

# 综合应用题说明文档

---

## 1. 概要

---

综合应用题共3道题目，3道题目围绕一个系统进行展开，分别是列表展示、删除数据、添加数据三个功能。

在开始之前请先仔细阅读本文档。

1. 应用题旨在检验大家的 Ajax 编程应用能力
2. 已为大家准备好测试题需要的服务端及数据接口等资源，请自行按照本文档第2小节《部署说明》将应用题项目部署到自己本地
3. 部署成功以后，请阅读第3小节《项目结构说明》
4. 开始根据题目需求进行答题

## 2. 部署说明

---

环境依赖：

- Node.js > 8.x.x
- npm

部署操作：

1. 在命令行中进入 `crud-ajax` 目录
2. 在该目录下分别执行以下命令

```
1 # 安装项目的所有依赖项
2 npm install
3
4 # 启动接口服务
5 npm start
```

执行完上述命令后，如果看到以下输出则说明项目部署启动成功：

```
crud-ajax — npm start — npm — node • npm TMPDIR=/var/folders/7b/2p5y0cmx55g33mzzhqwxblxr0000gn/T/ XPC_FLAGS=0x0 — Sol...
➔ crud-ajax npm start

> client@1.0.0 start /Users/lpz/Desktop/Web阶段测试题/前端_就业班_Node&Ajax阶段_
试卷/前端_就业班_Node&Ajax阶段_编程题1-2-3-4素材/crud-ajax
> node app.js

Server running at http://127.0.0.1:3000/
```

接下来在浏览器中访问 `http://127.0.0.1:3000/` 进行预览。

## 3. 项目结构说明

### 3.1. 静态页面

项目中已将 `views` 目录开放为静态资源。

页面	请求路径	备注
views/index.html	<code>/</code> 或者 <code>/index.html</code>	首页
views/new.html	<code>/new.html</code>	添加页面

### 3.2. 数据接口

项目中已提供好考试题需要的所有数据接口，具体使用细节查阅第4小节《接口说明》。

### 3.3. 其他资源

为了大家考试方便，所有静态页面中都已加载 `jQuery` 和 `art-template`。如需要其他第三方库，请自行使用 `npm` 下载使用。

## 4. 接口说明

注意：

- 请严格按照接口文档的要求的数据及格式请求接口，如果数据或格式错误接口会返回 400 错误状态码。

例如添加员工要求 `name` 字段为必须的，则必须传递，否则返回 `400` 错误状态码，并给出如下提示：

```
1 {  
2   "statusCode": 400,  
3   "error": "Bad Request",  
4   "message": "body should have required property 'name'"  
5 }
```

所以请认真阅读接口文档，按照接口文档要求使用接口。

所以请认真阅读接口文档，按照接口文档要求使用接口。

所以请认真阅读接口文档，按照接口文档要求使用接口。

### 4.1. 获取所有员工

GET <http://localhost:3000/employees/list>

请求示例：

- 请求地址：`http://localhost:3000/employees/list`

响应结果示例：

- 状态码：`200`
- 结果：

```
1 {  
2   "success": true,  
3   "data": [  
4     {  
5       "name": "Leanne Graham",  
6       "gender": "m",  
7       "birthday": "2018-12-18",  
8       "employmentDate": "2018-12-18",  
9       "mobilePhone": "12345678910",  
10    }  
11  ]  
12 }
```

```
10     "email": "graham@itcast.cn",
11     "id": "f546c570-044a-47f5-9259-e84eae47b066"
12 },
13 {
14     "name": "Ervin Howell",
15     "gender": "f",
16     "birthday": "2018-12-18",
17     "employmentDate": "2018-12-18",
18     "mobilePhone": "12345678910",
19     "email": "graham@itcast.cn",
20     "id": "6e4f69d4-0849-4af4-bdf2-a7492afb4d5d"
21 },
22 {
23     "name": "Clementine Bauch",
24     "gender": "m",
25     "birthday": "2018-12-18",
26     "employmentDate": "2018-12-18",
27     "mobilePhone": "12345678910",
28     "email": "graham@itcast.cn",
29     "id": "de85e12b-b14b-4137-a557-3046556d2ec9"
30 },
31 {
32     "name": "Patricia Lebsack",
33     "gender": "m",
34     "birthday": "2018-12-18",
35     "employmentDate": "2018-12-18",
36     "mobilePhone": "12345678910",
37     "email": "graham@itcast.cn",
38     "id": "4729f515-54e6-4546-9a08-7c9a1673c3b3"
39 },
40 {
41     "name": "Chelsey Dietrich",
42     "gender": "f",
43     "birthday": "2018-12-18",
44     "employmentDate": "2018-12-18",
45     "mobilePhone": "12345678910",
46     "email": "dietrich@itcast.cn",
47     "id": "af843025-ce71-4726-b058-bcf0f46c9ef2"
48 }
49 ]
50 }
```

## 4.2. 获取单个员工

GET <http://localhost:3000/employees>

查询参数：

名称	类型	是否必须	备注
id	String	是	唯一标识

请求示例：

- 请求地址：`http://localhost:3000/employees?id=f546c570-044a-47f5-9259-e84eae47b066`

响应结果示例：

- 状态码：`200`
- 结果：

```
1  {
2    "success": true,
3    "data": {
4      "name": "Leanne Graham",
5      "gender": "m",
6      "birthday": "2018-12-18",
7      "employmentDate": "2018-12-18",
8      "mobilePhone": "12345678910",
9      "email": "graham@itcast.cn",
10     "id": "f546c570-044a-47f5-9259-e84eae47b066"
11   }
12 }
```

## 4.3. 添加员工

`POST` <http://localhost:3000/employees/create>

请求体：

名称	类型	是否必须	备注
name	String	是	姓名
gender	String	是	性别，取值范围必须是 m 或 f，m 表示男，f 表示女
birthday	String	是	出生日期，类型必须符合“年-月-日”日期格式
employmentDate	String	是	入职日期，类型必须符合“年-月-日”日期格式
mobilePhone	String	是	手机号码
email	String	是	邮箱，类型必须是邮箱格式

请求示例：

- 请求地址：`http://localhost:3000/employees/create`
- 请求体：

```
1 {
2   "name": "Chelsey Dietrich",
3   "gender": "f",
4   "birthday": "2018-12-18",
5   "employmentDate": "2018-12-18",
6   "mobilePhone": "12345678910",
7   "email": "dietrich@itcast.cn"
8 }
```

响应结果示例：

- 状态码：`201`
- 结果：

```
1 {
2   "success": true,
3   "data": {
4     "name": "Chelsey Dietrich",
5     "gender": "f",
6     "birthday": "2018-12-18",
7     "employmentDate": "2018-12-18",
8     "mobilePhone": "12345678910",
9     "email": "dietrich@itcast.cn",
10    "id": "af843025-ce71-4726-b058-bcf0f46c9ef2"
11  }
12 }
```

## 4.4. 修改员工

`POST` <http://localhost:3000/employees/update>

查询参数：

名称	类型	是否必须	备注
id	String	是	唯一标识

请求体：

名称	类型	是否必须	备注
name	String	否	姓名
gender	String	否	性别，取值范围必须是 m 或 f，m 表示男，f 表示女
birthday	String	否	出生日期，类型必须符合“年-月-日”日期格式
employmentDate	String	否	入职日期，类型必须符合“年-月-日”日期格式
mobilePhone	String	否	手机号码
email	String	否	邮箱，类型必须是邮箱格式

请求示例：

- 请求地址：`http://localhost:3000/employees/update?id=f546c570-044a-47f5-9259-e84eae47b066`
- 请求体：

```
1 {  
2   "name": "Leanne Graham",  
3   "gender": "m",  
4   "birthday": "2018-12-18",  
5   "employmentDate": "2018-12-18",  
6   "mobilePhone": "12345678910",  
7   "email": "111@222.com"  
8 }
```

响应结果示例：

- 状态码：`200`
- 结果：

```
1 {
2   "success": true,
3   "data": {
4     "name": "Leanne Graham",
5     "gender": "m",
6     "birthday": "2018-12-18",
7     "employmentDate": "2018-12-18",
8     "mobilePhone": "12345678910",
9     "email": "111@222.com",
10    "id": "f546c570-044a-47f5-9259-e84eae47b066"
11  }
12 }
```

## 4.5. 删除员工

GET <http://localhost:3000/employees/delete>

查询参数：

参数名	类型	是否必须	备注
id	String	是	唯一标识

请求示例：

- 请求地址：`http://localhost:3000/employees/delete?id=af843025-ce71-4726-b058-bcf0f46c9ef2`

响应结果示例：

- 状态码：`200`
- 结果：

```
1 {
2   "success": true,
3   "data": {}
4 }
```