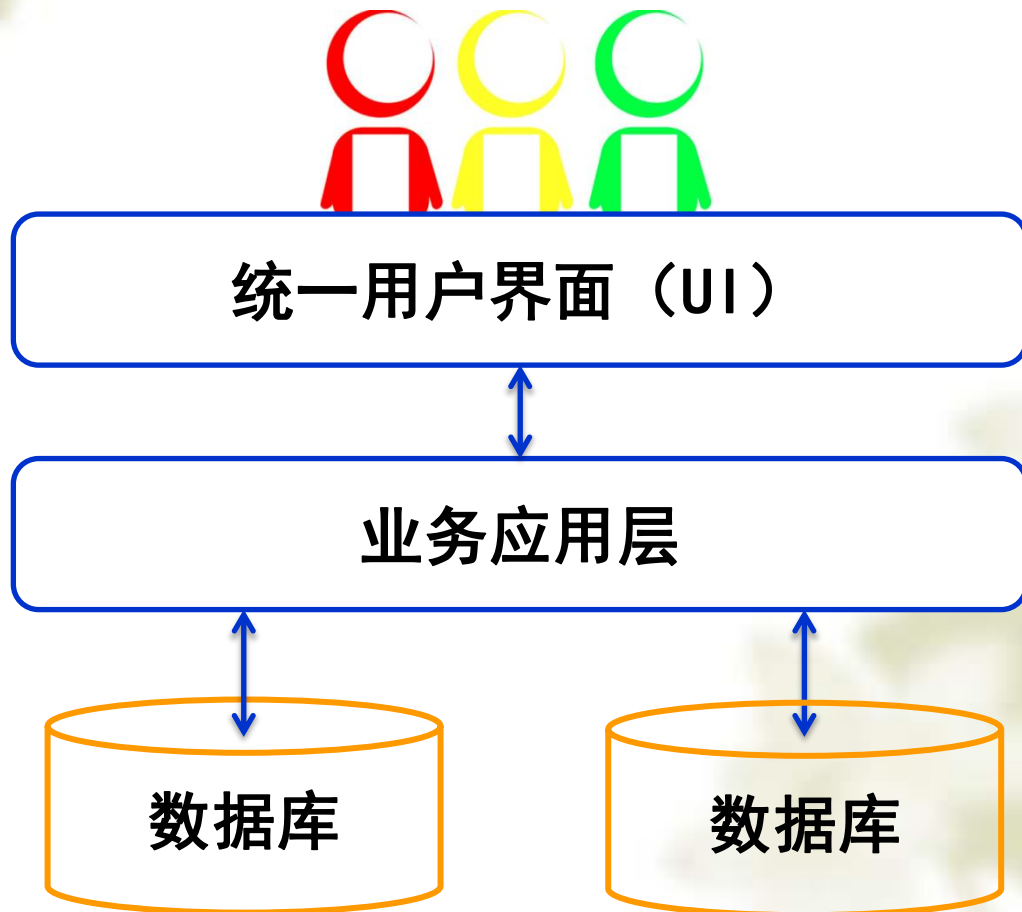


课程介绍

- ❖ 主要内容：**ASP.NET、C#、ADO.NET**，通过实验加强理解与应用！
- ❖ 上课要求：**理解原理，抓住要点，加强练习！**
- ❖ 考核方式：
 - ✎ 平时（含作业）+ 实验 + 期末笔试
- ❖ 学习方法：理论与实践相结合！

课程目标：应用Web开发技术完成一个网站



课程考核说明

❖ 课程成绩评定方法：

- 平时**20%**，实验**30%**，期末考试**50%**

❖ 平时成绩**20%**为学生考勤、课堂表现、课堂测试；玩手机或玩游戏发现一次扣**5**分，扰乱课堂秩序或其他与课堂学习无关的表现每次酌情扣**5~10**分。

❖ 特别提醒：根据学生手册，旷课**3**次及以上取消考试资格！

课程内容



在线资源

❖ 浙江省高等学校在线开放课程共享平台

The screenshot displays the user interface of the Zhejiang Provincial Online Open Course Sharing Platform. At the top, a dark blue header bar contains the platform's logo and name on the left, navigation links for '我的课堂' (My Classroom) and '账户设置' (Account Settings) in the center, and a user greeting '王华琼老师, 欢迎进入教师系统' (Teacher Wang Hualing, welcome to the teacher system) on the right, accompanied by '退出' (Logout) and '返回平台' (Return to Platform) buttons.

Below the header, the main content area is divided into a left sidebar and a central workspace. The sidebar, titled '管理信息系统开发' (Management Information System Development), lists various course management functions: '课程信息' (Course Information), '班级管理' (Class Management), '课程资料' (Course Materials), '教学计划' (Teaching Plan), '章节目录' (Chapter Directory), '章节发布' (Chapter Release), '练习考试' (Practice Exam), and '统计分析' (Statistical Analysis). The '章节发布' (Chapter Release) option is currently selected and highlighted in blue.

The central workspace is dedicated to chapter management. It features a top navigation bar with buttons for '公共班级' (Public Class), '17信管' (17 Information Management), and '17软工' (17 Software Engineering). A '发布所有章节' (Release All Chapters) button is positioned on the right. The main area lists course chapters with expandable sections. Each chapter entry includes a dropdown arrow, the chapter title, and a '发布本章' (Release This Chapter) button. The visible chapters are: '课程介绍' (Course Introduction), '第一章 Web应用基础' (Chapter 1: Web Application Basics), '第二章 Visual Studio集成开发环境' (Chapter 2: Visual Studio Integrated Development Environment), and '第三章 ASP.NET技术基础' (Chapter 3: ASP.NET Technology Basics). Each chapter is further broken down into sub-sections, such as '第一部分 Web应用概述' (Part 1: Overview of Web Applications) and '第二部分 网页编程基础' (Part 2: Basics of Web Programming) under Chapter 1, and '第一节' (Section 1) under Chapter 2.

在线资源

❖ QQ交流群



群名称:Web应用开发技术-18网工
群 号:950338950

第1章 Web应用基础

内容

❖ Web应用概述

- ☞ B/S与C/S架构模式

- ☞ Web应用相关技术的发展

❖ Web应用相关技术

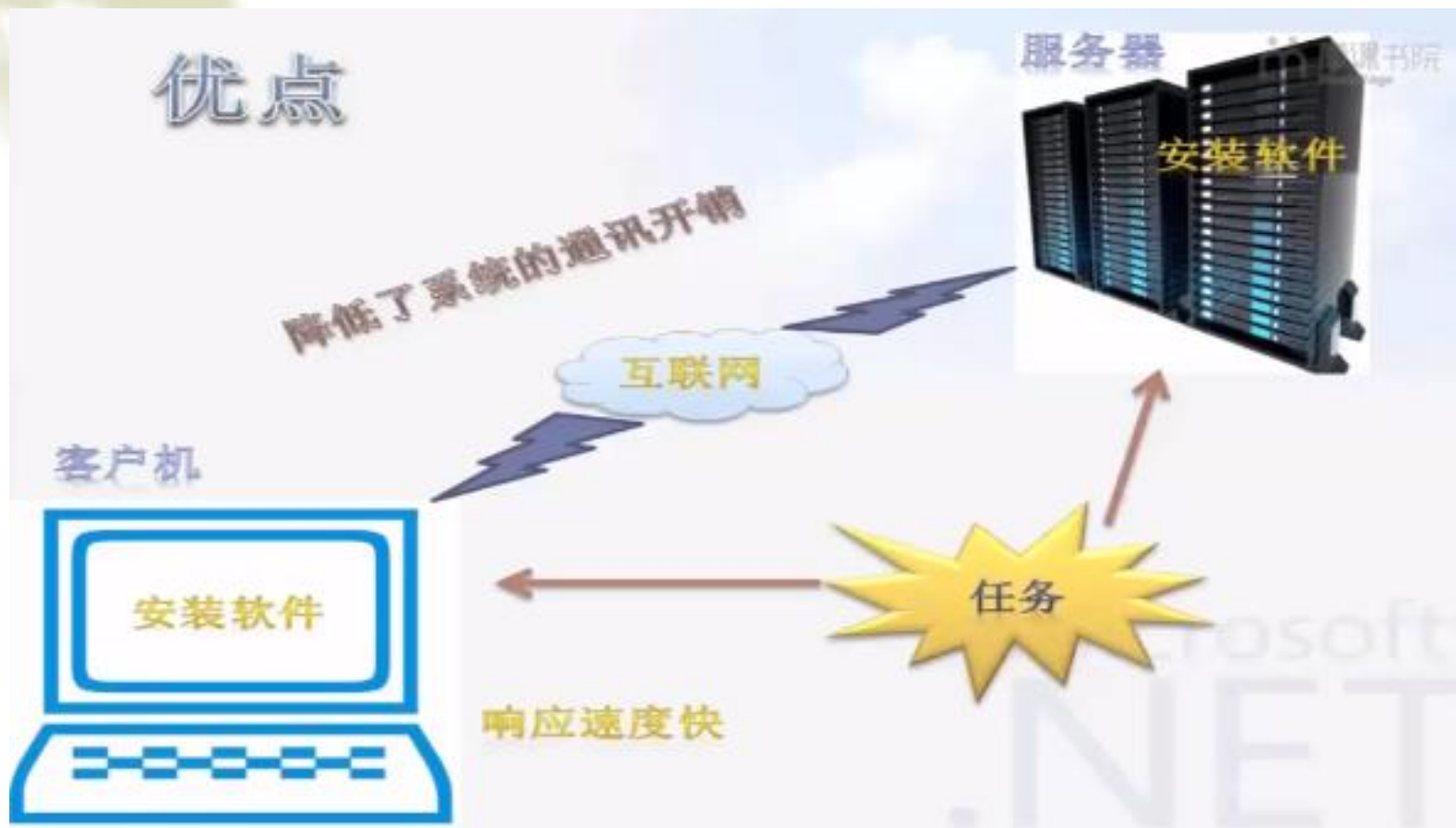
1.1 Web应用概述

- ❖ **B/S与C/S架构模式**
- ❖ **Web应用相关技术的发展**

1.1.1 B/S与C/S架构模式

- ❖ 在企业级应用软件中，若按系统部署的体系结构来分，往往可将其分为**B/S(Browser/Server)**和**C/S(Client/Server)**两种架构模式。
 - ✓ **C/S**架构是指通过在客户端安装一个软件，通过该软件访问服务端资源的一种结构体系。
 - ✓ **B/S**模式是指通过在服务端安装一些应用程序，在客户端只要通过浏览器访问服务器，就可以查看相关内容。
- ❖ **Web应用**就是指在**B/S**架构体系下的应用软件系统。

C/S架构模式（1）



C/S架构模式（2）

❖ C/S架构模式的缺点

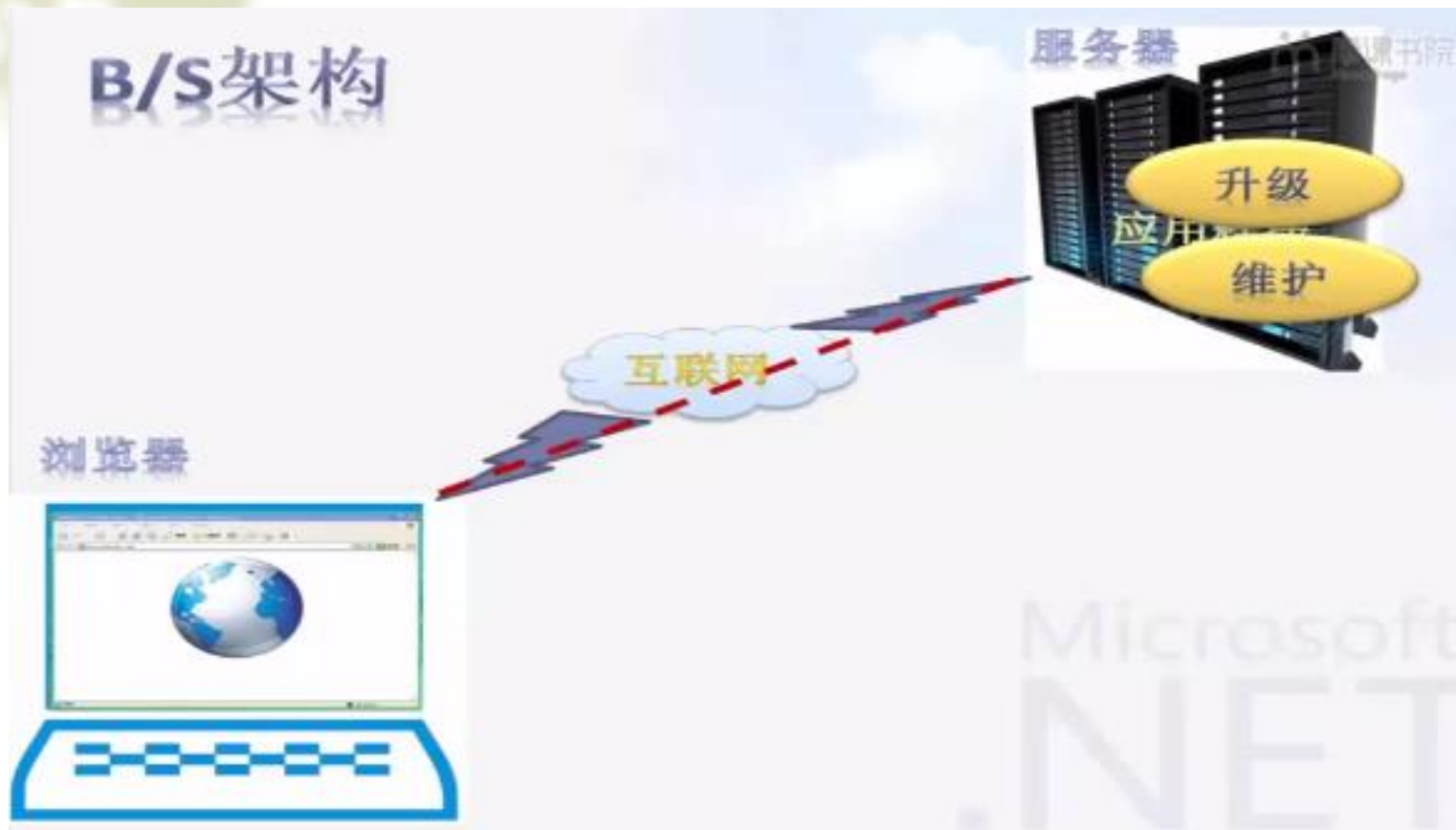


C/S架构模式（3）

❖ C/S架构模式的缺点

- ❧ 不同操作系统开发不同版本的软件
- ❧ 产品更新换代非常快，软件开发成本较高

B/S架构模式（1）



B/S架构模式（2）

❖ B/S架构模式的优点

维护和升级方式简单

使用起来会越来越简单



B/S架构模式（3）

❖ B/S架构模式的缺点



1.1.2 Web应用相关技术发展

- ❖ 从技术层面看，**Web**技术核心有三点：
 - ❧ 超文本传输（**HTTP**）协议，实现万维网的信息传输；
 - ❧ 统一资源定位符（**URL**），实现互联网信息的定位统一标识；
 - ❧ 超文本标记语言（**HTML**），实现信息的表示与存储

Web应用相关技术发展（续）

- ❖ 静态网页：**HTML**的诞生
- ❖ 客户端动态**Web**页面
- ❖ 服务器端动态**Web**页面

静态WEB网页

- ❖ 是用**HTML**书写的一种特殊类型的文件。
- ❖ **HTML**是超文本标记语言(**Hypertext Markup Language**)
- ❖ 浏览器解释执行**HTML**代码，并显示相应的页面

如何辨别静态网页？

- 编写**WEB**页面时已经用**HTML**代码完全确定了网页的具体内容；
- 无论谁或何时访问页面，静态**WEB**页面的内容和外观总是保持不变。

静态Web页面的工作原理

1、作者编写
HTML



静态Web页面的局限性

- ❖ 是否实时显示时间？
- ❖ 是否能针对不同用户定制Web页面？
- ❖ 能否阻止其他人复制自己的HTML代码？
- ❖ ...

客户端动态WEB页面

- ❖ 动态地改变网页的内容，为不同用户提供个性化的服务。

客户端的动态Web页面工作原理

Web服务器

1、作者编写的
网页文件

3、Web服务器定位HTML文件和指令文件

4、HTML和指令被返回到浏览器

2、客户请求网页

5、浏览器处理
指令并将指令
转换为HTML

6、浏览器处理HTML
流并显示页面

客户机

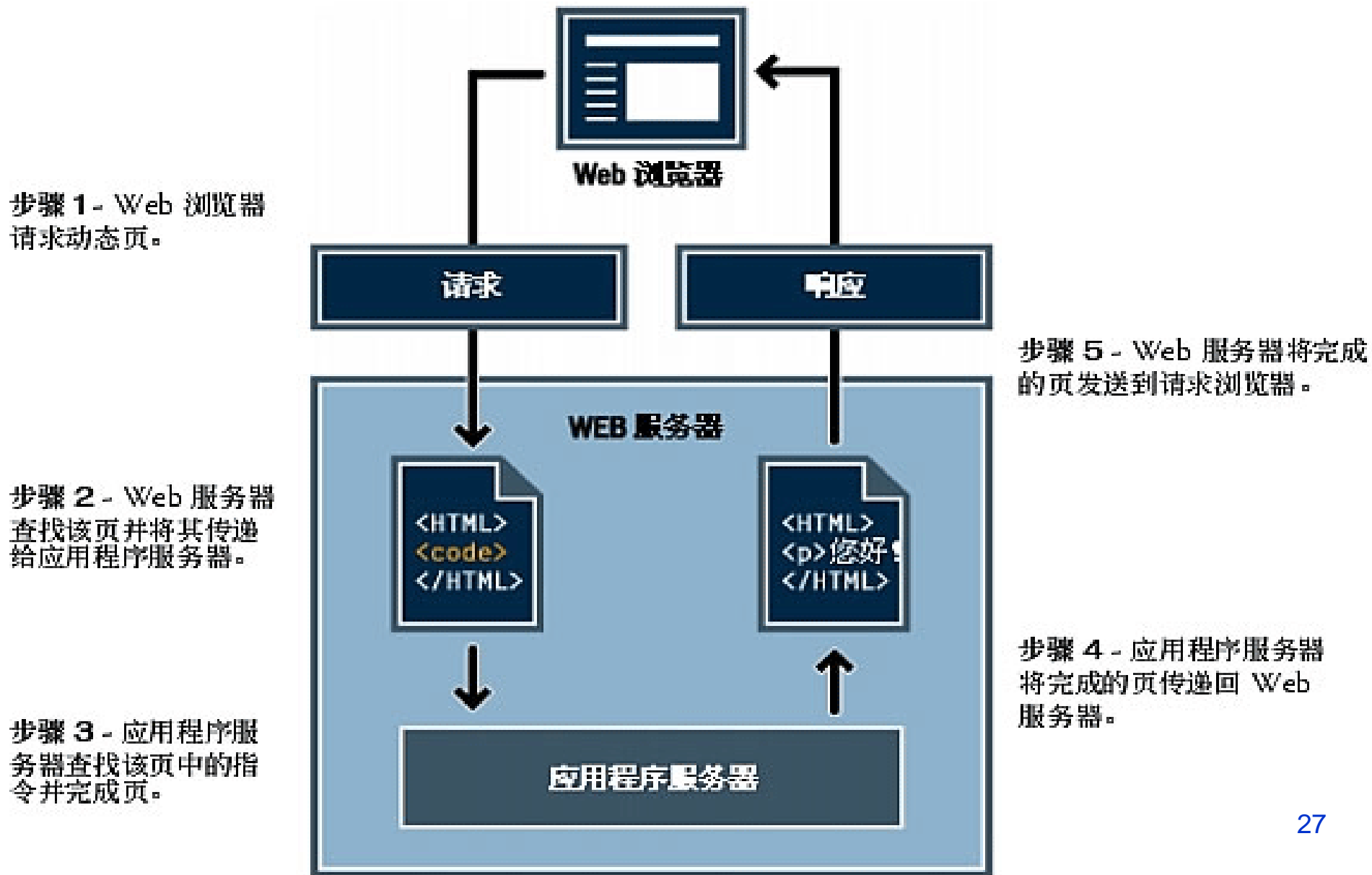
客户端的动态Web页面特点

- ❖ 采用脚本语言编程，在网页加载后对**HTML**中的元素进行操作。
- ❖ 脚本语言：**JavaScript** 和 **VBScript**
- ❖ 通常用于客户端的验证或页面的动画效果
- ❖ 能减轻服务器负担，充分利用客户机资源

客户端编程的缺点

- ❖ 所有代码都要下载到客户端才能执行，因此网页打开慢；
- ❖ 需对不同浏览器进行测试
- ❖ 如需用到服务器资源（如数据库中的数据），无法实现
- ❖ 无法阻止其他人复制自己的代码

服务器端的动态Web页面工作原理



服务器端编程技术

- ❖ 能克服客户端编程的缺点
- ❖ 在静态**Web**页面的基础上，增加服务器端代码。
- ❖ 服务器端编程技术有：

ASP、PHP、JSP、CGI、ASP.NET

❖ **ASP与ASP.NET**

- ✓ **ASP(Active Server Pages动态服务器页面)**：用脚本编程方法，添加代码的方法与客户端脚本中添加方法基本相同
- ✓ **ASP.NET**：是**ASP**的升级版本，提供了一种以**Microsoft .NET Framework**为基础开发**Web**应用程序的全新编程模式

Web应用开发技术体系

浏览器端技术

HTML
XHTML
Javascript
Css
Applet
ActiveX

HTTP

服务器端技术

CGI
ISAPI
ASP
PHP
JSP
ASP.NET

数据访问

数据库技术

Oracle
Sql Server
DB2

1.2 Web应用的相关技术

- ❖ **HTML**
- ❖ **XHTML**
- ❖ **可扩展标记语言XML**

网页前端开发基础

HTML

内容

CSS

样式

JavaScript

行为

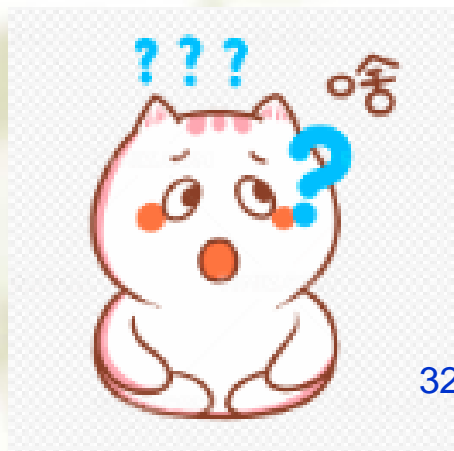
测一测

1. 用HTML标记语言编写一个简单的网页，网页最基本的结构是（ ）。

- A. `<html> <head>...</head> <frame>...</frame> </html>`
- B. `<html> <title>...</title> <body>...</body> </html>`
- C. `<html> <title>...</title> <frame>...</frame> </html>`
- D. `<html> <head>...</head> <body>...</body> </html>`

2. 以下标记符中，用于设置页面标题的是（ ）。

- A. `<title>`
- B. `<caption>`
- C. `<head>`
- D. `<html>`



一、HTML入门

❖ HTML是一种用来制作超文本文档的简单**标记**语言。



The screenshot shows a code editor window titled "Default.aspx" with a list of 18 lines of HTML code. The code is as follows:

```
1 <%@ Page Language="C#" AutoEventWireup="true" CodeFile="De
2
3 <!DOCTYPE html>
4
5 <html xmlns="http://www.w3.org/1999/xhtml">
6 <head runat="server">
7 <meta http-equiv="Content-Type" content="text/html; charse
8 <title></title>
9 </head>
10 <body>
11 <form id="form1" runat="server">
12 <div>
13
14 </div>
15 </form>
16 </body>
17 </html>
18
```

Two red arrows point from text annotations to the code:

- An arrow points from the text "标记的属性 空格分隔" to the space between the opening angle bracket and the first attribute in line 11: `<form id="form1" runat="server">`.
- An arrow points from the text "<标记> </标记>" to the opening and closing tags in line 12: `<div>` and `</div>`.

HTML结构

```
Default.aspx  + x
1  <%@ Page Language="C#" AutoEventWireup="true" C
2
3  <!DOCTYPE html>
4
5  <html xmlns="http://www.w3.org/1999/xhtml">
6  <head runat="server">
7  <meta http-equiv="Content-Type" content="text/h
8      <title></title>
9  </head>
10 <body>
11     <form id="form1" runat="server">
12     <div>
13
14     </div>
15     </form>
16 </body>
17 </html>
18
```

头部

主体

常用HTML标记符及其功能

- ❖ **<head>...</head>**: 首部标记符, 不包含网页的内容, 仅提供一些与网页相关的信息。
- ❖ **<body>...</body>**: 主体标记符, 包含网页内的所有内容, 如文字、图片及超链接等。
- ❖ **<title>...</title>**: 设置网页的标题, 一般在浏览器的顶部标题栏中显示。
- ❖ **...**: 控制字符的样式, **size**是它的一个常用属性, 用来控制字符大小。
- ❖ ****: 插入图片, 常用**alt**属性设置图片简单文字说明, 用**src**属性指明图片所在位置。(本地图片、网络图片)
- ❖ **<a>...**: 定义一个超链接, 用**href**属性可创建多种形式的超链接。

常用HTML标记符及其功能（续）

- ❖ **<hr>**: 换行并绘制一条水平直线，直线的上下两端都会留出一定的空白。
- ❖ **
: 强行中断当前行，多个
**标记可以创建多个空行。
- ❖ **<p>...</p>**: 在网页中分段，遇到**</p>**另起一个新的段落。
- ❖ **<center>...</center>**: 将内容居中。
- ❖ **...**: 定义一个有序列表，列表项的条目用标记符**li**创建，**ol**中可包含一个或多个**li**标记。
- ❖ **...**: 定义一个无序列表，列表项的条目用标记符**li**创建，**ul**中可包含一个或多个**li**标记。
- ❖ **<audio>...</audio>**: 播放音频文件；
- ❖ **<video>...</video>**: 播放视频文件；
- ❖ **<div>...</div>**: 定义一个块，块内可以有文本、图像等，目的是为了控制该块的样式。

常用HTML标记符及其功能（续）

- ❖ `<table>... </table>`: 创建一个表格。
- ❖ `<tr>...</tr>`: 开始表格中的每一行。
- ❖ `<td>...</td>`: 开始一行中的每一个单元格。
- ❖ `<th>...</th>`: 设置表格头

二、CSS入门

- ❖ 如何选择设置CSS样式的HTML元素？
- ❖ CSS与HTML如何关联？
- ❖ 常用的CSS属性有哪些？

CSS选择器

1、基本选择器包括HTML标记选择器、class（类）选择器和id（标识符）选择器。

（1）标记选择器

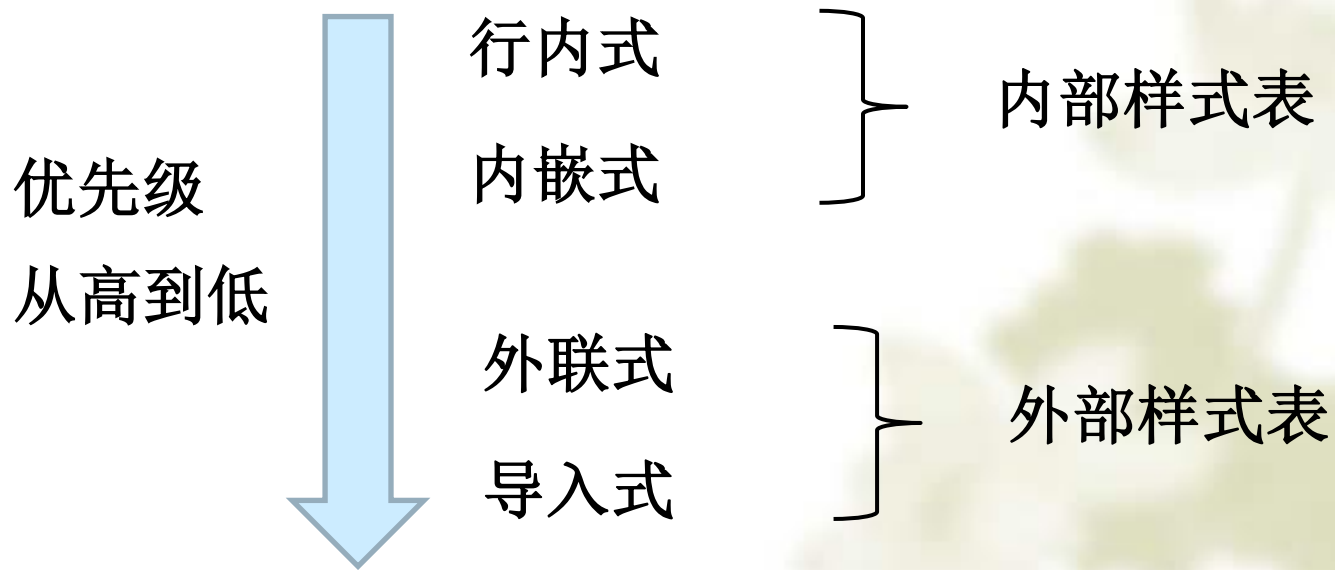
标记选择器是给指定标记定义样式的。

（2）Class（类）和id（标识符）选择器

方法与标记选择器定义样式表相似，如样式表代码的位置和样式表代码的组成都是相同的。其中，Class（类）选择器名以圆点开头，而id（标识符）选择器名则以#号开头。

CSS样式表在网页中的应用

CSS样式表包括内部样式表和外部样式表，具体应用方法为：行内式、内嵌式、外链式和导入式方法。



外部样式表是指CSS样式表的代码是置于HTML文件外部的，并以独立于HTML文件的形式单独保存在扩展名为.css的CSS样式表文本文件。

外链式方法是指把已经建立好的外部“.css样式表”文件，使用<link>标记链接到当前页面，以便应用其中的样式。

```
<head>
```

```
    <link rel="stylesheet" href="样式表文件名.css"  
    type="text/css" />
```

```
</head>
```


CSS字体属性

CSS 字体属性

属 性	属 性 含 义	属 性 值
font-family	使用什么字体	<字体名称> <字体系列>
font-size	定义字体的大小	<绝对大小> <相对大小> <长度>
font-style	字体是否斜体	normal italic blique
font-weight	定义字体的粗细	normal bold bolder lighter
font-variant	字体是否用小体大写	normal small-caps
font	定义各种字体属性	<以上各种字体属性的属性值>

CSS文本属性

CSS 文本属性

属 性	属 性 含 义	属 性 值
color	设置文本颜色	<颜色名称> <十六进制数 > < rgb 函数 >
word-spacing	定义单词之间的间距	normal <长度值>
letter-spacing	定义字符之间的间距	normal <长度值>
text-align	定义文字的对齐方式	left right center justify
text-indent	定义文本的首行缩进	<长度值> <百分比>
text-decoration	定义文字的修饰样式	none underline overline line-through blink
text-transform	使文本中的字母转换为其他形式	none capitalize uppercase lowercase
line-height	设置文本所在行的行高	normal <数值> <长度值> <百分比>

CSS文本样式——火焰字效果

❖ [E08_03.html](#)

设置文本颜色为黑
色，营造黑夜环境，便
于查看火焰效果

网页版面布局

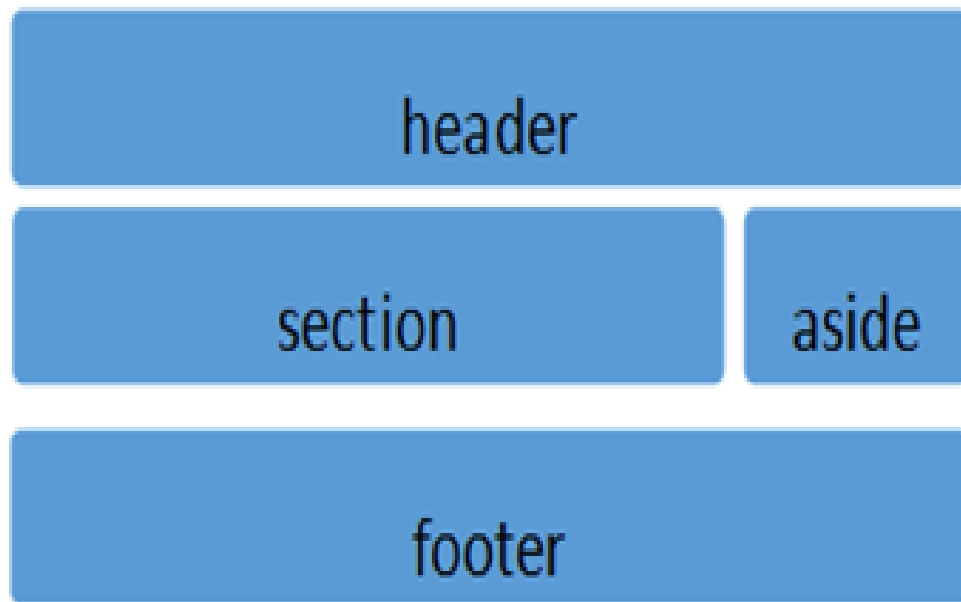
❖ 页面区域

🌀 标题栏 (Header)

🌀 页尾栏 (Footer)

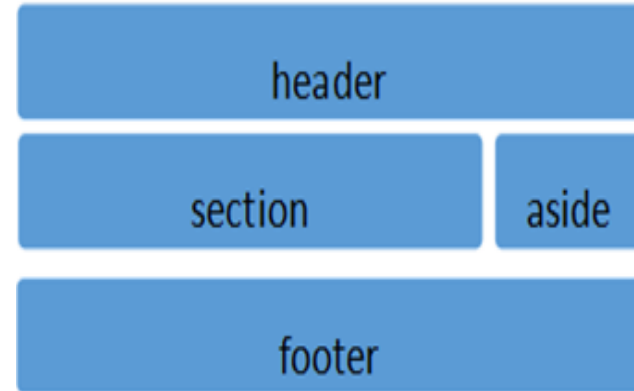
🌀 菜单栏 (aside)

🌀 内容栏 (Main)



网页版面布局

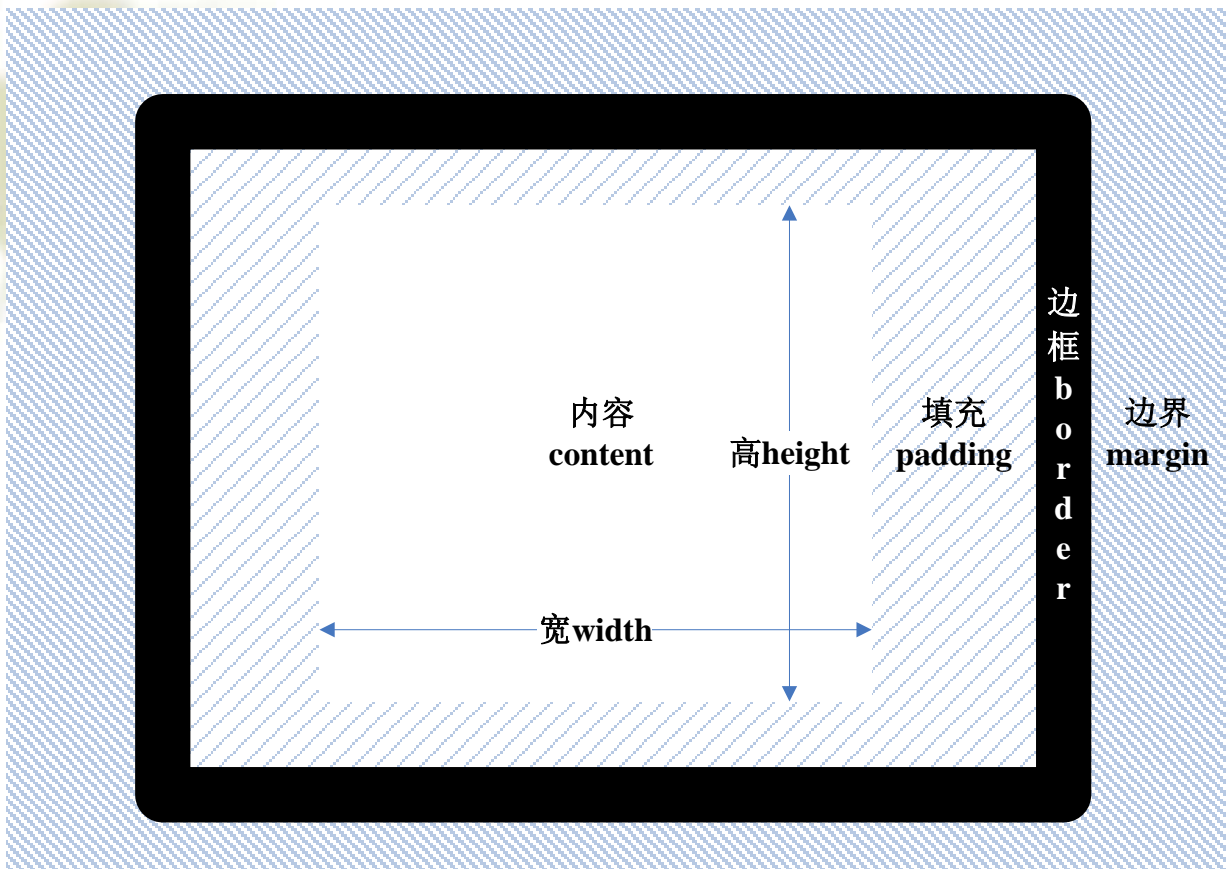
浮动属性 `float: none | left | right;`



清除浮动属性 `clear: none | left | right | both;`

定位属性 `position: static | absolute | relative | fixed;`

Web框



1. web框的宽度（width）、高度（height）样式
2. web框的填充距（padding）样式
3. web框的外边距（margin）样式

回顾

- ❖ **Web应用概述**
 - ❖ **B/S和C/S架构模式**
 - ❖ **Web技术的发展**
- ❖ **Web应用相关技术**
 - ❖ **HTML**
 - ❖ **CSS**
 - ❖ **JavaScript**



本章结束!