

Python 01



Ce este programarea?

- Programarea este procesul prin care oferim unei platforme (PC, Sistem de operare, Lift, etc) instructiuni specifice.
- Aceste instructiuni se pot face in diferite limbaje de programare, la fel cum noi putem comunica cu diferiti oameni in limbi diferite.
- Putem considera programarea ca fiind modul prin care comunicam cu un computer.
- Gruparea si ordinea de executie a instructiunilor pentru rezolvarea unei probleme se numeste algoritm.

Exemplu Algoritm

- Trebuie sa aranjam instrucțiunile astfel încât ordinea să fie una logică.

- merg către magazin
- plătesc pâinea
- ajung acasă
- merg către casă
- iau banii necesari
- ajung și cer pâinea
- iau pâinea

Rezolvare :



Compiler vs Interpreter

- **Compilerul** scanează și analizează tot codul sursă, apoi îl transformă pe tot în cod mașină sub forma unui program executabil. **Erorile programului sunt afișate la final**, depanarea fiind puțin mai dificilă.
Exemple: Java
- **Interpreterul** transformă în cod mașină linie cu linie codul, nefiind necesar un fișier executabil generat la final. **La prima eroare analiza se oprește**, fiind astfel mai ușor de depanat programele.
Exemple: Python

Python



Detalii oficiale:

<https://www.python.org/>
<https://pypi.org/>
<https://www.google.com/>



De ce Python?

- Codul este mai ușor de citit, înțeles, si învățat comparat cu multe alte limbaje
- Număr imens de librării, suport online excepțional și comunități active
- Simplitatea este cel mai mare avantaj



Unde este aplicat Python?

Application Development

Scripting

Automation

DevOps

Web Development

Artificial Intelligence

Hacking (White Hat = 👍 Black Hat = 🚫)

Robotics

Command line

- Command line, sau CLI (command line interface) este unul din mecanismele principale de interacțiune cu un computer/sistem de operare/program.
Windows - Command prompt
Mac - Terminal
- GUI vs CLI în programare

Installing Python

- Download link:

Google download python:

[Download Python](#)

- Check:

Add Python to PATH

Disable path length limit

Hardcore Hello World

- Check python version in CLI:

ATENȚIE! - V uppercase: `python -V` sau `python3 -V` sau `py -V`

- First script:

WINDOWS/MAC:

Comanda: `notepad numeleTau.py` `touch numeleTau.py`

Confirmam "Yes", pentru că vrem să salvăm noul fișier.py `open -e numeleTau.py`

În acest file, scriem:

```
print("hello itschool")
```

```
print()
```

```
print("hello python")
```

Ctrl + S pentru a salva, și putem închide fișierul

- Execution

Executăm scriptul python din command line:

`python scriptulTau.py` (sau `python3` sau `py`)

- Celebration



IDE - Integrated Development Environment

Aplicație în care noi putem interacționa cu fișiere în care avem cod

Acele fișiere pot fi executate direct din IDE, în funcție de limbajul de programare folosit

Locul în care putem centraliza mai multe unelte care ne pot ajuta în procesul de development

Aici vorbim de versionare (GIT), remote deployment, database tools

Debugging/Testing

Mediu în care putem afla informații extra despre programul care rulează

Platforma în care testarea unitară a codului (Unit Testing) poate fi configurată detaliat

Custom Layout/Editor/Code Formatting/etc

Majoritatea IDE-urilor ne permit foarte multe schimbări în layout

Sunt multe IDE-uri, unele comune, altele specifice pentru un limbaj

PyCharm Community Edition

[Download PyCharm: Python IDE for Professional Developers by JetBrains](#)

print function



- `print()` este o funcție Python care afișează informațiile dintre paranteze
- În mod normal, `print()` afișează în CLI (fie `command prompt`, `terminal` sau în terminalul din IDE)

Variable

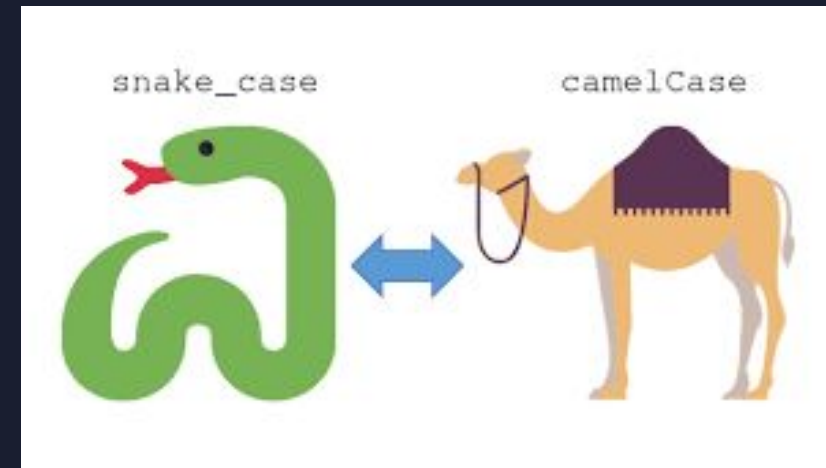
- Variabilele sunt un container pentru valori
- Python este un limbaj de programare dynamically typed, înseamnă că nu trebuie să menționăm ce tip de variabilă este când o definim
- pentru a crea o variabilă avem nevoie de 3 elemente
 - label
 - operator de alocare (assignment operator)
 - valoare
- Numele variabilelor pot conține doar caractere alfanumerice și underscore (A-z, 0-9, și _)

Variable

- Variabilele sunt case-sensitive, adica conteaza cum sunt denumite.
- Pentru formatul denumirilor in python o sa folosim snake_case

```
my_age = 20  
my_Age = 25  
MY_age = 30
```

Acestea sunt variabile diferite.



Variable

- Asignarea multiplelor valori la multiple variabile
- Asignarea unei singure valori la multiple variabile

`x, y, z = "Volvo", "BMW", "Audi"`

`x = y = z = "Audi"`

Exercitii

- Creeaza o variabila numita **car_name** si asigneaza valoarea **Volvo**
 - ? = “?”
- Creeaza o variabila numita **x** si asigneaza valoarea **50**
 - ? = ?
- Corecteaza numele variabilelor
 - 2first_name = “John”
 - myName = “Razvan”
 - email+_address = “test@itschool.ro”
- Creeaza variabilele **a, b, c** cu aceeasi valoare **“IT School”**

Next session

1 Simple data types (strings, boolean, numbers)

2 Operators

Thank you!