

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. 获取首页天气数据接口

URL : http://127.0.0.1:8000/api/v1/weather/<location>?index=1

1.1 请求方式

GET

1.2 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 index,

值恒定为 1,表示获取首页天气数据

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 1.4

data	具体数据均在 data 中	{}	{'weather': 天气, 'temperature': 温度}见下方展开
weather	实时天气(data 中数据)	char	当前天气情况
temperature	温度(data 中数据)	{}	{'time': 时间, 'temp': 对应温度}件下方展开
time	时间	[char, char, ...]	依次为从现在到未来 4 小时的整点时刻,例如 ['now', '09:00', '10:00', '11:00', '12:00']
temp	对应温度	[int, int, ..]	与时间相对应的温度值

响应示例:

```
{'code':200, data: {'weather':'多云', 'temperature': {'time': ['now', '09:00', '10:00', '11:00', '12:00'], 'temp': [4, 5, 7, 9, 4]}}}
```

1.4 异常码

异常响应示例:

2. 获取实时天气数据接口

URL : <http://127.0.0.1:8000/api/v1/weather/<location>?now=1>

2.1 请求方式

GET

2.2 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 now,

值恒定为 1,表示获取实时天气数据接口

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	{}	{‘weather’: 天气, ‘temperature’: 温度, ‘wind’: 风力风向}见下方展开
weather	实时天气(data 中数据)	char	当前天气情况
temperature	温度(data 中数据)	int	当前气温
wind	风力风向(data 中数据)	{}	{‘scale’: 风力, ‘direction’: 风向}见下方展开
scale	风力(wind 中数据)	int	
direction	风向(wind 中数据)	char	

响应示例:

```
{'code':200,data: {'weather':' 多云 ', 'temperature': 20, 'wind': {'scale': 2, 'direction: '东南'}}}
```

2.4 异常码

异常响应示例:

3. 获取生活指数数据接口

URL : http://127.0.0.1:8000/api/v1/weather/<location>?living_index=1

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

living_index,值恒定为 1,表示获取生活指数数据

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	{}	{'clothing':穿衣指数 'uv': 紫外线指数 , 'car_washing': 洗车指数, 'air_pollution_diffusion': 空气污染扩散指数}见下方展

			开
clothing	穿衣指数(data 中数据)	char	
uv	紫外线指数(data 中数据)	char	
car_washing	洗车指数(data 中数据)	char	
air_pollution_diffusion	空气污染扩散指数 (data 中数据)	char	

响应示例:

```
{'code':200,data: {'clothing': '最 弱 ', 'uv': '高 ', 'car_washing': '不 宜 ', 'air_pollution_diffusion': '良'}}
```

3.4 异常码

异常响应示例:

4. 获取 24 小时温度数据接口

URL : <http://127.0.0.1:8000/api/v1/weather/<location>?temp24=1>

4.1 请求方式

GET

4.2 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

temp24,值恒定为 1,表示获取 24 小时温度数据

4.3 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 4.4
data	具体数据均在 data 中	{}	{'time': 时间数组 'temperature': 温度数组} 见下方展开
time	时间(data 中数据)	[char,char,...]	未来 24 小时的整点时刻,详细请见下方示例
temperature	温度(data 中数据)	[int,int,...]	未来 24 小时温度,详细请见下方示例

响应示例:

```
{'code':200,data: {'time': ['01:00', '02:00', '03:00', '04:00', '05:00', '06:00', '07:00', '08:00', '09:00', '10:00', '11:00', '12:00','13:00', '14:00', '15:00', '16:00', '17:00', '18:00','19:00', '20:00', '21:00', '22:00', '23:00', '24:00'], 'temperature': [23, 32, 19, 34, 29, 33, 23, 32, 19, 34, 29, 33, 23, 32, 19, 34, 29, 33, 23, 32, 19, 34, 29, 33]}}
```

4.4 异常码

异常响应示例:

5. 获取 24 小时风力数据接口

URL : http://127.0.0.1:8000/api/v1/weather/<location>?wind_scale24=1

5.1 请求方式

GET

5.2 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

wind_scale24,值恒定为 1,表示获取 24 小时风力数据

5.3 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 5.4
data	具体数据均在 data 中	{}	{‘time’:时间数组 ‘wind’:风力值数组}见下方展开
time	时间(data 中数据)	[char,char,...]	未来 24 小时的整点时刻,详细请见下方示例
wind	温度(data 中数据)	[int,int,...]	未来 24 小时风力值,详细请见下方示例

响应示例:

```
{‘code’:200,data: {‘time’: ['01:00', '02:00', '03:00', '04:00', '05:00', '06:00', '07:00', '08:00', '09:00', '10:00', '11:00', '12:00','13:00', '14:00', '15:00', '16:00', '17:00', '18:00','19:00', '20:00', '21:00', '22:00', '23:00', '24:00'], ‘wind’: [2, 3, 1, 3, 2, 3, 2, 2, 1, 4, 2, 3, 3, 3, 1, 4, 2, 3, 2, 3, 1, 4, 2, 3]}}
```

5.4 异常码

异常响应示例:

6. 获取 7 天预报数据接口

URL : http://127.0.0.1:8000/api/v1/weather/<location>?forecast_7d=1

6.1 请求方式

GET

6.2 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

forecast_7d,值恒定为 1,表示获取未来 7 天预报数据

6.3 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 6.4
data	具体数据均在 data 中	{}	{‘date’: 时间数组 , ‘weather’: 天气数组 , ‘temperature’: 温度数组 , ‘wind_direction’:风向数组, ‘wind_scale’:风力值数组} 见下方展开
date	时间(data 中数据)	[char,char, ...]	未来 7 天的星期表示, 子元素数据类型为 char, 详细请见下方示例
weather	天气(data 中数据)	[char,char, ...]	未来 7 天的天气数据, 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, ...]	未来 7 天的温度数据, 子元素

			数据类型为 char,详细请见下方示例
wind_direction	风向(data 中数据)	[char,char,...]	未来 7 天的风向数据, 子元素数据类型为 char,详细请见下方示例
wind_scale	风力(data 中数据)	[int,int,...]	未来 7 天的风力数据, 子元素数据类型为 int,详细请见下方示例

响应示例:

```
{'code':200,data: {'date': ['星期日', '星期一', '星期二', '星期三', '星期四', '星期五', '星期六'], 'weather': ['多云', '多云', '多云', '多云', '多云', '多云', '多云'], 'temperature': ['5/13', '5/13', '5/13', '5/13', '5/13', '5/13', '5/13'], 'wind_direction': ['西北', '西北', '西北', '西北', '西北', '西北', '西北'], 'wind_scale': [2, 3, 1, 3, 2, 3, 2]}}
```

6.4 异常码

异常响应示例:

7. 获取周边天气数据接口

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

7.1 请求方式

GET

7.2 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

surrounding_weather,值恒定为 1,表示获取未来 7 天预报数据

7.3 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': 地区数组 , 'weather': 天气数组 , 'temperature': 温度数组} 见下方展开
location	地区(data 中数据)	[char,char, ...]	周边地区的中文名称数据 , 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, ...]	各周边地区对应的温度数据 , 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, ...]	各周边地区对应的天气数据 , 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': ['和平区', '河东区', '河西区', '河北区'], 'weather': ['多

云', '多云', '多云', '多云'], 'temperature': ['5/13', '5/13', '5/13', '5/13']}]}

7.4 异常码

异常响应示例: