

Введення у Python

№ уроку: 1 Курс: Python Starter

Засоби навчання: PyCharm

Огляд, мета та призначення уроку

Після завершення уроку учні матимуть уявлення про особливості мови програмування Python, історію її появи, переваги та недоліки, області застосування, навчатися використовувати інтерпретатор і створять свою першу програму цією мовою.

Вивчивши матеріал даного заняття, учень зможе:

- Використовувати інтерпретатор мови Python для запуску скриптів
- Використовувати інтерпретатор мови Python в інтерактивному режимі
- Створювати проекти та запускати код в інтегрованому середовищі розроблення

Зміст уроку

1. Загальні відомості
2. Знайомство з Python
3. Актуальність мови програмування
4. Особливості Python
5. Історія мови
6. Области застосування
7. Переваги і недоліки
8. Встановлення Python та IDE
9. Створення найпростіших програм на Python

Резюме

Мова програмування Python – це інструмент для створення програм найрізноманітнішого призначення. Мова з низьким порогом входження, що робить її відмінним вибором у якості першої мови.

Мова програмування – це формальна знакова система, що використовується для написання комп'ютерних програм. Мова визначає набір лексичних, синтаксичних та семантичних правил. Це дозволяє чітко визначити, як слід спілкуватися з комп'ютером.

Python – високорівнева, інтерпретована, об'єктно-орієнтована, скриптова мова програмування зі строгою динамічною типізацією.

Python є мультипарадигмальною мовою програмування, що підтримує імперативне, процедурне, структурне, об'єктно-орієнтоване програмування, метапрограмування та функціональне програмування. Задачі узагальненого програмування вирішуються за рахунок динамічної типізації.

Розробники мови зробили акцент на читабельності та зручності написання коду. Нині вона використовується у всіх IT галузях.

Python – найпопулярніша мова програмування 2021 року за версією авторитетного індексу TIOBE станом на січень 2022 року.

Ця мова програмування стає найпопулярнішою вже другий рік поспіль. Нагорода присуджується мові програмування, що набрала найбільшого приросту рейтингу за рік.

Історія мови

Python розроблявся з кінця 80-х. Його творець, Гвідо ван Россум, назвав мову на честь британського телешоу "Літаючий цирк Монті Пайтона", а не тому, що любив змії.

На той час Россум працював у голландському інституті CWI. Створював Python для студентів, оскільки мови на той час були складними для студентів. Взявши найкращі можливості з інших мов програмування, Python було представлено 20 лютого 1991 року.

У 2018 році Гвідо пішов у постійну відпустку з проекту Python, залишивши за собою право бути звичайним розробником. Зараз Гвідо ван Россум працює в Dropbox і серед розробників Python відомий як "великодушний довічний диктатор" проекту, що означає, що він продовжує спостерігати за процесом розроблення Python.

Python мав дві актуальні гілки: 2.x, 3.x. На початку 2020 року підтримка Python 2.x припинилася. Нові проекти слід створювати на Python 3 – це гарантує наявність актуальних бібліотек та виправлення старих помилок інтерпретатора.

Застосування

Python застосовується для створення скриптів та утиліт. Широко відомий у науковій сфері та використовується для аналізу даних. Іноді Python можна побачити серед мов для створення ігор. У нашому курсі ми будемо розглядати Python як мову для створення веб-додатків.

Ось ряд компаній, що успішно застосували Python у своїх продуктах: Dropbox, Blender, Mercurial, Instagram, Pinterest, EVE Online, World of Tanks, Civilization IV.

На жаль, Python не можна використовувати для низькорівневого програмування. Це означає, що нею драйвери не написати.

Особливості

Серед особливостей Python можна виділити такі:

- * простота синтаксису та вивчення;
- * підтримка різних парадигм програмування;
- * інтерпретована;
- * динамічна типізація;
- * відкритість;
- * кросплатформність;
- * існування альтернативних реалізацій, які вирішують певні проблеми (Stackless Python, PyPy) або інтегрують його в певну платформу (IronPython для .NET, Jython для Java).

Творець Python - досвідчений програміст. Він постарався усунути всі недоліки мов, що існували тоді. Йому це вдалося. Python може похвалитися: * простотою та читабельністю * великою стандартною бібліотекою та величезним вибором сторонніх бібліотеки та модулів * стислістю коду та економією часу розробника * можливістю зв'язування з кодом іншими мовами

Але не обійшлося без недоліків:

- * низька порівняно з компільованими мовами швидкість роботи;
- * GIL.

Щоб розпочати вивчення Python, необхідно вивчити поняття трансляції, компіляції та інтерпретації.

Трансляція — перетворення програми, написаної однією мовою програмування, на програму іншою мовою.

Компіляція — трансляція програми, що написана мовою високого рівня, в еквівалентну програму низькорівневою мовою, близькою до машинного коду.

Інтерпретація — аналіз рядок за рядком (порядковий аналіз), оброблення та виконання коду. Python є інтерпретованою мовою.

Інтерпретатор — програма (різновид транслятора), що виконує інтерпретацію.

Встановлення Python та IDE

Щоб виконати наш код, треба встановити програму-перекладач (інтерпретатор), яка пояснюватиме комп'ютеру, що означає наш код.

Інтерпретатор можна скачати з офіційного сайту мови: <https://python.org>. Оберіть відповідний до вашої системи інсталятор і дотримуйтесь інструкцій.

Для комфортного розроблення нам знадобиться **IDE — інтегроване середовище розробки**. Іншими словами, це система програмних засобів, що використовується програмістами для розроблення програмного забезпечення.

Середовище розробки обов'язково включає в себе:

- текстовий редактор;
- компілятор та/або інтерпретатор або засоби інтеграції з ним;
- засоби автоматизації збирання;
- відлагоджувач.

Також IDE може включати:

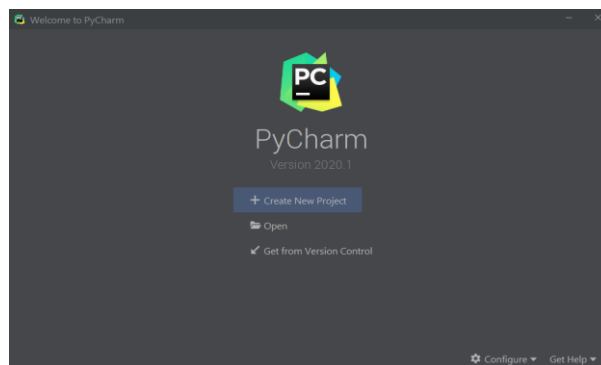
- засоби інтеграції із системами контролю версій;
- інструменти конструювання графічного інтерфейсу користувача

Офіційний сайт IDE PyCharm (рекомендується): <http://www.jetbrains.com/pycharm/PyCharm>

Практика

Давайте створимо наш перший Python проект. У прикладах використовуватиметься середовище розроблення PyCharm.

У стартовому діалозі необхідно обрати → "Create New Project", обрати тип проекту "Pure Python" та вказати шлях до папки, де він зберігатиметься.



Стартове діалогове вікно

Потім створимо перший файл вихідного коду. Для цього потрібно обрати команду меню File → New, натиснути праву кнопку миші в дереві проекту ліворуч і обрати New, після чого обрати Python File і ввести його ім'я.

Програма на Python, яка виводить вітання "Hello, World!":

```
print("Hello, World!")
```

Hello, World!

Тут *print* – функція, а *"Hello, World!"* – рядок.

Більш детально ці поняття будуть розглянуті на наступних уроках. Для першого запуску скрипта потрібно натиснути Alt+Shift+F10 і вибрати потрібний файл вихідного коду зі списку. В наступні рази просто Shift+F10.

Можна виводити кілька значень:

```
print("first", "second")
```

first second

І [привласнювати значенням імена](#):

```
name = "Alex"
```

```
print(name)
```

Alex

Також можна [ввести значення з клавіатури](#):

```
text = input("Enter some text : ")
```

```
print(text)
```

Enter some text : CyberBionic Systematics

CyberBionic Systematics

Будь-який файл вихідного коду на Python – це звичайний текстовий файл із розширенням.py. Скрипти можна запускати за допомогою командного рядку: `python file_name.py`.

Для експериментів та навчання зручно використовувати інтерпретатор в інтерактивному режимі. Для цього можна відкрити командою “python” інтерпретатора в командному рядку або, що зручніше, скористатися графічним інтерфейсом до нього, запустивши додаток IDLE.

Закріплення матеріалу

- Що таке програмування?
- Що таке інтерпретація?
- Що таке компіляція?
- Мова програмування Python інтерпретована чи компільована?
- Навіщо використовують мову програмування Python? Для чого її не можна використовувати?
- Які переваги та недоліки Python ви знаєте?
- Що таке інтегроване середовище розроблення?

Додаткове завдання

Завдання 1

Створіть новий проект в інтегрованому середовищі PyCharm. Створіть файл вихідного коду та напишіть програму, яка виводить ваше ім'я, прізвище та вік. Запустіть його.

Завдання 2

Створіть другий файл з кодом і напишіть програму, яка запитує у користувача, як його звуть, вітається з ним і бажає йому гарного дня. Запустіть його. Переключіться на перший скрипт і запустіть його. Перейдіть назад на другий скрипт

Самостійна діяльність учня

Завдання 1

Встановіть Python та PyCharm (або іншу IDE за допомогою Python, яка вам зручна). Дослідіть налаштування, налаштуйте середовище розроблення для себе: підберіть колірну схему та шрифт редактора, які вам подобаються, увімкніть або відключіть відображення номерів рядків, підсвічування поточного рядка, відображення роздільників між ділянками коду тощо.

Завдання 2

Створіть скрипт мовою Python, використовуючи звичайний текстовий редактор (можна використовувати редактор коду, такий як, наприклад, Sublime Text). Запустіть його за допомогою консолі. Запустіть його подвійним клацанням у провіднику Windows. Придумайте, як, використовуючи те, що ви вже вивчили, зробити так, щоб вікно консолі не закривалося відразу після запуску скрипту подвійним клацанням.

Завдання 3

Відкрийте IDLE (під Windows та OS X цей додаток встановлюється разом із Python). Поекспериментуйте зі звичайними арифметичними виразами. Спробуйте задати ім'я значенню якого-небудь виразу. Спробуйте вивести значення виразу з пояснюючим текстом за допомогою функції print, використовуючи як імена, так і безпосередньо вирази в якості параметрів функції.

Рекомендовані ресурси

Офіційний сайт Python

<https://python.org>

Документація з Python

<https://docs.python.org/3/>

Статті у Вікіпедії про ключові поняття, розглянуті на цьому уроці

<https://uk.wikipedia.org/wiki/Python>

https://uk.wikipedia.org/wiki/Мова_програмування

<https://uk.wikipedia.org/wiki/Інтерпретатор>

https://uk.wikipedia.org/wiki/Інтегроване_середовище_розробки