

一. 开发规范:

1. 后端环境

xxx

2. 通信协议

http

3. 通信格式

json

4. API 规范

一定程度上符合 RESTful 定义

二. 数据库结构

xxx

三. 接口说明

1. 获取周边景点天气数据接口

URL : http://127.0.0.1:8000/api/v1/tour/<location>?surrounding_weather=1

1.1 请求方式

GET

1.2 请求格式

http://127.0.0.1:8000/api/v1/tour/<location> 地址后方添加查询字符串

surrounding_weather,值恒定为 1,表示获取周边景点天气数据

1.3 响应格式

Json 具体参数如下

字段	含义	类型	备注
code	状态	int	默认正常为 200,异常请见 7.4

data	具体数据均在 data 中	{}	{'location': 景点名数组, 'weather': 天气数组, 'temperature': 温度数组} 见下方展开
location	景点名(data 中数据)	[char,char,...]	周边 5 个景点的中文名称数据, 子元素数据类型为 char, 详细请见下方示例
temperature	温度(data 中数据)	[char,char,...]	周边 5 个景点对应的温度数据, 子元素数据类型为 char, 详细请见下方示例
weather	风向(data 中数据)	[char,char,...]	周边 5 个景点对应的天气数据, 子元素数据类型为 char, 详细请见下方示例

响应示例:

```
{
  'code': 200,
  data: {
    'location': ['天塔', '天津之眼', '水上公园', '盘山', '五大道'],
    'weather': ['多云', '多云', '多云', '多云', '多云'],
    'temperature': ['5/13', '5/13', '5/13', '5/13', '5/13']
  }
}
```

1.4 异常码

异常响应示例:

2. 获取十大景点天气列表接口

URL : http://127.0.0.1:8000/api/v1/tour/<location>?top10=1

2.1 请求方式

GET

2.2 请求格式

http://127.0.0.1:8000/api/v1/tour/<location> 地址后方添加查询字符串 top10,值恒定为 1,表示获取十大景点天气数据接口

2.3 响应格式

Json 具体参数如下

字段	含义	类型	备注
code	状态	int	默认正常为 200,异常请见 2.4
data	具体数据均在 data 中	{}	{‘scenic_site_id’: 景点 ID 数组 , ‘scenic_site_name’: 景点名称数组,‘img_src’: 图片路径数组}见下方展开
scenic_site_id	景点 ID(data 中数据)	[int, int, ...]	顺序存储
scenic_site_name	景点名称(data 中数据)	[char, char, ...]	顺序存储
img_src	图片路径(data 中数据)	[char, char, ...]	注意：此路径须为 URL 格式的相对地址

响应示例:

```
{
  'code':200,
  'data':{
    'scenic_site_id':[1,2,3,4,5,6,7,8,9,10],
    'scenic_site_name':['天津之眼','水上公园','津湾广场','天津之眼','水上公园','
津湾广场','天津之眼','水上公园','津湾广场','天津之眼'],
    'img_src':['/static/images/tour/img1.png',
                '/static/images/tour/img2.png',
                '/static/images/tour/img3.png',
                '/static/images/tour/img4.png',
                '/static/images/tour/img5.png',
                '/static/images/tour/img6.png',
                '/static/images/tour/img7.png',
                '/static/images/tour/img8.png',
                '/static/images/tour/img9.png',
                '/static/images/tour/img10.png'
    ]
  }
}
```

2.4 异常码

异常响应示例:

3. 获取十大景点天气详情接口

URL : http://127.0.0.1:8000/api/v1/tour/<location>?weather_info=1&scenic_site_id=1111

3.1 请求方式

GET

3.2 请求格式

<http://127.0.0.1:8000/api/v1/weather/<location>> 地址后方添加查询字符串

weather_inf 及 scenic_site_id, weather_inf 值恒定为 1 , scenic_site_id 值为具体

景点 ID , 表示获取具体景点的天气数据

3.3 响应格式

Json 具体参数如下

字段	含义	类型	备注
code	状态	int	默认正常为 200,异常请见 3.4
data	具体数据均在 data 中	{}	{'week': 星期数组 , 'date': 日期数组 , 'weather': 天气数组 , 'temperature': 温度数组} , 见下方展开
week	星期(data 中数据)	[char, char, ...]	顺序存储
date	日期(data 中数据)	[char, char, ...]	顺序存储
weather	天气(data 中数据)	[char, char, ...]	顺序存储
temperature	温度(data 中数据)	[char, char, ...]	顺序存储

响应示例:

```
{
  'code':200,
  'data':{
    'week':['星期三','星期四','星期五','星期六','星期日','星期一','星期二'],
    'date':['11 月 20 日','11 月 21 日','11 月 22 日','11 月 23 日','11 月 24 日','11
月 25 日','11 月 26 日'],
    'weather':['多云转阴','多云转阴','多云转阴','多云转阴','多云转阴','多云转阴','
多云转阴'],
    'temperature':['0°C~9°C','0°C~9°C','0°C~9°C','0°C~9°C','0°C~9°C',
'0°C~9°C','0°C~9°C']
  }
}
```

3.4 异常码

异常响应示例: