**Stringuri (șiruri de caractere)**

Se specifică între ghilimele sau apostrofuri (same)

Lungimea unui string este dată de funcția len(string)

Caracterele sunt indexate de la 0. Accesarea unui caracter se face cu s[i] (dacă s este stringul).

!!! Accesarea unui caracter se poate face DOAR în evaluare, adică NU îi putem atribui unui s[i]

o valoare !!!

Pentru a modifica un caracter într-un string, acesta trebuie re-compus din părți.

Din fericire, există câteva forme de indexare care permit extragerea foarte elegantă de subsecvențe:

\* string[st:dr+1] ne ia secvența dintre indicii st și dr inclusiv.

Ex:

s="teleferic"

s[2:5] = "lef"

\* string [:dr] ne ia primele dr caractere

\* string [st:] ne ia de la indicele st la final

\* string [:-nr] ne ia stringul FĂRĂ ultimele 4 caractere

\* string [-nr:] ne ULTIMELE 4 caractere

Deci, dacă ne trebuie ca într-un string să ÎNLOCUIM caracterul de la indicele i cu '@' facem astfel:

s = s[:i]+'@'+s[i+1:]

Separarea cuvintelor dintr-un string în funcție de un separator:

string.split('separator') - asta ne dă un șir (listă) de stringuri

Dacă separatorii apar de mai multe ori, multe stringuri din listă vor fi vide.

string.split() - la fel, doar că separă în funcție de spații și tab-uri și NU mai bagă stringuri vide

string.find('secvență') - caută prima apariție pentru 'secvență' în stringul dat, întorcând indicele la care apare.

Dacă NU apare - întoarce -1.

La fel ca în C++, ca să vedem dacă un caracter este vocală (de exemplu)

putem folosi

if "aeiou".find(caracter)!=-1

Convertirea unui string (sau char) la litere mari:

string.upper()

Analog, la litere mici:

string.lower()