INFORMATICĂ · PYTHON · CLASA A VII-A · MEDIUL DE PROGRAMARE · LECȚIA 2

Structura unui program, comentarii și erori

Durată: 50 min Nivel: gimnaziu · Python Înainte: lecțiile py7g-01

CE VEI ȘTI DUPĂ LECȚIE

- Știi că un program este o secvență de instrucțiuni executate de sus în jos.
- Scrii comentarii cu # și le folosești ca să explici sau să oprești un rând.
- Deosebești cele trei feluri de erori: de sintaxă, de execuție și de logică.

1. Un program = instrucțiuni în ordine

Un program Python este o **secvență de instrucțiuni**. Python le citește **de sus în jos**, câte una, și nu sare peste nimic.

```
print("1. Pregătesc ingredientele")
print("2. Amestec")
print("3. Gust!")
```

```
1. Pregătesc ingredientele
2. Amestec
3. Gust!
```

Ordinea contează. Dacă schimbi rândurile între ele, schimbi și ce se afișează.

2. Comentariile

Un **comentariu** începe cu `#`. Python îl **ignoră**; este doar o notiță pentru oameni.

```
# Programul meu de test
print("Salut!")   # acest rând afișează un salut
# print("Pa!")
```

```
Salut!
```

Cu `#` poți face două lucruri:

- **explici** ce face programul (util pentru tine și pentru profesor);
- **oprești temporar** un rând: pui `#` în fața lui și el nu mai rulează (ca la ultimul `print` de mai sus).

SFAT

Începe fiecare program cu un comentariu care spune ce face. Peste o lună vei fi bucuros că l-ai scris.

3. Indentarea: nu pune spații la întâmplare

În Python, **spațiile de la începutul rândului** au sens. Deocamdată, **toate rândurile încep chiar din margine**. Mai târziu vom folosi spațiile ca să arătăm ce aparține unei decizii sau unei repetiții.

```
print("a")
  print("b")
```

```
IndentationError: unexpected indent
```

Traducere: „ai pus spații unde nu trebuia". Șterge spațiile din fața lui `print`.

4. Trei feluri de erori

Fel de eroare	Când apare	Exemplu
de sintaxă	înainte să ruleze programul; ai scris greșit o instrucțiune	`print("Salut)`
de execuție	programul pornește, dar se oprește la un rând	`print(10 / 0)`
de logică	programul merge până la capăt, dar **rezultatul e greșit**	calcul greșit

Eroare de execuție: nu poți împărți la zero.

```
print("Încep")
print(10 / 0)
print("Termin")
```

```
Încep
ZeroDivisionError: division by zero
```

Se afișează „Încep", apoi programul se oprește. Ultimul `print` nu mai rulează. Mesajul îți spune și numărul rândului cu problema.

Eroare de logică: este cea mai greu de găsit, pentru că Python nu se plânge de nimic!

```
# perimetrul unui dreptunghi cu laturile 3 și 4
print(3 + 4 + 3 + 3)
```

```
13
```

Programul rulează fără erori, dar perimetrul este de fapt `3 + 4 + 3 + 4 = 14`. Doar **tu** poți observa greșeala, dacă îți verifici rezultatul cu un exemplu cunoscut.

DE ȘTIUT

Cum depanezi: 1) citește ultimul rând al mesajului de eroare; 2) uită-te la rândul indicat; 3) verifică literele, parantezele, ghilimelele; 4) rulează din nou. Dacă programul nu dă eroare, dar rezultatul e greșit, testează-l cu un caz la care știi răspunsul.

5. Verifică-te

1. Ce afișează programul?

```
print("A")
# print("B")
print("C")
```

2. Ce fel de eroare este: a) ai scris `print("x)` (fără ghilimelele de la sfârșit); b) programul împarte la 0; c) programul afișează 5 în loc de 6?
3. Ce face `#` la începutul unui rând?
4. De ce programul următor dă eroare?

```
print("Bună")
   print("ziua")
```

Răspunsuri

1. Afișează `A`, apoi `C`. Rândul cu `B` este comentariu și este ignorat.
2. a. de sintaxă (`SyntaxError`); b) de execuție; c) de logică.
3. Transformă rândul (sau restul rândului) în comentariu, ignorat de Python.
4. Al doilea rând are spații la început, deci apare `IndentationError: unexpected indent`.