**Module 6 Challenge - python-api-challenge**

**Part 1: WeatherPy**

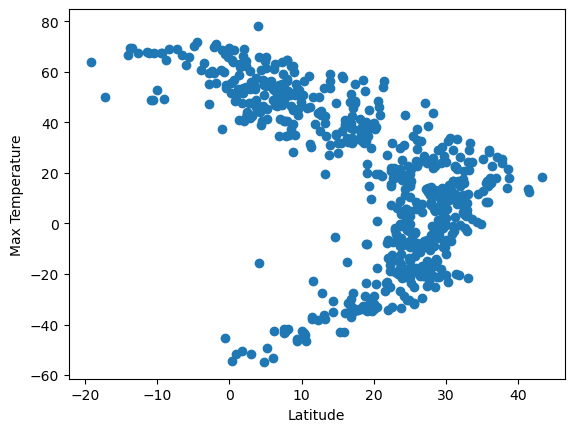
**Purpose:**

In this deliverable, you'll create a Python script to visualise the weather of over 500 cities of varying distances from the equator to create a representative model of weather across cities using the citipy Python library and the OpenWeatherMap API.

**Requirement 1: Create Plots to Showcase the Relationship Between Weather Variables and Latitude**

*WeatherPy\_Fig1.png*

Latitude vs Temperature



*WeatherPy\_Fig2.png*

Latitude vs Humidity

A diagram of blue dots

Description automatically generated

*WeatherPy\_Fig3.png*

Latitude vs Cloudiness

A diagram of blue dots

Description automatically generated

*WeatherPy\_Fig4.png*

Latitude vs Wind speed

A diagram of blue dots

Description automatically generated

**Requirement 2: Compute Linear Regression for Each Relationship**

Relationship between Temperature and Latitude

*Plot 1 - Linear regression in the Northern Hemisphere on Latitude vs Temperature*

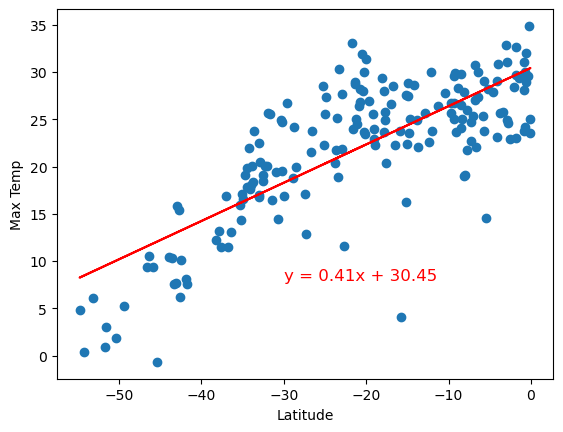
A red line and blue dots

Description automatically generated

**The r-value is**: -0.8419155328579567

An r value of -0.8419155328579567 indicates a strong negative correlation between temperature and latitude in the Northern Hemisphere.

*Plot 2 – Linear regression in the Southern Hemisphere on Latitude vs Temperature*



**The r-value is:** 0.7799615088336724

An r value of 0.7799615088336724 indicates a strong positive correlation between temperature and latitude in the Southern Hemisphere.

**Relationship between Humidity and Latitude:**

*Plot 3 – Linear regression in the Northern Hemisphere on Latitude vs Humidity*

A diagram of a graph

Description automatically generated

**The r-value is:** 0.1857695628962896

An r value of 0.1857695628962896 suggests a weak positive correlation between humidity and latitude in the Northern Hemisphere.

*Plot 4 – Linear regression in the Southern Hemisphere on Latitude vs Humidity*

A diagram of a line and a red line

Description automatically generated

**The r-value is:** 0.051201830053899215

An r value of 0.051201830053899215 indicates a very weak positive correlation between humidity and latitude in the Southern Hemisphere.

**Relationship between Cloudiness and Latitude:**

*Plot 5 – Linear regression in the Northern Hemisphere on Latitude vs Cloudiness*

A diagram of a graph

Description automatically generated

**The r-value is:** 0.037774314864813446

An r value of 0.037774314864813446 also suggests a very weak positive correlation between cloudiness and latitude in the Northern Hemisphere.

*Plot 6 – Linear regression in the Southern Hemisphere on Latitude vs Cloudiness*

A graph with blue dots and red line

Description automatically generated

**The r-value is:** 0.09002908351354302

An r value of 0.09002908351354302 indicates a weak positive correlation between cloudiness and latitude in the Southern Hemisphere.

**Relationship between Wind Speed and Latitude**

*Plot 7 – Linear regression in the Northern Hemisphere on Latitude vs Wind Speed*

A graph of blue dots

Description automatically generated

**The r-value is:** 0.03800283099030384

An r value of 0.03800283099030384 also suggests a very weak positive correlation between wind speed and latitude in the Northern Hemisphere.

*Plot 8 – Linear regression in the Southern Hemisphere on Latitude vs Wind Speed*

A graph of blue dots and red line

Description automatically generated

**The r-value is:** -0.31767288376934627

An r value of -0.31767288376934627 indicates a moderate negative correlation between wind speed and latitude in the Southern Hemisphere.

**Part 2: VacationPy**

**Purpose:**

In this deliverable, you'll use Jupyter notebook the geoViews Python library, and the Geoapify API to plan future vacations

1. Create a map that displays a point for every city in the city\_data\_df

A map of the world with many colored circles

Description automatically generated

1. Narrow down the city\_data\_df DataFrame to find your ideal weather condition

* A max temperature lower than 25 degrees
* A minimum temperature higher than 20
* Cloudiness less than 2
* Wind speed less than 3 m/s

A computer screen shot of a code

Description automatically generated

1. Create a new DataFrame called hotel\_df to store the city, country, coordinates, and humidity

A screenshot of a computer

Description automatically generated

1. For each city, use the Geoapify API to find the first hotel located within 10,000 metres of your coordinates.

A screenshot of a computer program

Description automatically generated

1. Add the hotel name and the country as additional information in the hover message for each city in the map as in the following image:

A map with different colored circles

Description automatically generated