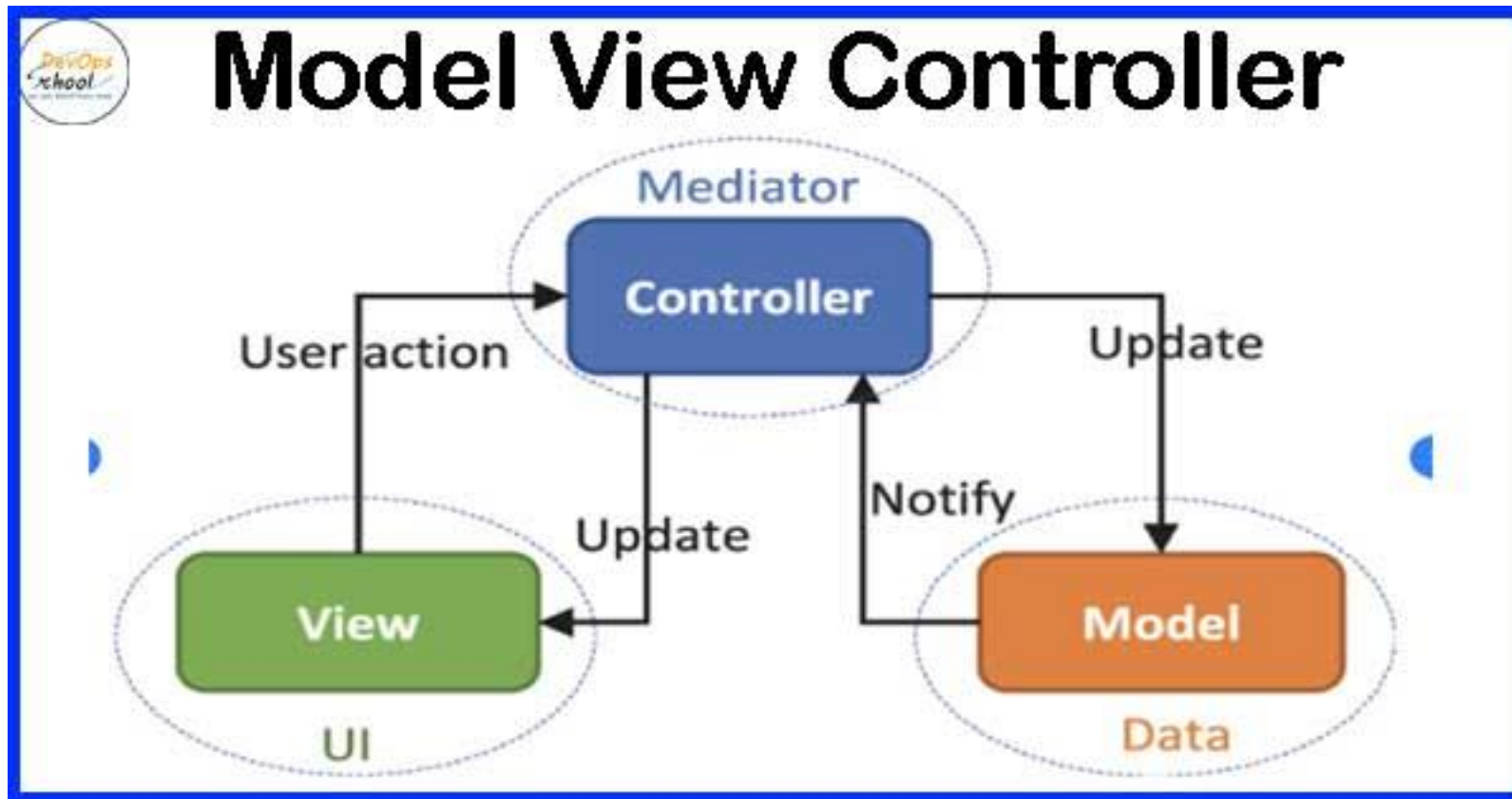


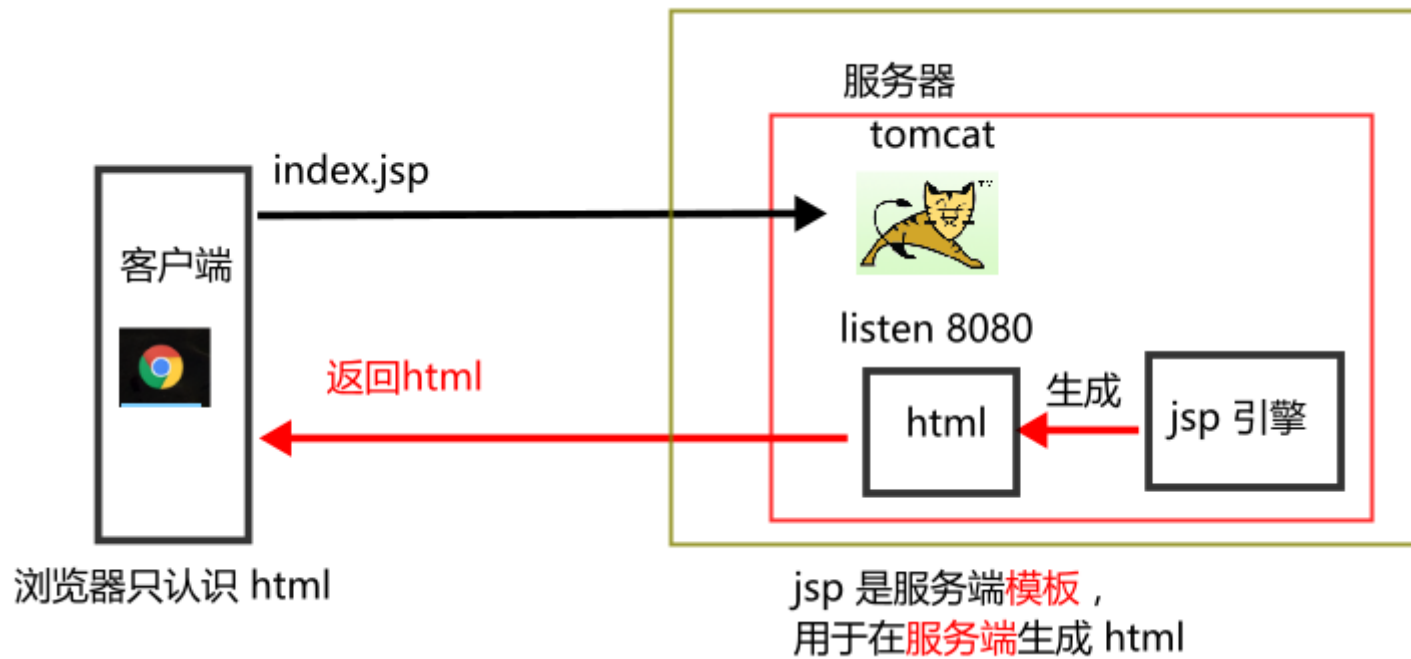
前端培训

杜晓宇

MVC架构



网页开发中的MVC



JSP-传统MVC实现

```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%! int day = 3; %>
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>菜鸟教程(runoob.com)</title>
</head>
<body>
  <h3>IF...ELSE 实例</h3>
  <% if (day == 1 || day == 7) { %>
    <p>今天是周末</p>
  <% }
  else { %>
    <p>今天不是周末</p>
  <% } %>
</body>
</html>
```

JSP前端模板的调试

- 问题： 前后端开发无法分离。

```
{% extends 'base.html' %} {% block content %} 创建任务 返回主页 {% for day in next_days %} {% endfor %} {% for day,res in next_resources %} {% endfor %}
```

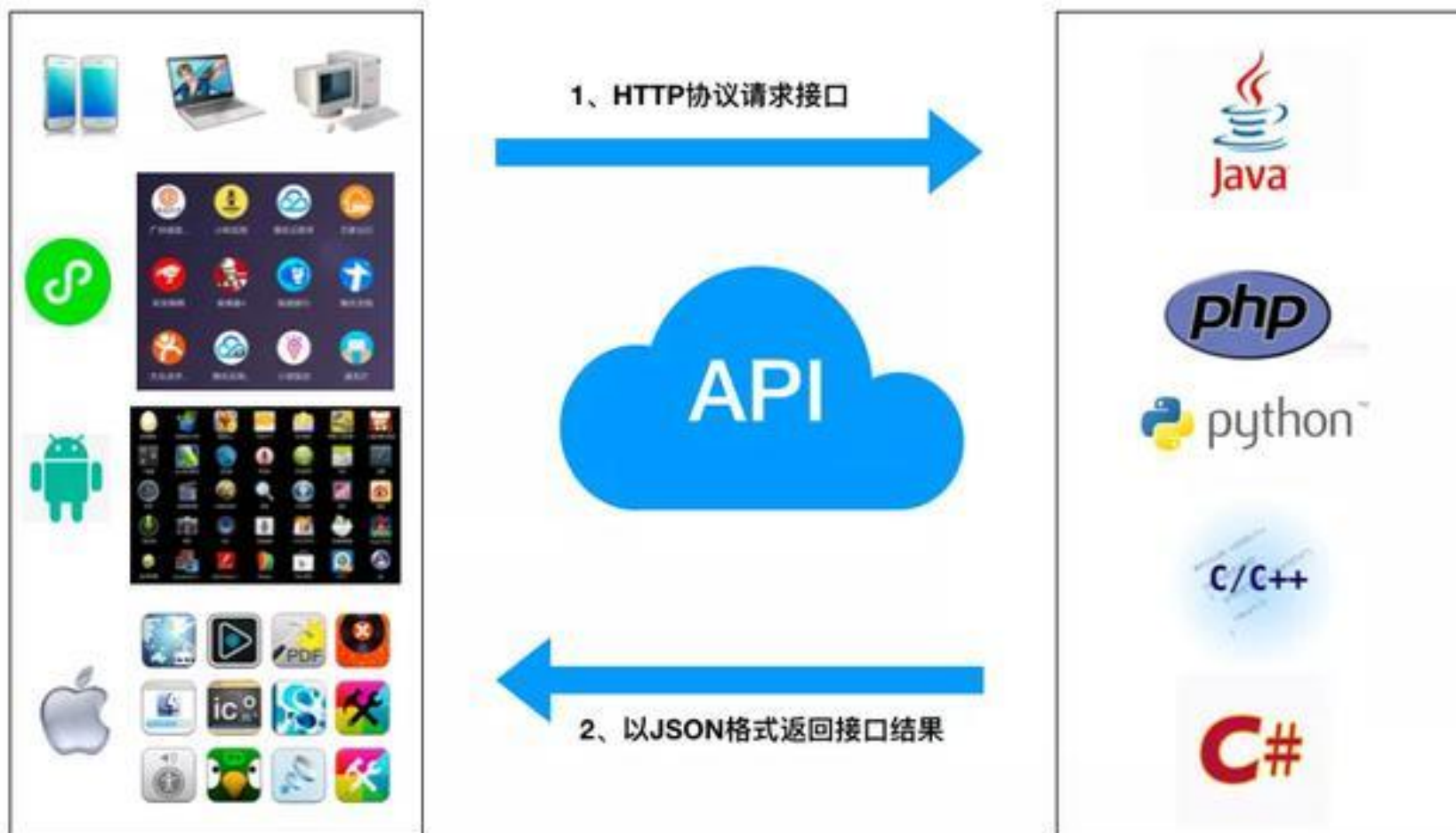
{{day}}
{{res}}

```
{% for task in tasks %} {% endfor %}
```

用户名	开始时间	结束时间	占用资源数	操作
{{task['username']}}	{{task['beg_time']}}	{{task['end_time']}}	{{task['resources']}}	删除

```
{% endblock %}
```

API开发-经典MVC



开发过程

Booking system

创建项目

2021-07-19

日期	7.19	7.20	7.21	7.22	7.23	7.24	7.25
group1	1/8	0/8	2/8	2/8	2/8	2/8	2/8
group2	1/8	0/8	2/8	2/8	2/8	2/8	2/8
group3	1/8	0/8	2/8	2/8	2/8	2/8	2/8
group4	1/8	2/8	2/8	2/8	2/8	2/8	4/8

前端开发

- 桌面开发
 - C#
 - QT
- 手机开发
 - Android
 - Object-C
- **网页开发**
 - 浏览器
 - 小程序
 - 公众号

网页开发

- HTML
- CSS
- Javascript (jQuery)

DOM树

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <title>示例页面</title>
```

```
</head>
```

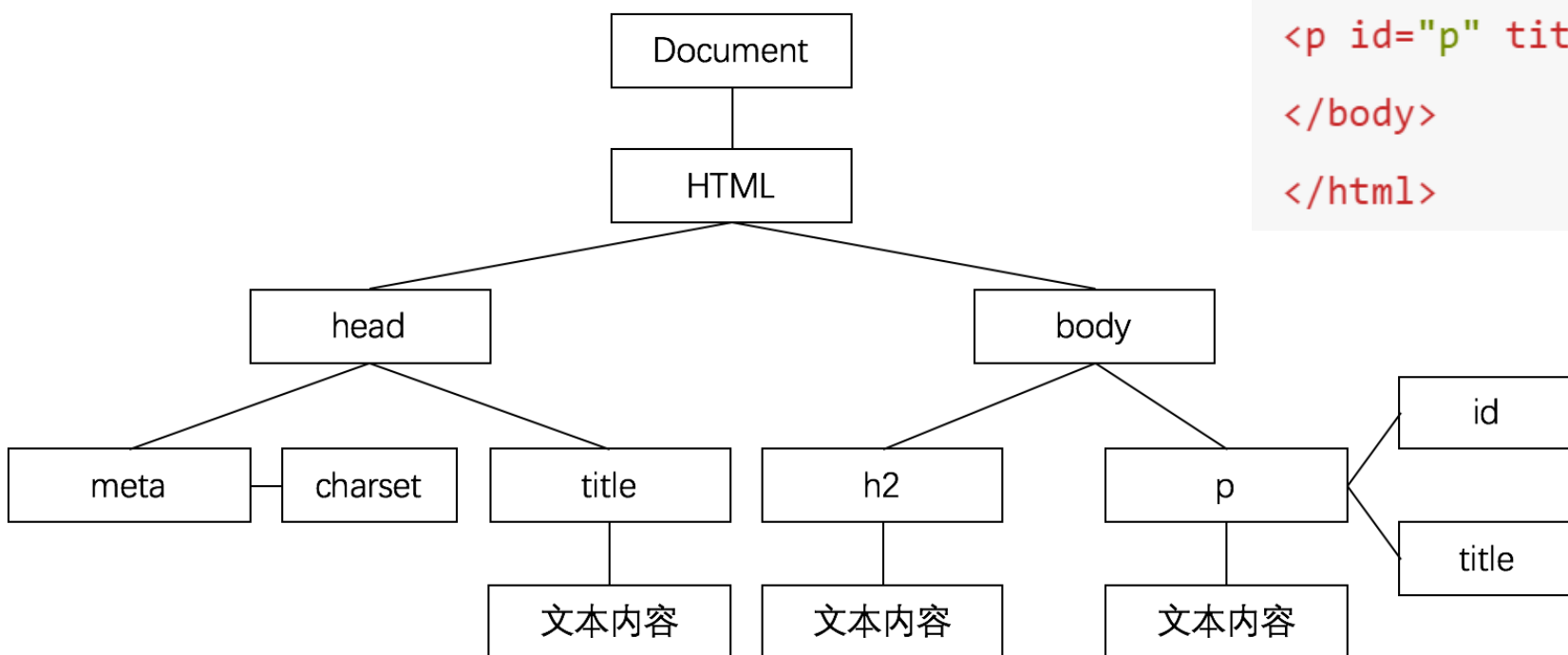
```
<body>
```

```
<h2>这是一个示例页面</h2>
```

```
<p id="p" title="this is p.">这是一个段落内容.</p>
```

```
</body>
```

```
</html>
```



选择器

https://www.w3school.com.cn/css/css_selectors.asp

所有简单的 CSS 选择器

选择器	例子	例子描述
<u><i>.class</i></u>	.intro	选取所有 class="intro" 的元素。
<u><i>#id</i></u>	#firstname	选取 id="firstname" 的那个元素。
<u><i>*</i></u> —	*	选取所有元素。
<u><i>element</i></u>	p	选取所有 <p> 元素。
<u><i>element,element,..</i></u>	div, p	选取所有 <div> 元素和所有 <p> 元素。

CSS与jQuery的选择器是一样的!!!

选择器

- 标签名
- id
- class

- List item 1
 1. List item 1-1
 2. List item 1-2
 3. List item 1-3
 1. List item 1-3-1
 2. List item **1-3-2**
 3. List item 1-3-3
 4. List item 1-4
- List item 2
- List item 3

```
<body>
<ul>
  <li>List item 1
    <ol>
      <li class='spec'>List item 1-1</li>
      <li class='spec'>List item 1-2</li>
      <li>List item 1-3
        <ol>
          <li>List item 1-3-1</li>
          <li>List item <em>1-3-2</em></li>
          <li>List item 1-3-3</li>
        </ol>
      </li>
      <li>List item 1-4</li>
    </ol>
  </li>
  <li id='l2'>List item 2</li>
  <li>List item 3</li>
</ul>
</body>
</html>
```

```
<html>
<head>
<style type="text/css">
ul em {
  color:red; font-weight:bold;
}
li.spec {
  color:blue;
}
li#l2 {
  color: yellow;
}
</style>
</head>
```

CSS

<https://www.w3school.com.cn/css/index.asp>

- 三种用法:

- 在标签的style属性中定义

- `<div style="background-color:#fcfdf8; padding:0;"></div>`

- 在head中使用<style>标签定义

- 在head中引入外部css文件

- `<link rel="stylesheet" type="text/css" href="/c7.css" />`

- 基本格式

```
选择器 {  
  属性: 值;  
}
```

```
<style>  
body {  
  background-color: lightblue;  
}  
</style>
```

```
<!DOCTYPE html>  
<html>  
<head>  
<style>  
body {  
  background-color: lightblue;  
}  
  
h1 {  
  color: white;  
  text-align: center;  
}  
  
p {  
  font-family: verdana;  
  font-size: 20px;  
}  
</style>  
</head>
```

CSS布局

https://www.w3school.com.cn/css/css_float_examples.asp

- 难点： 布局
- 主要标签： div
- 主要css属性： float



块布局

https://www.w3school.com.cn/tiy/t.asp?f=css_layout_cols

- 块级元素(block)是布局主体
 - 一个块级元素默认占一行，行末直接换行
 - h1~h6, div, p, hr, ol, ul, dl, pre, section, table, form, dd
- 行内元素(inline)一般也较少用于布局
 - 行内元素宽度由内容决定，行末不换行
 - Spam, button, input, label, select, textarea, img
- block与 inline可以通过style设置改变。
 - style='display: block;'
- Block放在一行，需要让块级元素浮动。
 - float: left;
- 让float的Block后换行，需要clear: both;

布局样例

Shanghai

The Flight

The City

The Island

The Food

The City

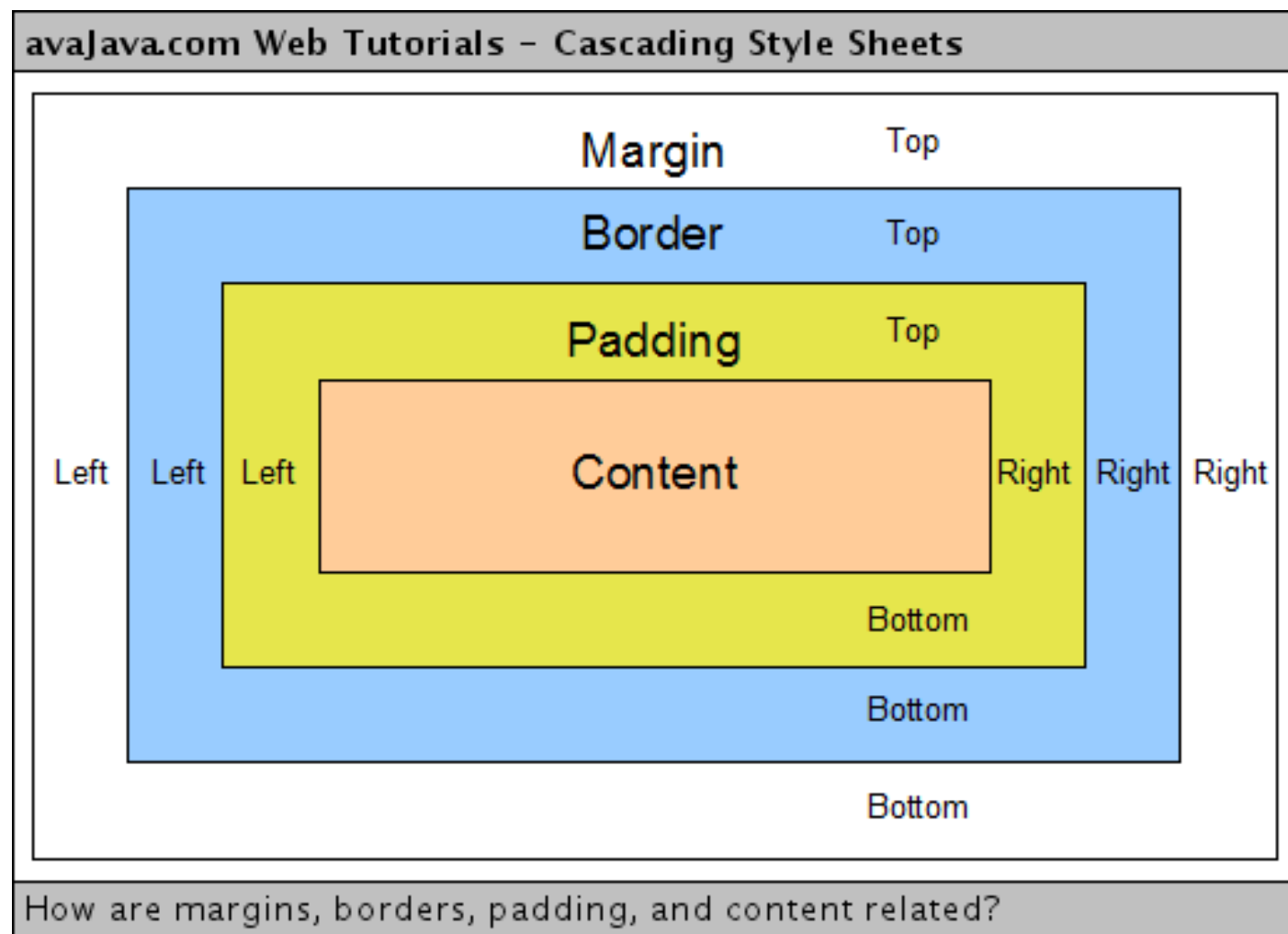
Shanghai is one of the four direct-administered municipalities of the People's Republic of China. Welcome to Shanghai!

You will learn more about web layout and responsive web pages in a later chapter.

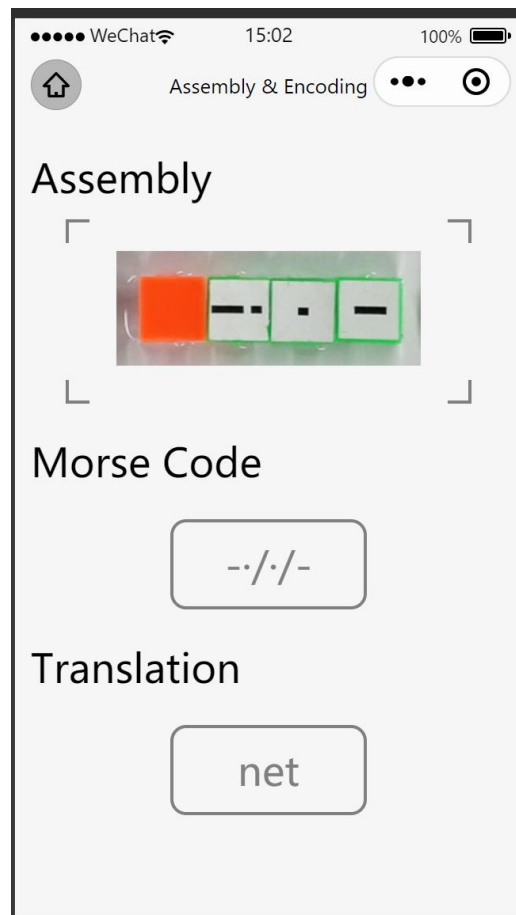
Footer Text

元素内容布局

- Margin: 外部空间
- Padding: 内部空间
- Border: 边框



内容结构样例



静态前端 -> 动态前端

- Javascript (JS)
- jQuery是JS的封装

JS基本格式

- 三种使用方式
- 在head中定义

```
<html>
<head>
<script type="text/javascript" src="jquery.js"></script>
<script type="text/javascript">
$(document).ready(function(){
    $("p").click(function(){
        $(this).hide();
    });
});
</script>
</head>

<body>
<p>If you click on me, I will disappear.</p>
</body>

</html>
```

- 在标签的事件中定义
 - `<button onclick="javascript:myFunction();" >Click me</button>`
- 在head中引入外部定义
 - `<script type="text/javascript" src="jquery.js"></script>`

JS常用事件(jQuery描述)

- `$(document).ready(function() {})`
- `$(selector).click(function(){})`
- 一些常用事件:

Event 函数	绑定函数至
<code>\$(document).ready(function)</code>	将函数绑定到文档的就绪事件 (当文档完成加载时)
<code>\$(selector).click(function)</code>	触发或将函数绑定到被选元素的点击事件
<code>\$(selector).dblclick(function)</code>	触发或将函数绑定到被选元素的双击事件
<code>\$(selector).focus(function)</code>	触发或将函数绑定到被选元素的获得焦点事件
<code>\$(selector).mouseover(function)</code>	触发或将函数绑定到被选元素的鼠标悬停事件

JS事件常见处理

- 保存信息到变量中：
 - `var clicked = 0;`
 - `$('#button').click(function() {clicked+=1;});`
- 将信息写入到页面中：
 - `$('#view').text('confirmed.')`
- 添加或修改DOM结点：
 - `$('#node').append($('#<div>new node</div>');`
- 发送远程访问请求：

```
$("#button").click(function(){
    $.get("demo_test.asp",function(data,status){
        alert("Data: " + data + "\nStatus: " + status);
    });
});
```

前端开发实例

- 步骤一：实现HTML结构
- 步骤二：通过CSS定义样式
- 步骤三：添加JS动态效果

- 调试工具： Chrome

前端学习网站

- <https://www.w3school.com.cn/>
- 一定要亲自试一试!!!

一些资源

- GITHUB免费主页空间: <https://pages.github.com/>
- 团队GITLAB项目管理: <http://10.10.109.103/>

实现中容易遇到的问题

- **问题整理:**

- 1.<head>和<body>css格式文件和js脚本文件尽量放在head中或body后定义。例: <script src="a.js"></script>or<link href="style.css" rel="stylesheet" type="text/css">。
- 2.JavaScript DOM选择器实现元素控制。例: function (\$dom){}, 这样定可以提高函数的复用性和灵活性。
- 3.css中的focus伪类: 伪类在元素获得焦点时向元素添加特殊的样式。
- 4.css中的一些名词问题: padding, border, margin, background, 布局常用<p></p>分行,float:left分块, 以及<div class="clear"></div>消除格式。
- 5.css选择器: 类选择器, id选择器, 注意格式 (不空格会报错) 。
- 6.\$(document).ready(function(){})与windows.onload=function(){})的区别,还有(function(){})()与上两者的区别。
- 7.组件化开发思想 (项目大框架, 通过分块组件化来实现); 前端模板与渲染方式: (1) 页面级渲染, 如: <div>{% \$a %}</div> 后端\$smarty->assign('a', 'give data'); (2) 异步的请求与新增模板, 如ajax获取字符串直接渲染方式。
- 8.其他就是各种函数的使用和各个模块的选择, 以及css格式和js函数怎么写, 需要自己去练一练。