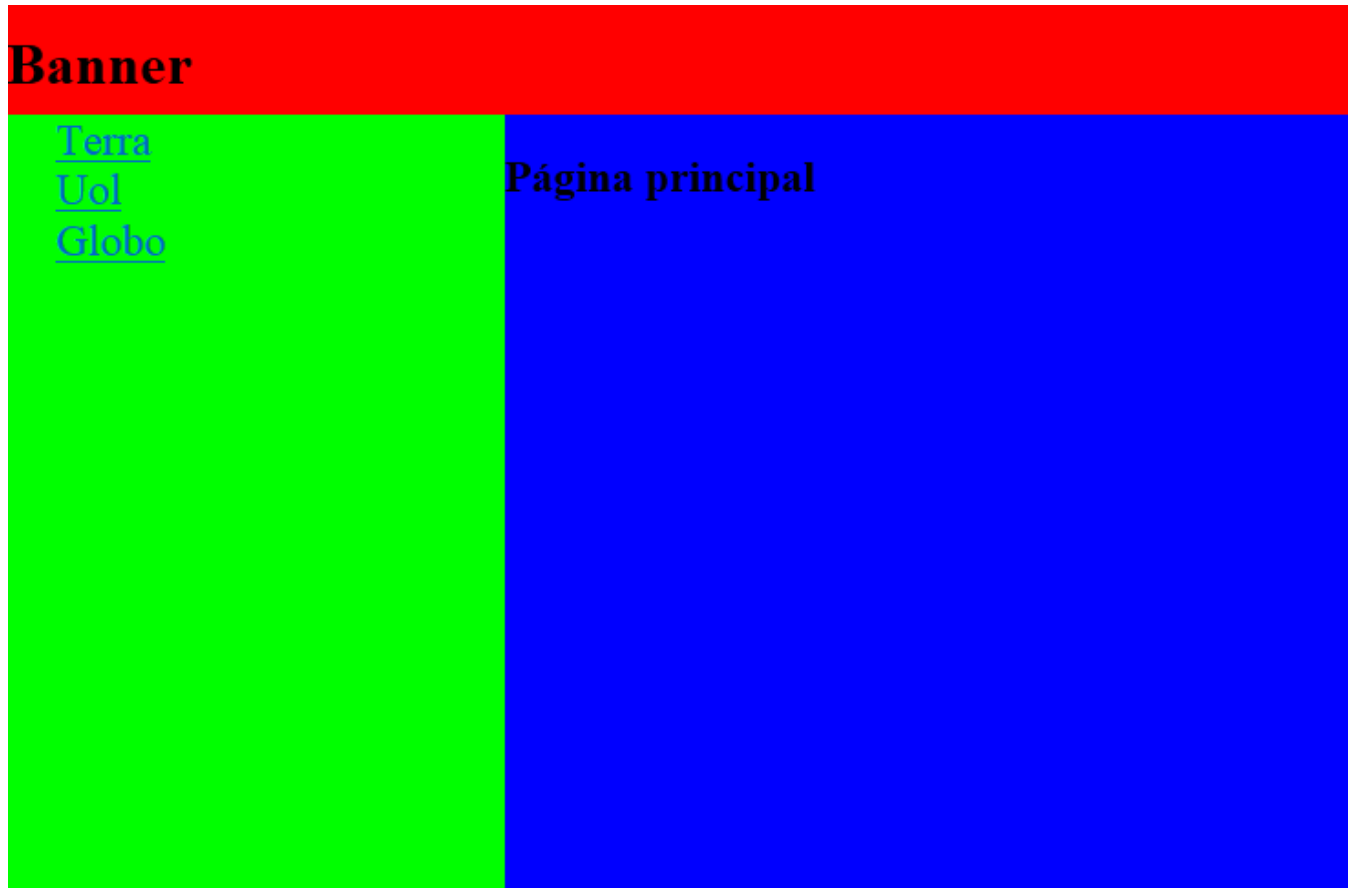
- Verifique um exemplo de Overflow no endereço:
- http://www.w3schools.com/cssref/playit.asp?filename=playcss_overflow

Sintaxe: CSS EXTERNO

- Até o momento a maioria dos exemplos e exercícios criamos o CSS dentro do próprio arquivo HTML.
- Para criar o CSS em um arquivo separado, utilize o comando abaixo:
- ```
<head>
<link rel="stylesheet" type="text/css"
href="meuarquivo.css">
</head>
```

# Exercício

- Faça o CSS em um arquivo em separado
- Faça a seguinte página:



# Exercicio - Dicas

- Criar 3 classes ou IDS para os DIVs (ou Ids, como serão chamados uma única vez dentro deste html)
- BODY:
  - margin: 0; padding: 0;
- ID 1 – banner
  - Altura 10% largura 100% cor de fundo vermelha
- ID2 – menu esquerda
  - Altura 90% largura 20% top 10% cor de fundo verde
- ID3 – centro
  - Altura 90% largura 80% top 10% left 20% cor de fundo azul
- Classe dos LINKs
  - left: 10%;
  - font-size: 24px;

# Exercício 02

Esquerda em  
Cima

Direita em  
Cima

Esquerda em  
Baixo

Direita em  
Baixo

## Exercício 03 - formatação de formulários

Nome:

Mensagem:

Digite suas dúvidas

**Enviar**

# Exercício 04 - formatação de formulários

- para formulários (elemento form): cor de fundo #FFFFCC; largura 380px; tamanho da font: 20px; Tipo de fonte: Verdana, sans-serif; cor da fonte: #003399; Borda de largura 2px, sólida e de cor azul; espaçamento a esquerda de 10px e espaçamento bottom de 20px.
- Para campos do tipo text (dica: crie uma classe com as configurações, sem relação com elemento e atribua esta classe aos elementos text): cor de fundo :#BoEoE6; fonte de tamanho 20px, do tipo georgia, sans-serif e cor #0033CC; margem a direita de 20px;
- Para botões: cor de fundo #CC3300; fonte de 20px do tipo Arial, sans-serif e cor da fonte #CCFFFF;
- Para campos do tipo textarea: cor de fundo: #BoEoE6; fonte de tamanho 16px do tipo: verdana, arial, helvetica, sans-serif e cor #003399; largura do textarea (propriedade width) = 320px;