

API Document.

Value of base_url = **http://127.0.0.1:8000**

1. {{base_url}}/api/weather/

- This a Get method which accepts [year, region, page, year]
- region can be [East_Anglia, England, England_E_and_NE, England_N, England_NW_and_N_Wales, England_S, England_SE_and_Central_S, England_SW_and_S_Wales, England_and_Wales, Midlands, Northern_Ireland, Scotland, Scotland_E, Scotland_N , Scotland_W , UK , Wales]
- weather_param ['Tmax', 'Tmin', 'Tmean', 'AirFrost', 'Raindays1mm', 'Sunshine', 'Rainfall']
- year range can be from 1884-2023
- page value is auto generated by Django Pagination.
- The above api can display all records from database.
- Pagination is set 30 records.
- We can also apply filter based on region, weather parameter and year.
- To apply filter pass the filter as query parameter.

```
{
  "code": 100,
  "msg": "Record details fetched successfully.",
  "total_count": 140,
  "page_count": 30,
  "next": "http://127.0.0.1:8000/api/weather/?page=2&region=England&weather_parameter=Tmax",
  "previous": null,
  "result": [
    {
      "id": 82851,
      "region": "England",
      "weather_parameter": "Tmax",
      "year": 1884,
      "jan": 8.2,
      "feb": 7.6,
      "mar": 9.6,
      "apr": 10.7,
      "may": 15.9,
      "jun": 18.4,
      "jul": 20.3,
      "aug": 22.2,
      "sep": 18.4,
      "oct": 12.6,
      "nov": 8.2,
      "dec": 6.4,
      "win": 0.0,
    }
  ]
}
```

```

        "spr": 12.06,
        "sum": 20.3,
        "aut": 13.05,
        "ann": 13.22
    }
]
}

```

2. {{base_url}}/api/region_param_details/

- This a Get method which accepts [region, weather_param, lower_range_year, upper_range_year]
- region can be [East_Anglia, England, England_E_and_NE, England_N, England_NW_and_N_Wales, England_S, England_SE_and_Central_S, England_SW_and_S_Wales, England_and_Wales, Midlands, Northern_Ireland, Scotland, Scotland_E, Scotland_N , Scotland_W , UK , Wales]
- weather_param ['Tmax', 'Tmin', 'Tmean', 'AirFrost', 'Raindays1mm', 'Sunshine', 'Rainfall']
- year range can be from 1884-2023
- The API return seasonal and annual values for requested weather_param and region for mentioned range.

```

{
  "code": 100,
  "msg": "Data Got",
  "results": [
    {
      "region": "England",
      "weather_parameter": "Tmax",
      "year": 2000,
      "ann": 13.63,
      "aut": 13.79,
      "sum": 19.7,
      "spr": 12.84
    },
    {
      "region": "England",
      "weather_parameter": "Tmax",
      "year": 2001,
      "ann": 13.45,
      "aut": 14.52,
      "sum": 20.27,
      "spr": 12.18
    },
    {

```

```

        "region": "England",
        "weather_parameter": "Tmin",
        "year": 2000,
        "ann": 6.23,
        "aut": 6.96,
        "sum": 10.83,
        "spr": 4.76
    },
    {
        "region": "England",
        "weather_parameter": "Tmin",
        "year": 2001,
        "ann": 5.83,
        "aut": 7.69,
        "sum": 11.07,
        "spr": 4.06
    }
]
}

```

The above api will return seasonal and annual weather data for single region, single weather parameter for specified year for range of years. It takes region, weather_parameter and year in the query parameter.

3. {{base_url}}/api/get_value/

- This is a Get method which accepts [calculate, weather_param, field]
- field can be ['jan', 'feb', 'mar', 'apr', 'may', 'jun', 'jul', 'aug', 'sep', 'oct', 'nov', 'dec', 'win', 'spr', 'sum', 'aut', 'ann']
- calculate ['max', 'min']
- weather_param ['Tmax', 'Tmin', 'Tmean', 'AirFrost', 'Raindays1mm', 'Sunshine', 'Rainfall']
- **weather_param** should be a single value.
- **calculate** should be a single value.
- **field** should be a single value.
- even if multiple values are passed, last values for each param will be considered.
- The above api will get the min/max value for any specific weather_param.

```

{
    "code": 100,
    "msg": "Max Value for Tmax",
    "results": {
        "region": "East_Anglia",
        "year": 2007,
        "Max_value": 9.9
    }
}

```

```
}
```

4. `{{base_url}}/api/region_weather_param/`

- This a Get method which accepts [calculate, weather_param, month]
- month can be ['jan', 'feb', 'mar', 'apr', 'may', 'jun', 'jul', 'aug', 'sep', 'oct', 'nov', 'dec', 'win', 'spr', 'sum', 'aut', 'ann']
- calculate ['sum', 'avg']
- weather_param ['Tmax', 'Tmin', 'Tmean', 'AirFrost', 'Raindays1mm', 'Sunshine', 'Rainfall']
- **month** key have all month's name, seasonal and Annual mapping present.
- weather_param and month is not mandatory Query Param fields.
- calculate is mandatory Query Param field.
- This api will return list of dictionary where ' ' will be key and value will be sum/avg of the month parameter for all regions. (all years(140+) jan value average.)

```
{
  "code": 100,
  "msg": "Data Got",
  "results": {
    "AirFrost": [
      {
        "region": "East_Anglia",
        "jan_avg": 11.1625
      },
      {
        "region": "England",
        "jan_avg": 11.403124999999998
      }
    ],
    "Raindays1mm": [
      {
        "region": "East_Anglia",
        "jan_avg": 11.178947368421055
      },
      {
        "region": "England",
        "jan_avg": 13.218796992481211
      }
    ]
  }
}
```

5. {{base_url}}/api/weather__param__year/

- This a Get method which accepts [calculate, weather__param, month]
- month can be ['jan', 'feb', 'mar', 'apr', 'may', 'jun', 'jul', 'aug', 'sep', 'oct', 'nov', 'dec', 'win', 'spr', 'sum', 'aut', 'ann']
- calculate ['sum', 'avg']
- weather__param ['Tmax', 'Tmin', 'Tmean', 'AirFrost', 'Raindays1mm', 'Sunshine', 'Rainfall']
- **month** key have all month's name, seasonal and Annual mapping present.
- weather__param and month is not mandatory Query Param fields.
- calculate is mandatory Query Param field.
- This api will return list of dictionary where ' ' will be key and value will be sum/avg of the month parameter for all years.

(all 17 regions ann value average.)

```
{
  "code": 100,
  "msg": "Data Got",
  "results": {
    "Tmax": [
      {
        "year": 1884,
        "ann_avg": 12.26529411764706
      },
      {
        "year": 1885,
        "ann_avg": 11.158235294117643
      }
    ],
    "Raindays1mm": [
      {
        "year": 1884,
        "ann_avg": 4.963529411764706
      },
      {
        "year": 1885,
        "ann_avg": 3.8241176470588236
      }
    ]
  }
}
```

6. {{base_url}}/api/region__year/

- This a Get method which accepts [calculate, weather__param, month]
- month can be ['jan', 'feb', 'mar', 'apr', 'may', 'jun', 'jul', 'aug', 'sep', 'oct', 'nov', 'dec', 'win', 'spr', 'sum', 'aut', 'ann']
- calculate ['sum', 'avg']
- weather__param ['Tmax', 'Tmin', 'Tmean', 'AirFrost', 'Raindays1mm', 'Sunshine', 'Rainfall']
- **month** key have all month's name, seasonal and Annual mapping present.
- month is not mandatory Query Param field.
- calculate is mandatory Query Param field.
- weather__param is mandatory filed and should be a single field.
- This api will return list of dictionary where ' ' will be key and value will be sum/avg of the month parameter for all year. (all years(140+) jan value average.)

```
{
  "code": 100,
  "msg": "Data Got",
  "results": {
    "East_Anglia": {
      "jan_avg": 6.399999999999995,
      "ann_avg": 13.655000000000001
    },
    "England": {
      "jan_avg": 6.319999999999999,
      "ann_avg": 12.965285714285713
    }
  }
}
```