

Приказ излаза са функцијом print

Функција `print` се користи за приказ излаза у Пајтон програмима на екрану рачунара. Функција је део претходно дефинисаног и написаног кода који изводи одређену операцију. Пајтон има велики број предефинисаних функција које изводе различите операције. Када се функција стартује, каже се да се функција позива. Део кода функције `print` унутар заграде се назива аргумент функције. А то је заправо податак који треба да се прикаже на екрану као резултат позива функције `print`. Апострофи у примеру само дају почетак и крај текста који треба да се прикаже на екрану.

Пример 01: Испис откуцаног текста у интерпретеру (интерактивном моду)

```
>>> print('Zdravo, svete!')
Zdravo, svete!
>>>
```

Пример 02: Испис откуцаних линија текста у скрипт моду

```
print('Ovo je veoma')
print('veoma')
print('veoma')
print('kratak tekst.')
```

После куцања претходног кода, клик на опцију Debug па на Start Without debugging. Тиме се стартује превођење целог откуцаног кода (програма) у машински језик и предаје се процесору на извршење. Искази (statements) овог програма су се извршавали по редоследу по којем су се у куцали у коду, од линије 1 до линије 4.

Основе рада са стринговима и стринг литералима

Подаци: `'Ovo je veoma'`, `'veoma'`, `'kratak tekst.'` су низови карактера (знакова) и у програмирању низови знакова се називају стрингови.

Када се стринг појави као део кода, он се назива стринг литерал.

У Пајтону, стринг литерали се могу појавити између апострофа или између наводника.

Ако је потребно да стринг литерал садржи апостроф као део стринга, онда се стринг затвара са наводницима.

Ако је потребно да стринг литерал садржи наводнике као део стринга, онда се стринг затвара са апострофима.

Пајтон такође допушта да се стринг литерал затвори унутар троструких наводника или апострофа.

На овај начин могу се користити и наводници и апострофи у стрингу истовремено.

Такође, троструки наводници или апострофи се користе када је потребно приказати стринг литерал у више линија.

Пример 03: Приказ апострофа и наводника као део стринга

```
print('Rekao sam mu: "Nestle" je dobra cokolada.')
print("A on meni: Znas, 'decko', u pravu si.")
Излаз:
```

Rekao sam mu: "Nestle" je dobra cokolada.

A on meni: Znas, 'decko', u pravu si.

Пример 04: Приказ апострофа и наводника истовремено као део стринга

```
print(''Rekao sam mu: 'Decko',"Nesle" je dobra cokolada.'')
Излаз:
```

Rekao sam mu: 'Decko',"Nesle" je dobra cokolada.

Пример 05: Приказ стринга у више линија на излазу

```
print(''Jedan
Dva
Tri''')
```

Излаз:

Jedan

Dva

Tri

Коментари

Сваки коментар започиње симболом #, који се назива знаком бројева. Било шта после тог знака до краја исте линије кода представља коментар. Када напише компјутерски програм, своје идеје програмер преточи у велики број линија кода. Програмерске идеје често нису очигледне било коме другоме сем самом аутору. Дешава се да аутор кода после неког времена се врати написаном коду али ни њему самом више није лако да одреди смисао његових идеја. Једини начин да се избегну овакви проблеми је писање коментара. Коментари у коду помажу читању кода, олакшавају праћење развоја програмерских идеја и дају смисао и иначе тешко разумљивим линијама кода програмских језика.

Пример 01: Употреба коментара у коду

```
print('Naslov: Novi kod') #komentar zauzima deo linije koda
#komentar zauzima celu liniju bez koda
```

Излаз:

```
Naslov: Novi kod
```

Промењиве

Програми обично смештају податке у меморију рачунара и изводе операције над тим подацима.

Програми користе промењиве за приступ и манипулацију подацима смештеним у меморији.

Промењива је име које представља вредност у компјутерској меморији.

Каже се да промењива упућује (reference) на вредност.

Идентификатор и оператор доделе

`a = 23` Промењива **a** указује на меморијску локацију у којој се налази вредност **23**.

То је постигнуто помоћу оператора доделе (=, знак једнакости) (assignment operator).

Промењиве се представљају помоћу идентификатора.

```
a = 23
```

```
b = 11
```

```
name = "GOOD"
```

Промењива **a** указује на меморијску локацију у којој се налази вредност **23**, промењива **b** указује на локацију са вредности **11**, промењива **name** указује на локацију са вредности **GOOD**.

Свака промењива може да промени вредност током рада програма.

Промењива садржи њој прво додељену вредност све док се промењивој не додели нека друга вредност. Прво додељена вредност више није у употреби па Пајтон интерпретер аутоматски је брише из меморије преко процеса који се зове сакупљање ђубрета (*garbage collection*), чиме се ослобађа меморијска локација за даље коришћење.

Када се промењивој додељује вредност коју већ има нека друга промењива, интерпретер ће упутити да обе промењиве показују на исту меморијску локацију са жељеном вредности.

На овај начин се штеди употреба меморијских локација.

Провера указивања на исту локацију

Постоје два начина за проверу да ли промењиве указују на исту меморијску локацију:

Први је коришћењем инструкције `id(a)` унутар функције `print`, чиме се на екрану исписује адреса меморијске локације на коју упућује промењива **a**.

Други је коришћењем логичког оператора `is`.

Оператор упоређује адресе меморијских локација на које упућују промењиве **a** и **b** и ако су адресе идентичне, даје као резултат тачно (**True**) а у супротном нетачно (**False**).

Правила давања имена промењивима(избор идентификатора)

- не може се користити службена реч
- у имену не може постојати празно место
- први карактер може бити било које велико или мало слово или доња линија (`_`) (*underscore*)
- следећи карактери у имену могу бити слова, цифре или доње линије
- постоји разлика између малих и великих слова (*ja* није иста промењива као *Ja* или *jA*)
- име промењиве би требало да има описно значење о сврси промењиве (*tezina_u_uncama*, *baza_podataka_ucenici*, *tabela_repertoar_bioskopa*)

Задатак 014: Рад са стринг литералима

1. а) Отворити нови Пајтон пројекат и назвати га `konan_inventar`. Како се зове фајл (сорт код)?
 б) Додати следећу линију у скрипти: `print("Imam sandale")`. Шта је то `print` ?
 в) Кликнути на `Debug-> Start Without Debugging`. Шта се десило ?
 г) У коду отворити нову линију кода пре већ попуњене и унети: `print('Zovem se Konan.')`. Шта ће бити сада написано у конзоли ?
 д) Има ли разлике између стрингова у коду ? Зашто ?
2. Следећа линија скрипте треба да изгледа овако:
`Bravo Konan.`
 Написати код да би се то и приказало. Преписати код у свеску.
3. Потребно је да се садржај коментара из кода појави на излазу. Како то урадити ?
4. Коришћењем троструких наводника приказати следећи стринг:
`nivo 1`
`nivo 0`
`nivo -1`

Задатак 015: Писање коментара

1. Нека се у скрипти налази следећи код:

```
print()
print("Moja omiljena knjiga je:")
print('')
print("Gospodar prstenova")
```

 Убацити у сваку постојећу линију кода коментар
2. Написати мишљење о следећем делу кода који се налази на самом почетку скрипте:

```
#+-----+
#| Naziv programa:  Interaktivni_potez      |
#| Autor:          Marko Markovic          |
#| Namena:         Igranje saha protiv kompjutera |
#| Vlasnik:        autor                    |
#| Datum izrade:   20.08.2020.              |
#+-----+
```

Која је намена датог коментара ? Зашто је стављен на почетак кода ? Које је намењен ? Има ли аутор кода корист од оваквог коментара и зашто ?

Задатак 016: Промењиве

1. Којим од следећих идентификатора су неправилно додељена имена и зашто:

units_per_day	print
dayOfWeek	Print
3dGraph	xxx
June1997	ja*ti
Mixture#3	danas sutra
2. Без куцања кода одговорити шта ће се појавити на излазу:


```
a = 20
b = 40
c = 30
x, y, z = a, b, c
print(x, y, z)
```
3. Куцањем следећег кода проверити излаз:


```
a = 20
b = 40
c = 30
a = b = c
print(a, b, c)
```

 објаснити резултат излаза.

Прожебати следеће задатке:

1. Написати скрипт који даје на излазу следеће:

Bioskop	Film	Datum	Vreme
Dom sindikata	Ratovi zvezda	20.05.2018.	14:00
Sinapleks	Park iz doba jure	03.10.2018.	20:00
2. Коришћењем знака '@' нацртати квадрат, димензија 4x4.
3. Нека x и y добијају наизменично вредности од 1 до 6, тако што x добија само парне вредности а у само непарне вредности. Написати скрипту која на излазу приказује како се x и y мењају.
4. Нека су почетни подаци за промењиву a највећи двоцифрени паран број а за промењиву b стринг "x". Разменити вредности промењивих a и b коришћењем помоћне промењиве тако да промењиве a и b увек указују на почетне податке.