

```

{
  "cells": [
    {
      "cell_type": "code",
      "execution_count": 1,
      "id": "banner-europe",
      "metadata": {},
      "outputs": [],
      "source": [
        "import requests"
      ]
    },
    {
      "cell_type": "code",
      "execution_count": 2,
      "id": "official-symphony",
      "metadata": {},
      "outputs": [],
      "source": [
        "class getWeather:\n",
        "\n",
        "    data = None\n",
        "    count = None\n",
        "\n",
        "    def __init__(self, apiKey, city, unit):\n",
        "        self.apiKey = apiKey\n",
        "        self.unit = unit\n",
        "        self.city = city\n",
        "\n",
        "    def requestForecast(self, apiKey, city, unit, cast =
'forecast'):\n",
        "        url =
f'http://api.openweathermap.org/data/2.5/{cast}?appid={apiKey}&q={city}&m
ode=json&units={unit}' \n",
        "\n",
        "        response = requests.get(url).json()\n",
        "        self.count = response['cnt']\n",
        "        self.data = response\n",
        "\n",
        "        print('New data has been retrieved')\n",
        "\n",
        "        return response\n",
        "\n",
        "    def printForecast(self):\n",
        "        for day in range(self.count):\n",
        "            date = self.data['list'][day]['dt_txt']\n",
        "            temp = self.data['list'][day]['main']['temp']\n",
        "            humidity =
self.data['list'][day]['main']['humidity']\n",
        "            windSpeed = self.data['list'][day]['wind']['speed']\n",
        "            windDeg = self.data['list'][day]['wind']['deg']\n",
        "            weather =
self.data['list'][day]['weather'][0]['description']\n",

```

```

        "                weatherId =
self.data['list'][day]['weather'][0]['id']\n",
        "                cloud = self.data['list'][day]['clouds']['all']\n",
        "\n",
        "                print('\nDate/time:\t{}'.format(date))\n",
        "                print('Temperature: \t{} Celsius'.format(temp))\n",
        "                print('Humidity:\t{}'.format(humidity))\n",
        "                print('windspeed:\t{} m/s'.format(windSpeed))\n",
        "                print('windDeg:\t{}'.format(windDeg))\n",
        "                print('Clouds: \t{}'.format(cloud))\n",
        "                print('Weather ID: \t{}'.format(weatherId))\n",
        "                print('Weather: \t{}'.format(weather))"
    ]
},
{
    "cell_type": "code",
    "execution_count": 4,
    "id": "periodic-electron",
    "metadata": {},
    "outputs": [],
    "source": [
        "apiKey = '#####'\t#change
key\n",
        "unit = 'metric'\n",
        "#city = input('Enter city: ')\n",
        "city = 'Arnhem'\n",
        "cast = 'forecast'"
    ]
},
{
    "cell_type": "code",
    "execution_count": 5,
    "id": "prospective-seeking",
    "metadata": {},
    "outputs": [],
    "source": [
        "weatherData = getWeather(apiKey, city, unit) #not needed"
    ]
},
{
    "cell_type": "code",
    "execution_count": 6,
    "id": "fresh-google",
    "metadata": {},
    "outputs": [
        {
            "name": "stdout",
            "output_type": "stream",
            "text": [
                "New data has been retrieved\n"
            ]
        }
    ],
    "source": [

```

```

    "data = weatherData.requestForecast(apiKey, city, unit) \n"
  ]
},
{
  "cell_type": "code",
  "execution_count": 7,
  "id": "equivalent-nurse",
  "metadata": {},
  "outputs": [
    {
      "name": "stdout",
      "output_type": "stream",
      "text": [
        "\n",
        "Date/time:\t2021-06-10 12:00:00\n",
        "Temperature: \t25.34 Celsius\n",
        "Humidity:\t42%\n",
        "windspeed:\t3.23 m/s\n",
        "windDeg:\t257\n",
        "Clouds: \t15%\n",
        "Weather ID: \t801\n",
        "Weather: \tfew clouds\n",
        "\n",
        "Date/time:\t2021-06-10 15:00:00\n",
        "Temperature: \t24.94 Celsius\n",
        "Humidity:\t46%\n",
        "windspeed:\t2.67 m/s\n",
        "windDeg:\t252\n",
        "Clouds: \t15%\n",
        "Weather ID: \t801\n",
        "Weather: \tfew clouds\n",
        "\n",
        "Date/time:\t2021-06-10 18:00:00\n",
        "Temperature: \t23.17 Celsius\n",
        "Humidity:\t60%\n",
        "windspeed:\t1.62 m/s\n",
        "windDeg:\t286\n",
        "Clouds: \t28%\n",
        "Weather ID: \t802\n",
        "Weather: \tscattered clouds\n",
        "\n",
        "Date/time:\t2021-06-10 21:00:00\n",
        "Temperature: \t16.79 Celsius\n",
        "Humidity:\t90%\n",
        "windspeed:\t2.47 m/s\n",
        "windDeg:\t346\n",
        "Clouds: \t100%\n",
        "Weather ID: \t804\n",
        "Weather: \tovercast clouds\n",
        "\n",
        "Date/time:\t2021-06-11 00:00:00\n",
        "Temperature: \t15.03 Celsius\n",
        "Humidity:\t92%\n",
        "windspeed:\t0.89 m/s\n",

```

"windDeg:\t299\n",
"Clouds: \t78%\n",
"Weather ID: \t803\n",
"Weather: \tbroken clouds\n",
"\n",
"Date/time:\t2021-06-11 03:00:00\n",
"Temperature: \t14.05 Celsius\n",
"Humidity:\t93%\n",
"windspeed:\t2.04 m/s\n",
"windDeg:\t237\n",
"Clouds: \t20%\n",
"Weather ID: \t801\n",
"Weather: \tfew clouds\n",
"\n",
"Date/time:\t2021-06-11 06:00:00\n",
"Temperature: \t17.04 Celsius\n",
"Humidity:\t81%\n",
"windspeed:\t3.31 m/s\n",
"windDeg:\t240\n",
"Clouds: \t16%\n",
"Weather ID: \t801\n",
"Weather: \tfew clouds\n",
"\n",
"Date/time:\t2021-06-11 09:00:00\n",
"Temperature: \t21.49 Celsius\n",
"Humidity:\t67%\n",
"windspeed:\t4.59 m/s\n",
"windDeg:\t240\n",
"Clouds: \t7%\n",
"Weather ID: \t800\n",
"Weather: \tclear sky\n",
"\n",
"Date/time:\t2021-06-11 12:00:00\n",
"Temperature: \t24.28 Celsius\n",
"Humidity:\t60%\n",
"windspeed:\t5.18 m/s\n",
"windDeg:\t253\n",
"Clouds: \t38%\n",
"Weather ID: \t802\n",
"Weather: \tscattered clouds\n",
"\n",
"Date/time:\t2021-06-11 15:00:00\n",
"Temperature: \t23.76 Celsius\n",
"Humidity:\t62%\n",
"windspeed:\t4.45 m/s\n",
"windDeg:\t265\n",
"Clouds: \t87%\n",
"Weather ID: \t804\n",
"Weather: \tovercast clouds\n",
"\n",
"Date/time:\t2021-06-11 18:00:00\n",
"Temperature: \t21.68 Celsius\n",
"Humidity:\t72%\n",
"windspeed:\t4.74 m/s\n",

"windDeg:\t276\n",
"Clouds: \t73%\n",
"Weather ID: \t803\n",
"Weather: \tbroken clouds\n",
"\n",
"Date/time:\t2021-06-11 21:00:00\n",
"Temperature: \t16.95 Celsius\n",
"Humidity:\t85%\n",
"windspeed:\t2.83 m/s\n",
"windDeg:\t265\n",
"Clouds: \t100%\n",
"Weather ID: \t804\n",
"Weather: \tovercast clouds\n",
"\n",
"Date/time:\t2021-06-12 00:00:00\n",
"Temperature: \t15.07 Celsius\n",
"Humidity:\t93%\n",
"windspeed:\t2.99 m/s\n",
"windDeg:\t240\n",
"Clouds: \t92%\n",
"Weather ID: \t804\n",
"Weather: \tovercast clouds\n",
"\n",
"Date/time:\t2021-06-12 03:00:00\n",
"Temperature: \t15.7 Celsius\n",
"Humidity:\t87%\n",
"windspeed:\t2.88 m/s\n",
"windDeg:\t242\n",
"Clouds: \t86%\n",
"Weather ID: \t804\n",
"Weather: \tovercast clouds\n",
"\n",
"Date/time:\t2021-06-12 06:00:00\n",
"Temperature: \t16.82 Celsius\n",
"Humidity:\t89%\n",
"windspeed:\t3.45 m/s\n",
"windDeg:\t284\n",
"Clouds: \t93%\n",
"Weather ID: \t804\n",
"Weather: \tovercast clouds\n",
"\n",
"Date/time:\t2021-06-12 09:00:00\n",
"Temperature: \t19.49 Celsius\n",
"Humidity:\t62%\n",
"windspeed:\t5 m/s\n",
"windDeg:\t293\n",
"Clouds: \t100%\n",
"Weather ID: \t804\n",
"Weather: \tovercast clouds\n",
"\n",
"Date/time:\t2021-06-12 12:00:00\n",
"Temperature: \t21.12 Celsius\n",
"Humidity:\t50%\n",
"windspeed:\t5.43 m/s\n",

"windDeg:\t309\n",
"Clouds: \t98%\n",
"Weather ID: \t804\n",
"Weather: \tovercast clouds\n",
"\n",
"Date/time:\t2021-06-12 15:00:00\n",
"Temperature: \t19.54 Celsius\n",
"Humidity:\t53%\n",
"windspeed:\t6.11 m/s\n",
"windDeg:\t327\n",
"Clouds: \t100%\n",
"Weather ID: \t804\n",
"Weather: \tovercast clouds\n",
"\n",
"Date/time:\t2021-06-12 18:00:00\n",
"Temperature: \t16.27 Celsius\n",
"Humidity:\t65%\n",
"windspeed:\t4.85 m/s\n",
"windDeg:\t327\n",
"Clouds: \t67%\n",
"Weather ID: \t803\n",
"Weather: \tbroken clouds\n",
"\n",
"Date/time:\t2021-06-12 21:00:00\n",
"Temperature: \t11.07 Celsius\n",
"Humidity:\t80%\n",
"windspeed:\t2.67 m/s\n",
"windDeg:\t315\n",
"Clouds: \t8%\n",
"Weather ID: \t800\n",
"Weather: \tclear sky\n",
"\n",
"Date/time:\t2021-06-13 00:00:00\n",
"Temperature: \t9.23 Celsius\n",
"Humidity:\t91%\n",
"windspeed:\t2.44 m/s\n",
"windDeg:\t302\n",
"Clouds: \t16%\n",
"Weather ID: \t801\n",
"Weather: \tfew clouds\n",
"\n",
"Date/time:\t2021-06-13 03:00:00\n",
"Temperature: \t8.57 Celsius\n",
"Humidity:\t94%\n",
"windspeed:\t1.62 m/s\n",
"windDeg:\t311\n",
"Clouds: \t5%\n",
"Weather ID: \t800\n",
"Weather: \tclear sky\n",
"\n",
"Date/time:\t2021-06-13 06:00:00\n",
"Temperature: \t13.13 Celsius\n",
"Humidity:\t76%\n",
"windspeed:\t1.47 m/s\n",

"windDeg:\t328\n",
"Clouds: \t4%\n",
"Weather ID: \t800\n",
"Weather: \tclear sky\n",
"\n",
"Date/time:\t2021-06-13 09:00:00\n",
"Temperature: \t18.05 Celsius\n",
"Humidity:\t58%\n",
"windspeed:\t1.92 m/s\n",
"windDeg:\t300\n",
"Clouds: \t2%\n",
"Weather ID: \t800\n",
"Weather: \tclear sky\n",
"\n",
"Date/time:\t2021-06-13 12:00:00\n",
"Temperature: \t21.28 Celsius\n",
"Humidity:\t52%\n",
"windspeed:\t2.43 m/s\n",
"windDeg:\t298\n",
"Clouds: \t8%\n",
"Weather ID: \t800\n",
"Weather: \tclear sky\n",
"\n",
"Date/time:\t2021-06-13 15:00:00\n",
"Temperature: \t23.01 Celsius\n",
"Humidity:\t51%\n",
"windspeed:\t2.34 m/s\n",
"windDeg:\t326\n",
"Clouds: \t15%\n",
"Weather ID: \t801\n",
"Weather: \tfew clouds\n",
"\n",
"Date/time:\t2021-06-13 18:00:00\n",
"Temperature: \t20.41 Celsius\n",
"Humidity:\t68%\n",
"windspeed:\t2.74 m/s\n",
"windDeg:\t27\n",
"Clouds: \t9%\n",
"Weather ID: \t800\n",
"Weather: \tclear sky\n",
"\n",
"Date/time:\t2021-06-13 21:00:00\n",
"Temperature: \t14.79 Celsius\n",
"Humidity:\t88%\n",
"windspeed:\t2.38 m/s\n",
"windDeg:\t50\n",
"Clouds: \t0%\n",
"Weather ID: \t800\n",
"Weather: \tclear sky\n",
"\n",
"Date/time:\t2021-06-14 00:00:00\n",
"Temperature: \t13.56 Celsius\n",
"Humidity:\t91%\n",
"windspeed:\t1.79 m/s\n",

"windDeg:\t79\n",
"Clouds: \t0%\n",
"Weather ID: \t800\n",
"Weather: \tclear sky\n",
"\n",
"Date/time:\t2021-06-14 03:00:00\n",
"Temperature: \t12.77 Celsius\n",
"Humidity:\t92%\n",
"windspeed:\t2.18 m/s\n",
"windDeg:\t121\n",
"Clouds: \t0%\n",
"Weather ID: \t800\n",
"Weather: \tclear sky\n",
"\n",
"Date/time:\t2021-06-14 06:00:00\n",
"Temperature: \t16.8 Celsius\n",
"Humidity:\t79%\n",
"windspeed:\t2.65 m/s\n",
"windDeg:\t146\n",
"Clouds: \t0%\n",
"Weather ID: \t800\n",
"Weather: \tclear sky\n",
"\n",
"Date/time:\t2021-06-14 09:00:00\n",
"Temperature: \t21.83 Celsius\n",
"Humidity:\t57%\n",
"windspeed:\t3.27 m/s\n",
"windDeg:\t167\n",
"Clouds: \t0%\n",
"Weather ID: \t800\n",
"Weather: \tclear sky\n",
"\n",
"Date/time:\t2021-06-14 12:00:00\n",
"Temperature: \t25.29 Celsius\n",
"Humidity:\t47%\n",
"windspeed:\t3.37 m/s\n",
"windDeg:\t194\n",
"Clouds: \t0%\n",
"Weather ID: \t800\n",
"Weather: \tclear sky\n",
"\n",
"Date/time:\t2021-06-14 15:00:00\n",
"Temperature: \t26.66 Celsius\n",
"Humidity:\t43%\n",
"windspeed:\t3.68 m/s\n",
"windDeg:\t248\n",
"Clouds: \t5%\n",
"Weather ID: \t800\n",
"Weather: \tclear sky\n",
"\n",
"Date/time:\t2021-06-14 18:00:00\n",
"Temperature: \t23.29 Celsius\n",
"Humidity:\t63%\n",
"windspeed:\t4.02 m/s\n",


```

"windDeg:\t292\n",
"Clouds: \t3%\n",
"Weather ID: \t800\n",
"Weather: \tclear sky\n",
"\n",
"Date/time:\t2021-06-14 21:00:00\n",
"Temperature: \t16.96 Celsius\n",
"Humidity:\t74%\n",
"windspeed:\t3.68 m/s\n",
"windDeg:\t315\n",
"Clouds: \t65%\n",
"Weather ID: \t803\n",
"Weather: \tbroken clouds\n",
"\n",
"Date/time:\t2021-06-15 00:00:00\n",
"Temperature: \t14.82 Celsius\n",
"Humidity:\t93%\n",
"windspeed:\t4.12 m/s\n",
"windDeg:\t331\n",
"Clouds: \t74%\n",
"Weather ID: \t803\n",
"Weather: \tbroken clouds\n",
"\n",
"Date/time:\t2021-06-15 03:00:00\n",
"Temperature: \t14.38 Celsius\n",
"Humidity:\t75%\n",
"windspeed:\t3.29 m/s\n",
"windDeg:\t339\n",
"Clouds: \t100%\n",
"Weather ID: \t804\n",
"Weather: \tovercast clouds\n",
"\n",
"Date/time:\t2021-06-15 06:00:00\n",
"Temperature: \t14.41 Celsius\n",
"Humidity:\t72%\n",
"windspeed:\t3.32 m/s\n",
"windDeg:\t330\n",
"Clouds: \t100%\n",
"Weather ID: \t804\n",
"Weather: \tovercast clouds\n",
"\n",
"Date/time:\t2021-06-15 09:00:00\n",
"Temperature: \t17.92 Celsius\n",
"Humidity:\t53%\n",
"windspeed:\t3.3 m/s\n",
"windDeg:\t324\n",
"Clouds: \t65%\n",
"Weather ID: \t803\n",
"Weather: \tbroken clouds\n"
]
}
],
"source": [
"weatherData.printForecast()"
]

```

```

]
},
{
  "cell_type": "code",
  "execution_count": 8,
  "id": "stunning-australian",
  "metadata": {},
  "outputs": [],
  "source": [
    "from fastai.collab import *\n",
    "from fastai.tabular.all import *"
  ]
},
{
  "cell_type": "code",
  "execution_count": 9,
  "id": "supported-magnet",
  "metadata": {},
  "outputs": [],
  "source": [
    "forecast_raw = pd.DataFrame(columns =
['date_time', 'temp', 'humid', 'wspd', 'wdeg', 'cloud', 'weather_id'])"
  ]
},
{
  "cell_type": "code",
  "execution_count": 10,
  "id": "flexible-native",
  "metadata": {},
  "outputs": [],
  "source": [
    "number_of_rows = data['cnt']\n",
    "\n",
    "for i in range(number_of_rows):\n",
    "    date_time = data['list'][i]['dt_txt']\n",
    "    date = date_time[:10]\n",
    "    temp = data['list'][i]['main']['temp']\n",
    "    humidity = data['list'][i]['main']['humidity']\n",
    "    windSpeed = data['list'][i]['wind']['speed']\n",
    "    windDeg = data['list'][i]['wind']['deg']\n",
    "    weatherId = data['list'][i]['weather'][0]['id']\n",
    "    cloud = data['list'][i]['clouds']['all']\n",
    "    \n",
    "    forecast_raw = forecast_raw.append({\n",
    "        'date_time':date_time,\n",
    "        'temp':temp,\n",
    "        'humid':humidity,\n",
    "        'wspd':windSpeed,\n",
    "        'wdeg':windDeg,\n",
    "        'cloud':cloud,\n",
    "        'weather_id':weatherId\n",
    "    }, ignore_index=True)"
  ]
},

```

```

{
  "cell_type": "code",
  "execution_count": 11,
  "id": "cellular-preference",
  "metadata": {},
  "outputs": [
    {
      "data": {
        "text/html": [
          "<div>\n",
          "<style scoped>\n",
          "    .dataframe tbody tr th:only-of-type {\n",
          "        vertical-align: middle;\n",
          "    }\n",
          "\n",
          "    .dataframe tbody tr th {\n",
          "        vertical-align: top;\n",
          "    }\n",
          "\n",
          "    .dataframe thead th {\n",
          "        text-align: right;\n",
          "    }\n",
          "</style>\n",
          "<table border=\"1\" class=\"dataframe\">\n",
          "  <thead>\n",
          "    <tr style=\"text-align: right;\">\n",
          "      <th></th>\n",
          "      <th>date_time</th>\n",
          "      <th>temp</th>\n",
          "      <th>humid</th>\n",
          "      <th>wspd</th>\n",
          "      <th>wdeg</th>\n",
          "      <th>cloud</th>\n",
          "      <th>weather_id</th>\n",
          "    </tr>\n",
          "  </thead>\n",
          "  <tbody>\n",
          "    <tr>\n",
          "      <th>0</th>\n",
          "      <td>2021-06-10 12:00:00</td>\n",
          "      <td>25.34</td>\n",
          "      <td>42</td>\n",
          "      <td>3.23</td>\n",
          "      <td>257</td>\n",
          "      <td>15</td>\n",
          "      <td>801</td>\n",
          "    </tr>\n",
          "    <tr>\n",
          "      <th>1</th>\n",
          "      <td>2021-06-10 15:00:00</td>\n",
          "      <td>24.94</td>\n",
          "      <td>46</td>\n",
          "      <td>2.67</td>\n",
          "      <td>252</td>\n",

```

```

"      <td>15</td>\n",
"      <td>801</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>2021-06-10 18:00:00</td>\n",
"      <td>23.17</td>\n",
"      <td>60</td>\n",
"      <td>1.62</td>\n",
"      <td>286</td>\n",
"      <td>28</td>\n",
"      <td>802</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>2021-06-10 21:00:00</td>\n",
"      <td>16.79</td>\n",
"      <td>90</td>\n",
"      <td>2.47</td>\n",
"      <td>346</td>\n",
"      <td>100</td>\n",
"      <td>804</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>4</th>\n",
"      <td>2021-06-11 00:00:00</td>\n",
"      <td>15.03</td>\n",
"      <td>92</td>\n",
"      <td>0.89</td>\n",
"      <td>299</td>\n",
"      <td>78</td>\n",
"      <td>803</td>\n",
"    </tr>\n",
"  </tbody>\n",
"</table>\n",
"</div>"
],
"text/plain": [
"      date_time    temp humid  wspd wdeg cloud
weather_id\n",
"0  2021-06-10 12:00:00  25.34    42   3.23  257    15
801\n",
"1  2021-06-10 15:00:00  24.94    46   2.67  252    15
801\n",
"2  2021-06-10 18:00:00  23.17    60   1.62  286    28
802\n",
"3  2021-06-10 21:00:00  16.79    90   2.47  346   100
804\n",
"4  2021-06-11 00:00:00  15.03    92   0.89  299    78      803"
]
},
"execution_count": 11,
"metadata": {},
"output_type": "execute_result"

```

```

    }
  ],
  "source": [
    "forecast_raw.head()"
  ]
},
{
  "cell_type": "code",
  "execution_count": 12,
  "id": "introductory-major",
  "metadata": {},
  "outputs": [],
  "source": [
    "#remove data at night\n",
    "for i in range(len(forecast_raw)):\n",
    "    if int(forecast_raw['date_time'][i][11:13]) < 4 or\nint(forecast_raw['date_time'][i][11:13]) > 22: \n",
    "        forecast_raw.drop(index=i, axis=0, inplace=True)"
  ]
},
{
  "cell_type": "code",
  "execution_count": 13,
  "id": "tough-married",
  "metadata": {},
  "outputs": [
    {
      "data": {
        "text/html": [
          "<div>\n",
          "<style scoped>\n",
          "    .dataframe tbody tr th:only-of-type {\n",
          "        vertical-align: middle;\n",
          "    }\n",
          "\n",
          "    .dataframe tbody tr th {\n",
          "        vertical-align: top;\n",
          "    }\n",
          "\n",
          "    .dataframe thead th {\n",
          "        text-align: right;\n",
          "    }\n",
          "</style>\n",
          "<table border=\"1\" class=\"dataframe\">\n",
          "  <thead>\n",
          "    <tr style=\"text-align: right;\">\n",
          "      <th></th>\n",
          "      <th>index</th>\n",
          "      <th>date_time</th>\n",
          "      <th>temp</th>\n",
          "      <th>humid</th>\n",
          "      <th>wspd</th>\n",
          "      <th>wdeg</th>\n",
          "      <th>cloud</th>

```

```

"      <th>weather_id</th>\n",
"    </tr>\n",
"  </thead>\n",
"  <tbody>\n",
"    <tr>\n",
"      <th>0</th>\n",
"      <td>0</td>\n",
"      <td>2021-06-10 12:00:00</td>\n",
"      <td>25.34</td>\n",
"      <td>42</td>\n",
"      <td>3.23</td>\n",
"      <td>257</td>\n",
"      <td>15</td>\n",
"      <td>801</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>1</th>\n",
"      <td>1</td>\n",
"      <td>2021-06-10 15:00:00</td>\n",
"      <td>24.94</td>\n",
"      <td>46</td>\n",
"      <td>2.67</td>\n",
"      <td>252</td>\n",
"      <td>15</td>\n",
"      <td>801</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>2</th>\n",
"      <td>2</td>\n",
"      <td>2021-06-10 18:00:00</td>\n",
"      <td>23.17</td>\n",
"      <td>60</td>\n",
"      <td>1.62</td>\n",
"      <td>286</td>\n",
"      <td>28</td>\n",
"      <td>802</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>3</td>\n",
"      <td>2021-06-10 21:00:00</td>\n",
"      <td>16.79</td>\n",
"      <td>90</td>\n",
"      <td>2.47</td>\n",
"      <td>346</td>\n",
"      <td>100</td>\n",
"      <td>804</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>4</th>\n",
"      <td>6</td>\n",
"      <td>2021-06-11 06:00:00</td>\n",
"      <td>17.04</td>\n",
"      <td>81</td>

```

```

        "        <td>3.31</td>\n",
        "        <td>240</td>\n",
        "        <td>16</td>\n",
        "        <td>801</td>\n",
        "    </tr>\n",
        " </tbody>\n",
        "</table>\n",
        "</div>"
    ],
    "text/plain": [
        "    index          date_time    temp humid  wspd wdeg cloud
weather_id\n",
        "0         0  2021-06-10 12:00:00  25.34   42   3.23  257   15
801\n",
        "1         1  2021-06-10 15:00:00  24.94   46   2.67  252   15
801\n",
        "2         2  2021-06-10 18:00:00  23.17   60   1.62  286   28
802\n",
        "3         3  2021-06-10 21:00:00  16.79   90   2.47  346  100
804\n",
        "4         6  2021-06-11 06:00:00  17.04   81   3.31  240   16
801"
    ]
},
    "execution_count": 13,
    "metadata": {},
    "output_type": "execute_result"
}
],
"source": [
    "forecast_raw.reset_index(inplace=True)\n",
    "forecast_raw.head()"
]
},
{
    "cell_type": "code",
    "execution_count": 20,
    "id": "eligible-essex",
    "metadata": {},
    "outputs": [],
    "source": [
        "#To find the value with the most apperance\n",
        "from collections import Counter\n",
        "\n",
        "def most_frequent(lst):\n",
        "    data = Counter(lst)\n",
        "    return data.most_common(1)[0][0]"
    ]
},
{
    "cell_type": "code",
    "execution_count": 21,
    "id": "fourth-section",
    "metadata": {},

```

```

"outputs": [],
"source": [
    "def merge_data(weather):\n",
    "    new_data = pd.DataFrame(columns = ['date',\n",
    "    '\n",
    "'mor_temp','mor_humid','mor_wspd','mor_wdeg','mor_cloud','mor_weather_id'\n",
    ',\n",
    "'noo_temp','noo_humid','noo_wspd','noo_wdeg','noo_cloud','noo_weather_id'\n",
    ',\n",
    "'aft_temp','aft_humid','aft_wspd','aft_wdeg','aft_cloud','aft_weather_id'\n",
    ',)\n",
    "    \n",
    "    mor_count, mor_temp, mor_humid, mor_wspd, mor_wdeg, mor_cloud =\n",
    "[0]*6\n",
    "    noo_count, noo_temp, noo_humid, noo_wspd, noo_wdeg, noo_cloud =\n",
    "[0]*6\n",
    "    aft_count, aft_temp, aft_humid, aft_wspd, aft_wdeg, aft_cloud =\n",
    "[0]*6\n",
    "    \n",
    "    mor_weather_id=[]\n",
    "    noo_weather_id=[]\n",
    "    aft_weather_id=[]\n",
    "    \n",
    "    current_date = weather.loc[0, 'date_time'][0:10]\n",
    "    \n",
    "    for i in range(len(weather)):\n",
    "        if weather.loc[i, 'date_time'][0:10] != current_date:\n",
    "            new_data = new_data.append({\n",
    "                'date':current_date,\n",
    "                'mor_temp':mor_temp/mor_count,\n",
    "                'mor_humid':mor_humid/mor_count,\n",
    "                'mor_wspd':mor_wspd/mor_count,\n",
    "                'mor_wdeg':mor_wdeg/mor_count,\n",
    "                'mor_cloud':mor_cloud/mor_count,\n",
    "                'mor_weather_id':most_frequent(mor_weather_id),\n",
    "                'noo_temp':noo_temp/noo_count,\n",
    "                'noo_humid':noo_humid/noo_count,\n",
    "                'noo_wspd':noo_wspd/noo_count,\n",
    "                'noo_wdeg':noo_wdeg/noo_count,\n",
    "                'noo_cloud':noo_cloud/noo_count,\n",
    "                'noo_weather_id':most_frequent(noo_weather_id),\n",
    "                'aft_temp':aft_temp/aft_count,\n",
    "                'aft_humid':aft_humid/aft_count,\n",
    "                'aft_wspd':aft_wspd/aft_count,\n",
    "                'aft_wdeg':aft_wdeg/aft_count,\n",
    "                'aft_cloud':aft_cloud/aft_count,\n",
    "                'aft_weather_id':most_frequent(aft_weather_id)\n",
    "            }, ignore_index=True)\n",
    "            \n",
    "            mor_count, mor_temp, mor_humid, mor_wspd, mor_wdeg,\n",
    "mor_cloud = [0]*6\n",

```



```

        "            noo_count, noo_temp, noo_humid, noo_wspd, noo_wdeg,
noo_cloud = [0]*6\n",
        "            aft_count, aft_temp, aft_humid, aft_wspd, aft_wdeg,
aft_cloud = [0]*6\n",
        "            \n",
        "            mor_weather_id=[]\n",
        "            noo_weather_id=[]\n",
        "            aft_weather_id=[]\n",
        "        \n",
        "            current_date = weather.loc[i, 'date_time'][0:10]\n",
        "        \n",
        "        if int(weather.loc[i, 'date_time'][11:13]) < 11:\n",
        "            mor_count += 1\n",
        "            mor_temp += weather.loc[i, 'temp']\n",
        "            mor_humid += weather.loc[i, 'humid']\n",
        "            mor_wspd += weather.loc[i, 'wspd']\n",
        "            mor_wdeg += weather.loc[i, 'wdeg']\n",
        "            mor_cloud += weather.loc[i, 'cloud']\n",
        "            mor_weather_id.append(weather['weather_id'][i])\n",
        "        \n",
        "        elif int(weather.loc[i, 'date_time'][11:13]) < 16:\n",
        "            noo_count += 1\n",
        "            noo_temp += weather.loc[i, 'temp']\n",
        "            noo_humid += weather.loc[i, 'humid']\n",
        "            noo_wspd += weather.loc[i, 'wspd']\n",
        "            noo_wdeg += weather.loc[i, 'wdeg']\n",
        "            noo_cloud += weather.loc[i, 'cloud']\n",
        "            noo_weather_id.append(weather['weather_id'][i])\n",
        "        \n",
        "        else:\n",
        "            aft_count += 1\n",
        "            aft_temp += weather.loc[i, 'temp']\n",
        "            aft_humid += weather.loc[i, 'humid']\n",
        "            aft_wspd += weather.loc[i, 'wspd']\n",
        "            aft_wdeg += weather.loc[i, 'wdeg']\n",
        "            aft_cloud += weather.loc[i, 'cloud']\n",
        "            aft_weather_id.append(weather['weather_id'][i])\n",
        "        \n",
        "        return new_data"
    ]
},
{
    "cell_type": "code",
    "execution_count": 22,
    "id": "found-cleanup",
    "metadata": {},
    "outputs": [],
    "source": [
        "forecast_data = merge_data(forecast_raw)"
    ]
},
{
    "cell_type": "code",
    "execution_count": 23,

```

```

"id": "unavailable-horizon",
"metadata": {},
"outputs": [
  {
    "data": {
      "text/html": [
        "<div>\n",
        "<style scoped>\n",
        "    .dataframe tbody tr th:only-of-type {\n",
        "        vertical-align: middle;\n",
        "    }\n",
        "\n",
        "    .dataframe tbody tr th {\n",
        "        vertical-align: top;\n",
        "    }\n",
        "\n",
        "    .dataframe thead th {\n",
        "        text-align: right;\n",
        "    }\n",
        "</style>\n",
        "<table border=\"1\" class=\"dataframe\">\n",
        "  <thead>\n",
        "    <tr style=\"text-align: right;\">\n",
        "      <th></th>\n",
        "      <th>date</th>\n",
        "      <th>mor_temp</th>\n",
        "      <th>mor_humid</th>\n",
        "      <th>mor_wspd</th>\n",
        "      <th>mor_wdeg</th>\n",
        "      <th>mor_cloud</th>\n",
        "      <th>mor_weather_id</th>\n",
        "      <th>noo_temp</th>\n",
        "      <th>noo_humid</th>\n",
        "      <th>noo_wspd</th>\n",
        "      <th>noo_wdeg</th>\n",
        "      <th>noo_cloud</th>\n",
        "      <th>noo_weather_id</th>\n",
        "      <th>aft_temp</th>\n",
        "      <th>aft_humid</th>\n",
        "      <th>aft_wspd</th>\n",
        "      <th>aft_wdeg</th>\n",
        "      <th>aft_cloud</th>\n",
        "      <th>aft_weather_id</th>\n",
        "    </tr>\n",
        "  </thead>\n",
        "  <tbody>\n",
        "    <tr>\n",
        "      <th>0</th>\n",
        "      <td>2021-06-10</td>\n",
        "      <td>25.340000</td>\n",
        "      <td>42.000000</td>\n",
        "      <td>3.230000</td>\n",
        "      <td>257.000000</td>\n",
        "      <td>15.000000</td>\n",

```

```

"      <td>801</td>\n",
"      <td>24.94</td>\n",
"      <td>46.0</td>\n",
"      <td>2.67</td>\n",
"      <td>252.0</td>\n",
"      <td>15.0</td>\n",
"      <td>801</td>\n",
"      <td>19.980</td>\n",
"      <td>75.0</td>\n",
"      <td>2.045</td>\n",
"      <td>316.0</td>\n",
"      <td>64.0</td>\n",
"      <td>802</td>\n",
"    </tr>\n",
"  <tr>\n",
"    <th>1</th>\n",
"    <td>2021-06-11</td>\n",
"    <td>20.936667</td>\n",
"    <td>69.333333</td>\n",
"    <td>4.360000</td>\n",
"    <td>244.333333</td>\n",
"    <td>20.333333</td>\n",
"    <td>801</td>\n",
"    <td>23.76</td>\n",
"    <td>62.0</td>\n",
"    <td>4.45</td>\n",
"    <td>265.0</td>\n",
"    <td>87.0</td>\n",
"    <td>804</td>\n",
"    <td>19.315</td>\n",
"    <td>78.5</td>\n",
"    <td>3.785</td>\n",
"    <td>270.5</td>\n",
"    <td>86.5</td>\n",
"    <td>803</td>\n",
"  </tr>\n",
"  <tr>\n",
"    <th>2</th>\n",
"    <td>2021-06-12</td>\n",
"    <td>19.143333</td>\n",
"    <td>67.000000</td>\n",
"    <td>4.626667</td>\n",
"    <td>295.333333</td>\n",
"    <td>97.000000</td>\n",
"    <td>804</td>\n",
"    <td>19.54</td>\n",
"    <td>53.0</td>\n",
"    <td>6.11</td>\n",
"    <td>327.0</td>\n",
"    <td>100.0</td>\n",
"    <td>804</td>\n",
"    <td>13.670</td>\n",
"    <td>72.5</td>\n",
"    <td>3.760</td>\n",

```

```

"      <td>321.0</td>\n",
"      <td>37.5</td>\n",
"      <td>803</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>2021-06-13</td>\n",
"      <td>17.486667</td>\n",
"      <td>62.000000</td>\n",
"      <td>1.940000</td>\n",
"      <td>308.666667</td>\n",
"      <td>4.666667</td>\n",
"      <td>800</td>\n",
"      <td>23.01</td>\n",
"      <td>51.0</td>\n",
"      <td>2.34</td>\n",
"      <td>326.0</td>\n",
"      <td>15.0</td>\n",
"      <td>801</td>\n",
"      <td>17.600</td>\n",
"      <td>78.0</td>\n",
"      <td>2.560</td>\n",
"      <td>38.5</td>\n",
"      <td>4.5</td>\n",
"      <td>800</td>\n",
"    </tr>\n",
"  </tbody>\n",
"</table>\n",
"</div>"
],
"text/plain": [
"      date      mor_temp  mor_humid  mor_wspd  mor_wdeg
mor_cloud  \\n",
"0  2021-06-10  25.340000  42.000000  3.230000  257.000000
15.000000  \n",
"1  2021-06-11  20.936667  69.333333  4.360000  244.333333
20.333333  \n",
"2  2021-06-12  19.143333  67.000000  4.626667  295.333333
97.000000  \n",
"3  2021-06-13  17.486667  62.000000  1.940000  308.666667
4.666667  \n",
"  \n",
"      mor_weather_id  noo_temp  noo_humid  noo_wspd  noo_wdeg
noo_cloud  \\n",
"0              801      24.94      46.0      2.67      252.0
15.0  \n",
"1              801      23.76      62.0      4.45      265.0
87.0  \n",
"2              804      19.54      53.0      6.11      327.0
100.0  \n",
"3              800      23.01      51.0      2.34      326.0
15.0  \n",
"  \n",

```

```

"    noo_weather_id  aft_temp  aft_humid  aft_wspd  aft_wdeg
aft_cloud  \\n",
"0          801      19.980      75.0      2.045      316.0
64.0  \n",
"1          804      19.315      78.5      3.785      270.5
86.5  \n",
"2          804      13.670      72.5      3.760      321.0
37.5  \n",
"3          801      17.600      78.0      2.560      38.5
4.5  \n",
" \n",
"    aft_weather_id  \n",
"0          802  \n",
"1          803  \n",
"2          803  \n",
"3          800  "
]
},
"execution_count": 23,
"metadata": {},
"output_type": "execute_result"
}
],
"source": [
    "forecast_data.head()"
]
},
{
    "cell_type": "code",
    "execution_count": null,
    "id": "dirty-correction",
    "metadata": {},
    "outputs": [],
    "source": [
        "forecast_data.info()"
    ]
},
{
    "cell_type": "code",
    "execution_count": 25,
    "id": "metric-brighton",
    "metadata": {},
    "outputs": [],
    "source": [
        "path = Path('.')"
    ]
},
{
    "cell_type": "code",
    "execution_count": 26,
    "id": "accepting-equity",
    "metadata": {},
    "outputs": [
        {

```

```

    "data": {
      "text/plain": [
        "(#28)
[Path('~lock.OpenWeather_Antwerp.csv#'), Path('training_data_v3.csv'), Pat
h('training_data_v3_1.csv'), Path('training_data_v2.csv'), Path('Timeseries
_52.012_6.131_SA_0deg_-
179deg_2005_2016_Doesburg.csv'), Path('EnergyProduction_LinearRegression_v
2.2.ipynb'), Path('.git'), Path('SolarEnergyDataRefine_v3.ipynb'), Path('Ene
rgyProduction_LinearRegression_v2.1.ipynb'), Path('SolarEnergy.zip')...]"
      ]
    },
    "execution_count": 26,
    "metadata": {},
    "output_type": "execute_result"
  }
],
"source": [
  "path.ls()"
]
},
{
  "cell_type": "code",
  "execution_count": 27,
  "id": "modified-storm",
  "metadata": {},
  "outputs": [],
  "source": [
    "learn = load_learner(path/'export_v2.pkl')"
  ]
},
{
  "cell_type": "code",
  "execution_count": 28,
  "id": "stock-tiger",
  "metadata": {},
  "outputs": [],
  "source": [
    "day_list = []\n",
    "month_list = []\n",
    "for i in range(len(forecast_data)):\n",
    "    day_list.append(int(forecast_data.date[i][8:]))\n",
    "    month_list.append(int(forecast_data.date[i][5:7]))\n",
    "forecast_data['day'] = day_list\n",
    "forecast_data['month'] = month_list"
  ]
},
{
  "cell_type": "code",
  "execution_count": 29,
  "id": "informed-middle",
  "metadata": {},
  "outputs": [
    {
      "data": {

```

```

"text/html": [
  "<div>\n",
  "<style scoped>\n",
  "    .dataframe tbody tr th:only-of-type {\n",
  "        vertical-align: middle;\n",
  "    }\n",
  "\n",
  "    .dataframe tbody tr th {\n",
  "        vertical-align: top;\n",
  "    }\n",
  "\n",
  "    .dataframe thead th {\n",
  "        text-align: right;\n",
  "    }\n",
  "</style>\n",
  "<table border=\"1\" class=\"dataframe\">\n",
  "  <thead>\n",
  "    <tr style=\"text-align: right;\">\n",
  "      <th></th>\n",
  "      <th>mor_temp</th>\n",
  "      <th>mor_humid</th>\n",
  "      <th>mor_wspd</th>\n",
  "      <th>mor_wdeg</th>\n",
  "      <th>mor_cloud</th>\n",
  "      <th>mor_weather_id</th>\n",
  "      <th>noo_temp</th>\n",
  "      <th>noo_humid</th>\n",
  "      <th>noo_wspd</th>\n",
  "      <th>noo_wdeg</th>\n",
  "      <th>noo_cloud</th>\n",
  "      <th>noo_weather_id</th>\n",
  "      <th>aft_temp</th>\n",
  "      <th>aft_humid</th>\n",
  "      <th>aft_wspd</th>\n",
  "      <th>aft_wdeg</th>\n",
  "      <th>aft_cloud</th>\n",
  "      <th>aft_weather_id</th>\n",
  "      <th>day</th>\n",
  "      <th>month</th>\n",
  "    </tr>\n",
  "  </thead>\n",
  "  <tbody>\n",
  "    <tr>\n",
  "      <th>0</th>\n",
  "      <td>25.340000</td>\n",
  "      <td>42.000000</td>\n",
  "      <td>3.230000</td>\n",
  "      <td>257.000000</td>\n",
  "      <td>15.000000</td>\n",
  "      <td>801</td>\n",
  "      <td>24.94</td>\n",
  "      <td>46.0</td>\n",
  "      <td>2.67</td>\n",
  "      <td>252.0</td>

```

```

"      <td>15.0</td>\n",
"      <td>801</td>\n",
"      <td>19.980</td>\n",
"      <td>75.0</td>\n",
"      <td>2.045</td>\n",
"      <td>316.0</td>\n",
"      <td>64.0</td>\n",
"      <td>802</td>\n",
"      <td>10</td>\n",
"      <td>6</td>\n",
"    </tr>\n",
"  <tr>\n",
"    <th>1</th>\n",
"    <td>20.936667</td>\n",
"    <td>69.333333</td>\n",
"    <td>4.360000</td>\n",
"    <td>244.333333</td>\n",
"    <td>20.333333</td>\n",
"    <td>801</td>\n",
"    <td>23.76</td>\n",
"    <td>62.0</td>\n",
"    <td>4.45</td>\n",
"    <td>265.0</td>\n",
"    <td>87.0</td>\n",
"    <td>804</td>\n",
"    <td>19.315</td>\n",
"    <td>78.5</td>\n",
"    <td>3.785</td>\n",
"    <td>270.5</td>\n",
"    <td>86.5</td>\n",
"    <td>803</td>\n",
"    <td>11</td>\n",
"    <td>6</td>\n",
"  </tr>\n",
"  <tr>\n",
"    <th>2</th>\n",
"    <td>19.143333</td>\n",
"    <td>67.000000</td>\n",
"    <td>4.626667</td>\n",
"    <td>295.333333</td>\n",
"    <td>97.000000</td>\n",
"    <td>804</td>\n",
"    <td>19.54</td>\n",
"    <td>53.0</td>\n",
"    <td>6.11</td>\n",
"    <td>327.0</td>\n",
"    <td>100.0</td>\n",
"    <td>804</td>\n",
"    <td>13.670</td>\n",
"    <td>72.5</td>\n",
"    <td>3.760</td>\n",
"    <td>321.0</td>\n",
"    <td>37.5</td>\n",
"    <td>803</td>\n",

```



```

"      <td>12</td>\n",
"      <td>6</td>\n",
"    </tr>\n",
"    <tr>\n",
"      <th>3</th>\n",
"      <td>17.486667</td>\n",
"      <td>62.000000</td>\n",
"      <td>1.940000</td>\n",
"      <td>308.666667</td>\n",
"      <td>4.666667</td>\n",
"      <td>800</td>\n",
"      <td>23.01</td>\n",
"      <td>51.0</td>\n",
"      <td>2.34</td>\n",
"      <td>326.0</td>\n",
"      <td>15.0</td>\n",
"      <td>801</td>\n",
"      <td>17.600</td>\n",
"      <td>78.0</td>\n",
"      <td>2.560</td>\n",
"      <td>38.5</td>\n",
"      <td>4.5</td>\n",
"      <td>800</td>\n",
"      <td>13</td>\n",
"      <td>6</td>\n",
"    </tr>\n",
"  </tbody>\n",
"</table>\n",
"</div>"
],
"text/plain": [
"      mor_temp  mor_humid  mor_wspd    mor_wdeg  mor_cloud
mor_weather_id  \\\n",
"0  25.340000  42.000000  3.230000  257.000000  15.000000
801  \n",
"1  20.936667  69.333333  4.360000  244.333333  20.333333
801  \n",
"2  19.143333  67.000000  4.626667  295.333333  97.000000
804  \n",
"3  17.486667  62.000000  1.940000  308.666667  4.666667
800  \n",
"  \n",
"      noo_temp  noo_humid  noo_wspd    noo_wdeg  noo_cloud
noo_weather_id  \\\n",
"0      24.94      46.0      2.67      252.0      15.0
801  \n",
"1      23.76      62.0      4.45      265.0      87.0
804  \n",
"2      19.54      53.0      6.11      327.0     100.0
804  \n",
"3      23.01      51.0      2.34      326.0      15.0
801  \n",
"  \n",

```

```

      "    aft_temp    aft_humid    aft_wspd    aft_wdeg    aft_cloud
aft_weather_id day  \\\n",
      "0    19.980      75.0      2.045      316.0      64.0
802    10    \n",
      "1    19.315      78.5      3.785      270.5      86.5
803    11    \n",
      "2    13.670      72.5      3.760      321.0      37.5
803    12    \n",
      "3    17.600      78.0      2.560      38.5      4.5
800    13    \n",
      "\n",
      "    month    \n",
      "0         6    \n",
      "1         6    \n",
      "2         6    \n",
      "3         6    "
    ]
  },
  "execution_count": 29,
  "metadata": {},
  "output_type": "execute_result"
}
],
"source": [
  "forecast_input = forecast_data.drop(['date'], axis=1,
inplace=False)\n",
  "forecast_input.head()"
]
},
{
  "cell_type": "code",
  "execution_count": 30,
  "id": "directed-vegetation",
  "metadata": {},
  "outputs": [
    {
      "data": {
        "text/html": [],
        "text/plain": [
          "<IPython.core.display.HTML object>"
        ]
      },
      "metadata": {},
      "output_type": "display_data"
    },
    {
      "data": {
        "text/html": [],
        "text/plain": [
          "<IPython.core.display.HTML object>"
        ]
      },
      "metadata": {},
      "output_type": "display_data"
    }
  ]
}

```

```

},
{
  "data": {
    "text/html": [],
    "text/plain": [
      "<IPython.core.display.HTML object>"
    ]
  },
  "metadata": {},
  "output_type": "display_data"
},
{
  "data": {
    "text/html": [],
    "text/plain": [
      "<IPython.core.display.HTML object>"
    ]
  },
  "metadata": {},
  "output_type": "display_data"
}
],
"source": [
  "predicted_value = []\n",
  "\n",
  "for i in range(len(forecast_data)):\n",
  "    row, clas, probs = learn.predict(forecast_input.iloc[i])\n",
  "    predicted_value.append(clas)"
]
},
{
  "cell_type": "code",
  "execution_count": 31,
  "id": "latter-thinking",
  "metadata": {},
  "outputs": [
    {
      "data": {
        "text/plain": [
          "[tensor([0.4472]), tensor([0.4932]), tensor([15.3369]),\ntensor([21.1662])]"
        ]
      },
      "execution_count": 31,
      "metadata": {},
      "output_type": "execute_result"
    }
  ],
  "source": [
    "predicted_value"
  ]
},
{
  "cell_type": "code",

```

```
    "execution_count": null,
    "id": "white-antique",
    "metadata": {},
    "outputs": [],
    "source": []
  }
],
"metadata": {
  "kernelspec": {
    "display_name": "Python 3",
    "language": "python",
    "name": "python3"
  },
  "language_info": {
    "codemirror_mode": {
      "name": "ipython",
      "version": 3
    },
    "file_extension": ".py",
    "mimetype": "text/x-python",
    "name": "python",
    "nbconvert_exporter": "python",
    "pygments_lexer": "ipython3",
    "version": "3.8.5"
  }
},
"nbformat": 4,
"nbformat_minor": 5
}
```