

Прикладне програмування в науках про Землю

Практична робота №1

доц. Онищук В.І.
доц. Демидов В.К.
аспір. Охрімчук Р.Ю.

Хід роботи

1. встановлення Anaconda;
2. встановлення PostgreSQL;
3. налаштування conda середовища та встановлення необхідних бібліотек;
4. створення файлу конфігурації та ініціалізація середовища ods у створеній базі даних – datacube;
5. знайомство з Jupyter Notebook. Тестування створення підключення до бази даних datacube;
6. зробити звіт про виконану роботу в електронному вигляді.

1. Встановлення Anaconda

1.1 Для встановлення Anaconda потрібно перейти за посиланням: <https://www.anaconda.com/download>;

1.2 Натиснути на Download і завантажити актуальну версію програмного забезпечення Anaconda на ПК;

The screenshot shows the Anaconda website's download page. The header includes the Anaconda logo, navigation links (Platform, Solutions, Pricing, Partners, Resources, Company), and buttons for Sign-in and Contact Sales. The main heading is 'Anaconda Distribution' followed by 'Free Download' in large white text. Below this is the tagline 'Everything you need to get started in data science on your workstation.' A list of four features with green checkmarks is displayed: 'Free distribution install', 'Thousands of the most fundamental DS, AI, and ML packages', 'Manage packages and environments from desktop application', and 'Deploy across hardware and software platforms'. A prominent green 'Download' button with a Windows icon is shown. Below it, a link 'Get Additional Installers' is accompanied by icons for Windows, macOS, and Linux. A 'We use cookies' banner is at the bottom, featuring 'Accept all' and 'Reject all' buttons, along with links for 'More choices' and 'See our privacy policy'. A small robot icon with a red notification bubble is in the bottom right corner.

ANACONDA

Platform Solutions Pricing Partners Resources Company

Sign-in Contact Sales

Anaconda Distribution

Free Download

Everything you need to get started in data science on your workstation.

- ✓ Free distribution install
- ✓ Thousands of the most fundamental DS, AI, and ML packages
- ✓ Manage packages and environments from desktop application
- ✓ Deploy across hardware and software platforms

Download

Get Additional Installers

Windows macOS Linux

We use cookies

By clicking "Accept all", you agree to the storing of cookies on your device for functional, analytics, and advertising purposes.

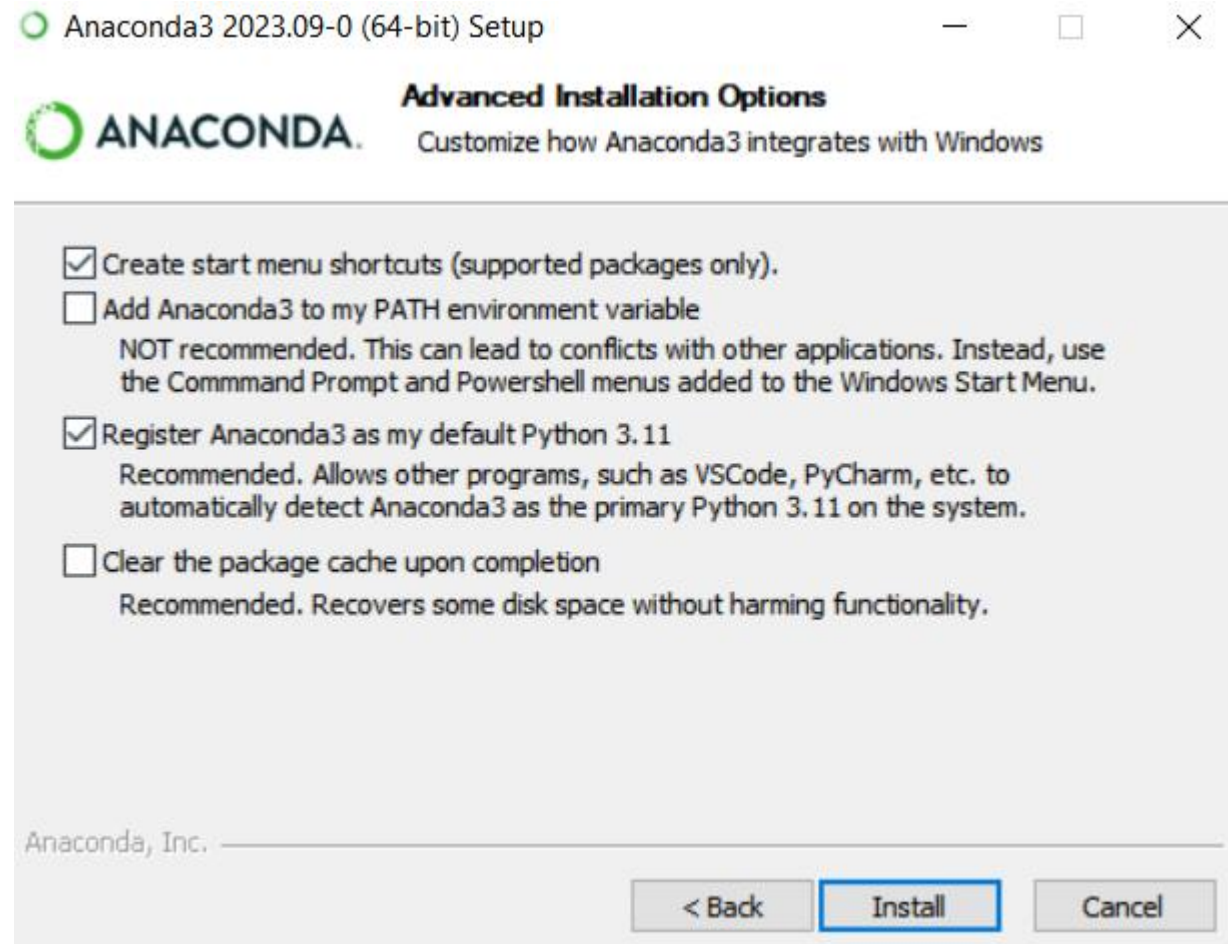
Accept all Reject all

More choices See our privacy policy

1

1. Встановлення Anaconda

1.3 Запустити застосунок із інсталяцією, натискати Next до моменту потрапляння на вікно із інсталяцією, перевірити чи всі опції обрані, як показано на картинці і далі почати процес встановлення;

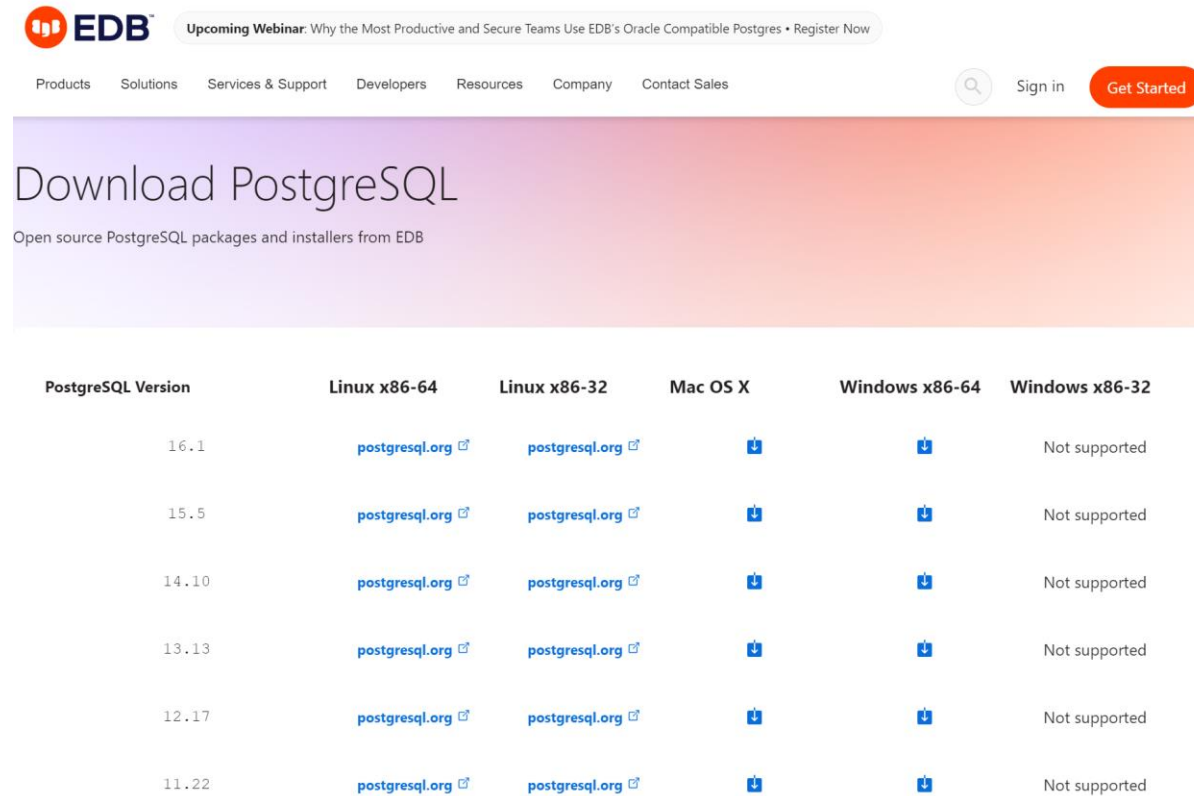


2. Встановлення PostgreSQL

2.1 Для встановлення PostgreSQL потрібно перейти за посиланням:

<https://www.enterprisedb.com/downloads/postgres-postgresql-downloads;>

2.2 Оберіть версію у залежності від вашої операційної системи (у рамках практичної роботи буде наведений прилад встановлення компонентів ODC для Windows x86-64);



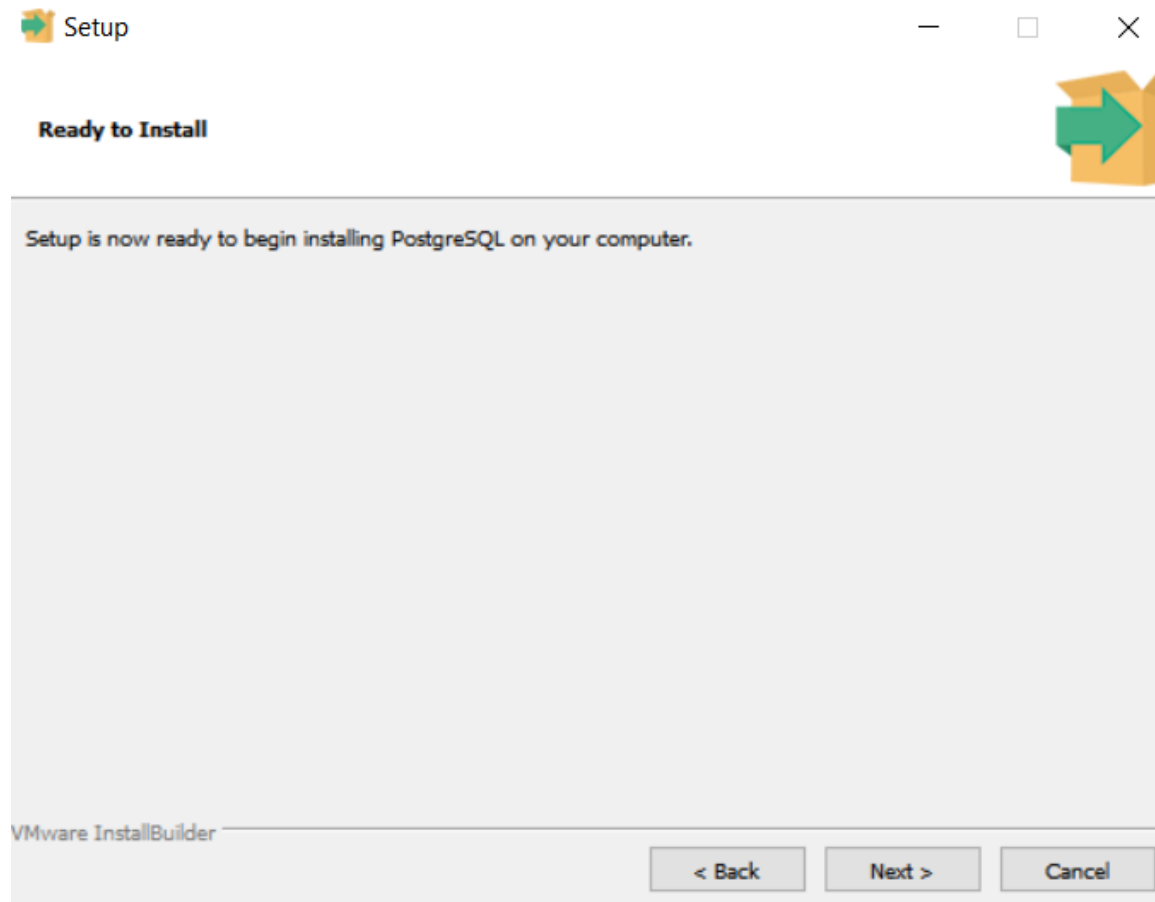
The screenshot shows the EDB PostgreSQL download page. At the top, there is a navigation bar with links for Products, Solutions, Services & Support, Developers, Resources, Company, and Contact Sales. A search icon and 'Sign in' link are also present. A prominent orange 'Get Started' button is on the right. Below the navigation bar, the main heading is 'Download PostgreSQL' with the subtitle 'Open source PostgreSQL packages and installers from EDB'. The main content area features a table with columns for PostgreSQL Version, Linux x86-64, Linux x86-32, Mac OS X, Windows x86-64, and Windows x86-32. The table lists versions 16.1, 15.5, 14.10, 13.13, 12.17, and 11.22. For each version, the Linux and Mac OS X columns provide download links to postgresql.org, while the Windows columns show download icons. The Windows x86-32 column consistently indicates 'Not supported'.

PostgreSQL Version	Linux x86-64	Linux x86-32	Mac OS X	Windows x86-64	Windows x86-32
16.1	postgresql.org	postgresql.org			Not supported
15.5	postgresql.org	postgresql.org			Not supported
14.10	postgresql.org	postgresql.org			Not supported
13.13	postgresql.org	postgresql.org			Not supported
12.17	postgresql.org	postgresql.org			Not supported
11.22	postgresql.org	postgresql.org			Not supported

2. Встановлення PostgreSQL

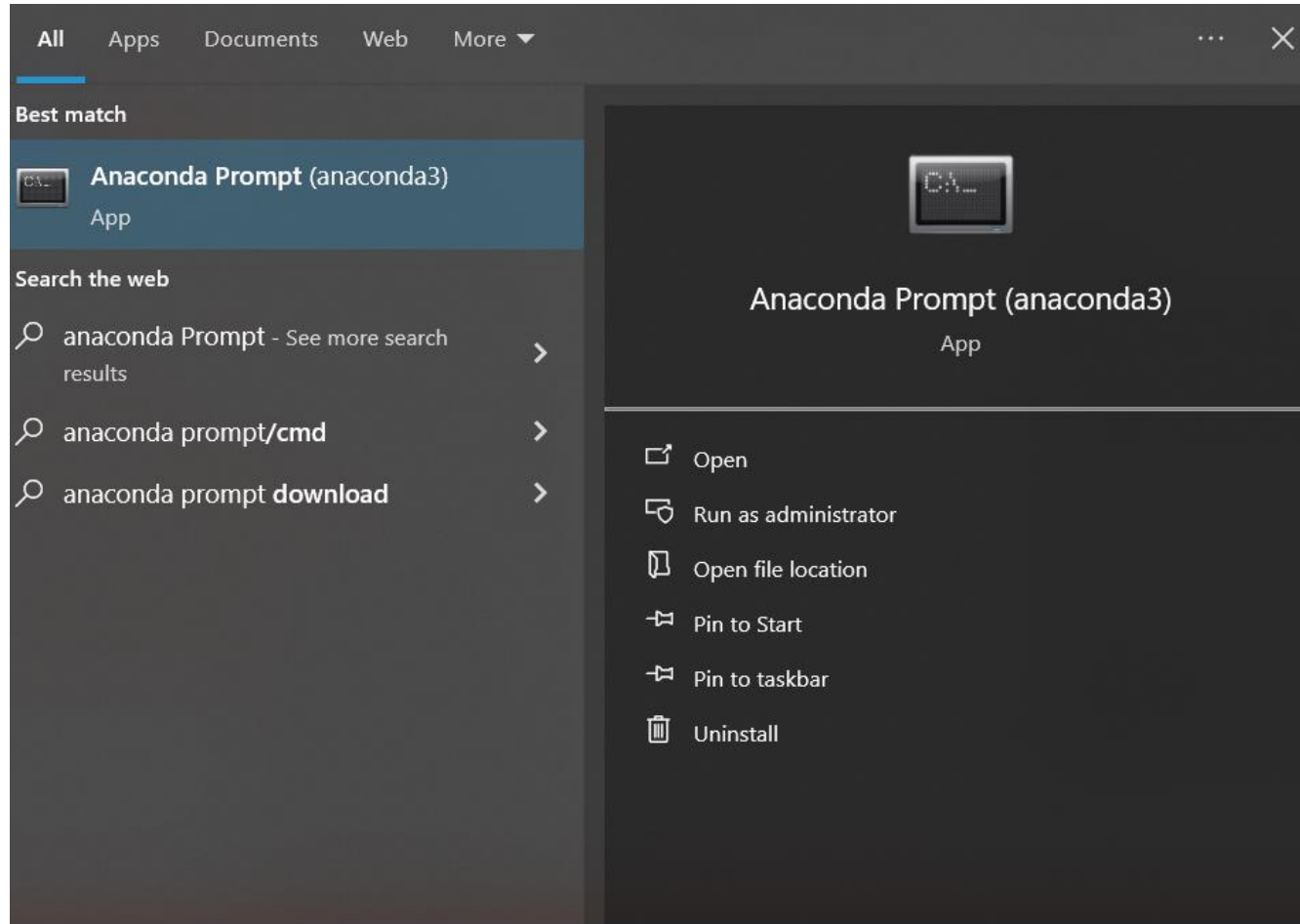
2.3 Запустити застосунок із інсталяцією, натискати Next до моменту початку інсталяції та почати встановлення.

*при створенні бази даних **postgres** встановіть пароль до бази – **root**.



3. Налаштування conda середовища та встановлення необхідних бібліотек

3.1 Із допомогою опції пошуку, знайти та запустити застосунок Anaconda Prompt



3. Налаштування conda середовища та встановлення необхідних бібліотек

3.2 Встановити всі необхідні бібліотеки для роботи із odc, для цього потрібно виконати такі команди:

- `conda config --add channels conda-forge`

Дана команда використовується для додавання каналу **conda-forge** до списку каналів в вашому конфігураційному файлі **conda**. **Conda** - це менеджер пакетів та система управління середовищами, яка дозволяє вам встановлювати, оновлювати та управляти пакетами програмного забезпечення та їх залежностями. **conda-forge** є одним із найпопулярніших каналів у спільноті **anaconda**, який надає велику кількість пакетів, що не завжди доступні в основному репозиторії Anaconda. Додавання **conda-forge** до списку каналів забезпечує ширший доступ до пакетів та часто оновлені версії.

- `conda create --name odc_env datacube`

```
xz                conda-forge/win-64::xz-5.2.6-h8d14728_0
yaml              conda-forge/win-64::yaml-0.2.5-h8ffe710_2
zict              conda-forge/noarch::zict-3.0.0-pyhd8ed1ab_0
zipp              conda-forge/noarch::zipp-3.17.0-pyhd8ed1ab_0
zlib              conda-forge/win-64::zlib-1.2.13-hcfcfb64_5
zstd              conda-forge/win-64::zstd-1.5.5-h12be248_0

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

(при необхідності підтвердження інсталяції, потрібно натиснути: **y**)

Дана команда використовується для створення нового середовища **conda** з назвою **odc_env** та встановлення пакета **datacube** та всіх пакетів для його стабільної роботи у це середовище. Також **conda** при встановленні пакетів вирішує проблеми сумісності бібліотек.

- `conda activate odc_env`

Дана команда використовується для активації раніше створеного середовища **conda** під назвою **odc_env**.

- `conda install jupyter notebook`

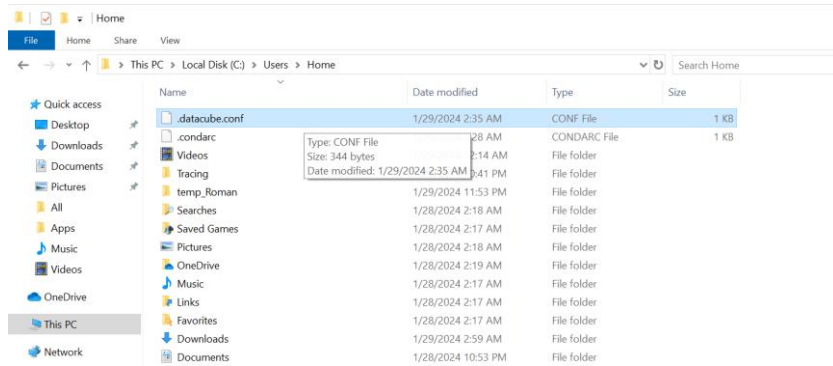
(також при необхідності підтвердження інсталяції, потрібно натиснути: **y**)

Дана команда встановить *Jupyter notebook*, який буде використовувати згодом, як середовище для написання і запуску коду

4. Створення бази даних datacube та файлу конфігурації для її синхронізації із ODC;

4.1 Потрібно перейти в дану директорію - C:\Users\%user_name%

4.2 Створити файл із назвою **.datacube.conf**



4.3 Додати до даного файлу дані конфігурацій та зберегти зміни

```
[datacube]
db_database: datacube

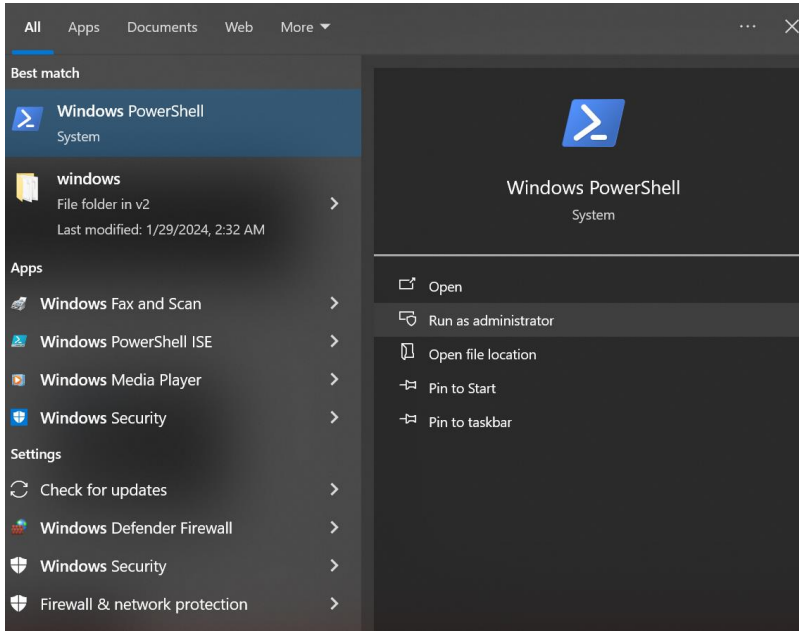
# A blank host will use a local socket. Specify a hostname (such as localhost) to use TCP.
db_hostname:localhost

# Credentials are optional: you might have other Postgres authentication configured.
# The default username otherwise is the current user id.
db_username:postgres
db_password:root # даний пароль, встановлюється користувачем при встановлені бази даних
```

```
.datacube.conf
1  [datacube]
2  db_database: datacube
3
4  # A blank host will use a local socket. Specify a hostname (such as localhost) to use TCP.
5  db_hostname:localhost
6
7  # Credentials are optional: you might have other Postgres authentication configured.
8  # The default username otherwise is the current user id.
9  db_username:postgres
10 db_password:root
11
```

4. Створення файлу конфігурації та ініціалізація середовища ods у створеній базі даних - datacube

4.4 Створити базу даних – datacube, для цього використаємо Windows PowerShell:



За допомогою команди **cd** потрібно перейти в директорію, де було інстальовано базу даних:

cd '.\Program Files\PostgreSQL\16\bin\'

Наступним кроком потрібно перейти до командного рядка бази даних, під користувачем postgres. Для цього потрібно виконати команду:

.\psql.exe -U postgres

Для входу потрібно ввести пароль, який був створений вами при інсталяції бази даних.

Наступним кроком потрібно запустити команду для створення бази даних datacube:

CREATE DATABASE datacube;

Команда **psql -U postgres** використовується для підключення до бази даних PostgreSQL за допомогою інтерфейсу командного рядка **psql**.

- **psql** - це інтерактивний термінал для роботи з PostgreSQL;
- **-U** - це параметр командного рядка, який вказує ім'я користувача для підключення;
- **postgres** після **-U** - це власне ім'я користувача, яке вказується.

Таким чином, команда **psql -U postgres** означає, що ви наказуєте **psql** підключитися до бази даних PostgreSQL, використовуючи ім'я користувача **postgres**. Це зазвичай стандартне ім'я користувача для PostgreSQL, особливо при виконанні адміністративних завдань. Користувач **postgres** зазвичай має широкі привілеї та часто використовується для початкової налаштування та конфігурації сервера баз даних.

```
PS C:\> cd '.\Program Files\PostgreSQL\16\bin\'
PS C:\Program Files\PostgreSQL\16\bin> .\psql.exe -U postgres
Password for user postgres:
psql (16.1)
WARNING: Console code page (850) differs from windows code page (1252)
8-bit characters might not work correctly. See psql reference
page "Notes for Windows users" for details.
Type "help" for help.

postgres=# CREATE DATABASE datacube;
CREATE DATABASE
postgres=#
```

4. Створення файлу конфігурації та ініціалізація середовища odc у створеній базі даних - datacube

4.5 Перевірити готовність всіх компонентів для ініціалізації середовища odc у базі даних:

Для виконання даної операції потрібно активувати раніше встановлене середовище odc_env, та виконати команду:

- **datacube system check**



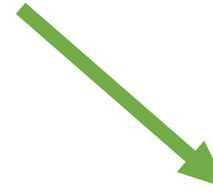
```
Anaconda Prompt
(base) C:\Users\Home>conda activate odc_env
(odc_env) C:\Users\Home>datacube system check
Version:      1.8.17
Config files: C:\Users\Home\.datacube.conf
Host:         localhost:5432
Database:     datacube
User:         postgres
Environment:  None
Index Driver: default

Valid connection: Error Connecting to Database: (psycopg2.OperationalError) connection to server at "localhost" (::1), port 5432 failed: FATAL: database "datacube" does not exist
ed: FATAL: database "datacube" does not exist

(Background on this error at: https://sqlalche.me/e/14/e3q8)

(odc_env) C:\Users\Home>
```

Якщо було отримано дану помилку або іншу, необхідно повернутися до попередніх кроків і перевірити правильність їх виконання



```
Anaconda Prompt
(base) C:\Users\Home>conda activate odc_env
(odc_env) C:\Users\Home>datacube system check
Version:      1.8.17
Config files: C:\Users\Home\.datacube.conf
Host:         localhost:5432
Database:     datacube
User:         postgres
Environment:  None
Index Driver: default

Valid connection: YES

(odc_env) C:\Users\Home>
```

Вітаю, все готове для наступного кроку!

4. Створення файлу конфігурації та ініціалізація середовища odc у створеній базі даних - datacube

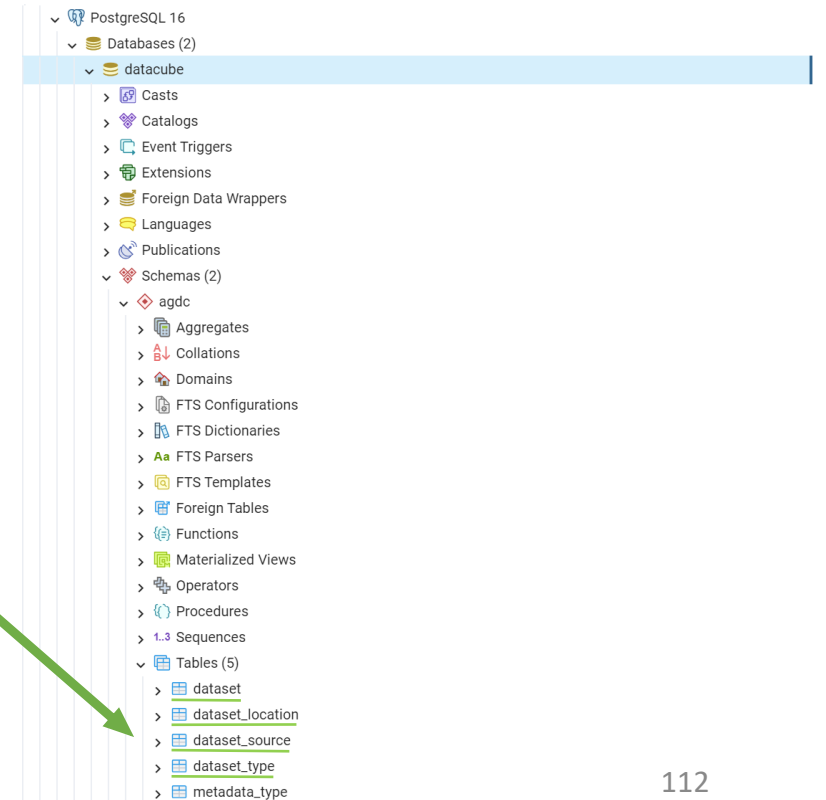
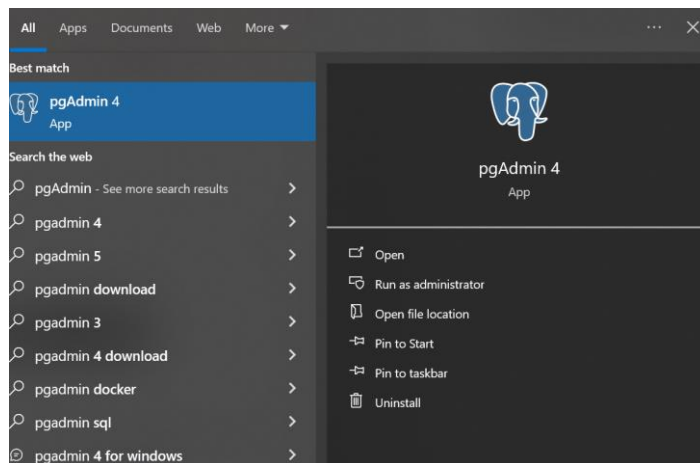
4.6 Для ініціалізації середовища **odc** у базі даних у базі даних **datacube**:

- **datacube -v system init**

```
(odc_env) C:\Users\Home>datacube -v system init
2024-01-29 03:20:00,958 21288 datacube INFO Running datacube command: C:\Users\Home\anaconda3\envs\odc_env\Scripts\datacube -v system init
Initialising database...
2024-01-29 03:20:01,323 21288 datacube.drivers.postgres._core INFO Ensuring user roles.
2024-01-29 03:20:01,430 21288 datacube.drivers.postgres._core INFO Creating schema.
2024-01-29 03:20:01,434 21288 datacube.drivers.postgres._core INFO Creating tables.
2024-01-29 03:20:01,552 21288 datacube.drivers.postgres._core INFO Creating triggers.
2024-01-29 03:20:01,567 21288 datacube.drivers.postgres._core INFO Creating added column.
2024-01-29 03:20:01,570 21288 datacube.drivers.postgres._core INFO Adding role grants.
2024-01-29 03:20:01,573 21288 datacube.index.postgres.index INFO Adding default metadata types.
Created.
Checking indexes/views.
2024-01-29 03:20:01,839 21288 datacube.drivers.postgres._api INFO Checking dynamic views/indexes. (rebuild views=True, indexes=False)
Done.
```

(при успішному виконанні команди, буде отримано дане повідомлення)

4.7 Наступним кроком, потрібно відкрити **pgAdmin**, відкрити базу даних **datacube** та впевнитися у тому, що були створені таблиці для роботи із odc:



5. Знайомство з Jupyter Notebook. Тестування створення підключення до бази даних datacube

5.1 Для запуску **jupyter notebook** потрібно активувати середовище **odc_env** і виконати у командному рядку:

- **jupyter notebook**

```
(odc_env) C:\Users\Home>jupyter notebook --notebook-dir='C:/Applied Programming in Earth Sciences'
[C 2024-01-30 01:34:35.961 ServerApp] Bad config encountered during initialization: No such directory: 'C:/Applied_Programming_in_Earth_Sciences'

(odc_env) C:\Users\Home>jupyter notebook --notebook-dir='C:/PhD/lectures/Applied Programming in Earth Sciences'
[W 2024-01-30 01:34:50.677 ServerApp] A `jupyter_server_extension_points` function was not found in notebook_shim. Instead, a `jupyter_server_extension_paths` function was found and will be used for now. This function name will be deprecated in future releases of Jupyter Server.
[I 2024-01-30 01:34:50.679 ServerApp] jupyter_lsp | extension was successfully linked.
[I 2024-01-30 01:34:50.709 ServerApp] jupyter_server_terminals | extension was successfully linked.
[I 2024-01-30 01:34:50.731 ServerApp] jupyterlab | extension was successfully linked.
[I 2024-01-30 01:34:50.752 ServerApp] notebook | extension was successfully linked.
[I 2024-01-30 01:34:51.353 ServerApp] notebook_shim | extension was successfully linked.
[I 2024-01-30 01:34:51.443 ServerApp] notebook_shim | extension was successfully loaded.
[I 2024-01-30 01:34:51.447 ServerApp] jupyter_lsp | extension was successfully loaded.
[I 2024-01-30 01:34:51.459 ServerApp] jupyter_server_terminals | extension was successfully loaded.
[I 2024-01-30 01:34:51.466 LabApp] JupyterLab extension loaded from C:\Users\Home\anaconda3\envs\odc_env\Lib\site-packages\jupyterlab
[I 2024-01-30 01:34:51.476 LabApp] JupyterLab application directory is C:\Users\Home\anaconda3\envs\odc_env\share\jupyter\lab
[I 2024-01-30 01:34:51.480 LabApp] Extension Manager is 'pypi'.
[I 2024-01-30 01:34:51.486 ServerApp] jupyterlab | extension was successfully loaded.
[I 2024-01-30 01:34:51.495 ServerApp] notebook | extension was successfully loaded.
[I 2024-01-30 01:34:51.496 ServerApp] Serving notebooks from local directory: C:/PhD/lectures/Applied Programming in Earth Sciences
[I 2024-01-30 01:34:51.510 ServerApp] Jupyter Server 2.12.5 is running at:
[I 2024-01-30 01:34:51.511 ServerApp] http://localhost:8888/tree?token=8146b4dfb1d53cbb65e30c8dcf78252f9fa94ac4f0ee968
[I 2024-01-30 01:34:51.519 ServerApp] http://127.0.0.1:8888/tree?token=8146b4dfb1d53cbb65e30c8dcf78252f9fa94ac4f0ee968
[I 2024-01-30 01:34:51.520 ServerApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 2024-01-30 01:34:51.583 ServerApp]

To access the server, open this file in a browser:
file:///C:/Users/Home/AppData/Roaming/jupyter/runtime/jpserver-15752-open.html
Or copy and paste one of these URLs:
http://localhost:8888/tree?token=8146b4dfb1d53cbb65e30c8dcf78252f9fa94ac4f0ee968
http://127.0.0.1:8888/tree?token=8146b4dfb1d53cbb65e30c8dcf78252f9fa94ac4f0ee968
```

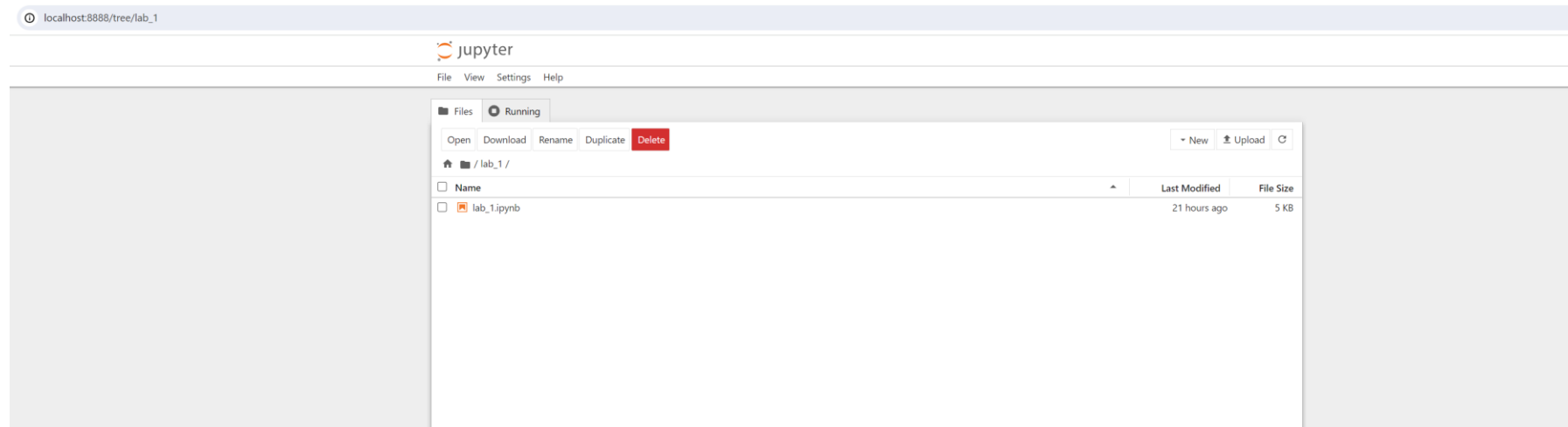
(при виконанні цієї команди має автоматично відкритися вікно веб-браузера, який був встановлений за замовчуванням. Якщо це не відбулося, скопіюйте згенерований URL-адресу разом із токеном і вставте її у адресний рядок будь-якого браузера, а потім перейдіть за посиланням)

5. Знайомство з Jupyter Notebook. Тестування створення підключення до бази даних datacube

*при необхідності можна налаштувати кореневу директорію, яка буде відкриватися при запуску Jupyter Notebook. Для цього потрібно використати параметр **--notebook-dir**.

Наприклад:

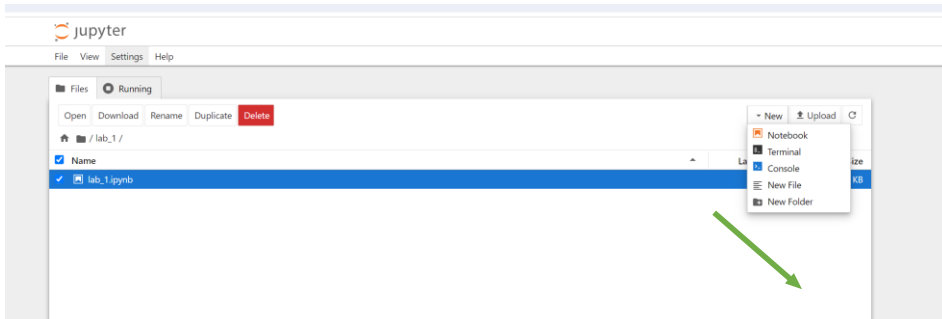
jupyter notebook --notebook-dir='C:/Applied_Programming_in_Earth_Sciences'.



(при відкритті Jupyter Notebook у вкладці Files, буде відкрито цільову директорію)

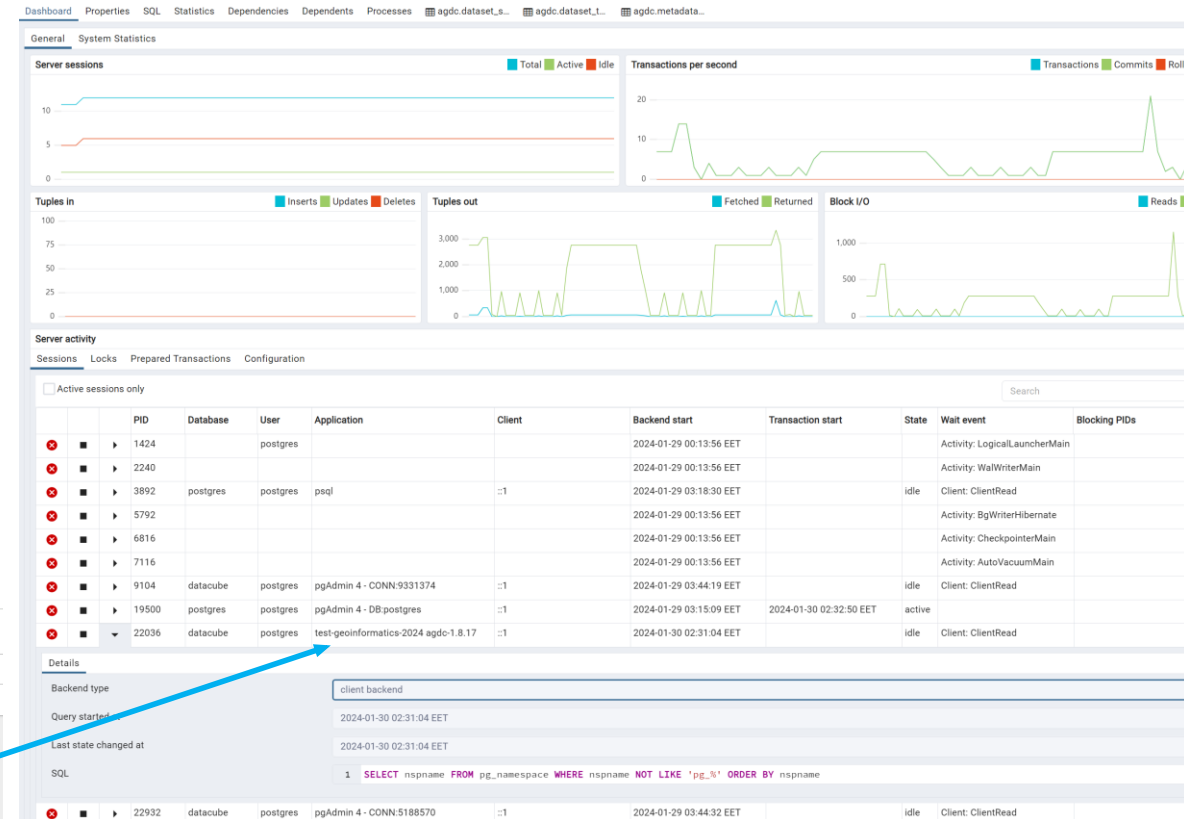
5. Знайомство з Jupyter Notebook. Тестування створення підключення до бази даних datacube

5.2 Створіть Notebook із назвою **lab_1**:



5.3 У створеному Notebook виконайте наступні команди:

- `import datacube`
 - `dc = datacube.Datacube(app="test_geoinformatics_2024")`
- (ця команда створює підключення до бази даних та дозволяє проводити маніпуляції з геоданими на основі раніше доданих метаданих)



(після виконання даної команди через застосунок **pgAdmin** можна побачити створення нової сесії - **test-geoinformatics-2024 agdc-1.8.17**)

6. Зробити звіт про виконану роботу в електронному вигляді.

6.1 Для успішного виконання практичної роботи команда `dc = datacube.Datacube(app="test")` має виконуватися без помилок. Це буде означати, що все готово для додавання продуктів та їх датасетів. Як завершальний етап виконання цієї роботи, потрібно підготувати звіт про виконану роботу і надіслати його в Google Classroom.