**员工转正考试试题——前端工程师**

姓名： 公司： 岗位： 成绩：

1. **填空题（每空2.5分，共计50分）**

1、次数：每天指纹考勤（外出时，考勤软件）（ 4 ）次。

2、作息时间：夏季：（ 8:00-12:00 14：00-18：00；）冬季：（8:00-12:00 13：30-17：30）体变更时间以公司通知为准。

3、公司设立全勤奖制度，每人每月50元。一个月内，迟到、早退、忘打卡累计（ 3）次（含）之内；无事假、病假、旷工视为全勤，享受全勤奖50元。

4、迟到、早退五次之内，每次扣款（ 10）元；超过五次的，每次扣（ 20 ）元；

5、忘打卡（ 3 ）次之内不扣款，超过（ 3 ）次每次扣10元；超过五次的，每次扣20元；

6、因工作需要无考勤记录，如因特殊情况无法提前申请，第二个工作日内补办相关流程，逾期申请的按（ 忘打卡 ）处理。

7、（ A ）类请假：不可抗力或直系亲属必须缘由请假。如：本人生病；家中直系亲属生病必要的看病、照顾时间；家中紧急重要事宜处理等，能提供对应的证明。（ B ）类请假：因个人原因或非直系亲属缘由请假。如：表兄弟姐妹、朋友结婚、满月宴；旁系亲属、朋友生病探望；考试（和工作职称相关或驾照）；可说明正当理由的其他请假。能提供对应的证明。（ C ）类请假：拒绝说明理由或理由极不充分且不能提供证明等请假。

8、员工（含管理人员）未经请假不到岗、不符合请假要求或离职交接未办理完毕就离岗的，记为（ 旷工）；

9、未经批准离岗2小时以内（含），记为（ 半天 ）旷工；2小时以上，记为（ 一 ）天旷工；

10、外勤打卡：宁津，临邑，禹城等100公里之内区域，要求上午上班到达时间（ 9：00 ），乐陵，齐河，庆云等100公里之外区域，要求上午上班到达时间（ 9：30）。

11、工龄：工龄以员工实际入职公司的当日计算，满周年的次月开始计算工龄（ 30 ）元/月；每年按照（ 30）元递增，集团内调动工龄连续计算；工龄补助最高发放（ 10 ）年。

12**、**社保费用缴纳：事、病假超（ ）天（含）及有旷工的员工，费用由个人全部承担。

1. **简答题（50分）**
2. get和post请求的区别？（15分）

答：

* GET在浏览器回退时是无害的，而POST会再次提交请求。
* GET产生的URL地址可以被Bookmark，而POST不可以。
* GET请求会被浏览器主动cache，而POST不会，除非手动设置。
* GET请求只能进行url编码，而POST支持多种编码方式。
* GET请求参数会被完整保留在浏览器历史记录里，而POST中的参数不会被保留。
* GET请求在URL中传送的参数是有长度限制的，而POST么有。
* 对参数的数据类型，GET只接受ASCII字符，而POST没有限制。
* GET比POST更不安全，因为参数直接暴露在URL上，所以不能用来传递敏感信息。
* GET参数通过URL传递，POST放在Request body中。

HTTP的底层是TCP/IP。所以GET和POST的底层也是TCP/IP，也就是说，GET/POST都是TCP链接。GET和POST能做的事情是一样一样的。你要给GET加上request body，给POST带上url参数，技术上是完全行的通的。浏览器通常都会限制url长度在2K个字节，而（大多数）服务器最多处理64K大小的url。超过的部分，恕不处理。如果你用GET服务，在request body偷偷藏了数据，不同服务器的处理方式也是不同的，有些服务器会帮你卸货，读出数据，有些服务器直接忽略，所以，虽然GET可以带request body，也不能保证一定能被接收到哦。

对于GET方式的请求，浏览器会把http header和data一并发送出去，服务器响应200（返回数据）；

而对于POST，浏览器先发送header，服务器响应100 continue，浏览器再发送data，服务器响应200 ok（返回数据）。

1. 简述在ionic项目中var 定义的变量与$scope定义的变量有什么区别。（20分）

答： var定义的变量会有变量提升，var不能定义动态变量名，$scope可以定义动态变量名，

  for(let i=1;i<7;i++){   //ionic动态命名变量

            var res = 'res'+i;

            $scope[res] = i;

            console.log("$scope:",res);

        }

1. 请列举在前端开发中，编写代码有哪些基本规范，使你编写的代码可读性更高。（15分）

答：**A.基本原则**

符合web标准（UTF-8，HTML5），语义化html（HTML5新增要求，减少div和span等无特定语义的标签使用），结构表现行为分离（HTML-CSS-JS代码分离，不同行为代码高内聚低耦合），兼容性优良（早期版本浏览器兼容，移动端和PC端设备兼容）.页面性能方面（减少请求次数，例如使用精灵图和sass语法），代码要求简洁明了有序，尽可能的减小服务器负载，保证最快的解析速度（减小repaint和reflow）.

**B.文件命名规范**

1、html，css，js，lib，images文件均存放至项目的目录中。如果使用相关前端框架，根据框架的文件格式进行合理布局。

2、所有文件夹及文件使用英文命名（避免使用中文路径）。

3、html文件：入口文件使用index.html。如果有对应的设计组设计原稿，需要将对应的设计稿和html文件命名一致并合理存放。

4、css文件命名：后缀.css。通用initial.css，初始化base.css，首页index.css，其他页面按照对应的html命名。

5、Js文件命名：英文命名，后缀.js.通用common.js，初始化base.js。 其他页面按照对应的html命名。

**C.HTML规范**

1、文档类型声明及编码：统一为html5声明类型。书写时利用IDE实现层次分明的缩进（默认缩进4空格）。

2、非特殊情况下CSS文件放在body部分<meta>标签后。非特殊情况下大部分JS文件放在<body>标签尾部（如果需要界面未加载前执行的代码可以放在head标签后）避免行内JS和CSS代码。

3、所有编码需要遵循html（XML）标准，标签&属性&属性命名必须由小写字母及下划线数字组成，且所有标签必须闭合，包括br()，hr()等。属性值用双引号。

4、引入JS库文件，文件名须包含库名称及版本号及是否为压缩版，比如jquery-1.4.1.min.js。引入插件，文件名格式为库名称+插件名称，比如jQuery.bootstrap.js。

5、书写页面过程中，请考虑向后扩展性。class&id参见css书写规范.

6、需要为html元素添加自定义属性的时候，首先要考虑下有没有默认的已有的合适标签去设置，如果没有，可以使用须以"data-"为前缀来添加自定义属性，避免使用"data："等其他命名方式。

7、语义化html，如标题根据重要性用h\*(同一页面只能有一个h1)，段落标记用p，列表用ul，内联元素中不可嵌套块级元素。

8、尽可能减少div多层级嵌套。

9、书写链接地址时，必须避免重定向，例如：href="http：//myVue.com/"，即须在URL地址后面加上“/”；

10、在页面中尽量避免使用style属性，即style="…"。

11、必须为含有描述性表单元素(input，textarea)添加label，如姓名：须写成：姓名：

12、能以背景形式呈现的图片，尽量写入css样式中。

13、重要图片必须加上alt属性。给重要的元素和截断的元素加上title。

14、给区块代码及重要功能(比如循环)加上注释，方便后台添加功能。

15、特殊符号使用：尽可能使用代码替代：比如<(<)&>(>)&空格()&»(»)等等。

**D.CSS规范**

1、编码规范为utf-8。

2、协作开发及分工：i会根据各个模块，同时根据页面相似程序，事先写\*\*体框架文件，分配给前端人员实现内部结构&表现&行为。共用css文件base.css由i书写，协作开发过程中，每个页面请务必都要引入，此文件包含reset及头部底部样式，此文件不可随意修改。

3、class与id的使用：id是唯一的并是父级的，class是可以重复的并是子级的，所以id仅使用在大的模块上，class可用在重复使用率高及子级中。id原则上都是由我分发框架文件时命名的，为JS预留钩子的除外。

4、为JS预留钩子的命名，请以js\_起始，比如：js\_hide，js\_show。

5、class与id命名：大的框架命名比如header/footer/wrapper/left/right之类的在2中由i统一命名.其他样式名称由小写英文&数字&来组合命名，如i\_comment，fontred，width200。避免使用中文拼音，尽量使用简易的单词组合。总之，命名要语义化，简明化

6、规避class与id命名(此条重要，若有不明白请及时与i沟通)：a）通过从属写法规避，示例见d。b）取父级元素id/class命名部分命名，示例见d。c）重复使用率高的命名，请以自己代号加下划线起始，比如i\_clear。d）a，b两条，适用于在2中已建好框架的页面，如，要在2中已建好框架的页面代码中加入新的div元素，按a命名法则：...，样式写法：#mainnav.firstnav{.......}按b命名法则：...，样式写法：.main\_firstnav{.......}

7、css属性书写顺序，建议遵循：布局定位属性-->自身属性-->文本属性-->其他属性.此条可根据自身习惯书写，但尽量保证同类属性写在一起.属性列举：布局定位属性主要包括：display&list-style&position（相应的top，right，bottom，left）＆float&clear＆visibility＆overflow；

自身属性主要包括：width&height&margin&padding&border&background。

文本属性主要包括：color&font&text-decoration&text-align&vertical-align&white-space&

其他&content。

8、书写代码前，提高样式重复使用率。

9、充分利用html自身属性及样式继承原理减少代码量，例如：

即可实现日期居右显示

1. ul.list li {
2. position：relative
3. }
4. ul .list li span {
5. position：absolute；right：0
6. }

10、样式表中中文字体名，请务必转码成unicode码，以避免编码错误时乱码。

11、背景图片请尽可能使用精灵图技术，减小http请求，考虑到多人协作开发，精灵图按模块制作。

12、使用table标签时(尽量避免使用table标签)，请不要用width/height/cellspacing/cellpadding等table属性直接定义表现，应尽可能的利用table自身私有属性分离结构与表现，如thead，tr，th，td，tbody，tfoot，colgroup，scope。(cellspaing及cellpadding的css控制方法：table{border：0。margin：0。border-collapse：collapse。}tableth，tabletd{padding：0。}，base.css文件中我会初始化表格样式)

13、杜绝使用兼容ie8。

14、用png图片做图片时，要求图片格式为png-8格式，若png-8实在影响图片质量或其中有半透明效果，请为ie6单独定义背景：\_background：none。\_filter：progid：DXImageTransform.Microsoft.AlphaImageLoader(sizingMethod=crop，src=’img/bg.png’)。

15、避免兼容性属性的使用，比如text-shadow||css3的相关属性。

16、减少使用影响性能的属性，比如position：absolute||float。

17、必须为大区块样式添加注释，小区块适量注释。

18、代码缩进与格式：建议单行书写，可根据自身习惯，后期优化会统一处理。

**E. JS书写规范**

1、文件编码统一为utf-8，书写过程过，每行代码结束必须有分号。原则上所有功能均根据XXX项目需求原生开发，以避免网上down下来的代码造成的代码污染(沉冗代码||与现有代码冲突||...)。

2、库引入：原则上仅引入jQuery库，若需引入第三方库，须与团队其他人员讨论决定。

3、变量命名：驼峰式命名.原生JS变量要求是纯英文字母，首字母须小写，如myVue。jQuery变量要求首字符为'\_'，其他与原生JS规则相同，如：\_myVue。另，要求变量集中声明，避免全局变量.

4、类命名：首字母大写，驼峰式命名.如MyVue。

5、函数命名：首字母小写驼峰式命名.如myVue()。

6、命名语义化，尽可能利用英文单词或其缩写。

7、尽量避免使用存在兼容性及消耗资源的方法或属性，比如eval\_r()&innerText。

8、后期优化中，JS非注释类中文字符须转换成unicode编码使用，以避免编码错误时乱码显示。

9、代码结构明了，加适量注释.提高函数重用率。

10、注重与html分离，减小reflow，注重浏览器性能.

F.图片规范

1、所有页面元素类图片均放入img文件夹，测试用图片放于demo文件夹。

2、图片格式gif/png/jpg。提倡使用webp文件格式，使用软件进行图片压缩。

3、命名全部用小写英文字母||数字||\_的组合，其中不得包含汉字||空格||特殊字符；尽量用易懂的词汇，便于团队其他成员理解。另，命名分头尾两部分，用下划线隔开，比如ad\_left01.gif||btn\_submit.gif。

4、在保证视觉效果的情况下选择最小的图片格式与图片质量，以减少加载时间。5、尽量避免使用半透明的png图片(若使用，请参考css规范相关说明)。

6、运用css精灵图技术集中小的背景图或图标，减小页面http请求，但注意，请务必在对应的精灵图psd源图中划参考线，并保存至img目录下

G.测试规范

开发及测试工具约定建议使用Aptana-Sublime-Vim-WebStrom，亦可根据自己喜好选择，但须遵循如下原则：

1、不可利用IDE的视图模式'画'代码。

2、不可利用IDE生成相关功能代码。

3、编码必须格式化，比如缩进。测试工具：前期开发仅测试FireFox&IE6&IE7&IE8，后期优化时加入Opera&Chrome&Safari。建议测试顺序：FireFox-->IE-->Opera-->Chrome-->Safari。