pyth 

МІНІСТЕРСТВО ОСВІТИ І НАУКИ, МОЛОДІ ТА СПОРТУ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра Інформаційної Безпеки

Засоби підготовки та аналізу даних

Лабораторна робота №1

Наука про дані: підготовчий етап

Мета роботи: ознайомитися з основними кроками по роботі з даними – workflow від постановки задачі до написання пояснювальної записки, зрозуміти постановку задачі та природу даних, над якими виконується аналітичні операції

Основні поняття: сирі дані (raw data), підготовка даних (data preparation)

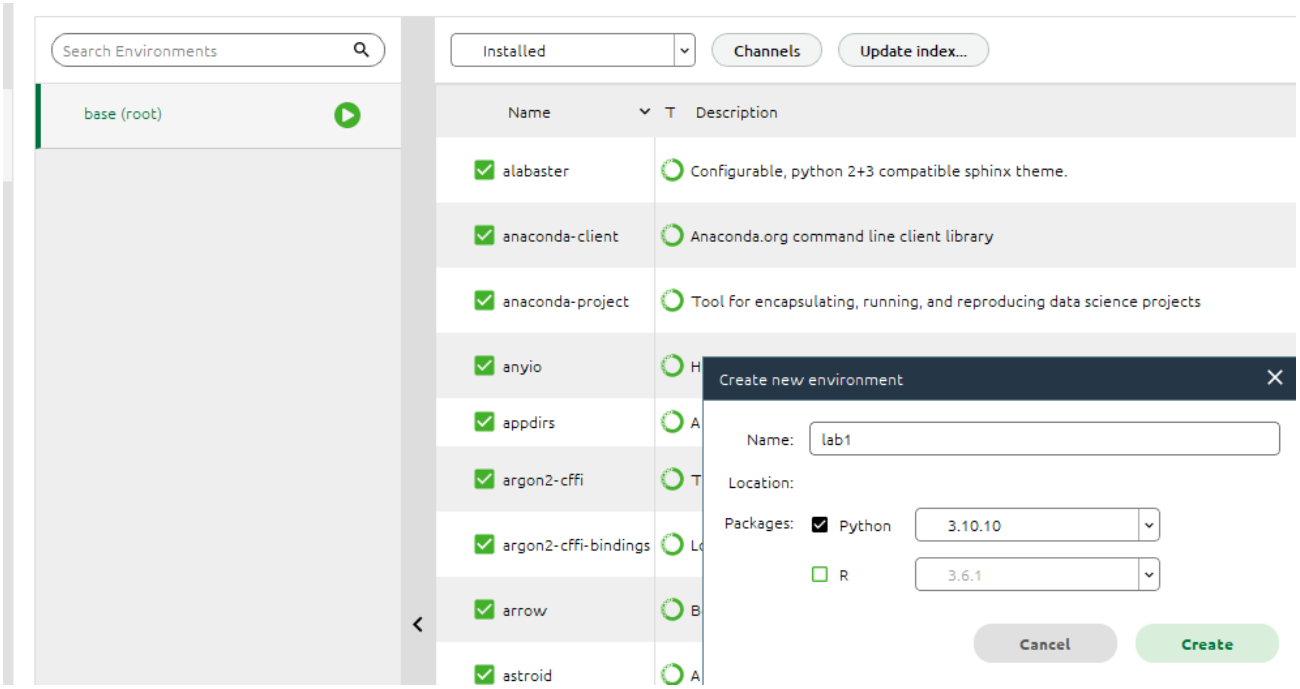
Перевірив:
_____

Виконав:
студент II курсу
групи ФБ-14
Хаща Іван

Київ 2023

Хід виконання роботи

- Створити env в якому будуть встановлені всі необхідні бібліотеки та налаштування для даної лабораторної роботи;




```
(base) PS C:\Users\os2fall> conda info --envs
# conda environments:
#
base                  *  C:\Users\os2fall\anaconda3
lab1                     C:\Users\os2fall\anaconda3\envs\lab1

(base) PS C:\Users\os2fall> conda activate lab1
(lab1) PS C:\Users\os2fall>
```

Для кожної із адміністративних одиниць України завантажити тестові структуровані файли, що містять значення VHI-індексу. Ця процедура має бути автоматизована, параметром процедури має бути індекс (номер) області. При зберіганні файлу до його імені потрібно додати дату та час завантаження;

```
lab1) PS C:\Users\os2fall> New-Item -Path "C:\Study\AD\lab1\lab1.py" -ItemType File


    Directory: C:\Study\AD\lab1


ode                 LastWriteTime         Length Name
---                 -------------         ------ ----
a----        04.04.2023     22:04              0 lab1.py
```

Програмний код для збереження тестових файлів з даними:

```
import urllib, urllib.request
from datetime import datetime

def get_data(province_id):
    url="https://www.star.nesdis.noaa.gov/smcd/emb/vci/VH/get_TS_admin.php?country=UKR&provinceID=[]&year1=1981&year2=2023&type=Mean".format(province_id)
    webpage = urllib.request.urlopen(url)
    text = webpage.read()
    now = datetime.now()
    date_and_time_time = now.strftime("%d.%m.%Y_%H^%M^%S")
    out = open('C:\\Study\\AD\\lab1\\' + 'NOAA_ID' + str(province_id) + '-' + date_and_time_time + '.csv', 'wb')
    out.write(text)
    out.close
```

```
>>> import sys
>>> sys.path.append("C:\Study\AD\lab1")
>>>
>>> from lab1 import get_data
>>> for id in range(1,28):
...     get_data(id)
```

Отримали:

__pycache__	04.04.2023 22:40	File folder
lab1	04.04.2023 22:36	Исходный файл P...
NOAA_ID1-04.04.2023_22^41^07	04.04.2023 22:41	Исходный файл C...
NOAA_ID2-04.04.2023_22^41^08	04.04.2023 22:41	Исходный файл C...
NOAA_ID3-04.04.2023_22^41^09	04.04.2023 22:41	Исходный файл C...
NOAA_ID4-04.04.2023_22^41^10	04.04.2023 22:41	Исходный файл C...
NOAA_ID5-04.04.2023_22^41^11	04.04.2023 22:41	Исходный файл C...
NOAA_ID6-04.04.2023_22^41^12	04.04.2023 22:41	Исходный файл C...
NOAA_ID7-04.04.2023_22^41^14	04.04.2023 22:41	Исходный файл C...
NOAA_ID8-04.04.2023_22^41^15	04.04.2023 22:41	Исходный файл C...
NOAA_ID9-04.04.2023_22^41^16	04.04.2023 22:41	Исходный файл C...
NOAA_ID10-04.04.2023_22^41^17	04.04.2023 22:41	Исходный файл C...
NOAA_ID11-04.04.2023_22^41^17	04.04.2023 22:41	Исходный файл C...
NOAA_ID12-04.04.2023_22^41^18	04.04.2023 22:41	Исходный файл C...
NOAA_ID13-04.04.2023_22^41^19	04.04.2023 22:41	Исходный файл C...
NOAA_ID14-04.04.2023_22^41^20	04.04.2023 22:41	Исходный файл C...
NOAA_ID15-04.04.2023_22^41^21	04.04.2023 22:41	Исходный файл C...

Зчитати завантажені текстові файли у фрейм (детальніше про роботу із фреймами буде розказано у подальших лабораторних роботах). Імена стовпців фрейму мають бути змістовними та легкими для сприйняття (не повинно бути спеціалізованих символів, пробілів тощо). Ця задача має бути реалізована у вигляді окремої процедури, яка на вхід приймає шлях до директорії, в якій зберігаються файли;

Фрагмент коду для підготовки відповідних дата фреймів:

```
def make_header(filepath):
    headers = ['Year', 'Week', 'SMN', 'SMT', 'VCI', 'TCI', 'VHI', 'empty']
    dataframe = pd.read_csv(filepath, header=1, names=headers)
    dataframe.drop(dataframe.loc[dataframe['VHI'] == -1].index)
    return dataframe
```

```
>>> import os.path
>>> from lab1 import make_header
>>> for id in range(1,28):
...     for second in range(21, 43):
...             if os.path.isfile(f"C:\\Study\\AD\\lab1\\NOAA_ID{id}-04.04.2023_22^41^{second}"):
...                     make_header(f"C:\\Study\\AD\\lab1\\NOAA_ID{id}-04.04.2023_22^41^{second}")
```

- Реалізувати процедуру, яка змінить індекси областей, які використані на порталі NOAA на наступні:

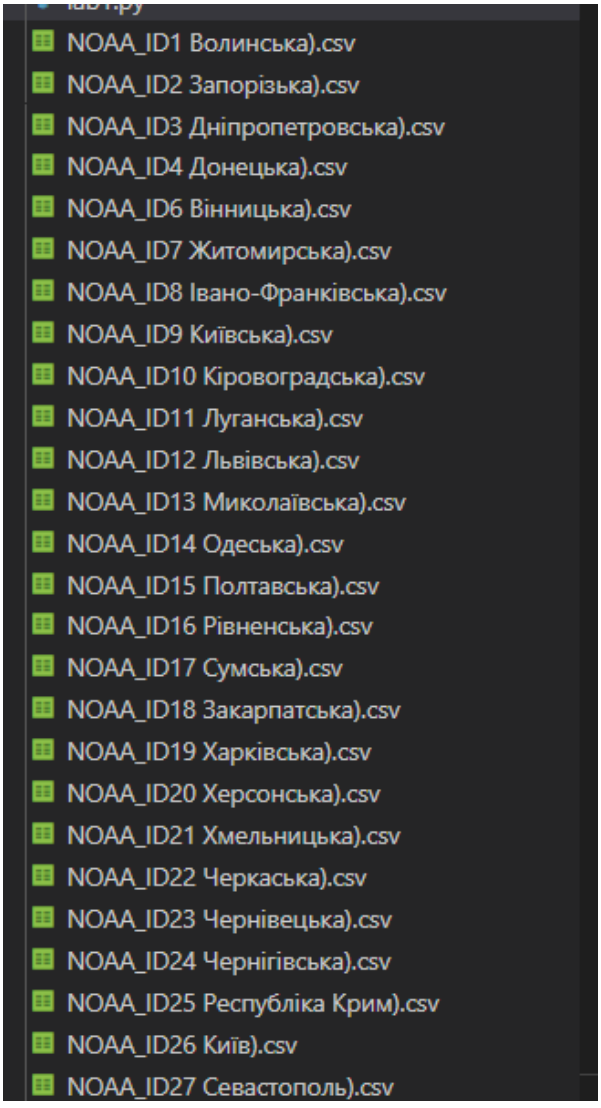
№ області	Назва	№ області	Назва
1	Вінницька	13	Миколаївська
2	Волинська	14	Одеська
3	Дніпропетровська	15	Полтавська
4	Донецька	16	Рівненська
5	Житомирська	17	Сумська
6	Закарпатська	18	Тернопільська
7	Запорізька	19	Харківська
8	Івано-Франківська	20	Херсонська
9	Київська	21	Хмельницька
10	Кіровоградська	22	Черкаська
11	Луганська	23	Чернівецька
12	Львівська	24	Чернігівська
		25	Республіка Крим

Програмний код процедури, що може змінювати індекси областей:

```python
def index_change(filepath, old, new, oblast):
    dataframe = make_header(filepath)

    dataframe['area'] = old
    dataframe['area'].replace({old: new}, inplace=True)

    dataframe.to_csv(f'C:\\Study\\AD\\lab1\\NOAA_ID{new} ({oblast}).csv', index=False)
    return dataframe
```

```python
area_list = ['1', '2', '3', '4', '5', '6', '7', '8', '9', '10', '11', '12', '13', '14', '15', '16', '17', '18', '19', '20', '21', '22', '23', '24', '25', '26', '27']
areas_new_list = ["22", "24", "23", "25", "3", "4", "8", "19", "20", "21", "9", "26", "10", "11", "12", "13", "14", "15", "16", "27", "17", "18", "6", "1", "2", "7", "
name_list = ["Черкаська", "Чернігівська", "Чернівецька", "Республіка Крим", "Дніпропетровська", "Донецька", "Івано-Франківська", "Харківська", "Херсонська","Хмельницьк

files = []
for i in range(1, 28):
    file_pattern = f'C:\\Study\\AD\\lab1\\NOAA_ID{i}-01.06.2023_*.csv'
    files.extend(glob.glob(file_pattern))
    for i in range(len(files)):
        index_change(files[i], area_list[i], areas_new_list[i], name_list[i])
```

Отримали:



```
Year,Week,SMN,SMT,VCI,TCI,VHI,empty,area
<tt><pre>1982,1.0,0.053,260.31,45.01,39.46,42.23,,22
1982,2.0,0.054,262.29,46.83,31.75,39.29,,22
1982,3.0,0.055,263.82,48.13,27.24,37.68,,22
```

- Реалізувати процедури для формування вибірок наступного виду (включаючи елементи аналізу):
  - Ряд VHI для області за рік, пошук екстремумів (min та max);
  - Ряд VHI за всі роки для області, виявити роки з екстремальними посухами, які торкнулися більше вказаного відсотка області;
  - Аналогічно для помірних посух

Програмний код процедури:

```python
def data_analysis(filepath, year):
    data = pd.read_csv(filepath)
    df = data[data['VHI'] != -1]

    ext_drought = df[df['VHI'] <= 15]  # Data for extreme drought periods
    max_val = ext_drought.loc[ext_drought['Year'].astype(str) == str(year), 'VHI'].max()
    print(f"{max_val} - maximum VHI for extreme drought in {year}")
    min_val = ext_drought.loc[ext_drought['Year'].astype(str) == str(year), 'VHI'].min()
    print(f"\t{min_val} - minimum VHI for extreme drought in {year}")

    this_year = int(ext_drought.loc[ext_drought['VHI'].idxmin(), 'Year'])
    print(f"\t\t{this_year} - the year with the most extreme drought period")

    drought = df[(15 < df['VHI']) & (df['VHI'] <= 35)]  # Data for moderate drought periods
    min_val = drought.loc[drought['Year'].astype(str) == str(year), 'VHI'].min()
    print(f"\t{min_val} - minimum VHI for moderate drought in {year}")
    max_val = drought.loc[drought['Year'].astype(str) == str(year), 'VHI'].max()
    print(f"{max_val} - maximum VHI for moderate drought in {year}")


data_analysis("C:\\Study\\AD\\lab1\\NOAA_ID1 Волинська).csv", 2000)
```

Отримали:

```
PS C:\Study\AD> python -u "c:\Study\AD\lab1\lab1.py"
14.2 - maximum VHI for extreme drought in 2000
        11.25 - minimum VHI for extreme drought in 2000
                2000 - the year with the most extreme drought perio
        15.07 - minimum VHI for moderate drought in 2000
34.73 - maximum VHI for moderate drought in 2000
PS C:\Study\AD>
```