UN FAO Food Security

Selected data obtained from the Food and Agriculture Organization of the United Nations, https://fao.org.

The data selected describe world prevalence of undernourishment (%), number of people undernourished (millions), prevalence of severe food insecurity in the total population (%) and the number of severely food insecure people (million).

The goal of this project is to create two interactive choropleth maps, one showing the global prevalence of undernourishment and the other the global number of undernourished people.

```
In [1]:  # Import initial necessary packages
         import pandas as pd
         import numpy as np
```

```
In [2]:  # Read the csv file into a dataframe
         data = pd.read_csv('UN_food_security.csv')

         # Display the first few rows of the dataset
         data.head()
```

Out[2]:

	Domain Code	Domain	Area Code (M49)	Area	Element Code	Element	Item Code	Item	Year Code	Year	Unit	Value	Flag	Flag Description	Note
0	FS	Suite of Food Security Indicators	4	Afghanistan	6121	Value	210041	Prevalence of undernourishment (percent) (3-ye...	20002002	2000-2002	%	46.4	E	Estimated value	NaN
1	FS	Suite of Food Security Indicators	4	Afghanistan	6121	Value	210041	Prevalence of undernourishment (percent) (3-ye...	20012003	2001-2003	%	44.1	E	Estimated value	NaN
2	FS	Suite of Food Security Indicators	4	Afghanistan	6121	Value	210041	Prevalence of undernourishment (percent) (3-ye...	20022004	2002-2004	%	39	E	Estimated value	NaN
3	FS	Suite of Food Security Indicators	4	Afghanistan	6121	Value	210041	Prevalence of undernourishment (percent) (3-ye...	20032005	2003-2005	%	36.3	E	Estimated value	NaN
4	FS	Suite of Food Security Indicators	4	Afghanistan	6121	Value	210041	Prevalence of undernourishment (percent) (3-ye...	20042006	2004-2006	%	34.5	E	Estimated value	NaN

```
In [3]:  # Confirm the dataframe data types
         data.dtypes
```

```
Out[3]:  Domain Code        object
         Domain             object
         Area Code (M49)     int64
         Area               object
         Element Code        int64
         Element            object
         Item Code          object
         Item               object
         Year Code           int64
         Year               object
         Unit               object
         Value              object
         Flag               object
         Flag Description   object
         Note               object
         dtype: object
```

The most important columns for this analysis will be **Area**, **Item**, **Year**, **Unit** and **Value**. There are two immediate problems that need addressed:

1. The values in **Year** are listed as ranges, and not a single year. The actual year for these ranges is in the middle, with the ranges representing a rolling average for each year (i.e. the range 2000 - 2002 represents the rolling average for the year 2001). This will have to be fixed in order to have a discrete year for creating the plots.

2. When initially viewing the data in Excel, it was noticed that some of the data in **Value** have the < comparison operator which will need to be removed.

In [4]:
```python
# Convert 'Year' from a range to a single year (middle year of each range)
data['Single Year'] = data['Year'].apply(lambda x: int(x.split('-')[0]) + 1)

# Confirm the change
data.head(3)
```

Out[4]:

	Domain Code	Domain	Area Code (M49)	Area	Element Code	Element	Item Code	Item	Year Code	Year	Unit	Value	Flag	Flag Description	Note	Single Year
0	FS	Suite of Food Security Indicators	4	Afghanistan	6121	Value	210041	Prevalence of undernourishment (percent) (3-ye...	20002002	2000-2002	%	46.4	E	Estimated value	NaN	2001
1	FS	Suite of Food Security Indicators	4	Afghanistan	6121	Value	210041	Prevalence of undernourishment (percent) (3-ye...	20012003	2001-2003	%	44.1	E	Estimated value	NaN	2002
2	FS	Suite of Food Security Indicators	4	Afghanistan	6121	Value	210041	Prevalence of undernourishment (percent) (3-ye...	20022004	2002-2004	%	39	E	Estimated value	NaN	2003

In [5]:
```python
# Replace values in 'Value' containing a comparison operator with a numeric equivalent and convert to float
data['Value'] = data['Value'].str.replace('<', '', regex = False).astype(float)

# Confirm the operator has been removed
cleaned = data[data['Value'].astype(str).str.contains('<')]
cleaned
```

Out[5]:

Domain Code	Domain	Area Code (M49)	Area	Element Code	Element	Item Code	Item	Year Code	Year	Unit	Value	Flag	Flag Description	Note	Single Year

No data appears in the newly created 'cleaned' dataframe, confirming the removal of the < operator in the **Value** column

In [6]:
```python
# Filter the dataset for 'Prevalence of undernourishment (percent)'

undernourishment_data = data[data['Item'] == 'Prevalence of undernourishment (percent) (3-year average)'].copy()

# Confirm the filter
undernourishment_data.head()
```

Out[6]:

| | Domain Code | Domain | Area Code (M49) | Area | Element Code | Element | Item Code | Item | Year Code | Year | Unit | Value | Flag | Flag Description | Note | Single Year |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | FS | Suite of Food Security Indicators | 4 | Afghanistan | 6121 | Value | 210041 | Prevalence of undernourishment (percent) (3-ye... | 20002002 | 2000-2002 | % | 46.4 | E | Estimated value | NaN | 2001 |
| 1 | FS | Suite of Food Security Indicators | 4 | Afghanistan | 6121 | Value | 210041 | Prevalence of undernourishment (percent) (3-ye... | 20012003 | 2001-2003 | % | 44.1 | E | Estimated value | NaN | 2002 |
| 2 | FS | Suite of Food Security Indicators | 4 | Afghanistan | 6121 | Value | 210041 | Prevalence of undernourishment (percent) (3-ye... | 20022004 | 2002-2004 | % | 39.0 | E | Estimated value | NaN | 2003 |
| 3 | FS | Suite of Food Security Indicators | 4 | Afghanistan | 6121 | Value | 210041 | Prevalence of undernourishment (percent) (3-ye... | 20032005 | 2003-2005 | % | 36.3 | E | Estimated value | NaN | 2004 |
| 4 | FS | Suite of Food Security Indicators | 4 | Afghanistan | 6121 | Value | 210041 | Prevalence of undernourishment (percent) (3-ye... | 20042006 | 2004-2006 | % | 34.5 | E | Estimated value | NaN | 2005 |

Now that the data have been cleaned, create a choropleth map for world prevalence of undernourishment. Using plotly express will allow for the creation of a slider to change the year for the data being shown on the map.

In [7]:
```python
# Load plotly express
import plotly.express as px

# Create a choropleth map with a slider for years
fig = px.choropleth(undernourishment_data,
                    locations = 'Area',
                    locationmode = 'country names',
                    color = 'Value',
                    hover_name = 'Area',
                    animation_frame = 'Single Year',
                    color_continuous_scale = px.colors.sequential.Plasma,
                    title = 'Global Prevalence of Undernourishment (%) Over Time')
fig.show()
```

Global Prevalence of Undernourishment (%) Over Time

Single Year=2001
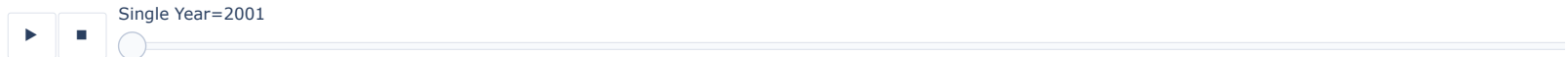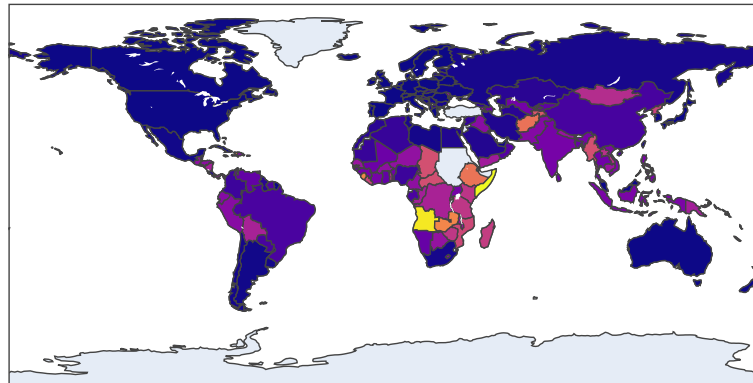
Next, a second choropleth map will be created to visualize the global number of undernourished people, using bubbles to show the magnitude of undernourishment for each country.
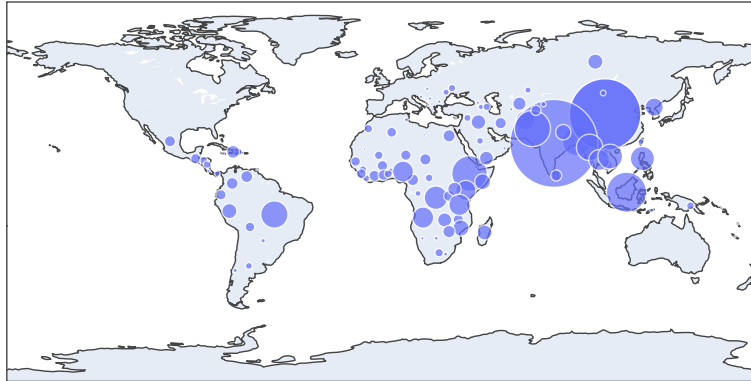
In [8]:
```python
# Filter the dataset for 'Number of people undernourished (million)'
undernourished_people = data[data['Item'] == 'Number of people undernourished (million) (3-year average)'].copy()

# Convert non-numeric values to numeric equivalents and convert 'Value' to float
undernourished_people['Value'] = pd.to_numeric(undernourished_people['Value'], errors = 'coerce').fillna(0)

# Create the bubble map
fig = px.scatter_geo(undernourished_people,
                     locations = 'Area',
                     locationmode = 'country names',
                     size = 'Value',
                     hover_name = 'Area',
                     animation_frame = 'Single Year',
```

```
                          title = 'Global Number of People Undernourished (Million) Over Time',
                          size_max = 50)
fig.show()
```

Global Number of People Undernourished (Million) Over Time

▶ ■   Single Year=2001

In [ ]: