1. 第 1 题

请写出一个符合 W3C 规范的 HTML 文件，要求

1. 页面标题为「我的页面」
2. 页面中引入了一个外部 CSS 文件，文件路径为 /style.css
3. 页面中引入了另一个外部 CSS 文件，路径为 /print.css，该文件仅在打印时生效
4. 页面中引入了另一个外部 CSS 文件，路径为 /mobile.css，该文件仅在设备宽度小于 500 像素时生效
5. 页面中引入了一个外部 JS 文件，路径为 /main.js
6. 页面中引入了一个外部 JS 文件，路径为 /gbk.js，文件编码为 GBK
7. 页面中有一个 SVG 标签，SVG 里面有一个直径为 100 像素的圆圈，颜色随意

```

<!DOCTYPE html>

<html lang="zh-CN">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<link rel="stylesheet" href="/style.css">

<link rel="stylesheet" href="/print.css" media="print">

<link rel="stylesheet" href="./mobile.css" media="screen and (max-device-width:500px)">

<script scr='/main.js'></script>

<script scr='/gbk.js' charset="GBK"></script>

<title>我的页面</title>

</head>

<body>

<svg version='1.1' xmlns='http://www.w3.org/2000/svg'>

<circle cx='150' cy='75' r='50'></circle>

</svg>

</body>

</html>

```

1. 第 2 题

[2016年腾讯前端面试题](https://github.com/Bless-L/MyBlog/blob/master/post/2016%E8%85%BE%E8%AE%AF%E5%AE%9E%E4%B9%A0%E7%94%9F%E5%89%8D%E7%AB%AF%E9%9D%A2%E8%AF%95%E7%BB%8F%E5%8E%86%E5%8F%8A%E6%80%BB%E7%BB%93%EF%BC%88%E4%BA%8C%EF%BC%89.md)：  
移动端是怎么做适配的？

回答要点：

1. meta viewport
2. 媒体查询（[CSS深入浅出第五课](https://xiedaimala.com/courses/003b1951-22af-4821-ad80-d2880c0074eb/)）
3. 动态 rem 方案（[CSS深入浅出第九课](https://xiedaimala.com/courses/003b1951-22af-4821-ad80-d2880c0074eb/)）

这三个知识点只有第一个我有讲过，你需要自学第二点和第三点。  
如果大家在这道题上都答得不好，我会在考试后直播这两节课的录屏。（因为我觉得视频里已经讲得很好了，没必要重新讲一遍）

1.通过设置meta的viewport,根据不同设备的宽度，不允许手机自动缩放，这样手机访问页面才不会缩小。

```

<meta name="viewport" content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">

```

2.通过写css的媒体查询，查询窗口不同宽度加载不同的样式比如:

```

@media (max-width:375px){

body{

background: red;

}

}

@media (min-width:375px){

body{

background: blue;

}

}

```

3.写长度单位运用rem，用JS获取设备宽度，来改变html的font-size，然后根据设备宽度进行同比例放大缩小，这样就可以适配不同的移动设备，写css也可以运用scss来写，这样直接写px。通过scss来转换成rem

1. 第 3 题

[2017年腾讯前端实习面试题（二面）](https://earthsplitter.github.io/2017/03/31/2017%E8%85%BE%E8%AE%AF%E5%AE%9E%E4%B9%A0%E7%BB%8F%E9%AA%8C%E6%80%BB%E7%BB%93/)：  
用过CSS3吗? 实现圆角矩形和阴影怎么做?

```

<style>

.box {

width: 100px;

height: 50px;

border: 1px solid red;

border-radius: 10px 10px;

box-shadow: 2px 2px 5px rgba(0, 0, 0, 0.5);

}

</style>

```

1. 第 4 题

出处同上（一面二面都问了）：  
什么是闭包，闭包的用途是什么？

我在课堂上讲过闭包是什么，也演示过闭包的用途，但是并没有帮大家总结。  
请自行搜索。

闭包：闭包就是能够读取其他函数内部变量的函数。

闭包的用途：1.可以调用函数内部的变量2.让这些变量的值始终保持在内存中，不被清除。

1. 第 5 题

出处同上：

call、apply、bind 的用法分别是什么？出处同上：

call、apply、bind 的用法分别是什么？

函数.call(对象,arg1,arg2....)

函数.apply(对象，[arg1,arg2,...])

var ss=函数.bind(对象,arg1,arg2,....)。ss();

功能上是没有区别的，都是改变this的指向，它们的区别主要是在于方法的实现形式和参数传递上的不同。

call和apply的this指向他们的第一个参数，apply的第二个参数是一个参数数组，call的第二个及其以后的参数都是数组里面的元素，都是直接执行。

bind会创建一个新函数，称为绑定函数，当调用这个函数的时候，绑定函数会以创建它时传入bind（）方法的第一个参数作为this，传入bind（）方法的第二个及以后的参数加上绑定函数运行时本身的参数按照顺序作为原函数的参数来调用原函数。

1. 第 6 题

出处同上：  
请说出至少 8 个 HTTP 状态码，并描述各状态码的意义。

例如：

状态码 200 表示响应成功。

200表示响应成功

201表示请求成功并且服务器创建了新的资源

202表示服务器已接受请求，但尚未处理

204服务器成功处理了请求，没有返回任何内容

301表示请求的网页已永久移动到新位置,会自动将请求者转到新位置。

400服务器不理解请求的语法

401未授权，请求要求身份验证。 对于需要登录的网页，服务器可能返回此响应。

403服务器已经理解请求，但拒绝执行

404表示请求失败，请求所希望得到的资源未被在服务器上发现

500服务器遇到错误，无法完成请求

503服务不可用，服务器目前无法使用

1. 第 7 题

出处同上：  
请写出一个 HTTP post 请求的内容，包括四部分。  
其中  
第四部分的内容是 username=ff&password=123  
第二部分必须含有 Content-Type 字段  
请求的路径为 /path

POST /path HTTP/1.1

Host : www.baidu.com

User-Agent: curl/7.54.0

Accept: *\*/\**

Content-Length: 24

Content-Tyoe: application/x-www-form-urlencoded

username=ff&password=123

1. 第 8 题

请说出至少三种排序的思路，这三种排序的时间复杂度分别为

1. O(n\*n)
2. O(n log2 n)
3. O(n + max)

冒泡排序：比较两个相邻的元素，排序错误的将换位，接着下一对进行排序（从后往前），排序完一轮后将确定一个最大或最小值，剩下的元素再进行排位，直到全部位置确定了。

选择排序：前一个元素和后一个元素进行排序，若后一个小于前一个，则记录后一个元素下标，后一个元素成为前一个元素再和下一个元素比较(从前往后）记录小的小标，直到比完一轮，得到了最小值的小标，然后和第一个元素换位，得到最小值。剩下元素开始下一轮比较。

堆排序：堆排序是一种树形选择排序，在排序过程中，将A[n]看成是完全二叉树的顺序存储结构，利用完全二叉树中双亲结点和孩子结点之间的内在关系来选择最小的元素。

插入排序：将一个元素插入到已经排好序的有序元素中，从而得到一个新的、个数加一的有序元素，以此类推所有元素都拍好。（起扑克牌法）

快速排序：选取一个基准元素，通过一趟排序将要排序的元素分割成独立的两部分，其中一部分的所有数据都比另外一部分的所有数据都要小，然后再按此方法，选取一个基准元素，对这两部分数据分别进行快速排序。

计数排序：对数组的所有元素数量进行计数，存到一个新的数组里，然后进行直接进行排序。

1. 第 9 题

著名前端面试题：  
一个页面从输入 URL 到页面加载显示完成，这个过程中都发生了什么？

这一题是在挖掘你的知识边界，所以你知道多少就要答多少。

可以先查阅一些资料再查，但是不要把自己不懂的东西放在答案里，面试官会追问的。

0.输入URL

1.解析URL

2.构造并发送HTTP请求

3.HTTP报文传输过程

4.服务器接受并处理HTTP报文

5.服务器构造并发送响应报文（传输过程略）

6.浏览器接收报文，并开始构建页面

7.（可选）浏览器发送静态资源请求（处理过程略）

8.（可选）浏览器发送Ajax请求（处理过程略）

9.页面构建完成

1. 第 10 题

著名面试题：  
如何实现数组去重？  
假设有数组 array = [1,5,2,3,4,2,3,1,3,4]  
你要写一个函数 unique，使得  
unique(array) 的值为 [1,5,2,3,4]  
也就是把重复的值都去掉，只保留不重复的值。

要求：

1. 不要做多重循环，只能遍历一次
2. 请给出两种方案，一种能在 ES 5 环境中运行，一种能在 ES 6 环境中运行（提示 ES 6 环境多了一个 Set 对象）

```

var array = [1,5,2,3,4,2,3,1,3,4] //ES5

function unique1(array) {

var newArray = []

for(var i = 0; i < array.length; i++) {

var number = array[i]

if(array[i] in newArray) {

continue

}

newArray.push(array[i])

}

return newArray

}

function unique2(array){ //ES5

var newArray = []

array.forEach((x,y)=>{

if(!(array[y] in newArray)) {

newArray.push(x)

}

})

return newArray

}

let mySet = new Set([1,5,2,3,4,2,3,1,3,4]) //ES6

function unique3() {

let a = []

mySet.forEach((x)=>{a.push(x)})

return a

}

```