

Web前端开发高级模拟测试（十）

68分

共54道题 总分：200分 形考总分：0分

形考：0分

单选题 多选题 判断题 客观填空题

一、单选题 共30题，60分

1

2分

nodejs的哪个模块可以搭建服务器？

- A Http
- B url
- C path
- D fs

作答结果：正确

2

2分

关于 javascript 模块化，下列描述错误的是（）。

- A AMD推崇依赖前置，CMD推崇依赖就近
- B Sea.js遵循AMD规范，RequireJS遵循CMD规范
- C 主流的模块化包括CommonJS,AMD,CMD等
- D 模块化有利于管理模块间的依赖，更依赖模块的维护

作答结果：正确

3

2分

关于html5说法错误的是()

- A canvas是HTML中你可以绘制图形的区域
- B SVG表示可缩放矢量图形
- C queryselector的功能类似于jquery的选择器
- D queryString是HTML5查找字符串的新方法

作答结果：正确

单选题
(30题，60分)

| | | | |
|----|----|----|----|
| 1 | 2 | 3 | 4 |
| 5 | 6 | 7 | 8 |
| 9 | 10 | 11 | 12 |
| 13 | 14 | 15 | 16 |
| 17 | 18 | 19 | 20 |
| 21 | 22 | 23 | 24 |
| 25 | 26 | 27 | 28 |
| 29 | 30 | | |

多选题
(15题，30分)

| | | | |
|----|----|----|----|
| 1 | 2 | 3 | 4 |
| 5 | 6 | 7 | 8 |
| 9 | 10 | 11 | 12 |
| 13 | 14 | 15 | |

判断题
(5题，10分)

| | | | |
|---|---|---|---|
| 1 | 2 | 3 | 4 |
|---|---|---|---|

正确

错误

批阅

未做

4

2分

JavaScript实现继承的方式，不正确的是：

- ☐ (A) 原型链继承
- ☐ (B) 构造函数继承
- ☒ (C) 组合继承
- ☐ (D) 关联继承

作答结果：错误

5

2分

下面的代码的输出是结果为： <?php \$s = '12345'; \$s[\$s[1]] = '2'; Echo \$s; ?>

- ☐ (A) 12345
- ☐ (B) 11345
- ☐ (C) 22345
- ☒ (D) 12245

作答结果：正确

6

2分

关于关键字const，下列说法错误的是：（）

- ☐ (A) 用于声明常量，声明后不可修改
- ☐ (B) 不会发生变量提升现象
- ☐ (C) 不能重复声明同一个变量
- ☒ (D) 可以先声明，不赋值。

作答结果：正确

7

2分

以下关于video说法正确的是（）

- ☐ (A) 当前，video元素支持三种视频格式，其中WebM=带有Theora视频编码和Vorbis音频编码的WebM文件。
- ☐ (B) source元素可以添加多个，具体播放哪个由浏览器决定。
- ☐ (C) video内使用img展示有视频封面
- ☒ (D) loop属性可以使媒介文件循环播放

作答结果：正确

vue生命周期中，DOM 渲染在哪个周期中就已经完成

8

2分

- ☐ (A) created
- ☒ (B) mounted
- ☐ (C) updated
- ☐ (D) beforeDestroy

作答结果：正确

9

2分

以下代码的运行结果是 () var obj={ a:1, b:function(){ console.log(this.a) } } var fun=obj.b; fun();

- ☐ (A) a
- ☐ (B) 1
- ☒ (C) undefined
- ☐ (D) 什么也看不到

作答结果：正确

10

2分

关于字符串扩展的新函数，下面描述错误的是： ()

- ☒ (A) includes函数用于判断字符串中是否含有指定的子字符串
- ☐ (B) repeat函数将目标字符串重复N次，目标字符串被修改
- ☐ (C) startsWidth函数判断指定的子字符串是否出现在目标字符串头部位置
- ☐ (D) endWidth函数判断指定的子字符串是否出现在目标字符串尾部位置

作答结果：错误

11

2分

关于html5说法错误的是()

- ☐ (A) canvas是HTML中你可以绘制图形的区域
- ☐ (B) SVG表示可缩放矢量图形
- ☐ (C) queryselector的功能类似于jquery的选择器
- ☒ (D) queryString是HTML5查找字符串的新方法

作答结果：正确

12

+new Array(017) 输出 ()

2分

- ☐ A 17
- ☒ B NaN
- ☐ C 15
- ☐ D error

作答结果：正确

13

2分

在vue中（）让CSS只在当前组件中起作用？

- ☐ A style
- ☐ B type
- ☒ C scoped
- ☐ D 以上都不是

作答结果：正确

14

2分

JavaScript实现继承的方式，不正确的是：

- ☐ A 原型链继承
- ☐ B 构造函数继承
- ☒ C 组合继承
- ☐ D 关联继承

作答结果：错误

15

2分

php 关于类继承说法正确的是

- ☐ A 子类可以直接调用基类的所有方法
- ☒ B 子类可以重写基类的方法
- ☐ C 子类可以直接继承多个基类
- ☐ D 子类可以重写基类 protected 级别方法为 private 级别

作答结果：正确

16

2分

关于对象的扩展，错误的是：（）

- ☐ A ES6中对象的表示法更加简洁
- ☐ B 对象的属性名可以是表达式
- ☒ C 对象的方法名不可以是表达式

☒ D 当属性名为表达式的时候，需要使用[]标识

作答结果：正确

17

2分

在vue中，关于keep-alive说法正确的是（）

- ☐ A 在使用时，需要用import先引入该组件
- ☐ B 是一个自定义的组件
- ☐ C 没有特殊的意义

☒ D <keep-alive></keep-alive> 包裹动态组件时，会缓存不活动的组件实例，主要用于保留组件状态或避免重新渲染

作答结果：正确

18

2分

jquery-mobile中“ui-content”类的作用是（）

- ☐ A 在页面顶部创建的工具条
- ☐ B 定义页面的内容
- ☒ C 在页面添加内边距和外边距
- ☐ D 创建页面底部工具条

作答结果：正确

19

2分

HTML5不支持的视频格式是（）

- ☒ A ogg
- ☐ B mp4
- ☐ C flv
- ☐ D webm

作答结果：错误

20

2分

关于关键字const，下列说法错误的是：（）

- ☐ A 用于声明常量，声明后不可修改
- ☐ B 不会发生变量提升现象
- ☒ C 不能重复声明同一个变量
- ☐ D 可以先声明，不赋值。

作答结果：错误

21

2分

关于数值的扩展, `window.isNaN("abc")` 和 `Number.isNaN("abc")`的结果分别是: ()

- ☒ A true false
- ☐ B false true
- ☐ C true true
- ☐ D false false

作答结果: 正确

22

2分

`sessionStorage`保存数据的方法正确的是 ()

- ☒ A `sessionStorage.setItem("键名","键值");`
- ☐ B `sessionStorage.saveltem("键名","键值");`
- ☐ C `sessionStorage.setItem("键值","键名");`
- ☐ D `sessionStorage.saveltem("键值","键名");`

作答结果: 正确

23

2分

关于Iterator遍历器的说法, 错误的是: ()

- ☐ A `next()`方法是Iterator遍历器的核心
- ☒ B 当`next()`返回对象的`done`属性为`false`, 遍历结束
- ☐ C 具有Iterator接口的对象视为可遍历对象
- ☐ D 可以自定义一个可遍历对象和其遍历行为

作答结果: 正确

24

2分

`XMLHttpRequest`对象的`status`的值, 当`status==404`是指什么 ()

- ☐ A 服务器端错误
- ☐ B url重定向
- ☒ C 客户端
- ☐ D 服务器端不知道客户端要请求哪个资源
- ☐ E 没有任何意义

作答结果: 正确

25

2分

在nodejs中，用来引入其它模块的函数是？

- ☐ (A) include
- ☒ (B) require
- ☐ (C) import
- ☐ (D) get

作答结果：正确

26

2分

下述有关HTML Doctype的描述，说法错误的是（）

- ☐ (A) 他的责任就是告诉浏览器文档使用哪种HTML或XHTML规范
- ☐ (B) 不同文档模式主要影响CSS内容的呈现，尤其是浏览器对盒模型的解析
- ☐ (C) 不同浏览器在混杂模式下的行为差异非常大，如果不是要某些hack技术，跨浏览器的行为根本就没有一致性可言
- ☒ (D) 如果在文档开始处没有发现文档类型声明，则所有浏览器都会默认开启严格模式

作答结果：正确

27

2分

在ES6语法中对module描述错误的是（）

- ☒ (A) export导出
- ☐ (B) import导入
- ☐ (C) export default默认导出
- ☐ (D) import * as重命名

作答结果：错误

28

2分

bootstrap的栅格系统描述错误的是（）

- ☐ (A) 系统默认每行划分12列
- ☐ (B) 网格不可以嵌套
- ☐ (C) 默认列间距是30px
- ☒ (D) 手机屏幕的列类前缀是.col-sm-x

作答结果：错误

29

2分

关于新特性Generator函数的描述，错误的是：（）

- ☐ (A) Generator函数，又称生成器函数
- ☒ (B) 声明Generator函数的关键字是：function*
- ☐ (C) Generator函数执行后得到的一个生成器
- ☐ (D) 使用return语句使Generator函数暂停执行，直到next方法的调用

作答结果：错误

30

2分

vue的设计模式是（）

- ☐ (A) MVC
- ☒ (B) MVVM
- ☐ (C) MVX
- ☐ (D) 以上都不对

作答结果：正确

二、多选题 共15题，30分

1

2分

在jQuery Mobile中，除了可以使用jquery提供的标准事件外，也提供了针对移动端浏览器的事件（）

- ☒ (A) 触摸事件 - 当用户触摸屏幕时触发
- ☒ (B) 滑动事件 - 当用户上下滑动时触发
- ☒ (C) 定位事件 - 当设备水平或垂直翻转时触发
- ☒ (D) 页面事件 - 当页面显示，隐藏，创建，加载或未加载时触发

作答结果：正确

2

2分

下列是jQuery Mobile的表单控件的有（）

- ☒ (A) 加载条
- ☒ (B) 文本输入框
- ☒ (C) 复选框
- ☒ (D) 滑动条

作答结果：错误

3

2分

下列哪些是ES6中对象的扩展（）

- ☒ A Object.is()
- ☒ B Object.assign()
- ☒ C Object.getOwnPropertyDescriptor
- ☒ D Object.keys()

作答结果：正确

4

2分

下列哪些是jQuery Mobile中提供的可用图标（）

- ☒ A ui-icon-arrow-l
- ☒ B ui-icon-arrow-r
- ☒ C ui-icon-info
- ☒ D ui-icon-delete

作答结果：正确

5

2分

常用的JavaScript代码调试语句有（）

- ☒ A console.log()
- ☒ B alert()
- ☒ C debugger
- ☒ D try...catch

作答结果：正确

6

2分

对于css样式的重置，说法正确的有（）

- ☒ A 去除标签的默认样式，如input、li等
- ☐ B HTML5的新标签设置display: block
- ☒ C 重置一些元素的样式，如超链接等
- ☒ D 把行内元素显示方式设置为block

作答结果：错误

7

2分

对于Express框架，下列描述正确的有（）

- ☒ A Express是nodejs web应用框架
- ☒ B Express可以设置中间件来响应HTTP请求
- ☒ C 定义了路由表用于执行不同的HTTP请求动作
- ☒ D 可以快速的搭建一个完整功能网站

作答结果：正确

8

2分

在jQuery Mobile中，可以使用data-position属性来设置页眉页脚的定位，可以使用的值（）

- ☒ A Inline
- ☒ B Fixed
- ☐ C Top
- ☐ D Bottom

作答结果：正确

9

2分

以下查询语句正确的有（）

- ☒ A select * from tableA where name='张三';
- ☐ B select * from tableA where name='张%' && age>20;
- ☐ C select * from tableA desc;
- ☒ D select * from tableA group by age;

作答结果：错误

10

2分

关于模板字符串，下列说法正确的是（）

- ☒ A 使用反引号标识
- ☒ B 插入变量的时候使用\${ }
- ☒ C 所有的空格和缩进都会被保留在输出中
- ☒ D \${ }中的表达式不能是函数的调用

作答结果：错误

11

2分

下列是虚拟 DOM 的优点（）

- ☒ A 保证性能下限
- ☒ B 无需手动操作DOM
- ☒ C 跨平台
- ☐ D 可以完全替代DOM

作答结果：正确

12

2分

除了webpack，还有哪些打包插件是用的比较多的（）

- ☒ A grunt
- ☒ B gulp
- ☒ C zip
- ☐ D bower

作答结果：错误

13

2分

下列哪些方式可以实现路由跳转（）

- ☒ A router-link
- ☐ B router-view
- ☐ C this.\$route.push({ path: '/user' })
- ☒ D this.\$router.push({ path: '/user' })

作答结果：正确

14

2分

下列哪些属性属于Vuex（）

- ☒ A State
- ☐ B Statu
- ☐ C Getter
- ☒ D Mutation

作答结果：错误

15

module模块中，对下列语句的描述，错误的是（）

2分

- ☐ A export default 默认导出
- ☐ B import导入
- ☒ C export、export default能直接导出变量表达式
- ☐ D import*as重命名

作答结果：错误

三、判断题 共5题，10分

1

2分

vue-router的导航守卫有全局导航守卫和组件内导航守卫两种

- ☐ A 对
- ☒ B 错

作答结果：正确

2

2分

express中，唯一内置的中间件是session中间件？

- ☐ A 对
- ☒ B 错

作答结果：正确

3

2分

使用webpack可以对后台代码进行打包（ ）

- ☒ A 对
- ☐ B 错

作答结果：错误

4

2分

webpack是前端唯一的模块化构建工具（ ）

- ☐ A 对
- ☒ B 错

作答结果：正确

5

2分

jQuery Mobile提供了5种不同的主题样式，从a到e，每一种主题的按钮、工具条、内容块等颜色都不一样（ ）

A 对

B 错

作答结果：正确

四、客观填空题 共4题，100分

1

30分

阅读下列说明、效果图和代码，进行动态网页开发，回答问题1至问题6。

试题三.zip

【说明】

该程序为一个商品展示系统，使用前后端分离模式，后端使用Node.js+Express框架，项目名为listProduct_server，前端使用Vue.js框架，使用Vue CLI构建项目,项目名为listProduct_client。其中后台采用MVC架构，核心文件包括服务器启动配置文件app.js，dao层文件product.dao.js，模型文件product.js，控制器文件product.controller.js，业务逻辑文件product.service.js，工具类文件夹util下的数据库配置文件db.js和自定义响应模块文件response.js。前端核心文件包括配置文件config/index.js，商品列表组件ProductList.vue，使用MySQL数据库存放商品信息。Vue.js框架采用Axios组件实现Ajax请求，项目API遵循RESTful API标准。

商品列表页面包括用户输入部分，商品列表部分。用户输入部分显示input输入框和搜索按钮，商品列表部分显示商品图片，商品名称，商品价格，删除按钮，

用户在input输入框中输入商品名称，点击“搜索”按钮，商品列表部分显示符合搜索条件的商品。点击“删除”按钮，删除对应到商品。

【效果图】

商品列表页面：对应模板文件ProductList.vue

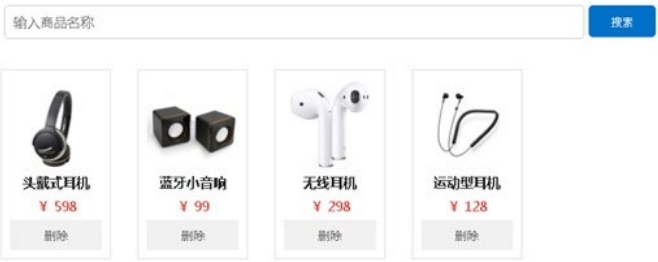


图4-1

【listProduct_server项目代码：app.js】

```
const express = require('express');
const app = express();
const bodyParser = require('body-parser');
const product = require('./routes/product.controller.js');
const http = require('http');
app.use(bodyParser.json());
app.use(bodyParser.urlencoded({extended: false}));
app.use('/', product);
http.(6).listen(3000, function() {
  console.log('listening on : 3000');
```

```
});
app.use(function(err, req, res, next) {
  console.error(err.stack)
  res.status(500).send('Something broke!')
})
```

【listProduct_server项目代码: product.js】

```
class product {
  constructor(id, name, price, img){
    this.id = id;
    this.name = name;
    this.price = price;
    this.img = img;
  }
  get id(){
    return this._id;
  }
  set id(id){
    this._id = id;
  }
  //仅展示部分get/set方法
};
module.exports = product;
```

【listProduct_server项目代码: product.controller.js】

```
const express = require('express');
const router = express.Router();
const productService = require('../service/product.service.js');
const back = require('../util/response.js');
router.get('/products/list', (req, res) => {
  productService.listProduct(req, (data) => {
    back.jsonWrite(res, data);
  })
});
router.get('/products/product', (req, res) => {
  productService.selectProduct(req, (data) => {
    back.jsonWrite(res, data);
  })
});
router.delete('/products/product', (req, res) => {
  productService.deleteProduct(req, (data) => {
    back.jsonWrite(res, data);
  })
});
module.exports = router;
```

【listProduct_server项目代码: product.service.js】

```
const productDao = require('../dao/product.dao.js');
const product = require('../model/product.js');

exports.listProduct = (req, callback_controller) => {
  productDao.listProduct((data) => {
    callback_controller(data);
  })
}
exports.selectProduct = (req, callback_controller) => {
  let model = new product(null, req.query.name, null, null);
  productDao.selectProduct(model, (data) => {
```

```
        callback_controller(data);
    })
}
exports.deleteProduct = (req,callback_controller) => {
    let model = new product(req.body["id"], null, null, null);
    productDao.deleteProduct(model, (data) => {
        callback_controller(data);
    })
}
```

【listProduct_client项目配置文件： config/index.js】

```
'use strict'
const path = require('path')
module.exports = {
  dev: {
    ...
    proxyTable: {
      '/': {
        // 配置跨域地址http://localhost:3000
        : 'http://localhost:3000',
        // 设置是否跨域
        : true,
        pathRewrite: {
          ".": ""
        }
      },
    },
    ...
  }
}
```

【listProduct_client项目项目代码： ProductList.vue】

```
<template>
  <div>
    <!-- 搜索框 -->
    <div>
      <input type="text" placeholder="输入商品名称" v-model="searchText">
      <button v-on:click="handleSearch">搜索</button>
    </div>
    <!-- 商品列表 -->
    <!--数据循环输出商品列表-->
    <div
      >
      <a href="javascript:;">
        
        <h4>{{ item.name }}</h4>
        <div>¥ {{ item.price }}</div>
        <div v-on:click="handleDel(item.id)">删除</div>
        <!-- <div>修改商品</div> -->
      </a>
    </div>
  </div>
</template>
<script>
// 引入请求方法
import axios from 'axios'
```

```
export default {
  data() {
    return {
      info: null, // 存放商品数据
      searchText: " // 存放用户搜索数据
    }
  },
  created() {
    // 请求列表数据，初始化
    this.getList();
  },
  methods: {
    getList() {
      // 请求全部商品列表数据
      axios.get('/products/list')
        .then(res => {
          if (res.data.state === 200){
            this.info = res.data.data;
          }
        })
        .catch(error => {
          console.log(error)
        })
    },
    handleSearch() {
      //通过url传参方式请求搜索商品的数据
      axios.      ('/products/product',{
        : {
          name: this.searchText
        }
      })
      .then(res => {
        if (res.data.state === 200){
          this.info = res.data.data;
        }
      })
      .catch(error => {
        console.log(error)
      })
    },
    handleDel(id){
      // 删除请求，将json对象发送到后台
      axios.      ('/products/product',{
        : {
          id: id
        }
      })
      .then(res => {
        if (res.data.state === 200){
          this.handleSearch();
        }
      })
      .catch(error => {
        console.log(error)
      })
    }
  }
}
```



```
        })
    }
}
}
</script>
```

【问题1】请根据MVC模型，回答下列问题。

- 1、在本项目中，请分析product.js中的代码回答product.js在MVC模型Model、View和Controller中属于哪层。
- 2、在本项目中，请分析product.controller.js中的代码回答product.controller.js在MVC模型Model、View和Controller中属于哪层。

【问题2】分析项目路由，回答下列问题。

在浏览器中输入网址，访问商品列表页面，请分析路由，回答下列问题：

- 1、在请求商品列表时，使用GET请求http://127.0.0.1:3000/products/list，以下哪个路由不能被调用

```
const express = require('express');
const app = express();
```

- A. app.get('/', function (req, res) {});
- B. app.use('/', function (req, res) {});
- C. app.get('/products/list', function (req, res) {});
- D. app.all('/products/list', function (req, res) {});
- 2、在定义URI时，以下哪个URI符合RESTful API规范

- A. GET /getProducts
- B. DELETE /deleteProduct/1
- C. GET /get_product?id=3
- D. GET /products/product

【问题3】基于Node.js和Express框架，回答下列问题。

- 1、在app.js文件中，下列哪种中间件没有被用到
 - A. 应用层中间件
 - B. 路由器级中间件
 - C. 错误处理中间件
 - D. 第三方中间件
- 2、在app.js中，使用http模块创建HTTP服务服务器，在app.js中填写 空。

【问题4】分析模型文件，回答下列问题。

在product.js文件中，定义了product类，在类中定义了constructor()方法。请分析product类，回答下列问题：

- 1、在product.service.js中执行let model = new product(null, req.query.name, null, null);代码时，在product类中依次执行了什么方法
 - A. set ->constructor
 - B. constructor->set
 - C. constructor->set->constructor
 - D. set
- 2、如果在product.service.js中执行let model = new product(null,

req.query.name, null, null);代码后，再加一行代码let name = model.name;; 那么这行代码在product类中执行了什么方法

- A. constructor
- B. set
- C. get
- D. constructor->get

【问题5】分析配置文件index.js，回答下列问题，填写（9）至（10）。

在listProduct_client项目中Ajax请求需要跨域，在config配置文件中的index.js中配置proxy代理来实现Ajax请求跨域。分析config配置文件中的index.js，在proxyTable中配置正确的proxy代理，项目中所有的Ajax请求实现跨域。填写（9）至（10）空。

【问题6】分析模板文件ProductList.vue，回答下列问题，填写（11）至（15）。

listProduct项目的后端和前端运行在同一个服务器下，分析ProductList.vue代码，实现用户在搜索商品时，向后台请求数据，渲染页面。点击删除按钮，删除对应的商品。填写（11）至（15）空。

1、学生答案：

2、学生答案：

3、学生答案：

4、学生答案：

5、学生答案：

6、学生答案：

7、学生答案：

8、学生答案：

9、学生答案：

10、学生答案：

- 11、学生答案：
- 12、学生答案：
- 13、学生答案：
- 14、学生答案：
- 15、学生答案：

2

30分

阅读下列说明、效果图，打开“考生文件夹\20002\quiz”文件夹中的素材，进行Laravel框架开发，在空（1）至（15）处填写正确代码，操作完成后保存素材。



【说明】
编写一个在线答题系统，试题共4道数学题，每题3个选项，都为单选题。每道25分，共100分。每做完一题，单击“下一题”按钮，提交当前题目答案，并显示下一题内容，如图1所示。在最后一题，按钮显示为“提交”。单击“提交”按钮，显示答对题数和得分情况，总分100分，如图2所示。



图 1
在线答题



图 2

文件设计如下表所示。

| 文件 | 说明 |
|-----------------------------------------|--------------|
| routes/web.php | 路由文件 |
| resources/views/quiz.blade.php | 答题模板页面，对应图 1 |
| resources/views/result.blade.php | 结果模板页面，对应图 2 |
| app/Http/Controllers/QuizController.php | Quiz 控制类文件 |
| public/css/quiz.css | 项目的 CSS 文件 |

【安装Laravel工程quiz】：参阅“考生文件夹\20002”文件夹中的“readme.txt”文件部署quiz工程，目录如图3所示，请打开“考生文件夹\20002”文件夹中的文件“问题1.txt”，根据目录中的内容回答问题，完成后保存文件“问题1.txt”。

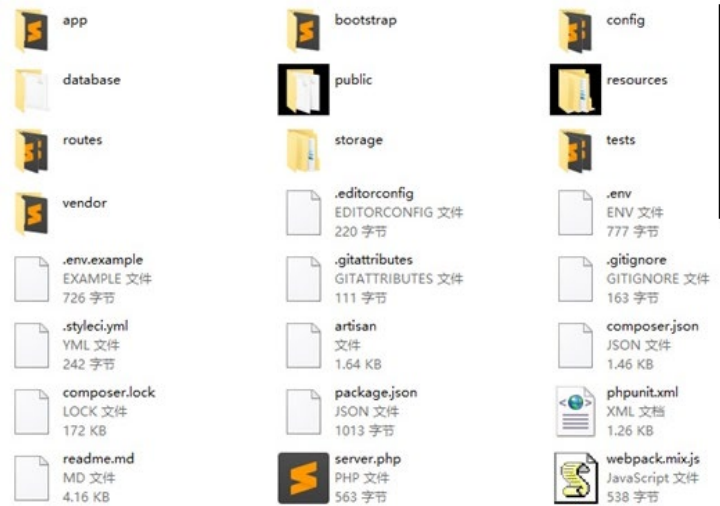


图 3

【配置路由】：请根据路由的配置信息，打开“考生文件夹\20002”文件夹中的文件“问题2.txt”，分析后回答问题，完成后保存文件“问题2.txt”。

编写quiz/routes/web.php。

- 1、进入答题系统路由；
- 2、提交当前题答案，进入下一题；
- 3、提交最后一题，并显示答题结果。

【修改控制类QuizController.php】：打开“考生文件夹\20002\quiz\app\Http\Controllers”文件夹中的文件“QuizController.php”，阅读代码，填写（4）至（10），完成后保存文件“QuizController.php”。

编写quiz/app/Http/Controllers/QuizController.php文件。

- 1、定义类变量\$questions保存试题数据；
- 2、定义保存到Session中的属性名常量；
- 3、创建getQuestion()函数，参数为题号，读取试题内容；
- 4、编写start()函数，读取题目内容，并传给模板页面；
- 5、编写next()函数，获取下一道题的题目内容；
- 6、编写submit()函数，从Session中取出用户的答案数组，计算数组元素个数，计算得分；
- 7、返回到result.blade.php模板页面，将答对的题数和得分传入。

注意：除删除（4）至（10）填入正确的内容外，不能修改或删除其它任何内容。

【编写模板文件quiz.blade.php】回答问题和填写代码（11）至（13）。

编写quiz/resources/views/quiz.blade.php文件，显示答题页面。

导入CSS文件时，使用内置URL类的方法来引入css文件，默认在Web根目录的css文件夹内，也就是public/css目录。

- 1、打开“考生文件夹\20002”文件夹中的文件“问题3.txt”，回答问题填写正确答案，完成后保存文件“问题3.txt”。
 - 2、打开“考生文件夹\20002\quiz\resources\views”文件夹中的文件“quiz.blade.php”，填写代码（12）至（13），完成后保存文件“quiz.blade.php”。
- 注意：除删除（12）至（13）填入正确的内容外，不能修改或删除其它任何内容。

【编写模板文件result.blade.php】：回答问题和填写代码（14）至（15）。

编写quiz/resources/views/result.blade.php文件，显示答题结果页面，包括答对的题数和最终得分。

- 1、打开“考生文件夹\20002”文件夹中的文件“问题4.txt”，回答问题填写正确答案，完成后保存文件“问题4.txt”。
 - 2、打开“考生文件夹\20002\quiz\resources\views”文件夹中的文件“result.blade.php”，填写代码（15），完成后保存文件“result.blade.php”。
- 注意：除删除（15）填入正确的内容外，不能修改或删除其它任何内容。

【问题】

进行Laravel项目开发，在空（1）至（15）处填入正确的内容。

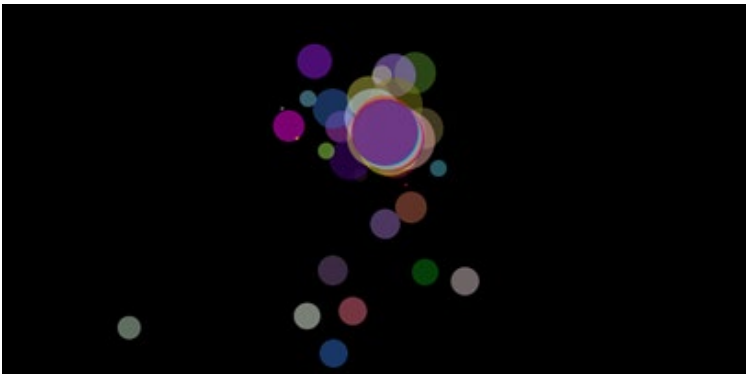
- 1、学生答案：
- 2、学生答案：
- 3、学生答案：
- 4、学生答案：
- 5、学生答案：
- 6、学生答案：
- 7、学生答案：
- 8、学生答案：
- 9、学生答案：


- 10、学生答案：
- 11、学生答案：
- 12、学生答案：
- 13、学生答案：
- 14、学生答案：
- 15、学生答案：

3

20分

阅读下列说明、效果图和代码，填写（1）至（10）代码。
【说明】
项目一为使用es6语法实现的页面效果，项目名为ball。项目下有index.html和js目录，js目录下面有index.js文件。



【素材】
 素材.rar

【代码： index.html】
`<!DOCTYPE html>
<html>
<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">
 <title>Document</title>
</head>
<body>
 <!-- 在（1）（2）处补齐合适的标签 -->
 <div id="canvas"></div>
 <!-- 在（3）处补齐代码，使用./开始的相对路径 -->
 <script src="index.js"></script>
</body>
</html>`

【代码： index.js】
// 定义一个取两个数之间的随机数的方法
`let newMath = (x, y) => {
 let tmp = y - x;`

```

        return Math.floor(Math.random() * tmp + x);
    }
    //生成一个随机的色值
    let randomRgbColor = () => { //随机生成RGBA颜色
        let r = Math.floor(Math.random() * 256); //随机生成256以内r值
        let g = Math.floor(Math.random() * 256); //随机生成256以
        内g值
        let b = Math.floor(Math.random() * 256); //随机生成256以内b值
        return `rgba(${r},${g},${b},.5)`; //返回rgb(r,g,b,a)格式颜色
    }
    //获取当前的画布
    const canvas = document.getElementById('canvas');
    // 在（4）处填写方法，返回一个用于在画布上绘图的环境
    const ctx = canvas.getContext('2d');
    canvas.width = 1000;
    canvas.height = 600;
    canvas.style.backgroundColor = '#000';
    //在（5）处补齐代码，创建小球的类
    (function () {
        class Ball {
            //在（6）处补齐代码，创建Ball类的构造函数
            constructor(x, y, color) {
                this.x = x;
                this.y = y;
                this.color = color;
                this.r = 40;
            }
            render() {
                ctx.save();
                ctx.beginPath();
                ctx.arc(this.x, this.y, this.r, 0, Math.PI * 2);
                ctx.fillStyle = this.color;
                ctx.fill();
                ctx.restore();
            }
        }
    })()
    // 在（7）（8）处补齐代码，MoveBall类继承Ball类
    (function () {
        class MoveBall extends Ball {
            constructor(x, y, color) {
                //在（9）处补齐代码，继承父类Ball的方法
                super(x, y, color);
                this.dX = new Math(-5, 5);
                this.dY = new Math(-5, 5);
                this.dR = new Math(1, 3);
            }
            update() {
                this.x += this.dX;
                this.y += this.dY;
                this.r -= this.dR;
                if (this.r < 0) {
                    this.r = 0;
                }
            }
        }
    })()
    //储存变化的小球
    let ballArr = [];
    //监听鼠标的移动
    canvas.addEventListener('mousemove', function (e) {
        // 在（10）处补齐代码，构造一个移动小球实例。
        ballArr.push(new MoveBall(e.offsetX, e.offsetY,
            randomRgbColor()));
    });
    // 开启定时器,并开始画图
    setInterval(function () {
        ctx.clearRect(0, 0, canvas.width, canvas.height);
        for (let i = 0, l = ballArr.length; i < l; i++) {
            ballArr[i].render();
            ballArr[i].update();
        }
    }, 50)
}

```

【问题】根据说明，代码提示补全代码。

- 1、 学生答案：
- 2、 学生答案：
- 3、 学生答案：
- 4、 学生答案：
- 5、 学生答案：
- 6、 学生答案：
- 7、 学生答案：
- 8、 学生答案：
- 9、 学生答案：
- 10、 学生答案：

4

20分

阅读下列说明，效果图和代码，填写1到10所空位置。

【说明】
本项目名称listpage，使用jQuery mobile布局的移动端项目。项目目录中有index.html,css目录和js目录。其中css目录存放样式文件，js目录存放jQuery.min.js和jQuery.mobile.min.js。
本项目实现首页、通信录页面和搜索页面切换功能。



(2) listProduct_server项目目录结构如下图所示：



【项目1效果图index.html】



【项目1代码：主页index.html】

```
<!DOCTYPE html>
<html>
<head>
  <meta name="viewport" content="width=device-width, initial-
scale=1">
  <link rel="stylesheet" href="/css/jquery.mobile-1.4.5.min.css">
  <script src="/js/jquery.min.js"> </script>
  <script src="/js/jquery.mobile-1.4.5.min.js"> </script>
</head>
<body>
<!-- 第一页 -->
<!-- 在 (1) 处补全代码，作用是页 -->
  <div data-role=" " id="pageone">
    <!--在 (2) 处补全代码，页面头部 -->
    <div data-role=" " ">
      <h1>欢迎使用jQuery Mobile</h1>
      <!-- 在 (3) 处补全代码，导航栏 -->
      <div data-role=" " ">
        <ul>
          <li><a href="#pageone" data-transition="slide"
data-direction="reverse" class="ui-btn-active"
data-icon="home">首页</a>
          </li>
          <li><a href="#pagetwo" data-transition="slide"
data-direction="reverse" data-icon="bars">通讯录</a>
          </li>
          <li><a href="#pagethree" data-transition="slide"
data-direction="reverse" data-icon="search">搜索</a></li>
        </ul>
```

```

    </div>
</div>
<!-- 在 (4) 处补全代码，页面主体 -->
<div data-role=" " class="ui-content">
    <!-- 在 (5) (6) 处补全代码，分别设置折叠框集合和折叠框 -->
    <div data-role=" " >
        <div data-role=" " >
            <h1>新闻标题1</h1>
            <p>新闻1的内容，新闻1的内容，新闻1的内容，新闻1的内容。</p>
            <p>新闻1的内容，新闻1的内容，新闻1的内容，新闻1的内容。</p>
            <p>新闻1的内容，新闻1的内容，新闻1的内容，新闻1的内容。</p>
        </div>
        <div data-role="collapsible">
            <h1>新闻标题2</h1>
            <p>新闻2的内容，新闻2的内容，新闻2的内容，新闻2的内容。</p>
            <p>新闻2的内容，新闻2的内容，新闻2的内容，新闻2的内容。</p>
            <p>新闻2的内容，新闻2的内容，新闻2的内容，新闻2的内容。</p>
            <p>新闻2的内容，新闻2的内容，新闻2的内容，新闻2的内容。</p>
        </div>
    </div>
    <div data-role="footer" data-position="fixed">
        <h1>底部文本</h1>
    </div>
</div>
<!-- 第二页 -->
<div data-role="page" id="pagetwo">
    <div data-role="header">
        <h1>通讯录</h1>
        <div data-role="navbar">
            <ul>
                <li><a href="#pageone" data-transition="slide" data-direction="reverse" data-icon="home">首页</a>
                </li>
                <li><a href="#pagetwo" data-transition="slide" data-direction="reverse" data-icon="bars" class="ui-btn-active">通讯录</a> </li>
                <li><a href="#pagethree" data-transition="slide" data-direction="reverse" data-icon="search">搜索</a> </li>
            </ul>
        </div>
    </div>
    <div data-role=" main" class="ui-content">
        <!-- 在 (7) 处补全代码，列表视图 -->
        <ul data-role=" " >
            <li><a href="#">列表项</a> </li>
            <li><a href="#">列表项</a> </li>
            <li><a href="#">列表项</a> </li>
        </ul>
    </div>
</div>
```

```

        </ul>
    </div>
    <!-- 在（8）处补全代码，固定在页面底部 -->
    <div data-role="footer" data-position="fixed">
        <h1>底部文本</h1>
    </div>
</div>
<!-- 第三页 -->
<div data-role="page" id="pagethree">
    <div data-role="header">
        <h1>搜索</h1>
        <div data-role="navbar">
            <ul>
                <li><a href="#pageone" data-transition="slide"
data-direction="reverse"
data-icon="home">首页</a>
                </li>
                <li><a href="#pagetwo" data-transition="slide"
data-direction="reverse"
data-icon="bars">通讯录</a></li>
                <!-- 在（9）处补全代码，图标是search -->
                <li><a href="#pagethree" data-transition="slide"
data-direction="reverse"
data-icon="search"
class="ui-btn-active">搜索</a></li>
            </ul>
        </div>
    </div>
    <div data-role="main" class="ui-content">
        <form method="post" action="">
            <label for="fname">请输入搜索内容</label>
            <input type="text" name="fname" id="fname">
            <!-- 在（10）处填写input类型为提交按钮 -->
            <input type="submit" value="提交按钮">
        </form>
    </div>
    <div data-role="footer" data-position="fixed">
        <h1>底部文本</h1>
    </div>
</div>
</body>
</html>
```

【问题】根据jQuery Mobile知识，正确填写所缺代码。

 [素材.zip](#)

1、学生答案：

2、学生答案：

3、学生答案：

4、学生答案：

5、学生答案：

6、学生答案：

7、学生答案：

8、学生答案：

9、学生答案：

10、学生答案：



技术支持：奥鹏教育

京公网安备 11010802029644号