ASSIGNMENT-2 REPORT

1.find_temperature_extremes(filename, city_name):

Purpose: Finds the hottest and coldest recorded months for a given city.
Inputs:

- filename: CSV file containing the dataset
- city_name: Name of the city

Outputs:
Dictionary containing the hottest and coldest months and their temperatures.

Logic:

1. Read rows for the given city
2. Extract month-year (YYYY-MM) and temperature values
3. Identify the maximum and minimum temperatures

2.get_seasonal_averages(filename, city_name, season)

Purpose: Computes the weighted average temperature of a city for one season (spring, summer, fall, winter).
Inputs:

- filename: Dataset CSV
- city_name: Target city
- season: Season to evaluate

Outputs: Dictionary with city, season, and average seasonal temperature.

Logic and Formula:

- Map seasons to months (e.g. summer = June to August)
- Select records matching the city and season
- Apply weighted average with uncertainty

Temp Average=($\sum(Ti/\sigma$**2)) / ($\sum(1/\sigma^{**}\ 2\ )$)

Ti = Temperature
$\sigma\ =\ Temperature\ uncertainity$

3.compare_decades(filename, city_name, decade1, decade2)

Purpose: Compares the long-term average temperature of a city across two decades and determines the trend (warming, cooling, stable)
Inputs:

- filename: Dataset CSV
- city_name: Target city

- decade1, decade2: Starting years of the decades (e.g. 1980, 2000)

Outputs: Dictionary containing each decade's average temperature, difference, and trend classification

Logic and Formula:

- Filter records for the city within each decade
- Compute weighted average using same formula as seasonal averages
- Compare averages:
  Positive difference → warming
  Negative difference → cooling
  Zero difference → stable

4.find_similar_cities(filename, target_city, tolerance=2.0)

Purpose: Finds other cities whose long-term average temperature is within a given tolerance compared to a target city

Inputs:

- filename: Dataset CSV
- target_city: City to compare
- tolerance: Max temperature difference (default 2°C)

Outputs: Dictionary with the target city's average and a list of similar cities.

Logic and Formula:

- Compute target city's weighted average temp
- For each city compute weighted average using the mentioned formula lately
- Compare with the target city. If difference ≤ tolerance, include in results.

5.get_temperature_trends(filename, city_name, window_size=5)

Purpose: Analyzes yearly temperature changes, computes moving averages, and detects warming/cooling streaks.

Inputs:

- filename: Dataset CSV
- city_name: Target city
- window_size: Number of years for moving average (default = 5)

Outputs:Dictionary containing annual averages, moving averages, and a trend analysis (overall slope, warming periods, cooling periods)

Formula:

- Moving averages of Ti = (Ti-windowsize+... + Ti-1 + Ti)/ window_size
- Overall slope= (T_end - T_start) / (Y_end - Y_start)
- Detect warming/cooling streaks by checking consecutive years of increasing or decreasing values, applying the slope formula to each streak.

Helper Funcitons :

- Country_name: returns country name for given city
- Avg_temp_with_uncertainity : calculates weighted average of temp using the formula mentioned in above text
- Get_available_years: returns the unique list of years in ascending order