



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

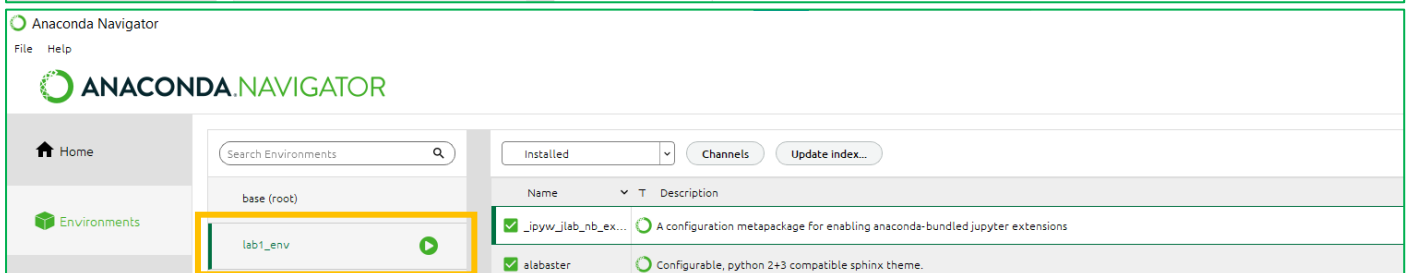
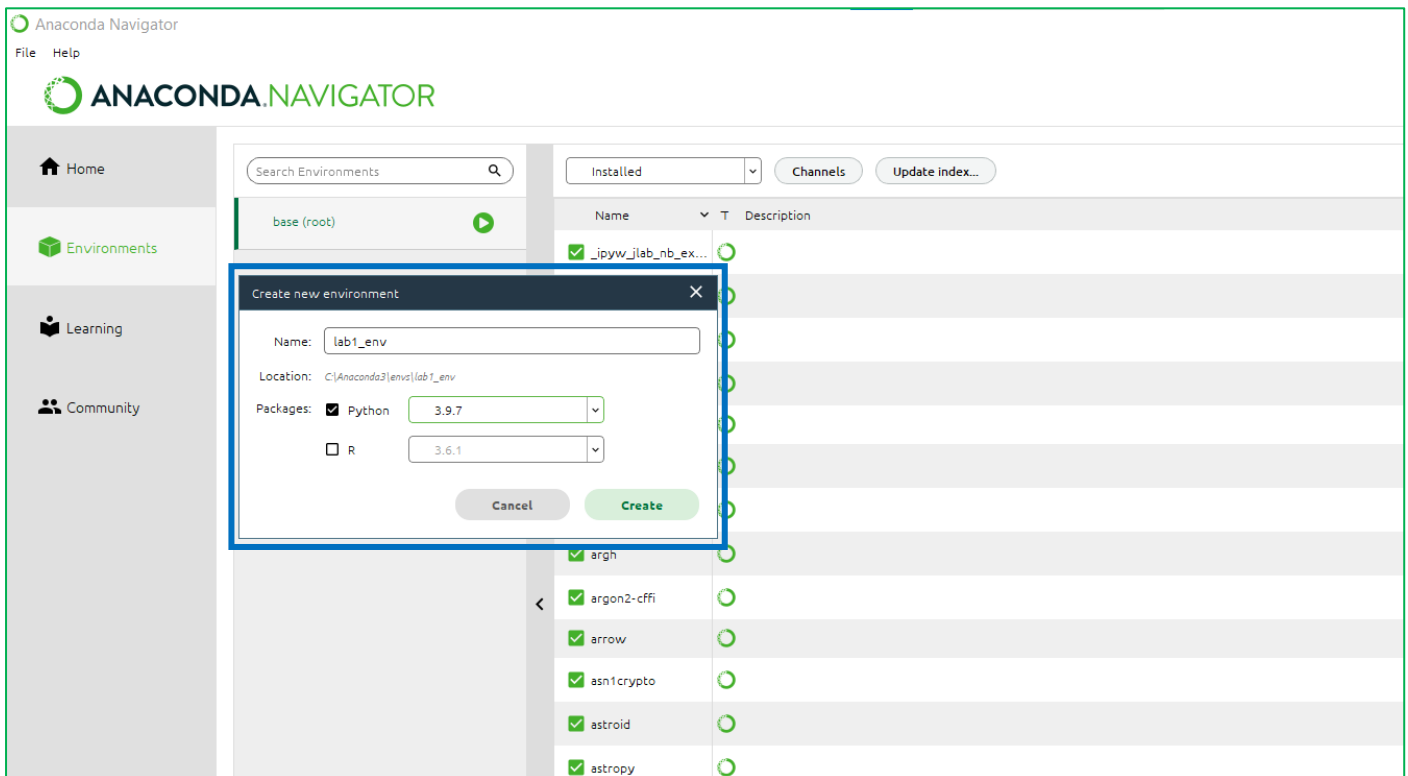
групи ФБ-01

Сахній Н.Р.

Київ 2022

Хід виконання роботи

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



Або...

```
(base) PS C:\Users\t-1000> conda info --envs
# conda environments:
#
base                  * C:\Anaconda3

(base) PS C:\Users\t-1000> conda create --name lab1_env python
WARNING: A directory already exists at the target location 'C:\Anaconda3\envs\lab1_env'
but it is not a conda environment.
Continue creating environment (y/[n])? y

Collecting package metadata (current_repodata.json): done
Solving environment: done


==> WARNING: A newer version of conda exists. <==
  current version: 4.10.3
  latest version: 4.12.0

Please update conda by running

  $ conda update -n base -c defaults conda

## Package Plan ##

  environment location: C:\Anaconda3\envs\lab1_env
  added / updated specs:
    - python
```



```
The following NEW packages will be INSTALLED:

bzip2                pkgs/main/win-64::bzip2-1.0.8-he774522_0
ca-certificates      pkgs/main/win-64::ca-certificates-2022.2.1-haa95532_0
certifi              pkgs/main/noarch::certifi-2020.6.20-pyhd3eb1b0_3
libffi               pkgs/main/win-64::libffi-3.4.2-h604cdeb4_1
openssl              pkgs/main/win-64::openssl-1.1.1n-h2bbff1b_0
pip                  pkgs/main/win-64::pip-21.2.4-py310haa95532_0
python               pkgs/main/win-64::python-3.10.0-h96c0403_3
setuptools           pkgs/main/win-64::setuptools-58.0.4-py310haa95532_0
sqlite               pkgs/main/win-64::sqlite-3.38.0-h2bbff1b_0
tk                   pkgs/main/win-64::tk-8.6.11-h2bbff1b_0
tzdata               pkgs/main/noarch::tzdata-2021e-hda174b7_0
vc                   pkgs/main/win-64::vc-14.2-h21ff451_1
vs2015_runtime       pkgs/main/win-64::vs2015_runtime-14.27.29016-h5e58377_2
wheel                pkgs/main/noarch::wheel-0.37.1-pyhd3eb1b0_0
wincertstore         pkgs/main/win-64::wincertstore-0.2-py310haa95532_2
xz                   pkgs/main/win-64::xz-5.2.5-h62dc97_0
zlib                 pkgs/main/win-64::zlib-1.2.11-hbd8134f_5
```

```
Proceed ([y]/n)? y
```

```
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
```

```
#
# To activate this environment, use
#
#     $ conda activate lab1_env
#
# To deactivate an active environment, use
#
#     $ conda deactivate
```

```
(base) PS C:\Users\t-1000> conda info --envs
# conda environments:
#
base                  * C:\Anaconda3
lab1_env              C:\Anaconda3\envs\lab1_env
```

```
(base) PS C:\Users\t-1000>
```

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;

```
(base) PS C:\Users\t-1000> conda activate lab1_env
(lab1_env) PS C:\Users\t-1000> New-Item -Path 'D:\KPI\Data Analysis\Lab 1\AD_lab1.py' -ItemType File
```

```
Directory: D:\KPI\Data Analysis\Lab 1
```

Mode	LastWriteTime	Length	Name
-a----	22.03.2022 21:35	0	AD_lab1.py

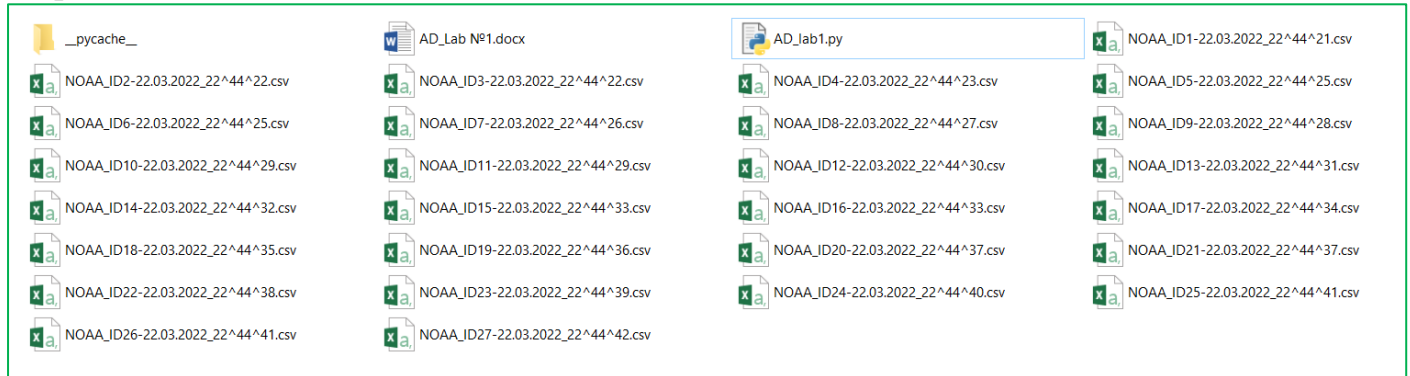
```
(lab1_env) PS C:\Users\t-1000> conda install urllib, urllib.request
```

Програмний код для збереження тестових файлів з даними:

```
1 import urllib, urllib.request
2 from datetime import datetime
3
4
5 def get_data(province_id):
6     url = 'https://www.star.nesdis.noaa.gov/smcd/emb/vci/VH/get_TS_admin.php?country=UKR&provinceID={}&year1=1981&year2=2020&type=Mean'.format(province_id)
7
8     # Відкриття WEB-сторінки можна зробити наступним чином:
9     webpage = urllib.request.urlopen(url)
10    text = webpage.read()
11
12    # Отримати поточну дату і час
13    now = datetime.now()
14    # Згенерувати строку з поточною датою і часом та необхідним форматуванням можна за допомогою методу strftime
15    date_and_time_time = now.strftime("%d.%m.%Y_%H%M%S")
16
17    # Створити новий файл за допомоги функції open
18    out = open('D:\KPI\Data Analysis\Lab 1\' + 'NOAA_ID' + str(province_id) + '-' + date_and_time_time + '.csv', 'wb')
19    # Після відкриття у змінній text міститься текст із WEB-сторінки, який тепер можна записати у файл
20    out.write(text)
21    out.close()
22
```

```
(lab1_env) PS C:\Users\t-1000> python
Python 3.10.0 | packaged by conda-forge | (default, Nov 10 2021, 13:20:59) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>
>>> import sys
>>> sys.path.append("D:\KPI\Data Analysis\Lab 1")
>>>
>>> from AD_lab1 import get_data
>>> for id in range(1, 28): # Завантажимо дані із 27 регіонів України, які є на сайті
...     get_data(id)
...
>>>
```

Отримали:



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:

```
23
24     import pandas as pd
25
26
27     def make_header(filepath):
28         headers = ['Year', 'Week', 'SMN', 'SMT', 'VCI', 'TCI', 'VHI', 'empty']
29         dataframe = pd.read_csv(filepath, header=1, names=headers)
30         dataframe.drop(dataframe.loc[dataframe['VHI'] == -1].index)
31         return dataframe
32
```

```
>>>
>>> import os.path
>>> from AD_lab1 import make_header
>>> for id in range(1, 28): # Для кожного файлу з даними зробити відповідні заголовки стовпців
...     for second in range(21, 43): # Усі файли, у яких в назві є значення від 21 до 42, що позначають секунди
...         if os.path.isfile(f"D:\\KPI\\Data Analysis\\Lab 1\\NOAA_ID{id}-22.03.2022_22^44^{second}"):
...             make_header(f"D:\\KPI\\Data Analysis\\Lab 1\\NOAA_ID{id}-22.03.2022_22^44^{second}")
...
>>>
```

Отримали (останній запис доданий у змінну dataframe):

```
>>> from AD_lab1 import dataframe
>>> dataframe
   Year  Week  SMN  SMT  VCI  TCI  VHI  empty
0  1982    1.0  0.053  260.31  45.01  39.46  42.23  NaN
1  1982    2.0  0.054  262.29  46.83  31.75  39.29  NaN
2  1982    3.0  0.055  263.82  48.13  27.24  37.68  NaN
3  1982    4.0  0.053  265.33  46.09  23.91  35.00  NaN
4  1982    5.0  0.050  265.66  41.46  26.65  34.06  NaN
...
```

```

2024      2020  49.0  0.078  266.01  48.41  38.06  43.23  NaN
2025      2020  50.0  0.073  264.76  49.34  37.58  43.46  NaN
2026      2020  51.0  0.067  263.19  48.87  37.09  42.98  NaN
2027      2020  52.0  0.063  261.35  48.73  39.69  44.21  NaN
2028      NaN    NaN    NaN    NaN    NaN    NaN    NaN
[2029 rows x 8 columns]
>>>

```

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:

```

33
34 def index_change(filepath, old, new, oblast):
35     dataframe = make_header(filepath)
36
37     dataframe['area'] = old
38     dataframe['area'].replace({old: new}, inplace=True)
39
40     dataframe.to_csv(f'D:\\KPI\\Data Analysis\\Lab 1\\NOAA_ID{new} ({oblast}).csv', index=False)
41     return dataframe
42

```

Отримали:

```

>>> from AD_lab1 import index_change
>>> index_change("D:\\KPI\\Data Analysis\\Lab 1\\NOAA_ID1-22.03.2022_22^44^21.csv", 1, 22, "Черкаська")
Year Week SMN SMT VCI TCI VHI empty area
0      1982  1.0  0.053  260.31  45.01  39.46  42.23  NaN  22
1      1982  2.0  0.054  262.29  46.83  31.75  39.29  NaN  22
2      1982  3.0  0.055  263.82  48.13  27.24  37.68  NaN  22
3      1982  4.0  0.053  265.33  46.09  23.91  35.00  NaN  22
4      1982  5.0  0.050  265.66  41.46  26.65  34.06  NaN  22
...      ...      ...      ...      ...      ...      ...      ...
2024      2020  49.0  0.078  266.01  48.41  38.06  43.23  NaN  22
2025      2020  50.0  0.073  264.76  49.34  37.58  43.46  NaN  22
2026      2020  51.0  0.067  263.19  48.87  37.09  42.98  NaN  22
2027      2020  52.0  0.063  261.35  48.73  39.69  44.21  NaN  22
2028      NaN    NaN    NaN    NaN    NaN    NaN    NaN  22
[2029 rows x 9 columns]
>>>

```

NOAA_ID22 (Черкаська).csv: Блокнот

NOAA_ID22 (Черкаська).csv

Файл Редагування Формат Вигляд Довідка

Year, Week, SMN, SMT, VCI, TCI, VHI, empty, area
<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:

```
43
44 def data_analysis(filepath, year):
45     data = pd.read_csv(filepath)
46     df = data[(data['VHI'] != -1)]
47
48     ext_drought = df[df['VHI'] <= 15] # Дані у періоди екстремальної засухи
49     max_val = ext_drought[ext_drought.Year.astype(str) == str(year)]['VHI'].max()
50     print(f"{max_val} - максимальний VHI екстремальної засухи в {year} році")
51     min_val = ext_drought[ext_drought.Year.astype(str) == str(year)]['VHI'].min()
52     print(f"\t{min_val} - мінімальний VHI екстремальної засухи в {year} році")
53
54     this_year = int(ext_drought[ext_drought['VHI'] == ext_drought['VHI'].min()]['Year'])
55     print(f"\t\t{this_year} - рік, в якому був найекстремальніший період засухи")
56
57     drought = df[(15 < df['VHI']) & (df['VHI'] <= 35)] # Дані у періоди помірної посухи
58     min_val = drought[drought.Year.astype(str) == str(year)]['VHI'].min()
59     print(f"\t\t{min_val} - мінімальний VHI помірної посухи в {year} році")
60     max_val = drought[drought.Year.astype(str) == str(year)]['VHI'].max()
61     print(f"{max_val} - максимальний VHI помірної посухи в {year} році")
62     pass
63
64
```

Отримали:

```
>>> from AD_lab1 import data_analysis
>>> data_analysis("D:\\KPI\\Data Analysis\\Lab 1\\NOAA_ID22 (Черкаська).csv", 2000)
14.64 - максимальний VHI екстремальної засухи в 2000 році
10.68 - мінімальний VHI екстремальної засухи в 2000 році
2000 - рік, в якому був найекстремальніший період засухи
15.71 - мінімальний VHI помірної посухи в 2000 році
34.78 - максимальний VHI помірної посухи в 2000 році
>>> _
```