

# Python

01

# Опис мови Python

Python — "високорівнева мова програмування, головною особливістю якого є читабельність та синтаксис, за допомогою якого можна легко описати будь-яку концепцію в декількох рядках коду."

© Гвідо Ван Россум (творець мови)

Python — інтерпретована високорівнева мова програмування загального призначення.

# Linux як середовище розробки на python.

Для програміста головна мета розробки на linux – гарантувати однакове виконання програми на сервері та на комп'ютері, на якому відбувалася розробка.

# Linux

Основні команди:

**cd [path]** - перехід до іншої категорії

**ls** – вміст поточної категорії

**mkdir** - створення папки

**pwd** - поточне розташування

**touch [filename]** - створення файлу

**cat [filename]** - відображення вмісту текстового файлу

**nano [filename]** - редагування файлу з консолі

**rm [filename]** - видалення файлу

**rm -rf [dir\_name]** - видалення папки

# Pip

pip - менеджер пакетів для Python

Основні команди:

pip install [packagename] - встановлення пакету

pip install -r [path\_to\_requirements] - встановлення переліку пакетів

pip freeze - відображення встановлених пакетів

pip uninstall [packagename] - видалення пакетів

# Virtualenv

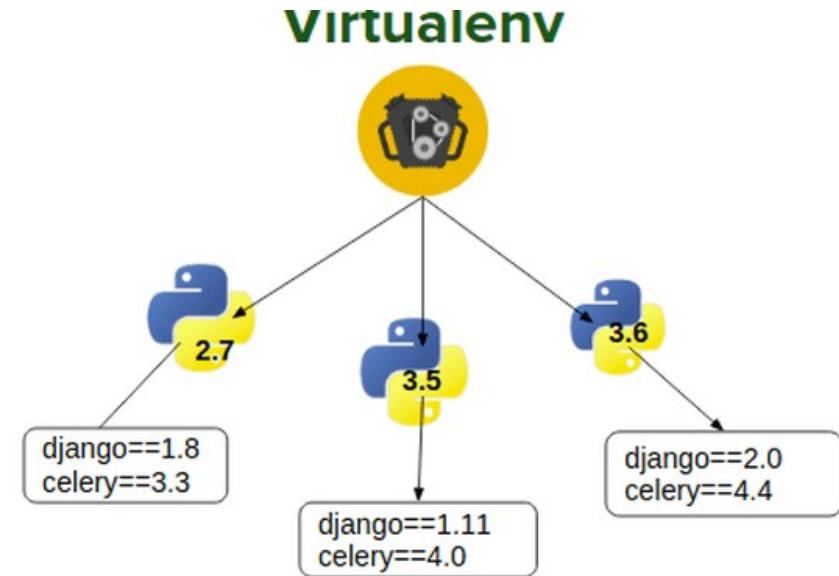
virtualenv - інструмент для створення ізольованого оточення Python.

Основні команди:

`python3 -m venv project_name_venv`

`source [virtualenvpath]/bin/activate` - активація оточення

`deactivate`



# **Основи Git**

**git <команда> --<опція> <аргументи>**

**git help <команда>**

**git config --global <поле> <значення>**

## **Встановлення та налаштування Git**

**sudo apt-get install git**

**git config --global user.name "My Name"**

**git config --global user.email myEmail@example.com**

# Основні команди для роботи з Git

**git add** додає вміст робочого каталогу до інексу для наступного комміту

**git status** показує стан файлів у робочому каталогі та індексі

**git diff** використовується для обчислення різниці між будь-якими двома GIT деревами

**git commit** бере всі дані, додані в індекс за допомогою **git add**, і зберігає їх зліпок у внутрішній базі даних, а потім зрушує покажчик поточної гілки на цей зліпок

**git reset**, як можна здогадатися з назви, використовується в основному для скасування змін

**git pull** отримання даних та інтеграція з іншими репозиторієм або локальною гілкою

**git rm** для видалення файлів з інексу та робочої копії. Вона схожа на **git add**, тільки вона видаляє, а не додає файли для наступного комміту

# Основні команди для роботи з Git

**git init** щоб перетворити звичайний каталог на Git репозиторій і почати версіонувати файли в ньому

**git clone** додає новий віддалений репозиторій для зазначеного URL на локальний комп'ютер

**git branch** створення нової гілки або перегляд існуючих гілок

**git checkout** перемикання на іншу гілку

**git merge** для злиття однієї або декількох гілок у поточну

**git log** для перегляду історії коммітів, починаючи з найсвіжішого

**git push** передача змін на віддалений репозиторій

## **Терміни Git**

гіт репозиторій (git repository);

коміт (commit);

гілка (branch);

смержити (merge);

конфлікти (conflicts);

спулити (pull);

запушити (push);

як ігнорувати якісь файли (.gitignore)

## **Стан у Git**

невідслідковується (untracked);

змінене (modified);

підготовлений (staged);

закомічене (committed).

## Typical Merge

