

Лабораторна робота №1

Вибір і тестування інтерпретатора Python

Мета роботи: набути навичок роботи у середовищі розробки Python. Ознайомитись з основними можливостями бібліотеки Numpy, Pandas, SciPy

Література

Python: <https://www.python.org/downloads/>

Jupyter <https://jupyter.org/>

Pandas: <https://pandas.pydata.org/>

SciPy <https://docs.scipy.org/doc/numpy/reference/routines.linalg.html>

Путівник мовою програмування Python- https://pythonguide.rozh2sch.org.ua/#_модулі_python

Зміст роботи

Завдання 1. Виберіть інтерпретатор мови Python.

Методичні рекомендації

Щоб почати працювати з Python, потрібно отримати доступ до інтерпретатору Python. Існує кілька загальних способів зробити це:

– Для установки інтерпретатора Python на комп'ютер потрібно завантажити дистрибутив. Завантажити його можна з офіційного сайту, перейшовши за посиланням <https://www.python.org/downloads/> Необхідно завантажити інсталяційний файл для операційної системи, встановленої на комп'ютері і запустити його (рис.1).

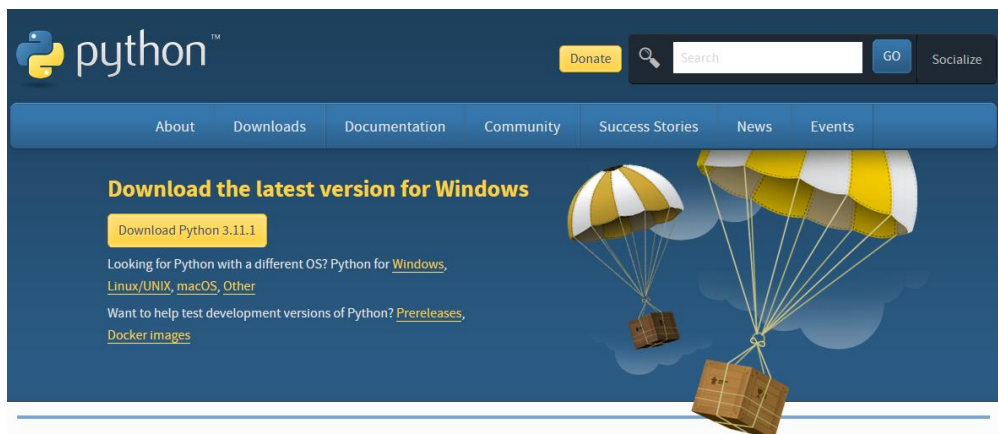


Рис.1. Офіційний сайт Python

– Деякі операційні системи, особливо Linux, надають менеджер пакетів, який можна запустити для установки Python.

– Для macOS, кращий спосіб встановити Python включає в себе установку менеджера пакетів під назвою Homebrew.

В якості альтернативи, існує ряд сайтів, які дозволяють отримати доступ до інтерпретатора Python он-лайн (Наприклад Colaboratory рис.2,

juvyter рис.3), без необхідності в установці чогось на вашому комп'ютері. Перш ніж почати вивчати Python потрібно мати стабільну версію інтерпретатора.

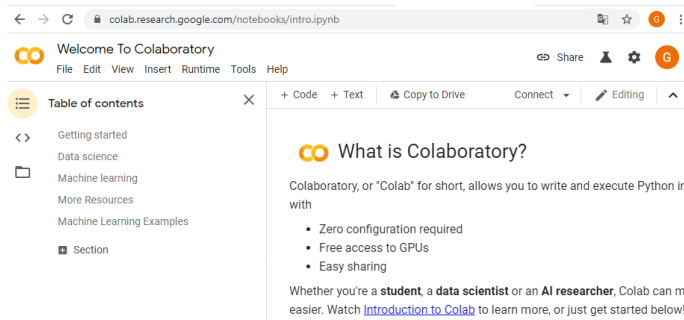


Рис.2. Colaboratory



Рис.3. jupyter

Завдання 2. Перевірити список доступних модулів та дізнатися версію встановлених бібліотек. Протягом навчання знадобляться наступні пакети:

- *Pandas*
- *NumPy*
- *Matplotlib*
- *Scipy*
- *Scikit-learn*

Методичні рекомендації

Перевірити список доступних модулів можна за допомогою команди (`help('modules')`).

Бібліотеку *NumPy*, досить часто використовують при вирішенні завдань в рамках машинного навчання і аналізу даних, вона також ефективно обробляє багатовимірні масиви. Багато інших бібліотеки побудовані на NumPy: *Pandas*, *Matplotlib*, *SciPy* або *scikit-learn*.

Бібліотека аналізу даних *Pandas* - це бібліотека з відкритим кодом, що має ліцензію BSD, що забезпечує високопродуктивні, прості у використанні структури даних та інструменти аналізу даних для мови програмування Python.

Pandas дуже добре підходить для роботи з одновимірними і двовимірними таблицями даних, є можливість працювати з файлами csv, таблицями Excel, може стикуватися з мовою R.

Matplotlib - бібліотека двовимірних числових побудов. Вона відмінно підходить для аналізу даних і створення високоякісних фігур різного формату, включаючи інтерактивні графіки і схеми.

Для того, щоб дізнатися версію бібліотеки, необхідно вбити наступну команду (наприклад для *Pandas* та *NumPy*):

```
import pandas as pd
print(pd.__version__)
```

Завдання 3. Створити csv-файл, використовував функцію *writer* або клас *DictWriter* (*ІоЗанписів*).

№	Завдання
1, 11	Довідник контактів (Прізвище, ім'я, телефон, дата, місяць, рік народження, стать тощо)
2, 12	Заробітна плата за 2 роки підрозділу (Прізвище, ім'я, стать, телефон, посада, заробітна плата, тощо)
3, 13	Деканат (Прізвище, ім'я, телефон, дата, місяць, рік народження, стать, група, бюджет/контракт, курс тощо)
4, 14	Адміністратор готелю (Список номерів: клас, число місць, ціна за номер сутки, зайнятий чи ні, якщо зайнятий скільки суток)
5, 15	Ріелтор (квартира/будинок, кількість кімнат, площа, ціна за 1кв.м, поверх, місто)
6, 16	Склад (найменування, кількість, ціна за одиницю, приналежність до класу фрукти/овочі тощо)
7, 17	Картотека фільмів (Назва, рік випуску, категорія, жанр, тривалість фільму, рейтинг)
8, 18	Затвердження житлового кредиту(Унікальний ідентифікатор позики, стать, заявник одружений (так/ні), утриманці, кількість утриманців, освіта, індивідуальний підприємець (так/ні), дохід заявника, сума кредиту в тисячах, термін кредиту в місяцях)
9, 19	Працевлаштування ((Прізвище, ім'я, телефон, дата, місяць, рік народження, стать, освіта, стаж, спеціальність)
10, 20	Студент (Прізвище, ім'я, дата, місяць, рік народження, стать, група, бюджет/контракт, курс, результати 4 семестрів (кожен семестр - 4 екзамени та 6 заліків)

Методичні рекомендації

Формат csv (Comma Separated Values) - це один з поширених файлових форматів, що зберігають інформацію в зручному вигляді. Кожен рядок у файлі представляє окремий запис або рядок, який складається з окремих стовпців, розділених комами (;, |). Python для спрощення роботи з ним надає спеціальний вбудований модуль csv.

Приклад коду для створення файлу з п'ятьма записами:

```
import csv
test = [
    ['Python', 'Guido van Rossum'],
    ['Scala', 'Martin Odersky'],
    ['PHP', 'Rasmus Lerdorf'],
    ['Ruby', 'Yukihiro Matsumoto'],
    ['C', 'Dennis Ritchie']
]
with open('test.csv', 'wt', newline='') as frecord:
    csvrecord = csv.writer(frecord)
    csvrecord.writerows(test)
```

Приклад коду для читання файлу:

```
with open('test.csv', 'rt') as freading:
    creading = csv.reader(freading)
    test = [row for row in creading]
print(test)
```

Результат:

```
[['Python', 'Guido van Rossum'], ['Scala', 'Martin Odersky'], ['PHP', 'Rasmus Lerdorf'], ['Ruby', 'Yukihiro Matsumoto'], ['C', 'Dennis Ritchie']]
```

Дані можуть мати формат списку словників, а не списку списків. Зчитаємо файл test.csv, цього разу використовуючи нову функцію *DictReader()*, вказуючи імена стовпців:

```
with open('test.csv', 'rt') as freading:
    creading = csv.DictReader(freading, fieldnames=['language', 'developer'])
    test = [row for row in creading]
print(test)
```

Результат:

```
[OrderedDict([('language', 'Python'), ('developer', 'Guido van Rossum')]),
OrderedDict([('language', 'Scala'), ('developer', 'Martin Odersky')]),
OrderedDict([('language', 'PHP'), ('developer', 'Rasmus Lerdorf')]),
OrderedDict([('language', 'Ruby'), ('developer', 'Yukihiro Matsumoto')]),
OrderedDict([('language', 'C'), ('developer', 'Dennis Ritchie')])]
```

Перепишемо csv-файл за допомогою функції *DictWriter()*. Також зробимо виклик функції *writeheader()*, щоб записати початковий рядок, що містить імена стовпців, у csv-файл:

```
test = [
    {'language': 'Python', 'developer': 'Guido van Rossum'},
    {'language': 'Scala', 'developer': 'Martin Odersky'},
    {'language': 'PHP', 'developer': 'Rasmus Lerdorf'},
    {'language': 'Ruby', 'developer': 'Yukihiro Matsumoto'},
    {'language': 'C', 'developer': 'Dennis Ritchie'},
]
with open('test.csv', 'wt', newline='') as frecord:
    crecord = csv.DictWriter(frecord, ['language', 'developer'])
    crecord.writeheader()
    crecord.writerows(test)
```

Цей код створює файл test.csv з рядком заголовку.

Завдання 4. Провести первинний аналіз створеного файлу.

- Завантажте файл і отримайте загальну інформацію про файл.
- Використовував функції head() і tail() можна продивитися п'ять перших і останніх записів.
- Розрахуйте статистичні характеристики (середнє арифмічне, моду, медіану, максимальне і мінімальне значення) за деякими параметрами
- Зробіть висновки щодо набору даних.

Завдання 5. З Kaggle завантажте файл:

<https://www.kaggle.com/datasets/salimwid/global-billionaire-wealth-and-sources-2002-2023> проведіть первинне дослідження набору даних.

Контрольні запитання

1. Звідки походить назва мови програмування Python?
2. Для яких цілей використовується Python?
3. Яке розширення мають файли із програмами, написаними на мові Python?
4. Як здійснюється установка сторонніх модулів?
5. Опишіть роботу з файлами xls, json, csv.
6. Як зчитувати і записувати список рядків, кожен з яких містить список стовпців у csv- файлі?
7. Виконайте наступну послідовність дій і поясните отримані результати.

```
a = [1, 2, 3, 4, 5, 6, 7, 6, 5, 4, 3, 2, 1]
b = [' ' * 2 * (7 - i) + 'Help' * i for i in a]
for line in b:
    print(line)
```

8. Що буде виведено

```
msg='decent Python'
print(msg[6:11])
print(msg[-12:-7])
```

9. Який символ використовується у мові Python для позначення коментарів?