# -\*- coding: utf-8 -\*-

"""

Created on Thu Nov 2 10:35:22 2023

@author: aishw

"""

import pandas as pd

import matplotlib.pyplot as plt

# Visualisation 1 : (Line Plots)

#Function1 : Reading file into DataFrame and customising the rows and columns

def process\_and\_print\_seasonal\_data(file\_path, start\_year=2000, end\_year=2022):

""" Defining the function1 to :

Read file into dataframe

filter data for the years 2000-2022

Calculate max and min rainfall during the seasons over the years (2000-2022)

Rename the columns"""

df\_UK = pd.read\_excel(file\_path)

df\_seasonal = pd.DataFrame(df\_UK[(df\_UK["year"] >= start\_year) & (df\_UK["year"] <= end\_year)],

columns=["year", "win", "spr", "sum", "aut"])

df\_seasonal["max"] = df\_seasonal[["win", "spr", "sum", "aut"]].max(axis=1)

df\_seasonal["min"] = df\_seasonal[["win", "spr", "sum", "aut"]].min(axis=1)

df\_seasonal.rename(columns={"year": "Year",

"win": "Winter",

"spr": "Spring",

"sum": "Summer",

"aut": "Autumn",

"max": "Maximum\_rainfall",

"min": "Minimum\_rainfall"}, inplace=True)

return df\_seasonal

#Function2: To Line plot1-Max and Min seasonal rainfall in UK from 2000 to 2022

def plot\_max\_min\_rainfall(df\_seasonal):

""" Defining the function2 to create a line plot1 to represent

max and min rainfall during seasons over the years with labels,

customizing visualization and to set title, X and Y axis labels

and show the legend for lineplot1"""

plt.subplot(1, 2, 1)

plt.plot(df\_seasonal["Year"], df\_seasonal["Maximum\_rainfall"], marker="o", label="Maximum rainfall", color="green")

plt.plot(df\_seasonal["Year"], df\_seasonal["Minimum\_rainfall"], marker="o", label="Minimum rainfall", color="red")

plt.legend(title="Max and Min rainfall", borderpad=0.5, fontsize=9)

plt.title("Max and Min seasonal rainfall in UK")

plt.xlabel("Years")

plt.ylabel("Rainfall precipitation in millimetres (mm)")

plt.xticks(df\_seasonal["Year"])

#Function3: To Line plot2- Seasonal rainfall in UK from 2000 to 2022

def plot\_seasonal\_rainfall(df\_seasonal):

""" Defining the function3 to create a line plot2 to represent

rainfall for four seasons in UK from 2000 to 2022 and customizing

visualization with labels,color. Also to Set title & X and Y axis labels

and show the legend for lineplot2"""

plt.subplot(1, 2, 2)

plt.plot(df\_seasonal["Year"], df\_seasonal["Winter"], label="winter", color="green")

plt.plot(df\_seasonal["Year"], df\_seasonal["Spring"], label="spring", color="red")

plt.plot(df\_seasonal["Year"], df\_seasonal["Summer"], label="summer", color="orange")

plt.plot(df\_seasonal["Year"], df\_seasonal["Autumn"], label="autumn", color="brown")

plt.legend(title="Seasonal Rainfall", borderpad=0.5, fontsize=9)

plt.title("Seasonal rainfall in UK")

plt.xlabel("Years")

plt.ylabel("Rainfall precipitation in millimetres (mm)")

plt.xticks(df\_seasonal["Year"])

# Defining filename

file\_path = "UK\_rainfall.xlsx"

# Assigning the functions to a variable

df\_seasonal\_data = process\_and\_print\_seasonal\_data(file\_path)

# print the data

print(df\_seasonal\_data)

# plot the line plots1 & 2

plt.figure(figsize=(30, 8))

plot\_max\_min\_rainfall(df\_seasonal\_data)

plot\_seasonal\_rainfall(df\_seasonal\_data)

# save and show the plot

plt.savefig("UK\_seasonal\_rainfall.png")

plt.show()

# Visualisation 2 : (Pie chart)

#Function4:To read the file into DataFrame

def read\_and\_print\_data(file\_path):

""" Defining the function4 to read the data from CSV file"""

df\_diversity = pd.read\_csv(file\_path)

print(df\_diversity)

#Function5:To plot pie of Ethnicity Percentage of Asians in England and Wales Regions

def plot\_ethnicity\_percentage(df\_diversity, ethnicity="Asian"):

""" Defining the function5 to filter data for the specified ethnicity and

to extract the necessary columns and region and percentage of ethnic

group. Setting explode values, title and show the legend to

plot the pie and save and show the pie"""

df\_ethnicity = pd.DataFrame(df\_diversity[df\_diversity["Ethnicity"] == ethnicity],

columns=["Ethnicity",

"Region",

"percentage of ethnic group"])

print(df\_ethnicity)

region = df\_ethnicity["Region"]

per\_ethnic = df\_ethnicity["percentage of ethnic group"]

explode = (0, 0, 0.1, 0, 0, 0, 0, 0, 0, 0, 0)

plt.figure(figsize=(40, 20))

plt.pie(per\_ethnic, explode=explode, labels=region, startangle=45, autopct='%1.1f%%')

plt.legend(region, title="Region", borderpad=1, fontsize=11)

plt.title(f"Ethnicity Percentage of {ethnicity}s in England and Wales Regions",

fontsize=25)

plt.savefig(f"Ethnicity\_Percentage\_{ethnicity}.png")

plt.show()

# Defining the filename and calling the functions

file\_path = "Diversity.csv"

read\_and\_print\_data(file\_path)

plot\_ethnicity\_percentage(pd.read\_csv(file\_path), ethnicity="Asian")

#Visualisation 3: (Bar chart)

#Function6:To read the file into DataFrame

def read\_and\_process\_data(file\_path, start\_year, end\_year):

""" Defining function6 to Read file into dataframe,

filtering data for the specified years, to

Calculate average precipitation per year and to rename the columns """

df\_UK = pd.read\_excel(file\_path)

df\_yearly = pd.DataFrame(df\_UK[(df\_UK["year"] >= start\_year) & (df\_UK["year"] <= end\_year)],

columns=["year", "jan", "feb",

"mar", "apr", "may",

"jun", "jul", "aug",

"sep", "oct", "nov", "dec"])

df\_yearly["Average"] = df\_yearly[["jan", "feb", "mar",

"apr", "may", "jun",

"jul", "aug", "sep",

"oct", "nov", "dec"]].mean(axis=1)

df\_yearly.rename(columns={"year": "Year",

"jan": "January",

"feb": "February",

"mar": "March",

"apr": "April",

"may": "May",

"jun": "June",

"jul": "July",

"aug": "August",

"sep": "September",

"oct": "October",

"nov": "November",

"dec": "December",

"Average": "Average"}, inplace=True)

return df\_yearly

#Function7 : To plot Bar Chart of Average rainfall per year (2000 – 2022)

def plot\_average\_rainfall(df\_yearly):

""" Defining function7 to plot the bar chart and custamizing it to set

a title,labelling X and Y axis and legend. Saving the output as png

and to Show the plot"""

plt.figure(figsize=(20, 8))

plt.bar(df\_yearly["Year"], df\_yearly["Average"])

plt.title("Average precipitation of rainfall per years (2000 - 2022)",

fontweight='bold', fontsize=10)

plt.xlabel("Years (2000-2022)", fontweight='bold', fontsize=13)

plt.ylabel("Rainfall precipitation (mm)", fontweight='bold', fontsize=13)

plt.xticks(df\_yearly["Year"])

plt.legend(df\_yearly["Year"], title="Max Average rainfall", borderpad=1, fontsize=11)

plt.savefig("Average\_rainfall\_per\_year.png")

plt.show()

# Defining the filename, start and end year

file\_path = "UK\_rainfall.xlsx"

start\_year = 2000

end\_year = 2022

# Assigning function to a variable and to print it

df\_yearly\_data = read\_and\_process\_data(file\_path, start\_year, end\_year)

print(df\_yearly\_data)

plot\_average\_rainfall(df\_yearly\_data)