

Q22. Consider the following HTTP request as received by a web server:

```
GET /about.html HTTP/1.1
Host: www.example.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:108.0) Firefox/108.0
Accept: text/html,application/xhtml+xml,application/xml
Accept-Language: en-GB,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
```

Assuming the request headers can be trusted, which of the following best describes what has happened to cause this request?

- A. Someone attempted to run gzip on the webpage about.html from their Linux terminal.
- B. Someone running the X11 windowing system navigated to `http://www.example.com/about.html`.**
- C. Someone running the Mozilla OS navigated to `www.example.com`.
- D. Several browsers made the same request for about.html at once, forcing the server to keep the connection alive.

Solution: A & D are nonsense, C involves a misreading of the user-agent and ignoring the path component of the GET request. The first lab involved studying these requests.

[1 mark]

解析: 1. 请求行:

- `GET /about.html HTTP/1.1`: 这里是**请求方法 (GET)**、请求的资源路径 (`/about.html`), 以及 HTTP 协议的版本号 (`HTTP/1.1`)。

2. 请求头:

- `Host: www.example.com`: 指定了服务器的主机名。
- `User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:108.0) Firefox/108.0`: 包含了发起请求的用户代理 (浏览器) 的信息。这个请求是由 Firefox 108.0 浏览器在 Linux 系统上发出的。
- `Accept: text/html,application/xhtml+xml,application/xml`: 表示**浏览器可以接受的媒体类型, 按优先级排列**。这里表示浏览器可以接受 **HTML、XHTML 和 XML** 格式的文档。
- `Accept-Language: en-GB,en;q=0.5`: 表示浏览器接受的**语言首选项, 按优先级排列**。这里表示首选英国英语 (`en-GB`), 其次是任何英语 (`en`), 权重为 0.5。
- `Accept-Encoding: gzip, deflate, br`: 表示浏览器**支持的内容编码方式, 按优先级排列**。这里表示浏览器可以接受 `gzip`、`deflate` 和 `br` (Brotli) 压缩。

3. Connection:

`Connection: keep-alive`: 表示客户端**请求保持 TCP 连接打开**, 以便可以发送更多的请求。

Q23. Consider the following HTTP request as received by a web server:

```
GET /hello.txt HTTP/1.1
User-Agent: curl/7.16.3 libcurl/7.16.3 OpenSSL/0.9.71 zlib/1.2.3
Host: www.example.com
Accept-Language: en, mi
```

Assuming the request headers can be trusted, which of the following best describes what has happened to cause this request?

- A. An internal server error has occurred.
- B. Someone accessed `http://www.example.com/hello.txt` with curl**
- C. Someone accessed `www.example.com` in Safari.
- D. The client has made an error in forming the request.

Solution: A & D are nonsense as they would relate to a response, C involves a misreading of the user-agent and ignoring the path component of the GET request. The first lab involved studying these requests.

Q23. You encounter an unfamiliar HTTP status code 413. Without knowing more than this code, which of the below might you reasonably conclude?

- A. The request has completed successfully, with some additional effects.
- B. The server needs to redirect you to another location.
- C. The server has encountered an internal error.
- D. There is an error in the request sent by the client.**

Solution: We covered HTTP status codes and the first-digit meanings in the HTTP lecture (slide 11).

[1 mark]

Q24. You encounter an unfamiliar HTTP status code 307. Without knowing more than this code, which of the below might you reasonably conclude?

- A. The request has completed successfully, with some additional effects.
- B. The server needs to redirect you to another location.**
- C. The server has encountered an internal error.
- D. There is an error in the request sent by the client.

Solution: We covered HTTP status codes and the first-digit meanings in the HTTP lecture (slide 11).

考点: HTTP 通讯的**状态码**

- 1xx: **information** 信息 (100 Continue)
- 2xx: **success** 成功 (200 OK)
- 3xx: **redirect** 重定向 (301 永久移动)
- 4xx: **client error** 客户端错误 (403 forbidden 404 page not found)
- 5xx: **server error** 服务器错误 (500 内部服务器错误)

Q25. Consider this snippet of HTML:

```
<p>
  I'm creating a link to
  <a
    href="https://www.mozilla.org/en-US/"
    title="The best place to find more information about Mozilla's
      mission and how to contribute">the Mozilla homepage</a>.
</p>
```

Which of the below reflects the text you would click on in the browser to follow the link?

- A. The whole paragraph.
- B. The contents of the anchor tag.**
- C. The href attribute.
- D. The title attribute.

Solution: This example comes from the required MDN reading, and answering it only requires understanding how anchor tags work, which is explained in the HTML5 lecture.

问的是**点击哪里会跳转链接**：href 里是链接，title 是鼠标放在上面会显示出来的一个方框，而 anchor tag（缩写为<a>这里的内容）就是会变蓝（点击会跳转!!）的地方。

Q24. Consider this snippet of HTML:

```
<p>
  I'm creating a link to
  <a
    href="https://www.mozilla.org/en-US/"
    title="The best place to find more information about Mozilla's
          mission and how to contribute">the Mozilla homepage</a>.
</p>
```

Which of the below reflects the text you would click on in the browser to follow the link?

- A. "I'm creating a link to the Mozilla homepage."
- B. "the Mozilla homepage"**
- C. "https://www.mozilla.org/en-US/"
- D. "The best place to find more information about Mozilla's mission and how to contribute"

Solution: This example comes from the required MDN reading, and answering it only requires understanding how anchor tags work, which is explained in the HTML5 lecture.

解释：另一种说法，在文本一样的情况下，点击 anchor tag 里的内容，即“the Mozilla homepage”，就可以跳转链接。

Q26. Client-side validation of HTML form fields...

- A. Protects the server from malicious requests.
- B. Provides a helpful aid for users to fill in a form correctly.**
- C. Cannot be disabled in most browsers.
- D. All of the above.

Solution: Client-side validation was part of the required reading in the second session. It provides no protection to the server, and is easily disabled in most browsers.

*有点搞不懂这道题在干什么

ChatGPT 的解释：这段文字讨论了 HTML 表单字段的**客户端验证**。客户端验证是指在用户填写表单时，通过 JavaScript 等前端技术在**用户的浏览器中对输入的数据进行验证**，而**不是在将数据发送到服务器之前**。

A. 保护服务器免受恶意请求。

这是**错误的**。客户端验证只在客户端（即用户的浏览器）执行，无法阻止恶意用户通过手动构造请求绕过验证发送恶意数据到服务器。

B. 为用户正确填写表单提供了有用的帮助。

这是**正确的**。客户端验证可以在用户填写表单时提供实时反馈，帮助用户发现并更正输入错误，从而提高用户体验。

C. 在大多数浏览器中无法禁用。

这是**错误的**。虽然大多数浏览器**默认启用客户端验证**，但用户可以**轻松**地通过禁用 JavaScript 或使用浏览器开发者工具等方法来绕过它。

Q27. Which of the following HTML elements should contain `<meta>` elements?

- A. `<header>`
- B. `<style>`
- C. `<nav>`
- D. None of the above.**

Solution: We covered the fact that metadata goes in the `<head>` in the first HTML lecture, and `<header>` was clearly covered as distinct from `<head>` in the MDN reading.

Q25. Which of the following HTML elements should contain `<meta>` elements?

- A. `<header>`
- B. `<head>`**
- C. `<nav>`
- D. None of the above.

Solution: We covered the fact that metadata goes in the `<head>` in the first HTML lecture. This should also have become apparent in several of the labs.

解析：HTML 的元素考察。

HTML 的 `<head>` tag 里面有： `<meta>`：字符集和视口

HTML 的 `<body>` tag 里面有： `<header>`、`<nav>`、`<div class="content">`和`<footer>`等元素，分别定义了页面的头部、导航、内容和页脚。

Q29. Which of these CSS selectors will apply a style to any `<p>` element that is the direct child of a `<div>` element with the class attribute 'info'?

- A. `div#info p`
- B. `div.info > p`**
- C. `div#info ~ p`
- D. `div.info + p`

Solution: Classes are identified as `div.info`, the `#info` would refer to an element with the ID of 'info', not the class. The '+' would refer to a following element, while '>' relates to a child. This was addressed in the first CSS lecture.

Q26. Which of these CSS selectors will apply a style to one `<p>` element immediately following an `` element with the class attribute 'insert'?

- A. `img#insert p`
- B. `img.insert > p`
- C. `img#insert ~ p`
- D. `img.insert + p`**

解析：选择器 (selector) + 继承 (inherited) 考察

`#info` -> # ID 选择器

`>info` -> > child (子代)

`+info` -> + following (紧跟 info 之后的一第一个 p 元素)

`div.info` 选择的是 class 为 info 的所有 `<div>` 元素。

`div#info` 选择的是 id 为 info 的 `<div>` 元素。

有可能会考： with ID attribute ,那样就是 `div#info` 了

Q30. A style with the selector of #aside, h2, h3 would be applied to:

- A. All elements with the class 'aside', and all second- and third- level headings.
- B. The element with the ID 'aside', and all second- and third- level headings.**
- C. Any third-level headings subordinate to a second-level heading inside an element with the ID 'aside'.
- D. Any third-level headings or second-level headings inside an element with the class 'aside'.

Solution: The comma indicates an 'and' relationship between selectors. Selector syntax was covered in the first CSS lecture.

Q27. A style with the selector of .aside, h2, h3 would be applied to:

- A. All elements with the class 'aside', and all second- and third- level headings.**
- B. The element with the ID 'aside', and all second- and third- level headings.
- C. Any third-level headings subordinate to a second-level heading inside an element with the ID 'aside'.
- D. Any third-level headings or second-level headings inside an element with the class 'aside'.

解析: . 是 class 选择器的同时, #是 id 选择器

Q31. When representing a colour in CSS, the keyword 'red' is equivalent to

- A. #FF0000**
- B. #00F000
- C. #00F00F
- D. #00FF00

Q28. When representing a colour in CSS, the keyword 'green' is equivalent to

- A. #FF0000
- B. #00F000
- C. #00F00F
- D. #00FF00**

解析: RGBA : CSS 上色是前三个 R (red) G (green) B (blue)
alpha 值表示颜色的透明度/不透明度级别。

Q32. You have been tweaking the style of an element so that it has a visible border. You want to create some visual space between the border and the image itself, with twice as much space below as there is above. Which of these rules should you use?

- A. padding: 10px 5px 20px 5px;**
- B. margin: 8px 8px 16px 8px;
- C. padding: 16px 8px 16px 8px;
- D. margin: 5px 5px 5px 10px;

Solution: Two pieces of information allow you to answer this question: you want the padding rather than the margin for space to the content, and the four dimensions are listed clockwise starting from the top. This was covered in the discussion of the box model in the first CSS lecture.

观察 space 是哪里的 space: between the border and the image itself -> padding
Padding 有两种上色方式, 指定 e.g. padding-top or 整个顺时针, 题目里使用了这种。
With twice as much space below as there above -> 下面的举例是上面的两倍。
根据顺时针: ↑ → ↓ ←, 可得 10 5 20 5 符合条件。

Q29. You have been tweaking the style of an element so that it has a visible border. You want to create some visual space between the border and other elements on the page, with twice as much space below as there is above. Which of these rules should you use?

- A. padding: 10px 5px 20px 5px;
- B. margin: 8px 8px 16px 8px;**
- C. padding: 16px 8px 16px 8px;
- D. margin: 5px 5px 5px 10px;

Space 是哪里的: between the border and other elements on the page -> margin 和 padding 做题思路相同。

Q30. Consider the following stylesheet:

```
html { font-size: 14px; }  
p {font-size: 0.8em; }  
div {font-size: 1.2em; }
```

Imagine this stylesheet is applied to a page which contains a <p> element inside a <div> element. What would be the size of the text in the <p> element, as measured in px?

- A. 14
- B. 12
- C. 16.80
- D. 13.44**

Solution: If you understand that the <p> size is defined relative to the <div> that contains it, which modifies the root font-size, then this becomes a simple test of arithmetic. $14 * 1.2 * 0.8 = 13.44$ (though just knowing it is not going to be 14, 12 or 16.8 would land you at the answer).

解析: 0.8em 和 1.2em 都是相对大小, 使用 $14 * 1.2 * 0.8$ 这样做。

Q31. Consider the following CSS rules:

```
.container {  
  display: grid;  
  gap: 20px;  
}  
.listitem {  
  grid-area: 1 / 2 / 3 / 4;  
}
```

Assuming an element corresponding to the second selector is placed in the grid container, on which row will the element end?

- A. 1
- B. 2
- C. 3**
- D. 4

解析: grid container (考查网格布局) 这个网格区域的含义是:

row 行的起始位置: 第 1 行

column 列的起始位置: 第 2 列

行的结束位置: 第 3 行

列的结束位置: 第 4 列

因此, 这个元素跨越了第 1 行到第 3 行, 但仅占据第 2 列到第 4 列。所以这个元素会结束在第 3 行。

Q34. Karen is using `wget` to download a website `example.com`. The index page of `example.com` contains the following HTML:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <link type="text/css" href="custom.css" rel="stylesheet">
  <link type="text/css" href="chrome.css" rel="stylesheet">
  <title>Example Page</title>
</head>
<body>
<h1>Example Page</h1>
<p>See the <a href="/about.html">about</a> page for more about this
site, or check out the <a href="https://www.iana.org/domains/example">
IANA resources</a> for more about example pages.</p>

</body>
</html>
```

If she downloads the page by invoking

```
wget -r -l 1 http://example.com
```

How many files will be retrieved in total?

Answer: 4 (因为这个网页上有四个 href 的直连链接)

Q32. Karen is using `wget` to download a website `example.com`. The index page of `example.com` contains the following HTML:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <link type="text/css" href="style.css" rel="stylesheet">
  <title>Example Page</title>
</head>
<body>
<h1>Example Page</h1>
<p>See the <a href="/about.html">about</a> page for more about this
site, or check out the <a href="https://www.iana.org/domains/example">
IANA resources</a> for more about example pages.</p>

</body>
</html>
```

If she downloads the page by invoking

```
wget -r -l 1 http://example.com
```

How many files will be retrieved in total?

Answer: 3 (因为这个网页上有三个 href 的直连链接)

当你使用 `wget -r -l 1 http://example.com` 命令时，`wget` 会下载指定网址上的页面，并查找这些页面中的 `<a>` 标签（具有 `href` 属性的链接），然后下载这些链接指向的页面。但是，由于指定了 `-l 1` 参数，`wget` 只会下载一级链接，也就是只下载指定网址上的直接链接，不会继续下载链接中的其他链接。

Q33. Cloud computing involves:

- A. Moving data to near-edge devices such as routers.
- B. The on-demand availability of resources such as storage and computational power.**
- C. A distributed architecture making use of a shared hash table to achieve consensus.
- D. All of the above.

Solution: A is fog computing, C is a hazy description of the blockchain. Cloud computing was introduced in the Cloud lecture.

解析：云计算涉及什么

A 选项描述的是边缘计算（fog computing），即将计算、存储和网络资源放置在离用户或数据源更近的位置，而不是云中心。在边缘计算中，数据被处理在接近数据源的地方，而不是传输到远程的云服务器上。

B 选项正确描述了云计算，它是指通过网络按需提供各种计算、存储和软件服务，而无需用户自己拥有或管理物理设备。

C 选项描述的是分布式账本技术（如区块链），它使用共享哈希表来实现分布式共识，而不是云计算的主要特征。

Q35. Which of the following statements is true?

- A. Functions are popped of the call stack after they finish executing.**
- B. Functions remain in the call stack after they finish executing.
- C. Call stacks are not used in JavaScript.
- D. JavaScript uses linked lists for navigating between functions.

陈述 A 是正确的。

在 JavaScript 中，当一个函数执行完成后，它会从调用栈中移除（或“弹出”）。

JavaScript 中确实使用调用栈。它表示函数调用的执行上下文。

JavaScript 不使用链表来在函数之间导航。它使用调用栈来管理函数调用。

Q34. Which of the following statements are true?

1. JavaScript is a interpreted language
 2. Javascript is a compiled language
- A. Both are correct
 - B. Only 2 is correct
 - C. Both are wrong
 - D. Only 1 is correct**

JavaScript→ interpreted

Q36. What will be the output of `console.log(a);`?

```
var a = "John Doe";  
console.log(a.length);
```

- A. 7
- B. 8**
- C. 6
- D. 0

Q36. What will be the output of `console.log(a);`?

```
console.log(a);  
var a = 10;
```

- A. Reference error
- B. Undefined**
- C. 10
- D. a

JavaScript 代码考查，就当他是 java 就好了。

Q35. Which of the following are syntactically correct ways to include Java Script code in your HTML?

- 1 `<script src="file.js"></script>`
- 2 `<script> code </script>`

这两个选项都是将 JavaScript 代码包含在 HTML 中的语法正确的方式：`<script src="file.js"></script>`：这样可以引入名为“file.js”的外部 JavaScript 文件。||
`<script> code </script>`：这样可以直接在 HTML 文件中包含 JavaScript 代码。

Q37. What will be the output of `console.log(a);` in each of the cases?

```
1      function hello(b){  
        console.log(a);  
      }  
      var a = "William";  
      hello();  
  
2      function hello(b){  
        console.log(a);  
        function Name(c){  
          var a = "William";  
        }  
      }  
      hello();
```

- A. William & William
- B. William & Undefined
- C. Undefined & William
- D. William & Reference Error**

看 var 变量在哪里。变量位置不同（全局还是局部）会影响结果不同。

Q38. What will be the output of `console.log();` below?

```
const obj1 = {Name: "John", Age: 20};  
console.log(obj1.Name);
```

- A. Undefined
- B. Reference error
- C. John**
- D. Not defined

Q38. What will be the output of `console.log()`; below?

```
const obj1 = {Name: "John", Age: 20};
const obj2 = {Name: "John", Age: 20};

console.log(obj1===obj2);
```

- A. True
- B. Undefined
- C. False**
- D. None of the above

Q39. What will be the output of the following function?

```
mult = () => {return(10*10);}
```

- A. 100**
- B. Exception will be thrown
- C. Memory leak
- D. Error

Function will work.

 **ChatGPT**

是的，箭头函数的语法确实是这样的，但更具体地说，这个函数工作的原因是箭头函数在语法上是有效的，并且函数体内只有一个表达式。箭头函数的语法如下：

```
javascript Copy code

(parameters) => {
  // 函数体
}
```

在这个例子中，函数体内有一个表达式 `return(10*5)`，它计算了 10 乘以 5 并返回结果。因为箭头函数的函数体只有一个表达式，所以可以省略花括号 `{}` 和 `return` 关键字，直接写表达式。所以，这个函数也可以写成更简洁的形式：

```
javascript Copy code

mult = () => 10 * 5;
```

这样的语法也是有效的，结果与原来的函数相同。箭头函数的这种简洁性使得它成为了 JavaScript 中常用的一种函数定义方式。

 

Q40. `JSON.stringify()` does one of the following.

- A. JSON object to JavaScript object.
- B. JavaScript object to JSON object**
- C. None of the above.
- D. This is not JavaScript methods.

`JSON.stringify()` 方法是将 JavaScript 对象转换为 JSON 字符串的，而不是将 JSON 对象转换为 JavaScript 对象。

在 JavaScript 中，对象是由键值对组成的集合，而 JSON (JavaScript Object Notation) 是一种轻量级的数据交换格式，常用于数据传输和存储。JSON 字符串是 JSON 对象的序列化形式，它由键值对组成，并且键名和字符串都使用双引号引起来。

`JSON.stringify()` 方法用于将 JavaScript 对象转换为符合 JSON 格式的字符串，而不是将 JSON 对象转换为 JavaScript 对象。

Q40. What will be the output of the following code?

```
const data = {
  "name": "John",
  "hobby": "Swimming",
  "language" : ["English", "German", "French"]
}
console.log(data.language[0]);
```

- A. John
- B. Swimming
- C. English**
- D. Prints empty array

Q41. `JSON.parse()` will output?.

```
const string = '{"Name":"Joe","City":"Bristol"}'
const myJSON = JSON.parse(string);
console.log(string);
```

- A. Name:"Joe", City:"Bristol"
- B. "Name": "Joe", "City": "Bristol"**
- C. Name: 'Joe', City: 'Bristol'
- D. None of the above.

貌似想考察 key 和 value 外面必须都得带上双引号 " " .

The output of the IDEA is : `{"Name": "Joe", "City": "Bristol"}`

Q41. What will be the output of the following code?

```
function printArray(a){
  var len = a.length, i=a.length-1;
  if (len==0)
    console.log("Empty Array");
  else
  {
    do
    {
      console.log(a[i]);
      i--;
    }while(i>=0);
  }
}
b = [1,2,3,4,5];
printArray(b);
```

- A. 1,2,3,4,5
- B. Prints 0 to length of the array
- C. 5,4,3,2,1**
- D. Prints empty array

Q42. What will be the output of the following code?

```
const numbers = [1,2,3,4,5];
numbers.forEach((number,index,array) => {
  console.log(index,number);
});
```

- A. 1,2,3,4,5
- B.

1	0	[1, 2, 3, 4, 5]
2	1	[1, 2, 3, 4, 5]
3	2	[1, 2, 3, 4, 5]
4	3	[1, 2, 3, 4, 5]
5	4	[1, 2, 3, 4, 5]
- C.

0	1
1	2
2	3
3	4
4	5
- D. Prints empty array

Q42. How would you refer to the parent class in REACT?

- A. this()
- B. inherits()
- C. self()
- D. super()**

Q44. What happens when you put this in a render() method?

```
let names = ["John","Matt","Joseph"];
function App() {
  return (
    <div className="App">
      {names.map(it => <p>{names[1]}</p>)}
    </div>
  )
}
```

- A.

Matt

- B.

Matt
Matt
Matt
- C. Joseph
- D. None of the above.

Q44. What happens when you put this in a render() method?

```
let names = ["John", "Matt", "Joseph"];  
function App() {  
  return (  
    <div className="App">  
      {names.map(it => <p>{it}</p>)}  
    </div>  
  )  
}
```

A. Displays Nothing

B. John
Matt
Joseph

C. Error

D. None of the above.

//不想看的一道题

Q28. A Thymeleaf application is acting as a frontend to a database of student grades. You see that one method in the Controller.java file has an annotation:

```
@GetMapping("/student/ids")
```

What do you know about the method this annotation is attached to?

- A. It accepts a student ID as a parameter.
- B. It returns a student ID.
- C. The method will not be called because the mapping is malformed.

D. None of the above.

Solution: The get mapping refers to the mapping of the get request to a method, and curly braces would indicate a parameter that the method will be passed. This exact syntax was a key part of the second exercise in the HTML lab.