



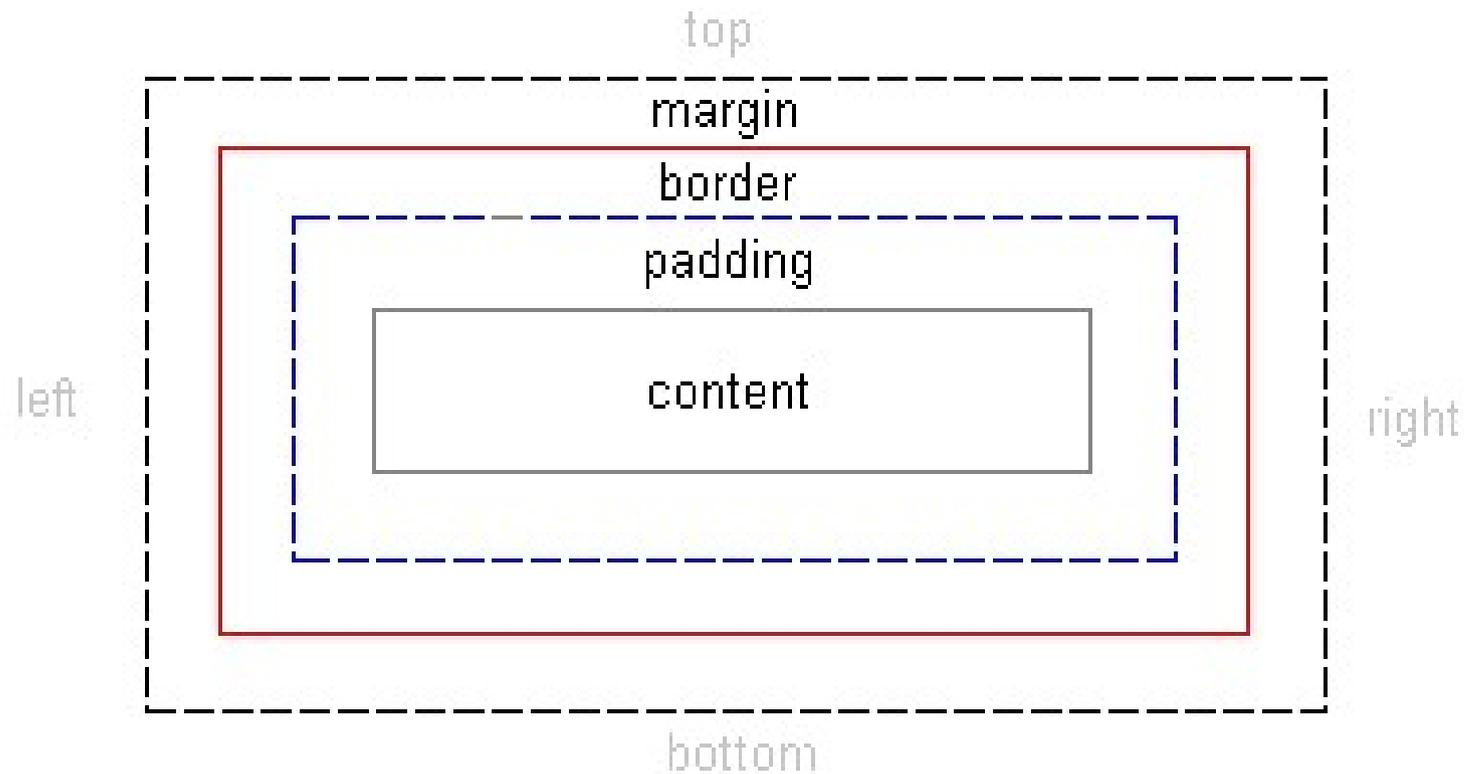
# INTRODUÇÃO AO DESENVOLVIMENTO WEB

PROF<sup>a</sup>. M.Sc. JULIANA H Q BENACCHIO

- O corpo (**body**) da página é a área total visível, mas é possível compartimentar a página em pequenas divisões com seus estilos próprios
- Tudo o que temos a fazer é englobar entre o par de tags `<div>...</div>` a área que pretendemos transformar em uma divisão
- Tais áreas, são como uma mini página que em CSS são chamadas de **boxes CSS**
- Boxes CSS se constituem na unidade básica de construção de layout de uma página Web

- O box model (modelo das caixas) em CSS, descreve os boxes (as caixas) geradas pelos elementos HTML
- O box model, detalha ainda, as opções de ajuste de margens, bordas, padding e conteúdo para cada elemento
- A seguir um diagrama representando a estrutura de construção do box model:

# Boxes CSS



- margin edge
- border edge
- - - padding edge
- content edge

- Boxes CSS podem ter uma largura e uma altura (width e height), uma cor de fundo ou até mesmo uma imagem de fundo que pode se repetir constituindo um padrão de preenchimento
- Um box CSS por padrão estende-se da margem esquerda à margem direita do corpo da página
- Se você não especificar uma margem para o corpo da página (**body**) um box CSS ocupará toda a largura da janela do navegador

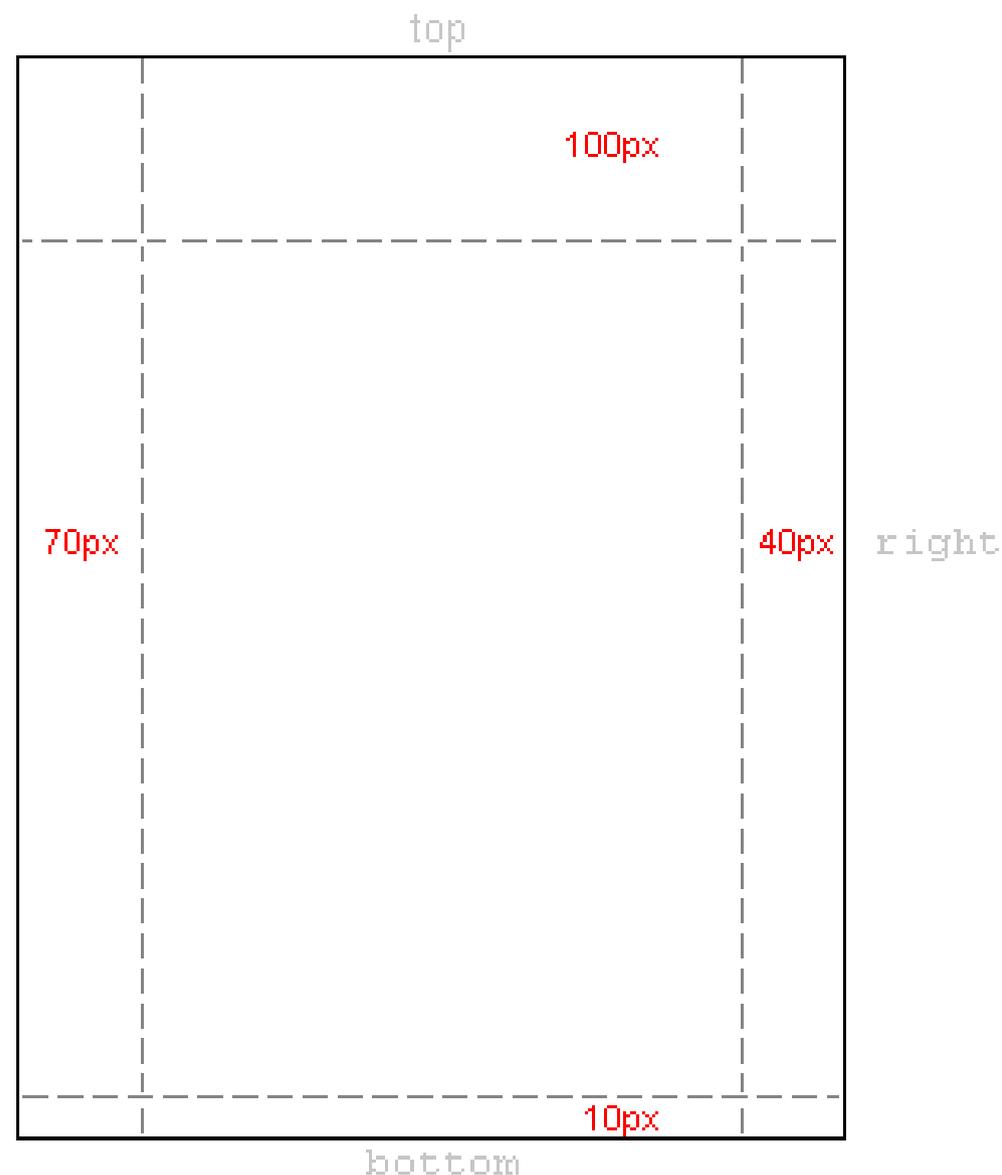
- Se você não especificar uma altura para um box CSS ele não terá altura (será considerada zero)
- Ao colocar texto dentro do box ele se expandirá em altura de modo a acomodar o texto colocado - ou uma imagem

- **Definindo margin de um elemento**
- Um elemento tem quatro lados: **right**, **left**, **top** e **bottom** (direito, esquerdo, superior e inferior)
- A **margin** é a distância entre os lados de elementos vizinhos (ou às bordas do documento)
- Vamos começar com um exemplo mostrando como definir **margins** para o documento, ou seja, para o elemento **<body>**. A ilustração a seguir mostra como serão as margens da página.

# Margin e padding

- No CSS:

```
body {  
  margin-top: 100px;  
  margin-right: 40px;  
  margin-bottom: 10px;  
  margin-left: 70px;  
}
```



# Margin e padding

- A propriedade **margin** é uma abreviação para todas as propriedades listadas anteriormente
- Desse modo é possível declarar várias propriedades de modo abreviado, economizando digitação, além de tornar a folha de estilo mais fácil de se ler e entender

```
body {  
    margin: 100px 40px 10px 70px;  
}
```

# Margin e padding

- As margens para a maioria dos elementos pode ser definida conforme o exemplo anterior
- Podemos então, por exemplo, definir margens para todos os parágrafos `<p>`:

```
p {  
    margin: 5px 50px 5px 50px;  
}
```

- **Definindo padding de um elemento**
- **padding** pode também ser entendido como "enchimento"
- Isto faz sentido, porque **padding** não é computado na distância entre elementos, **padding** define simplesmente a distância entre a borda e o conteúdo do elemento
- Ilustramos o uso de **padding** através de um exemplo onde todos os cabeçalhos têm uma cor de fundo definida

# Margin e padding

## Cães mais influentes da história da humanidade

### By Dicas do Cão

Desde o fim da Era do Gelo os cães acompanham a espécie humana. De lá para cá a cultura deu um salto gigante, e a evolução da civilização cada vez se tornou mais frequente. Diante desse fenômeno, é normal que os cães, que tanto nos acompanham, tenham feito alguma coisa pela aventura humana.

Conheça aqui a nossa lista de cães que mudaram a história da civilização, baseado no livro de Sam Stall. Os famosos cães aparecem desde a antiguidade, até o período (pós) moderno. O livro lista os 100 cães que marcaram a trajetória da humanidade. Conheça alguns que selecionamos aqui:

### Péritas

Era o cão favorito de Alexandre, o Grande, que o acompanha durante as explorações militares. Provavelmente o nome Péritas tenha se originado da palavra 'Janeiro' em Macedônio. Há indícios de que o cão possa ter pertencido aos molossos (avós dos atuais mastifes). O imperador gostava tanto desse cão que deu seu nome à uma cidade grega.

O fato marcante na vida desse cão, foi o fato dele ter salvo Alexandre de ter sido esmagado por um elefante. Isso aconteceu na batalha de Guagamela, e Péritas tinha mordido o elefante a tempo de salvar seu dono. Se isso não tivesse acontecido, seu dono não teria tempo de ter entrado na história como Alexandre, o Grande.

### Urian

Todos sabem da divisão da Igreja na Inglaterra. O rei Henrique VIII criou a Igreja Anglicana para poder se casar duas vezes. O que muitos não sabem é que tem um cãozinho envolvido na história. Trata-se de Urian, um cachorro que veio cometer um deslize, provocando a eterna separação da Igreja Católica com a Anglicana.

Conta-se que o Cardeal Wolsey, que era emissário de Henrique VIII, foi a um encontro do Papa Clemente VII, acompanhado do cão. O Papa e o Cardeal estavam quase chegando a um acordo, quando o cão Urian mordeu a canela do Papa. O resultado disso foi um desacordo, que dura até hoje.

```
h1 {  
    background : #FFFF00;  
}
```

```
h2 {  
    background : #FF9900;  
}
```

# Margin e padding

- Definindo **padding** para os cabeçalhos, alteramos a quantidade de enchimento existente ao redor de cada um deles:

```
h1 {  
    background: #FFFF00;  
    padding: 20px 20px 20px 80px;  
}  
h2 {  
    background: #FF9900;  
    padding-left: 120px;  
}
```

# Margin e padding

## Cães mais influentes da história da humanidade

### By Dicas do Cão

Desde o fim da Era do Gelo os cães acompanham a espécie humana. De lá para cá a cultura deu um salto gigante, e a evolução da civilização cada vez se tornou mais frequente. Diante desse fenômeno, é normal que os cães, que tanto nos acompanham, tenham feito alguma coisa pela aventura humana.

Conheça aqui a nossa lista de cães que mudaram a história da civilização, baseado no livro de Sam Stall. Os famosos cães aparecem desde a antiguidade, até o período (pós) moderno. O livro lista os 100 cães que marcaram a trajetória da humanidade. Conheça alguns que selecionamos aqui:

### Péritas

Era o cão favorito de Alexandre, o Grande, que o acompanha durante as explorações militares. Provavelmente o nome Péritas tenha se originado da palavra 'Janeiro' em Macedônio. Há indícios de que o cão possa ter pertencido aos molossos (avós dos atuais mastifes). O imperador gostava tanto desse cão que deu seu nome à uma cidade grega.

O fato marcante na vida desse cão, foi o fato dele ter salvo Alexandre de ter sido esmagado por um elefante. Isso aconteceu na batalha de Guagamela, e Péritas tinha mordido o elefante a tempo de salvar seu dono. Se isso não tivesse acontecido, seu dono não teria tempo de ter entrado na história como Alexandre, o Grande.

### Urian

Todos sabem da divisão da Igreja na Inglaterra. O rei Henrique VIII criou a Igreja Anglicana para poder se casar duas vezes. O que muitos não sabem é que tem um cãozinho envolvido na história. Trata-se de Urian, um cachorro que veio cometer um deslize, provocando a eterna separação da Igreja Católica com a Anglicana.

Conta-se que o Cardeal Wolsey, que era emissário de Henrique VIII, foi a um encontro do Papa Clemente VII, acompanhado do cão. O Papa e o Cardeal estavam quase chegando a um acordo, quando o cão Urian mordeu a canela do Papa. O resultado disso foi um desacordo, que dura até hoje.

### Laika

A história de Laika é digna de uma epopeia: ela era uma vira-lata que vivia pelas ruas de Moscou dos anos 50, até que cientistas do governo soviético a recolheram e treinaram-a para ser o primeiro cão-astronauta em

- **Atribuindo largura [width]**
- A propriedade **width** destina-se a definir a largura de um elemento.
- O exemplo a seguir constrói um box dentro do qual podemos digitar um texto

# Altura e largura

## `<div>` com 200px de largura e texto

Péritas

Era o cão favorito de Alexandre, o Grande, que o acompanha durante as explorações militares. Provavelmente o nome Péritas tenha se originado da palavra 'Janeiro' em Macedônio. Há indícios de que o cão possa ter pertencido aos molossos (avôs dos atuais mastifes). O imperador gostava tanto desse cão que deu seu nome à uma cidade grega.

- No HTML:

`<h1>&lt;div&gt; com 200px de largura e texto</h1>`

`<div class="box">Péritas<br>Era o cão favorito de Alexandre, o Grande, que o acompanha durante as explorações militares. Provavelmente o nome Péritas tenha se originado da palavra 'Janeiro' em Macedônio. Há indícios de que o cão possa ter pertencido aos molossos (avôs dos atuais mastifes). O imperador gostava tanto desse cão que deu seu o nome à uma cidade grega. </div>`

# Altura e largura

- No CSS:

```
div.box {  
    width: 200px;  
    border: 1px solid black;  
    background: #FF9900;  
}
```

- **Atribuindo altura [height]**
- No exemplo anterior a altura será determinada pelo conteúdo inserido no box
- Você pode definir a altura de um elemento com a propriedade **height**
- Como exemplo, vamos fazer a altura do box anterior igual a 500px

# Altura e largura

**<div> com 200px de largura 500px de altura e texto**

Péritas  
Era o cão favorito de Alexandre, o Grande, que o acompanha durante as explorações militares. Provavelmente o nome Péritas tenha se originado da palavra 'Janeiro' em Macedônio. Há indícios de que o cão possa ter pertencido aos molossos (avôs dos atuais mastifes). O imperador gostava tanto desse cão que deu seu nome à uma cidade grega.

- No HTML:

`<h1>&lt;div&gt; com 200px de largura 500px de altura e texto</h1>`

`<div class="box">Péritas<br>Era o cão favorito de Alexandre, o Grande, que o acompanha durante as explorações militares. Provavelmente o nome Péritas tenha se originado da palavra 'Janeiro' em Macedônio. Há indícios de que o cão possa ter pertencido aos molossos (avôs dos atuais mastifes). O imperador gostava tanto desse cão que deu seu o nome à uma cidade grega. </div>`

# Altura e largura

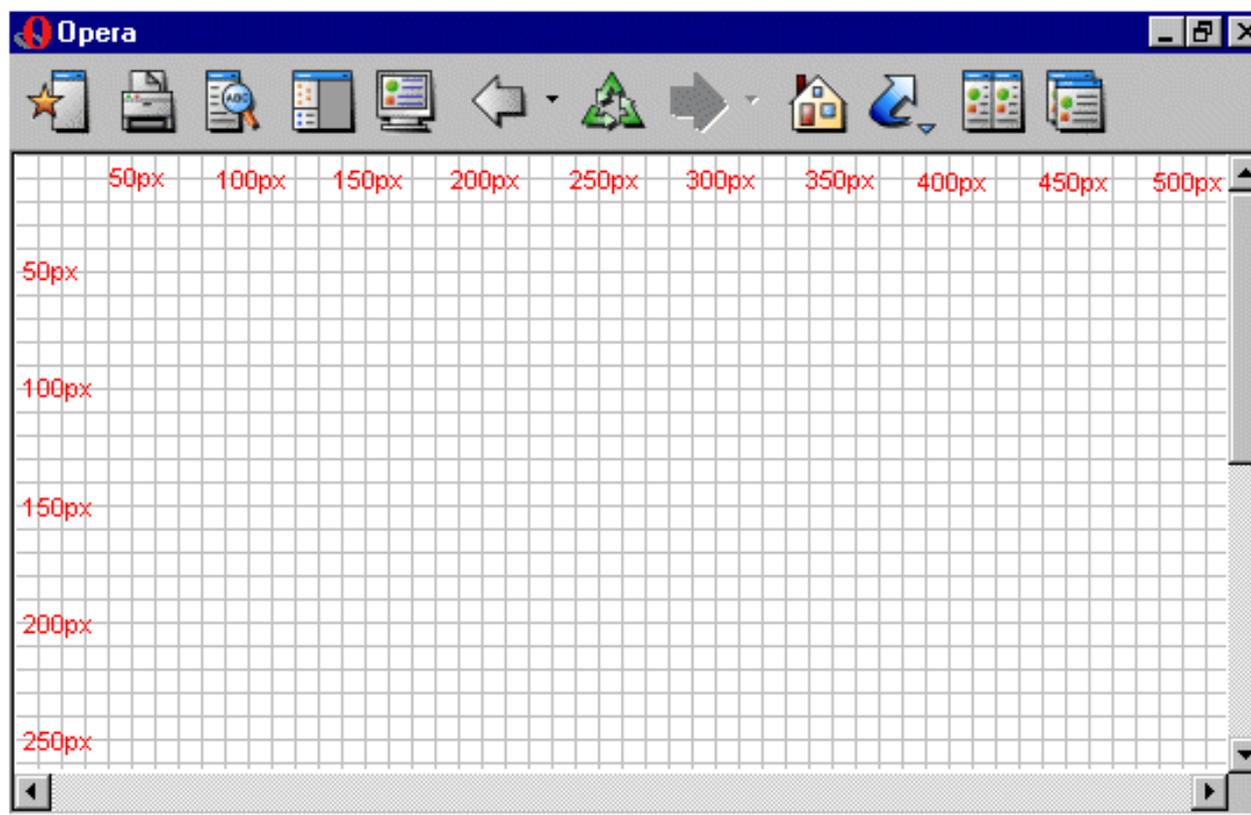
- No CSS:

```
div.box {  
    height: 500px;  
    width: 200px;  
    border: 1px solid black;  
    background: #FF9900;  
}
```

- Com posicionamento CSS podemos colocar um elemento em uma posição exata na página
- Combinado com **floats**, o posicionamento abre muitas possibilidades para criação de layouts precisos e avançados

# Posicionamento

- O princípio de posicionamento CSS
- Considere a janela do navegador como um sistema de coordenadas:



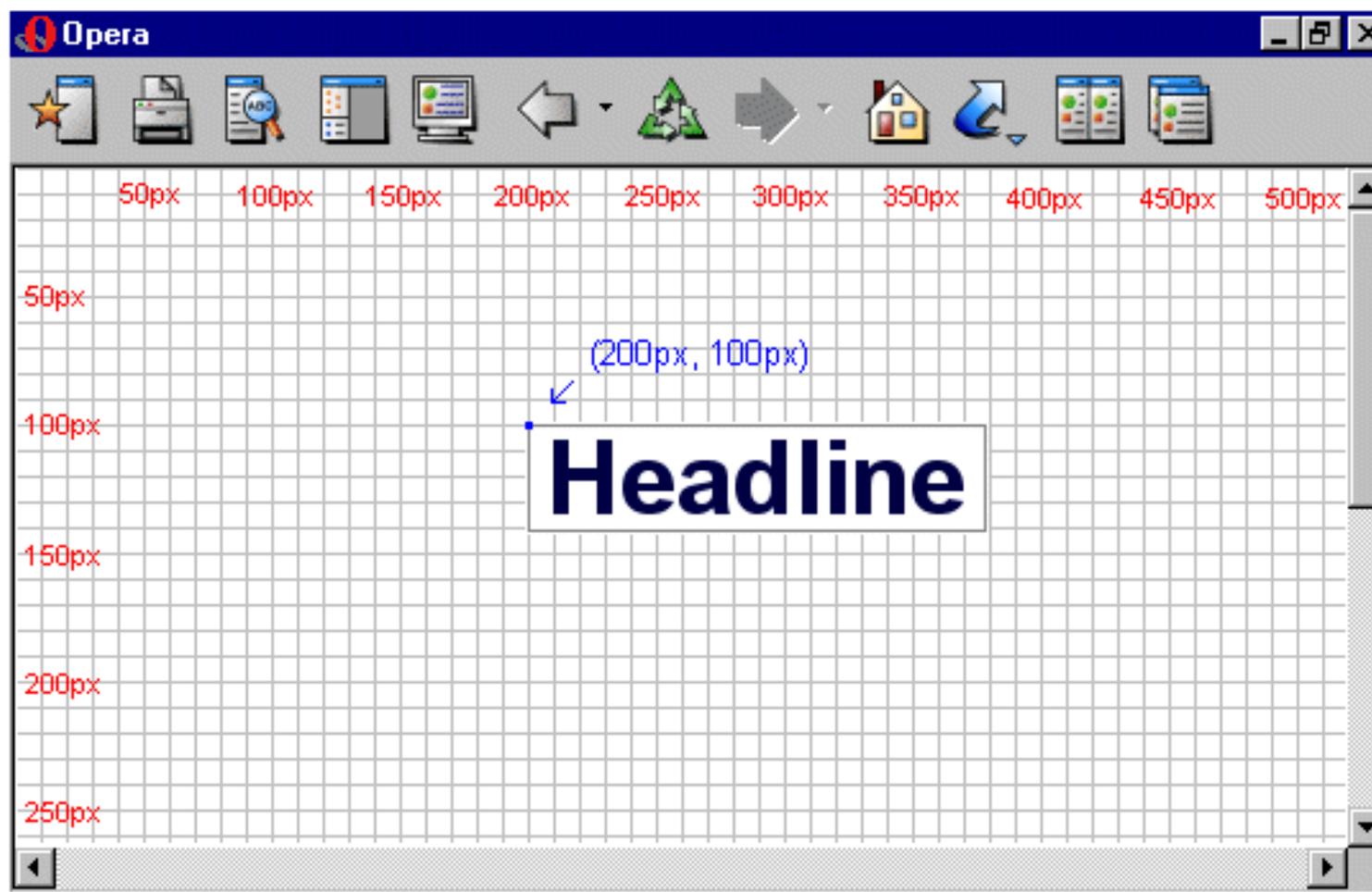
- O princípio de posicionamento CSS estabelece que você pode posicionar um elemento em qualquer lugar na tela usando um sistema de coordenadas
- Vamos supor que queremos posicionar um cabeçalho. Usando o box model o cabeçalho pode ser estilizado para ser apresentado como mostrado a seguir

- Se quisermos o cabeçalho posicionado a 100px do topo do documento e a 200px à esquerda, podemos usar o seguinte CSS:

```
h1 {  
    position: absolute;  
    top: 100px;  
    left: 200px;  
}
```

- O resultado é mostrado a seguir

# Posicionamento



- Posicionar com CSS é uma técnica precisa para colocar elementos. É muito mais fácil do que usar tabelas, imagens transparentes e tudo mais

# Posicionamento absoluto

- Um elemento posicionado absolutamente não cria nenhum espaço no documento
- Isto significa que não deixa nenhum espaço vazio após ser posicionado
- Para posicionar um elemento de forma absoluta a propriedade **position** deve ser definida para **absolute**
- Você pode então usar as propriedades left, right, top, e bottom para definir as coordenadas e posicionar o elemento

# Posicionamento absoluto

- Para exemplificar o posicionamento absoluto, temos quatro boxes nos quatro cantos da página.
- No HTML:

```
<div id="box1">Box 1</div>
```

```
<div id="box2">Box 2</div>
```

```
<div id="box3">Box 3</div>
```

```
<div id="box4">Box 4</div>
```

# Posicionamento absoluto

- No CSS:

```
#box1 {  
    position: absolute;  
    top: 50px;  
    left: 50px;  
}
```

```
#box2 {  
    position: absolute;  
    top: 50px;  
    right: 50px;  
}
```

# Posicionamento absoluto

```
#box3 {  
    position: absolute;  
    bottom: 50px;  
    right: 50px;  
}
```

```
#box4 {  
    position: absolute;  
    bottom: 50px;  
    left: 50px;  
}
```

# Posicionamento absoluto

Box 1

Box 2

Box 4

Box 3

# Posicionamento relativo

- Para posicionar um elemento de forma relativa a propriedade **position** deve ser definida para **relative**
- A diferença entre os dois tipos de posicionamento é a maneira como o posicionamento é calculado
- O posicionamento para posição relativa é calculado com base na posição original do elemento no documento
- Isto significa uma movimentação do elemento para a esquerda, para a direita, para cima ou para baixo. Assim fazendo o elemento ocupar um espaço após ser posicionado

# Posicionamento relativo

- Como exemplo de posicionamento relativo vamos tentar posicionar três imagens relativamente as suas posições originais na página
- Notar como as imagens deixam um espaço vazio nas suas posições originais no documento:

# Posicionamento relativo

## Cães mais influentes da história da humanidade

### By Dicas do Cão

Desde o fim da Era do Gelo os cães acompanham a espécie humana. De lá para cá a cultura deu um salto gigante, e a evolução da civilização cada vez se tornou mais frequente. Diante desse fenômeno, é normal que os cães, que tanto nos acompanham, tenham feito alguma coisa pela aventura humana.

Conheça aqui a nossa lista de cães que marcaram a história da civilização, baseado no livro de Sam Stall. Os famosos cães aparecem desde a antiguidade, até o período (pós) moderno. O livro lista os 100 cães que marcaram a trajetória da humanidade. Conheça alguns que selecionamos aqui:



### Péritas



Era o cão favorito de Alexandre, o Grande, que o acompanhava durante as explorações militares. Provavelmente o nome Péritas tenha se originado da palavra 'Janeiro' em Macedônio. Há indícios de que o cão possa ter pertencido aos molossos (avós dos atuais mastifes). O imperador gostava tanto desse cão que deu seu nome à uma cidade grega.

O fato marcante na vida desse cão, foi o fato dele ter salvo Alexandre de ter sido esmagado por um elefante. Isso aconteceu na batalha de Guagamela, e Péritas tinha mordido o elefante a tempo de salvar seu dono. Se isso não tivesse acontecido, seu dono não teria tempo de ter entrado na história como Alexandre, o Grande.

### Urian



Todos sabem que a Igreja Católica é a mais antiga do mundo. O rei Henrique VIII criou a Igreja Anglicana para poder se casar duas vezes. O que muitos não sabem é que tem um cãozinho envolvido na história. Trata-se de Urian, um cachorro que viveu no século XVI, provocando a eterna separação da Igreja Católica com a Anglicana.

Conta-se que o Cardeal de Inglaterra, que era emissário de Henrique VIII, foi a um encontro do Papa Clemente VII, acompanhado do cão. O Papa e o Cardeal estavam quase chegando a um acordo, quando o cão Urian mordeu a canela do Papa. O resultado disso foi um desacordo, que dura até hoje.

# Posicionamento absoluto

- No HTML:

...

```
<p>O fato marcante na vida desse cão, ...
```

```
... Alexandre, o Grande.</p>
```

```
<div id="dog1"></div>
```

...

```
<p>Conta-se que o Cardeal Wolsey, ...
```

```
... que dura até hoje.</p>
```

```
<div id="dog2"></div>
```

# Posicionamento absoluto

- No HTML:

...

```
<p>A história de Laika ...
```

```
... de regime comunista.</p>
```

```
<div id="dog3"></div>
```

...

# Posicionamento absoluto

- No CSS:

```
#dog1 {  
    position:relative;  
    left: 350px;  
    bottom: 150px;  
}
```

```
#dog2 {  
    position:relative;  
    left: 150px;  
    bottom: 500px;  
}
```

# Posicionamento absoluto

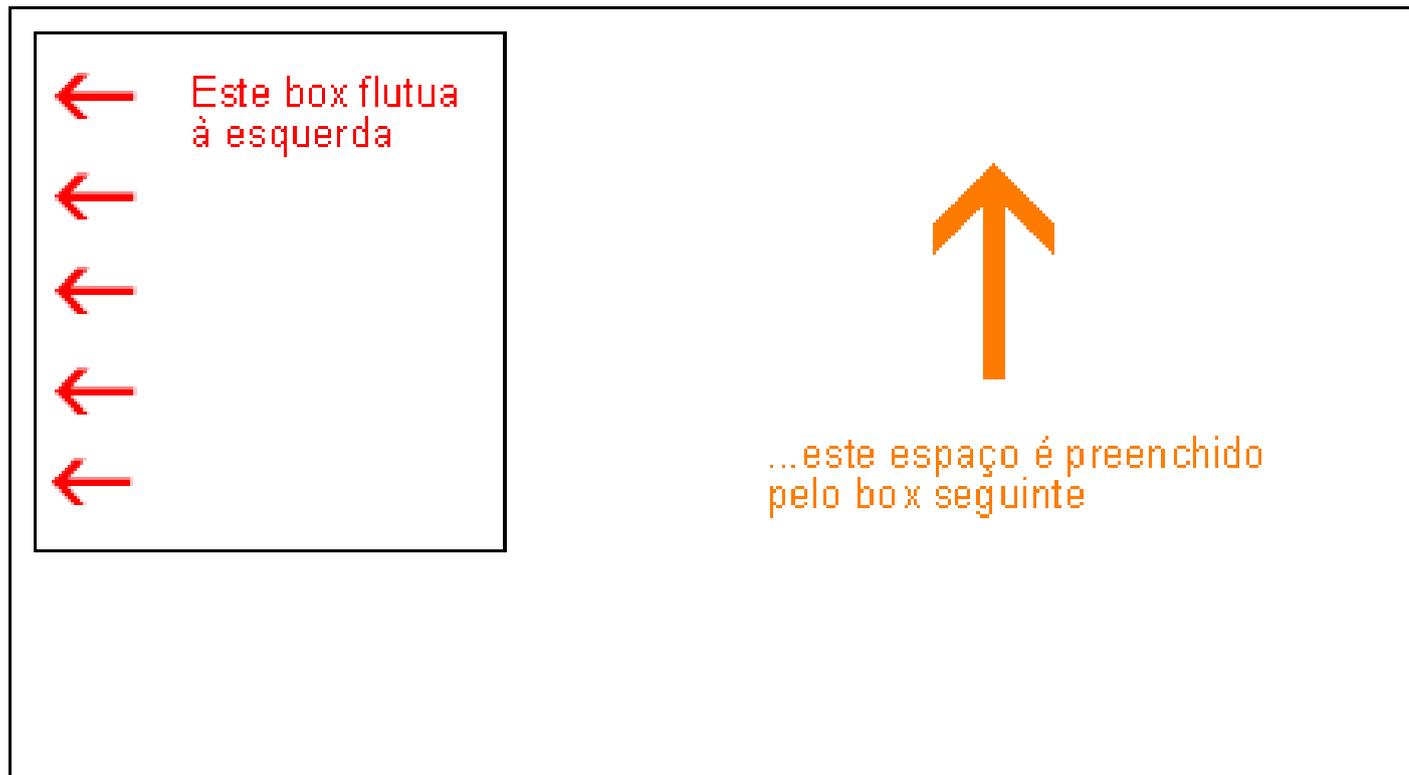
```
#dog3 {  
    position: relative;  
    left: 50px;  
    bottom: 700px;  
}
```

# Flutuando elementos (floats)

- Um elemento pode ser 'flutuado' à esquerda ou à direita com uso da propriedade **float**
- Isto significa que o box e seu conteúdo são deslocados para a direita ou para a esquerda do documento

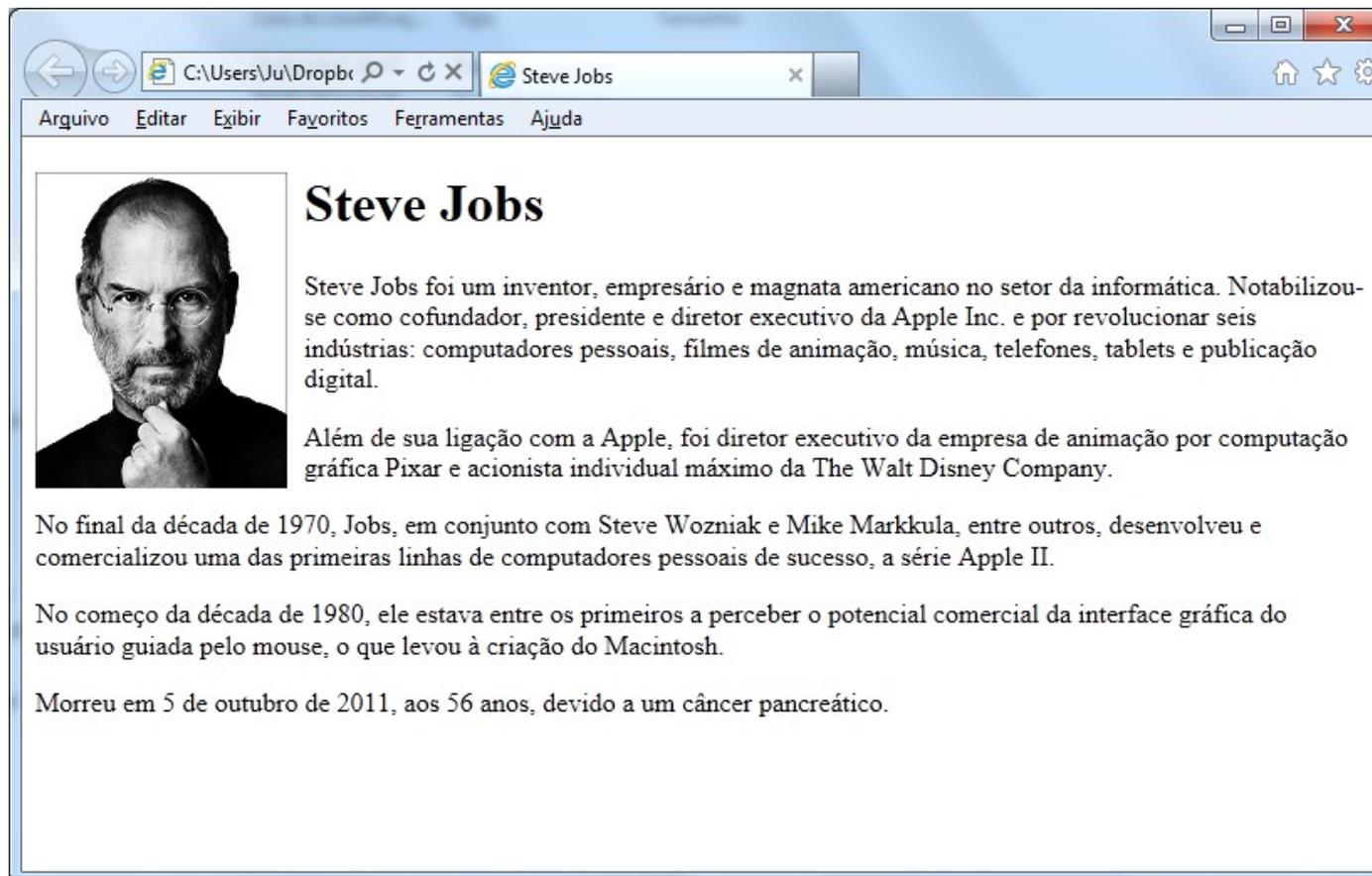
# Flutuando elementos (floats)

- A figura a seguir ilustra o princípio de float:



# Flutuando elementos (floats)

- Se desejamos que um texto seja posicionado em volta de uma figura como mostrado abaixo, basta flutuarmos a imagem:



# Flutuando elementos (floats)

- No HTML:

```
<div id="picture">
```

```
  
```

```
</div>
```

```
<h1>Steve Jobs</h1>
```

```
<p>Steve Jobs foi um inventor, empresário...
```

# Flutuando elementos (floats)

- No CSS: Para conseguir o efeito mostrado, basta definir uma largura para o box que o contém e declarar para ele **float: left;**

```
#picture {  
    float:left;  
    margin-right:10px;  
}
```

# Flutuando elementos (floats)

- Floats podem ser usados para construir colunas em um documento
- Para criar as colunas estruturamos as colunas no código HTML usando `<div>`

# Flutuando elementos (floats)



## Cloud Computing e o Linux

O Linux já representa hoje uma força econômica considerável, com um ecossistema estimado em 25 bilhões de dólares. Está em praticamente todos os lugares. Quando fazemos uma pesquisa no Google ou lemos um livro no Kindle, é o Linux que roda nos bastidores.

O Linux demonstrou de forma inequívoca o potencial do desenvolvimento de sistemas de forma colaborativa, que é o cerne do modelo Open Source. Seria praticamente impossível para qualquer empresa de software, sozinha, criar um sistema operacional de seu porte e complexidade. Estima-se que o custo de desenvolvimento de uma distribuição como a Fedora 9, com seus mais de 204 milhões de linhas de código, corresponda a quase 11 bilhões de dólares. Para chegar ao kernel 2.6.30 (mais de 11 milhões de linhas de código), o investimento seria de mais de 1,4 bilhão de dólares.

Para se ter uma idéia do volume de trabalho em cima do kernel, nos últimos quatro anos e meio, a média foi de 6.422 novas linhas de código adicionadas por dia, além de outras 1.687 alteradas e 3.285 removidas. Da versão 2.6.24 à 2.6.30 a média subiu para 10.923 linhas de código adicionadas por dia. Quem teria cacifê financeiro para sustentar, por si, um empreendimento bilionário desses?

Portanto, o Linux é uma força no presente. É usado não apenas em web servers e print servers, mas a cada dia vemos mais e mais aplicações core das empresas operando em plataformas Linux. A IBM, por exemplo, consolidou seu ambiente de computação interno (seus sistemas internos) em plataformas mainframe System Z, rodando Linux.

O Linux ainda tem muito espaço para crescer. É bem provável que vejamos uma migração mais intensa de ambientes Unix como Solaris e HP-UX para Linux, devido à incerteza quanto ao futuro dessas plataformas. Qual será o futuro dos servidores Solaris sob a nova administração de uma empresa de software como a Oracle, que não conhece os meandros dos modelos econômicos do setor de hardware?

E as plataformas baseadas em Itanium? Manter uma linha de processadores demanda muito investimento e é necessário que haja escala suficiente para haver retorno financeiro. Em comparação com a linha Xeon, o Itanium não é um produto de alto volume para a Intel. Embora a Intel não publique o seu volume de produção, analistas de indústria estimavam que em 2007 o ritmo de produção era de 200.000 processadores por ano. Segundo o Gartner, em 2007 o número de servidores Itanium vendidos foi de 55.000. Número muito pequeno quando comparado com os 417.000 servidores RISC (da família Power da IBM e das demais tecnologias) e os 8,4 milhões de servidores x86.

Mas e o futuro? Com as mudanças que já estão acontecendo, como a crescente disseminação da computação móvel e o modelo de computação em nuvem (Cloud Computing), como o Linux se posiciona?

É uma resposta fácil. O Linux, que não conseguiu muito espaço nos desktops, está se posicionando como plataforma dominante na computação móvel, como smartphones, netbooks e tablets. No seu último ano fiscal, que se encerrou em 30 de junho do ano passado, a Microsoft pela primeira vez reconheceu em seu relatório para os acionistas que os

E quanto aos servidores? Nessas máquinas o Linux está altamente alinhado com as tendências de virtualização e cloud computing. Alguns pontos positivos sobre o Linux chamam a atenção, quando falamos em cloud.

Primeiro, o **Linux opera em praticamente qualquer plataforma de hardware**, o que facilita provisionamento, alocação e gerenciamento de recursos computacionais em nuvem. Podemos criar desde uma nuvem baseada em plataforma x86 como o Google até nuvens em mainframes IBM, aproveitando o alto throughput e facilidade de virtualização dessas máquinas.

O **custo de licenciamento** é outro fator interessante. Embora existam distribuições licenciadas, um provedor de infraestrutura em nuvem, pelo grande número de servidores que deverá dispor (falamos aqui em milhares ou dezenas de milhares de máquinas), poderá adotar, pela escala, versões Linux não comerciais.

**Virtualização** é outro plus do Linux, com diversas tecnologias disponíveis como Xen (base da arquitetura de cloud da Amazon) e KVM.

Hoje, se olharmos Linux em cloud, já vemos seu uso como base tecnológica da nuvem do Google, da Amazon, do Force.com, do Salesforce e de startups como 3Tera (recentemente adquirida pela CA), Elastra e Mosso, entre outros. E com certeza seu uso se alastrará pelas futuras ofertas de nuvens.

# Flutuando elementos (floats)

- No HTML:

```
<div id="column1">  
  <p>0 Linux já representa hoje...  
    ...rodando Linux.</p>  
</div>  
<div id="column2">  
  <p>0 Linux ainda tem...  
    ...espaço a cada dia.</p>  
</div>  
<div id="column3">  
  <p>E quanto aos servidores....  
    ...ofertas de nuvens.</p>  
</div>
```

# Flutuando elementos (floats)

- No CSS:

```
#column1 {  
    float:left;  
    width: 30%;  
    margin-left:10px;  
}
```

```
#column3 {  
    float:left;  
    width: 30%;  
    margin-left:10px;  
}
```

```
#column2 {  
    float:left;  
    width: 30%;  
    margin-left:10px;  
}
```

# Flutuando elementos (floats)

- A seguir definimos a largura de cada coluna, por exemplo 30%, e declaramos float: left; para cada uma das colunas
- float pode ser declarado **left**, **right** ou **none**

# A propriedade clear

- A propriedade **clear** é usada para controlar o comportamento dos elementos que se seguem aos elementos floats no documento
- Por padrão, o elemento subsequente a um float, ocupa o espaço livre ao lado do elemento flutuado
- Por exemplo, o texto deslocou-se automaticamente para o lado da imagem

# A propriedade clear

- A propriedade clear pode assumir os valores **left**, **right**, **both** ou **none**
- A regra geral é: se clear, for por exemplo definido both para um box, a margem superior deste box será posicionada sempre abaixo da margem inferior dos boxes flutuados que estejam antes dele no código

# A propriedade clear

- Para evitar que o texto se posicione no espaço livre deixado pela imagem



# A propriedade clear

- No HTML:

```
<div id="picture"></div><h1>Steve Jobs</h1><p class="floatstop">Steve Jobs foi um inventor, empresário...
```

- No CSS

```
#picture {  
    float:left;  
    margin-right:10px;  
}  
.floatstop {  
    clear:both;  
}
```

# Usando z-index (Layers)

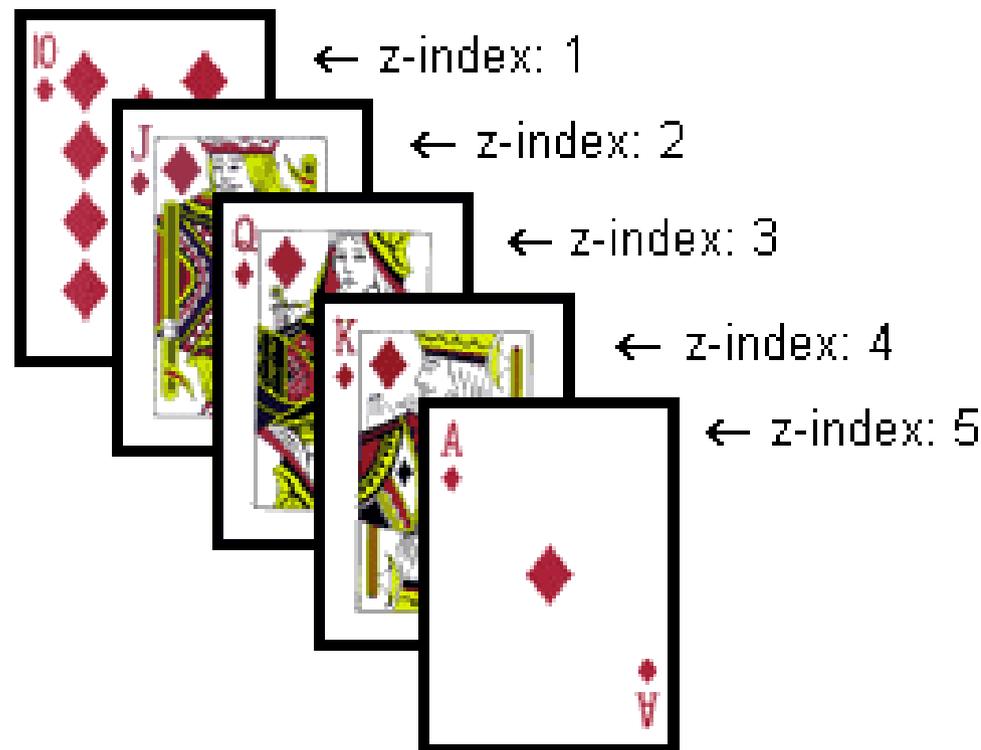
- CSS usa o espaço tri-dimensional - altura, largura e profundidade
- Já vimos as duas primeiras dimensões: altura e largura
- Agora aprenderemos como colocar elementos em **layers** (camadas)
- Resumindo, camadas significam como os elementos se sobrepõem uns aos outros

# Usando z-index (Layers)

- Para fazer isto definimos para cada elemento um número índice (z-index)
- O comportamento é que elementos com número índice maior se sobrepõem àqueles com menor número

# Usando z-index (Layers)

- Vamos supor um royal flush no jogo de poker. As cartas podem ser apresentadas como se cada uma delas tivesse um z-index:



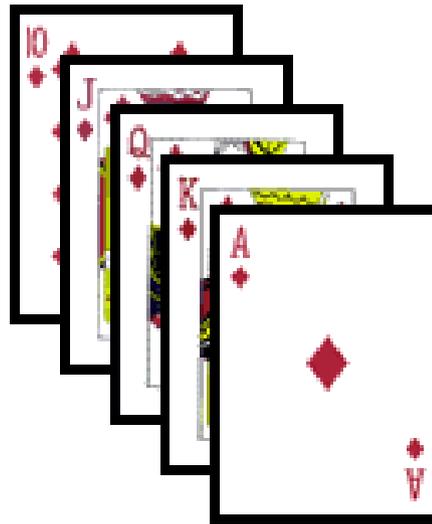
# Usando z-index (Layers)

- No caso mostrado, os números índice estão em uma sequência direta (de 1-5), contudo o mesmo resultado poderia ser obtido com uso de 5 diferentes números, não em sequência. O que conta é a cronologia dos números (a ordem).

# Usando z-index (Layers)

- Exemplo:

## Royal Flush



# Usando z-index (Layers)

- No HTML:

```
<h1>Royal Flush</h1>
```

```
<div id="ten_of_diamonds"></div>
```

```
<div id="jack_of_diamonds"></div>
```

```
<div id="queen_of_diamonds"></div>
```

# Usando z-index (Layers)

- No HTML:

```
<div id="king_of_diamonds"></div>
```

```
<div id="ace_of_diamonds"></div>
```

# Usando z-index (Layers)

- No CSS:

```
#ten_of_diamonds {  
  position: absolute;  
  left: 100px;  
  top: 100px;  
  z-index: 1;  
}
```

```
#jack_of_diamonds {  
  position: absolute;  
  left: 115px;  
  top: 115px;  
  z-index: 2;  
}
```

# Usando z-index (Layers)

```
#queen_of_diamonds {  
  position: absolute;  
  left: 130px;  
  top: 130px;  
  z-index: 3;  
}
```

```
#king_of_diamonds {  
  position: absolute;  
  left: 145px;  
  top: 145px;  
  z-index: 4;  
}
```

# Usando z-index (Layers)

```
#ace_of_diamonds {  
  position: absolute;  
  left: 160px;  
  top: 160px;  
  z-index: 5;  
}
```