09. FOR CIKLUSOK (FOR –RANGE)

A programozásban nagyon sokszor előfordul, hogy egy műveletet, vagy műveletsort ismételni kell. A programrészek ismétlését **ciklusok** segítségével oldjuk meg. A ciklusoknak több fajtája van. Mi a legalapvetőbbeket nézzük meg. Ezek a **számláló és a feltételes ciklus.**

Számláló ciklus

Ha a program írásakor el tudjuk dönteni, hogy hányszor szükséges lefutnia a programrészletnek, akkor számláló ciklust alkalmazunk. Például, ha egy karaktert, tízszer ki szeretnék íratni a képernyőre, akkor a számláló ciklust alkalmazunk.

- A megvalósításhoz szükségünk lesz egy úgynevezett **ciklusváltozó**ra.
- Ennek a típusának **megszámlálható**nak kell lennie!
- Programozói szokás, ezeket **a változókat i-nek és j-nek hívn**i.
- Ennek a változónak kell adni egy **kiinduló érték**et.
- A ciklus addig ismétel, amíg el nem éri a **végérték**et.
- Van egy úgynevezett **növekmény** is, amely meghatározza, hogy mekkora léptekkel halad a ciklus.
- Nagyon fontos a ciklus tagolása (beljebb kezéés).

A ciklus **mondatszerű leírás**a a következő:

ciklus i:=1-tól 10-ig 1-esével — i ➔a ciklusváltozó
Ki: „X” — 1➔a ciklusváltozó
ciklus vége — 10➔végérték
Ki: „Ciklus utáni utasítás.” — 1-esével➔növekmény

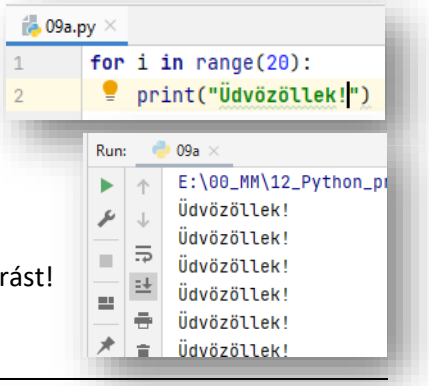
A Python programozáskor a „**for i in range(0,10,1)**” utasításra van szükségünk a ciklus megadásához.

- A for utasítás után megadjuk a ciklusváltozó nevét, majd az in szócska után a range utasítás zárójeles részében először a kezdő, majd a végértéket, végül a növekményt adjuk meg.
- Ha a növekmény 1, akkor elhagyhatjuk azt, elég csak az első két számot megadni!
- Ha a kezdőérték 0, akkor azt sem kell megadni, így csak a végértéket írjuk be a zárójelbe.
- Ha a range utasításnak egy értéke van, az mindig a végértéknél egyel nagyobb szám. Tehát ha valamit szeretnénk 10-szer lefuttatni, akkor 11-et aduk meg.
- A ciklus paramétereinek megadása után kettőspontot kell tenni, mert vezérlési szerkezet.
- A ciklus végét külön nem jelezzük, hanem a behúzást szüntetjük meg.

Megjegyzés	Mondatszerű leírás	Python
Egyparaméteres, kezdőérték: 0 végérték: 100 növekmény: 1	ciklus i:=0-től 100-ig <ciklusmag> ciklus vége	for i in range(101) <ciklusmag>
kétparaméteres, kezdőérték: 0 végérték: 100 növekmény: 1	ciklus i:=0-től 100-ig <ciklusmag> ciklus vége	for i in range(0,101) <ciklusmag>
háromparaméteres kezdőérték: 0 végérték: 100 növekmény: 10	ciklus i:=0-től 100-ig 10-esével <ciklusmag> ciklus vége	for i in range(0,101,10) <ciklusmag>

(09a.py)

A legegyszerűbb módja a ciklus kipróbálásának az az, hogy írassunk ki a képernyőre valamilyen szöveget n-szer!
Ebben a konkrét példában az „Üdvözöllek!” szöveget írassuk ki 20-szor!

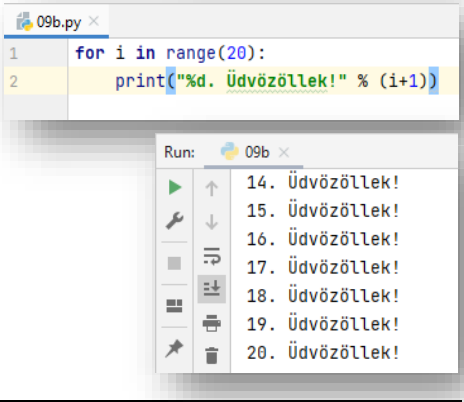


- A for parancs után megadjuk az „i” –t, mint változót, ami 0 –ról indul.
- A range(20) , az első sor végén megadja, hogy 20-szor hajtsa végre a kiírást!
- A print utasításban megadjuk a kiírandó szöveget!

(09b.py)

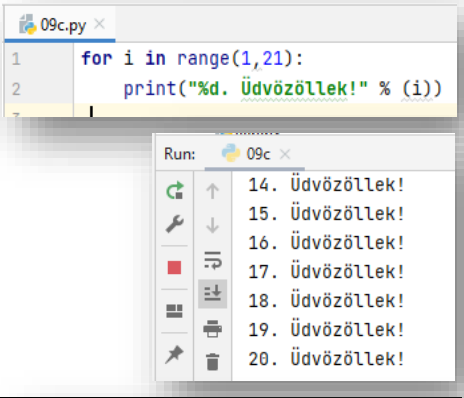
Módosítsuk az előző programot úgy, hogy a sor elején jelenjen meg, hogy éppen hányadszorra írta ki! Hányadik sorban vagyunk!

A szokott módon adjuk meg a formátumot, csak arra kell figyelni, hogy az „i” értéket egyel növelni kell!



(09c.py)

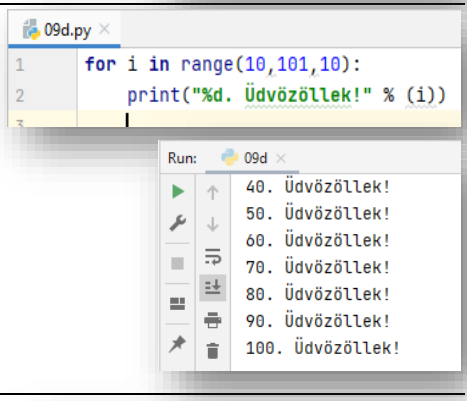
Ha a ciklusban kezdőértéket állítunk be, akkor nem kell egyesével növelni az „i”-t. Ilyenkor a ciklus kétparaméteres alakját használjuk. Mivel most 1-től szeretnének 20-ig kiíratni a számokat, ezért a végétéket 21-re kell állítani.



(09d.py)

Módosítsuk a programot úgy, hogy a program 10-től 100-ig tízesével írja ki a sorszámokat!

Mivel itt a növekményt is megadjuk, ezért háromparaméteres formában írjuk be a range parancsba, az adatokat.



(09e.py)

Végül nézzünk egy olyan példát, amit egyparaméteresen oldunk meg.
Írjuk ki 10-től ötösével 100-ig a sorszámokat!
Ha az „i” értékét eredetileg 1-esével emeljük, de bent a ciklusban 5-el megszorozzuk, akkor 5-ösével fog lépkedni és azt is írja ki.
Ha pedig hozzá adunk mindig 10-et, akkor a kiinduló szám az a 10-lesz.
Nézzük meg a példaprogramban, és értelmezzük, hogy hogyan működik!