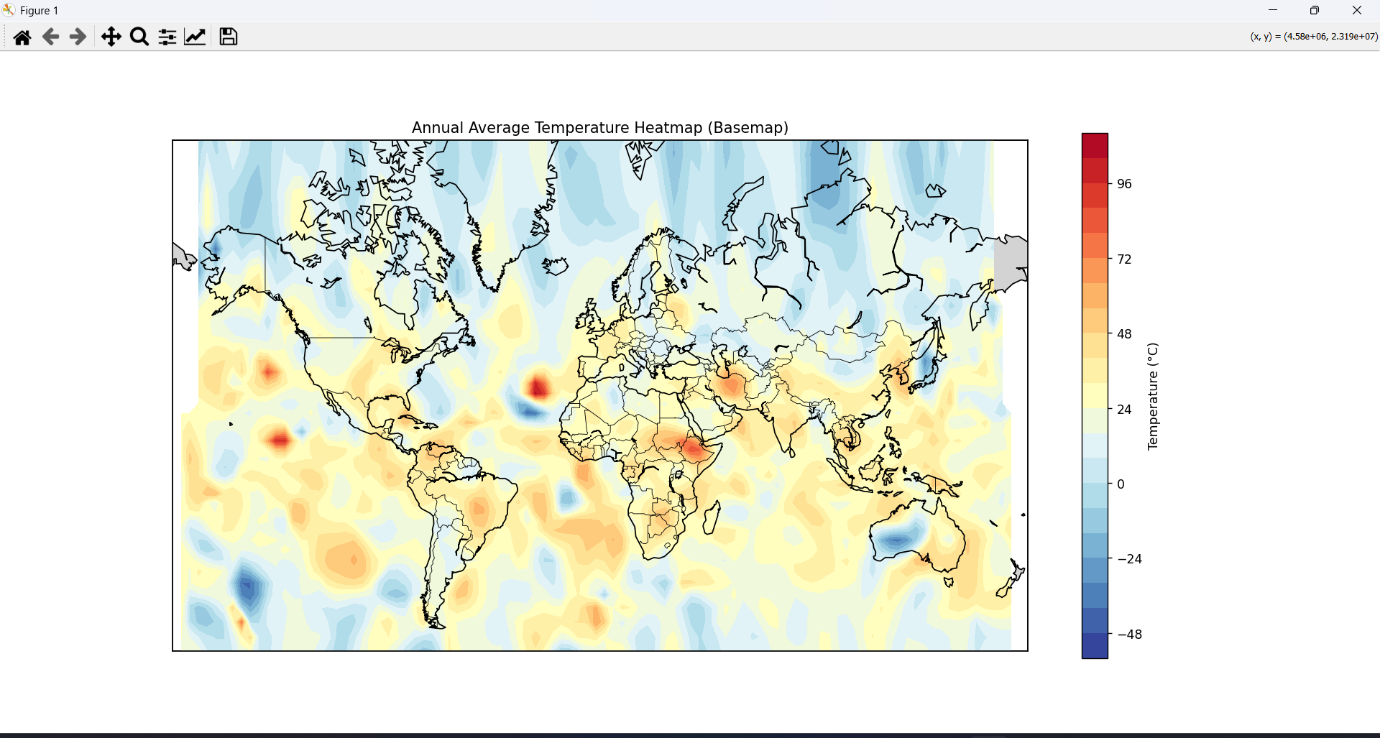
**Generating Geographical Heatmap**

**1. Task Description**

The task involves developing a Python-based application to generate a geographical heatmap that visualizes temperature variations across the globe. The project utilizes the `Basemap` library to create a Mercator projection map and employs data interpolation to produce a continuous heatmap. The application generates a sample dataset simulating temperature distribution based on latitude and adds random noise for realism. The output is saved as a PNG image file (`basemap\_heatmap.png`) for analysis. The goal is to demonstrate proficiency in geospatial data visualization and Python programming for an internship evaluation.

**2. Task Output Screenshot**



**3. Widget/Algorithm Used In Task**

* Basemap (from mpl\_toolkits.basemap): Used as the core mapping tool to create a Mercator projection of the world, draw coastlines, countries, and fill continents. It transforms geographical coordinates and masks water areas to ensure the heatmap is applied only to land.
* matplotlib.pyplot: Employed for plotting the heatmap, creating the figure, adding a colorbar, and saving the output image. The `contourf` function generates the continuous gradient effect.
* numpy: Utilized for numerical operations, including generating random sample data (`np.random.uniform`, `np.random.normal`), creating mesh grids (`np.meshgrid`), and handling array transformations.
* scipy.interpolate.griddata: Implements cubic interpolation to convert scattered temperature data points into a smooth grid for the heatmap.

**Algorithm Description:** The process involves generating random latitude and longitude points with temperature values derived from a cosine function of latitude (simulating warmer equatorial and colder polar regions) plus random noise. This data is interpolated onto a 100x50 grid using `griddata`. The `Basemap.transform\_scalar` method masks water areas, and `contourf` applies a color gradient (`RdYlBu\_r` colormap) to visualize temperature variations, with the result saved as an image.