

邂逅Web开发

王红元 coderwhy

目录

content



1 电脑配置、推荐软件

2 网站和网页的关系

3 网页的显示过程

4 网页的组成部分

5 浏览器和浏览器内核

电脑配置 – 显示隐藏文件和扩展名

文件夹选项



常规

查看

搜索

文件夹视图



你可以将此视图(如详细信息或图标)应用于这种类型的所有文件夹。

应用到文件夹(L)

重置文件夹(R)

高级设置:

- ☒ 隐藏空的驱动器
- ☒ 隐藏受保护的操作系统文件(推荐)
- ☒ 隐藏文件和文件夹
 - ☐ 不显示隐藏的文件、文件夹或驱动器
 - ☒ 显示隐藏的文件、文件夹和驱动器
- ☒ 隐藏文件夹合并冲突
- ☐ 隐藏已知文件类型的扩展名
- ☐ 用彩色显示加密或压缩的 NTFS 文件
- ☐ 在标题栏中显示完整路径
- ☐ 在单独的进程中打开文件夹窗口
- ☒ 在列表视图中键入时
 - ☒ 在视图中选中键入项
 - ☐ 自动键入到“搜索”框中

还原为默认值(D)

确定

取消

应用(A)

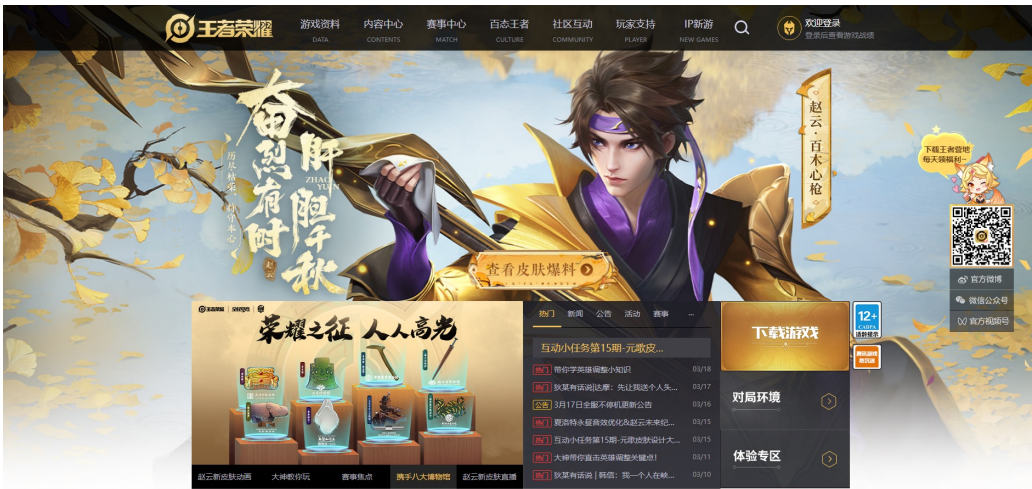
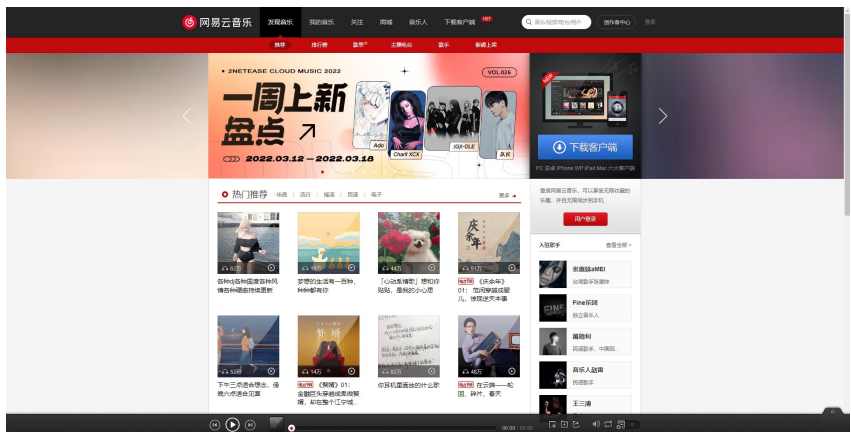
推荐安装的软件

- Chrome浏览器：开发必备浏览器
 - <https://www.google.cn/chrome/index.html>
- VSCode编辑器：开发推荐编辑器（编写代码）
 - <https://code.visualstudio.com/>
- Xmind Zen思维导图：思维导图笔记
 - <https://www.xmind.cn/>
- Typora：markdown笔记软件
 - 见资料分享中

认识网页和网站

■ 什么是网页？

- 网页的专业术语叫做 **Web Page**；
- 打开浏览器**查看到的页面**，是网络中的一“**页**”；
- 网页的内容可以非常丰富：包括**文字、链接、图片、音乐、视频**等等

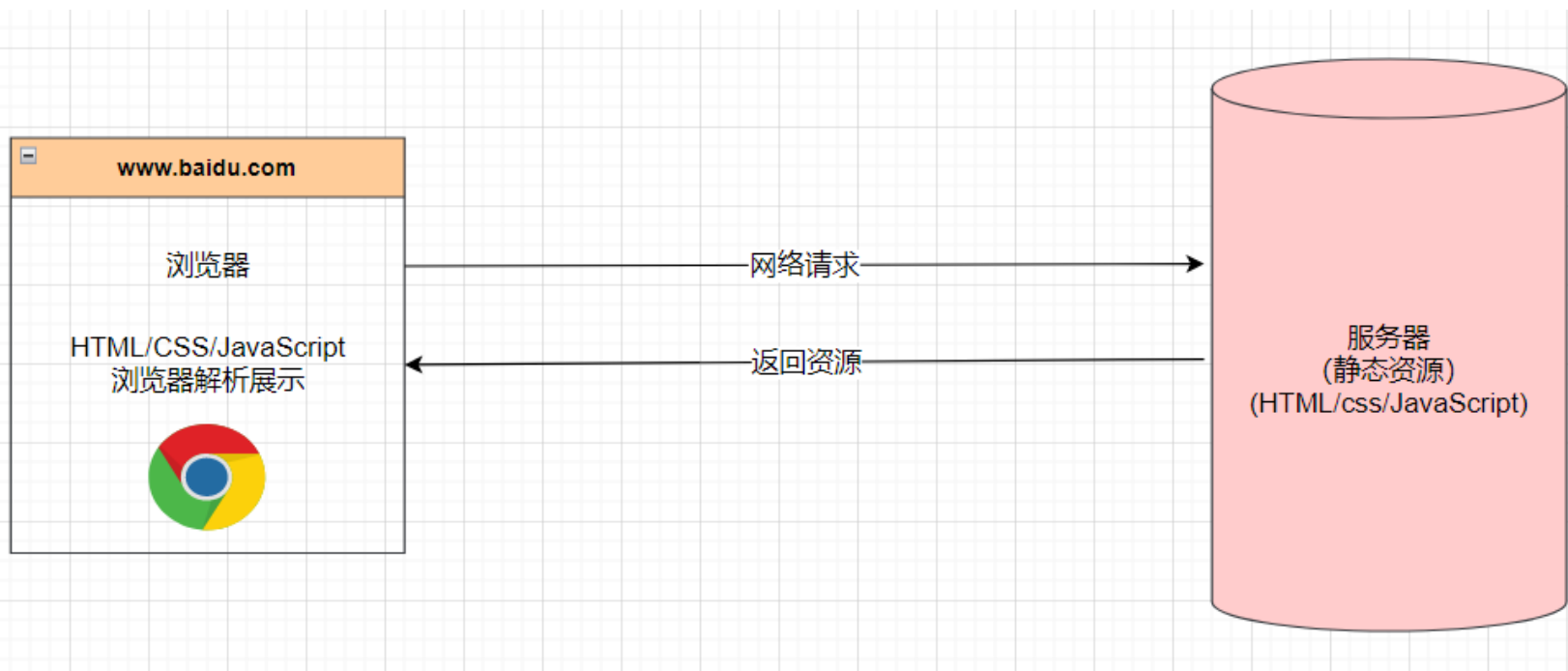


■ 网站是什么呢？

- 网站是由**多个网页组成的**；
- 通常一个网站由N个网页组成 ($N \geq 1$)；

网页的显示过程 – 用户角度

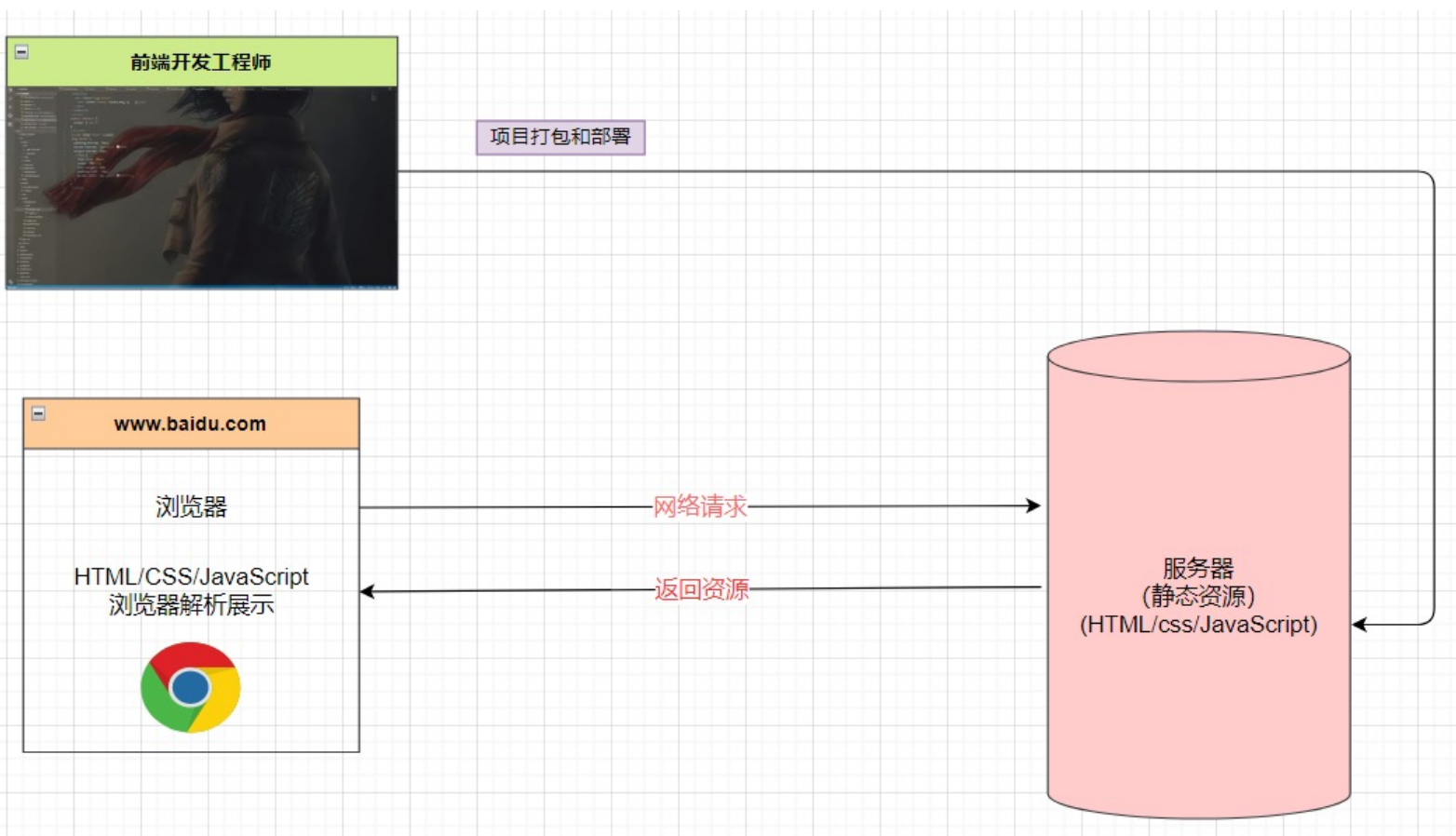
- 1.用户在浏览器输入一个网站；
- 2.浏览器会找到对应的服务器地址，请求静态资源（可以存放在世界上任何一个地方）；
- 3.服务器返回静态资源给浏览器；
- 4.浏览器对静态资源进行解析和展示；



问题：静态资源来自哪里？

网页的显示过程 – 前端工程师

- 1. 开发项目 (HTML/CSS/JavaScript/Vue/React)
- 2. 打包、部署项目到服务器里面



问题：服务器是什么？

服务器是什么？

- 我们日常生活接触到的基本都属于**客户端、前端**的内容：
 - 比如**浏览器、微信、QQ、小程序**；
- 我们知道自己的**手机并不可能存放哪些多的数据和资源**：
 - 比如你用《网易云听音乐》，音乐数据大部分都是存在 **“服务器”** 中的；
- 那么**服务到底是什么呢**？
 - 服务器本质上也是一**台类似于你电脑一样的主机**；
 - 但是这个主机有**几个特点**：
 - ✓ 二十四小时不关机的（稳定运行）；
 - ✓ 没有显示器的；
 - ✓ 一般装的是Linux操作系统（比如centos）；
- 那么我**以后到公司是不是就看得见服务器了呢**？
 - 目前公司大部分用的是**云服务器**（比如阿里云、腾讯云、华为云）；



世界上第一个网页

- 上世纪90年代，**Berners-Lee**上线了世界上第一个网站：

□ <http://info.cern.ch/hypertext/WWW/TheProject.html>

World Wide Web

The WorldWideWeb (W3) is a wide-area [hypermedia](#) information retrieval initiative aiming to give universal access to a large universe of documents.

Everything there is online about W3 is linked directly or indirectly to this document, including an [executive summary](#) of the project, [Mailing lists](#), [Policy](#), November's [W3 news](#), [Frequently Asked Questions](#).

[What's out there?](#)

Pointers to the world's online information, [subjects](#), [W3 servers](#), etc.

[Help](#)

on the browser you are using

[Software Products](#)

A list of W3 project components and their current state. (e.g. [Line Mode](#), X11 [Viola](#), [NeXTStep](#), [Servers](#), [Tools](#), [Mail robot](#), [Library](#))

[Technical](#)

Details of protocols, formats, program internals etc

[Bibliography](#)

Paper documentation on W3 and references.

[People](#)

A list of some people involved in the project.

[History](#)

A summary of the history of the project.

[How can I help?](#)

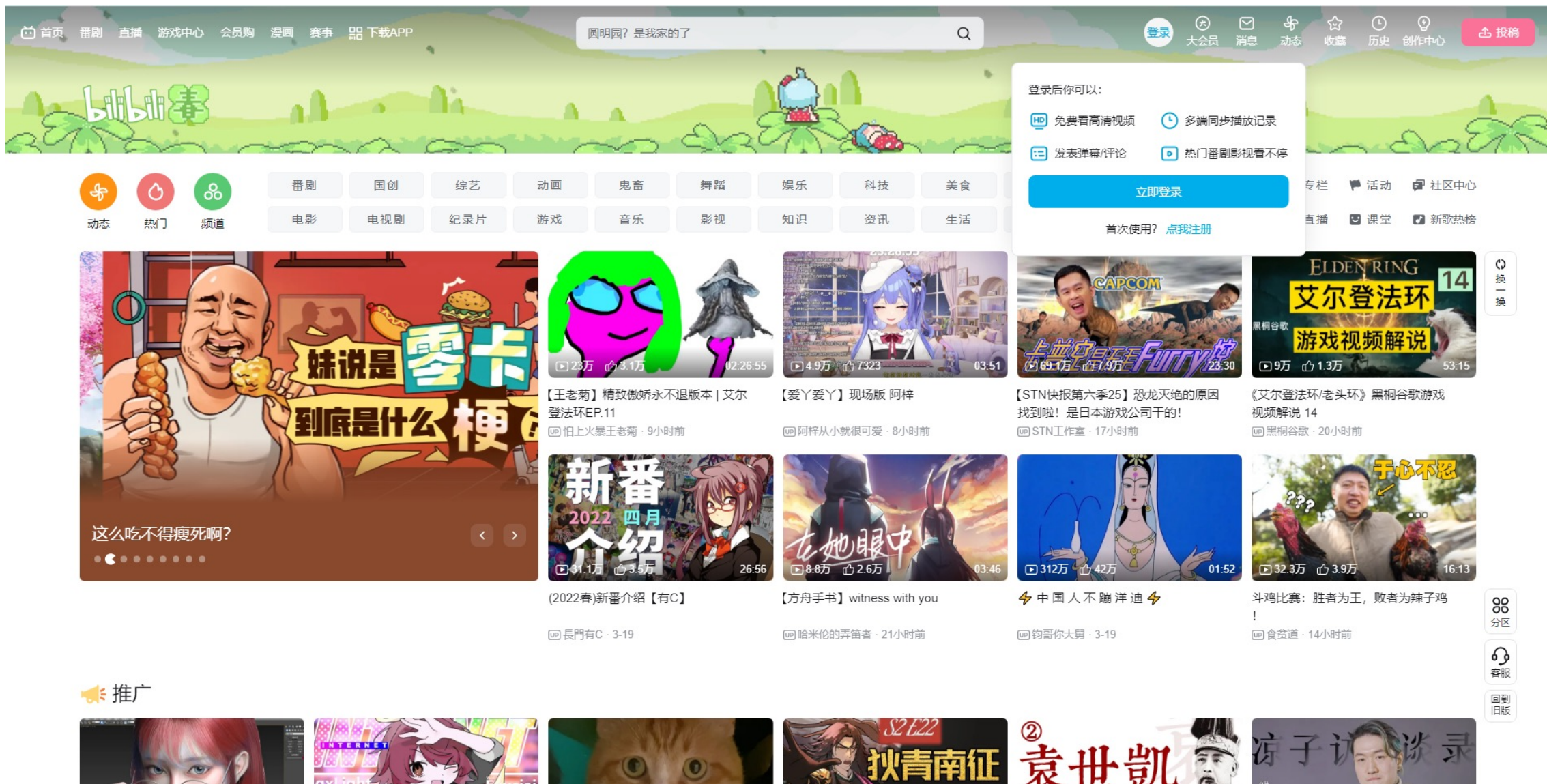
If you would like to support the web..

[Getting code](#)

Getting the code by [anonymous FTP](#), etc.

- 虽然目前我们会认为这个网页简单到不值一提，但是在当时它的发明是“天才之作”。

现代的网页已经非常复杂



网页的组成

■ 那么网页是由什么开发出来的呢？

- 阶段一：HTML元素；
- 阶段二：HTML元素 + CSS样式；
- 阶段三：HTML元素 + CSS样式 + JavaScript语言；



网页的组成

□ HTML

✓ 网页的**内容结构**



□ CSS

✓ 网页的**视觉体验**



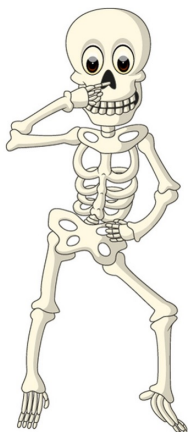
□ JavaScript

✓ 网页的**交互处理**



网页的组成

HTML



CSS



JavaScript



HTML



JS



CSS



网页源代码的角度



DevTools is now available in Chinese!

Always match Chrome's language

Switch DevTools to Chinese

Don't show again



Elements

Console

Sources

Network

Performance

>>

1

```
<style type="text/css">...</style>
```

```
<style type="text/css">...</style>
```

```
<style type="text/css">
```

```
/* 存放验证器全局样式 */
```

```
/* 验证器根容器 */
```

```
.global_container_10f0b * {
```

```
-webkit-box-sizing: border-box;
```

```
box-sizing: border-box;
```

```
font-family: "Helvetica Neue", Helvetica, Arial, "PingFang SC", "Hiragino
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div id="g-topbar" class="g-topbar" style="width: 1053px; top: 0px;">...</div>
```

```
<iframe name="contentFrame" id="g_iframe" class="g-iframe" scrolling="auto" fi
```

```
<script type="text/javascript">...</script>
```

```
<div class="g-btmbar"> == $0
```

```
<div class="m-playbar m-playbar-lock" style="top: -53px; visibility: visible
```

```
<div class="updn">...</div>
```

```
<div class="bg"></div>
```

```
<div class="hand" title="展开播放条"></div>
```

```
<div class="wrap" id="g_player" style="margin-left: -498.5px;">
```

```
<div class="btns">...</div>
```

```
<div class="head j-flag">
```

```
</a>
```

```
</div>
```

```
<div class="play">...</div>
```

CSS

JavaScript

HTML

html body div.g-btmbar

Styles

Computed

Layout

Event Listeners

DOM Breakpoints

Properties

Accessibility

Filter

```
element.style {
```

```
}
```

```
.g-btmbar {
```

```
position: fixed;
```

```
position: absolute;
```

pt_fr

网页开发的角度

JS index.js x JS style.js

src > components > top-ranking > JS index.js > default > HYTopRanking

```
22 return (  
23   <TopRankingWrapper>                                HTML  
24     <div className="header">  
25       <div className="image">  
26         <img src={getSizeImage(info.coverImgUrl)} alt="" />  
27         <a href="/todo" className="image_cover">ranking</a>  
28       </div>  
29       <div className="info">  
30         <a href="/todo">{info.name}</a>  
31         <div>  
32           <button className="btn play sprite_02"></button>  
33           <button className="btn favor sprite_02"></button>  
34         </div>  
35       </div>  
36     </div>  
37     <div className="list">  
38       {  
39         tracks.slice(0, 10).map((item, index) => {  
40           return (  
41             <div key={item.id} className="list-item">  
42               <div className="rank">{index + 1}</div>  
43               <div className="info">  
44                 <span className="name text-nowrap">{item.name}</span>  
45                 <div className="operate">  
46                   <button className="btn sprite_02 play"  
47                     onClick={e => playMusic(item)}></button>  
48                   <button className="btn sprite_icon2 addto"></button>  
49                   <button className="btn sprite_02 favor"></button>  
50                 </div>  
51               </div>  
52             </div>  
53           )  
54         }  
55       }  
56     </div>  
57   )  
58 )
```

```
.header {  
  height: 100px;  
  display: flex;  
  
  margin: 20px 0 0 20px;  
  
  .image {  
    width: 80px;  
    height: 80px;  
    position: relative;  
  
    img {  
      width: 80px;  
      height: 80px;  
    }  
  }  
}
```

```
// props and state  
const { info } = props;           JavaScript  
const { tracks = [] } = info;  
  
// redux hooks  
const dispatch = useDispatch();  
  
// other handle  
const playMusic = (item) => {  
  dispatch(getSongDetailAction(item.id));  
}
```

浏览器的作用

- 我们已经明确知道了网页的组成部分：HTML + CSS + JavaScript。
- 那么这些看起来枯燥的代码，是如何被渲染成多彩的网页呢？
 - 我们知道是通过浏览器来完成；



- 浏览器最核心的部分其实是 “浏览器内核” ；

浏览器的渲染引擎

■ 浏览器最核心的部分是渲染引擎（Rendering Engine），一般也称为“**浏览器内核**”

□ 负责**解析网页语法**，并**渲染（显示）网页**

■ 常见的浏览器有很多：



QQ浏览器



百度



UC浏览器



360浏览器



搜狗浏览器



猎豹浏览器



Chrome



百度浏览器

■ 课堂上必须安装的浏览器：**Chrome浏览器**（所有讲解也会基于这个浏览器）

不同浏览器的内核

■ 常见的浏览器内核有









□ **Trident**（三叉戟）：IE、360安全浏览器、搜狗高速浏览器、百度浏览器、UC浏览器；

□ **Gecko**（壁虎）：Mozilla Firefox；

□ **Presto**（急板乐曲）-> **Blink**（眨眼）：Opera

□ **Webkit**：Safari、360极速浏览器、搜狗高速浏览器、移动端浏览器（Android、iOS）

□ **Webkit** -> **Blink**：Google Chrome

浏览器	微软IE	谷歌Chrome	苹果Safari	火狐
				
内核				

■ 不同的浏览器内核有不同的解析、渲染规则，所以同一网页在不同内核的浏览器中的渲染效果也可能不同。