



STRING (NIZ ZNAKOVA)

STRING

- **stringovi su tip podataka u Pythonu koji se bavi tekстом odnosno omogućuje nam rad s nizom znakova**
- kreira se stavljanjem teksta unutra navodnika (jednostrukih ili dvostrukih)
- trostruki navodnici (omogućavaju protezanje niza znakova kroz nekoliko linija)
- svi uneseni razmaci uključeni su u niz znakova

OPERACIJE + | *

- operatori + i * mogu se koristiti sa stringovima
- operator + služi za kombiniranje (spajanje) dva stringa – concatenation
- operator * ponavlja zadani string određen broj puta

PRIMJER

```
>>> a = "informatika"  
>>> b = "predmet"  
>>> a+" "+b  
'informatika predmet'  
>>> 3*b  
'predmetpredmetpredmet'
```

DOHVAĆANJE VRIJEDNOSTI UNUTAR STRINGA

- svaki znak u stringu ima svoju poziciju – indeks –
- prvi znak ima indeks 0, drugi 1, treći 2 ...
- npr. `a[0]` – prvo slovo u varijabli `a`, `b[5]` – šesto slovo u varijabli `b`
- ako želimo dohvatiti znakove od, na primjer, 5. do 7. indeksa koristimo npr. `a[5:8]` (**NAPOMENA: 8. indeks nije uključen**)
- ako želimo dohvatiti zadnje znakove koristimo negativne brojeve npr. `a[-1]` je zadnji znak
- **ako dohvaćamo znak kojeg nema javit će nam grešku**

DOHVAĆANJE VRIJEDNOSTI UNUTAR STRINGA

- možemo ostaviti praznu početnu ili krajnju poziciju
- npr. `a[:4]` vraća nam prva četiri slova iz stringa `a` (znakove s indeksa od 0 do 3)
- npr. `a[5:]` vraća sva slova počevši od indeksa 5 pa do kraja stringa
- ako koristimo negativne indekse dobivamo krajnja slova stringa npr. `a[-2:]` će nam dati zadnja dva znaka stringa `a`
- opcionalni je treći argument, kojim definiramo korak pomaka npr. `a[1:7:2]` „korača” kroz string `s` korakom 2, i na taj način se selektiraju slova sa indeksom 1, 3 i 5 (ali ne i 7 jer nije uključen)
- **koristan je korak -1, kojim se kroz string prolazi natraške, pa se slova dobivaju u obrnutom redoslijedu**

NAPOMENA

- **ne možemo lako promijeniti znakove u stringu**
- pretpostavimo da imamo string `a` i da želimo promijeniti slovo sa indeksom 5 tako da bude 'X'
- **najlakše bi bilo: `a[5]='X'` - to ne radi**
- ne možemo "lako" modificirati pojedine dijelove stringa
- **mogli bi to na ovaj način: `a=a[:5]+'X'+a[6:]`**

PRIMJERI

- a='stringovi'

Kod	Rezultat	Opis
a[1:5]	trin	Tri znaka, slova sa indeksom 1, 2, 3, 4
a[:4]	stri	Prva četiri slova (s indeksom 0, 1, 2, 3)
a[5:]	govi	Slova od indeksa 5 do kraja
a[-3:]	ovi	Posljednja tri slova
a[:]	stringovi	Čitav string
a[1:7:2]	tig	Slova od indeksa 1 do 6, svako drugo (dodatno)
a[::-1]	ivognirts	Negativan korak unazad (unazad ispisani string)

IN i NOT IN u stringu

- vraćaju vrijednosti istina ili laž odnosno TRUE ili FALSE
- IN - provjerava nalazi li se dani znak u nizu
 - npr. 'd' in a – provjerava je li slovo d u stringu a
- NOT IN - provjerava da li se neki znak NE nalazi u nizu
 - npr. 'K' not in a – provjerava da li slovo k nije u stringu a

PRIMJERI

```
>>> a = "stringovi"  
>>> "stri" in a  
True  
>>> "e" in a  
False  
>>> "e" not in a  
True
```

OSNOVNE FUNKCIJE ZA STRINGOVE

- `len()` - vraća duljinu niza znakova
- `min()` - vraća znak s najmanjom kodnom vrijednosti
- `max()` - vraća znak s najvećom kodnom vrijednosti
- `ord()` – vraća dekadski kod pojedinog znaka
- `chr()` – vraća znak odgovarajućeg dekadskog koda
- `str()` - prima objekt i vraća niz znakova, odnosno pretvara nešto u string

PRIMJERI

```
>>> a = "string"
>>> len(a)
6
>>> max(a)
't'
>>> min(a)
'g'
>>> chr(56)
'8'
>>> chr(115)
's'
>>> ord("A")
65
>>> chr(65)
'A'
```

OSNOVNE FUNKCIJE ZA STRINGOVE

- `.count(x, poc, kraj)` - broji koliko puta se određeni znak `x` pojavljuje u zadanom nizu počevši od indeksa `poc` do indeksa `kraj`
- `.replace(x, y, koliko)` - zamjenjuje sve znakove `x` sa zadanim znakom `y` u nekom nizu, a ako navedemo koliko onda samo određen broj znakova `x` zamjenjuje
- `.find(x, poc, kraj)` – vraća indeks prvog pojavljivanja znaka `x` u zadanom stringu od indeksa `poc` do indeksa `kraj`
- `.rfind(x)` – vraća indeks zadnje pojave znaka `x` u zadanom stringu
- `.index(x)` - vraća indeks pozicije prvog pojavljivanja `x` u stringu

PRIMJERI

```
>>> a = "banana"
>>> a.count("a", 2)
2
>>> a.count("a")
3
>>> a.count("a", 2, 4)
1
>>> a.replace("a", "o")
'bonono'
>>> a.find("n")
2
```

OSNOVNE FUNKCIJE ZA STRINGOVE

- `.lower()` - pretvara cijeli niz znakova u mala slova
- `.upper()` - pretvara cijeli niz znakova u velika slova
- `.title()` – svaka riječ stringa počinje velikim slovom
- `.capitalize()` - u danom nizu prvo slovo stavlja u veliko slovo, a sva ostala slova u mala
- `.swapcase()` – sva velika slova u stringu mijenja s malim slovima i obrnuto mala s velikim slovima
- `.strip()` - miče sve praznine s desne i lijeve strane niza
- `.lstrip()` - miče sve praznine s lijeve strane niza
- `.rstrip()` - miče sve praznine s desne strane niza

OSNOVNE FUNKCIJE ZA STRINGOVE

- `.center(duljina, x)` – vraća string zadane duljine u kojem je zadan string centriran, a s lijeve i desne strane se nalazi potreban broj znakova `x`
- `.ljust(duljina, x)` – vraća string zadane duljine u kojem je zadan string lijevo poravnat, a s desne strane se nalazi potreban broj znakova `x`
- `.rjust(duljina, x)` – vraća string zadane duljine u kojem je zadan string desno poravnat, a s lijeve strane se nalazi potreban broj znakova `x`
- `.endswith(x, poc, kraj)` – vraća `True` ako string (ili podstring određen s `poc` i `kraj`) završava s `x`
- `.startswith(x, poc, kraj)` – vraća `True` ako string (ili podstring određen s `poc` i `kraj`) počinje s `x`

OSNOVNE FUNKCIJE ZA STRINGOVE

- `.isalnum()` - vraća `True` ako zadani string sadrži barem jedan znak ili i ako su svi znakovi alfanumarički (slova ili znamenke)
- `.isdigit()` - vraća `1` ako su svi znakovi u nizu znamenke
- `.isalpha()` - vraća `True` su svi znakovi u nizu slova
- `.isupper()` - vraća `True` su sva slova u nizu velika
- `.islower()` - vraća `True` su sva slova u nizu mala
- `.join(x)` – vraća string u kojem pojedine elemente stringa `x` poveže s zadanim stringom kao separatorom
- `.split(x)` – vraća listu s elementima koji nastaju tako da zadani string rastavimo s obzirom na podstring `x`
- `.format(argumenti)` – formatira znakovni niz u skladu s navedenim argumentima

PRIMJERI:

```
>>> a = "provjera"
>>> a.isdigit()
False
>>> a.isalpha()
True
>>> b = "1456"
>>> b.isdigit()
True
>>> b.isalnum()
True
>>> a.islower()
True
>>> a.isupper()
False
```

ZADACI ZA VJEŽBU

1. Napiši program koji će unijeti neku riječ i provjeriti počinje li zadana riječ slovom „a”.
2. Napiši program koji će unijeti neku riječ i provjeriti završava li zadana riječ s „ka”
3. Napiši program koji će unijeti neku riječ te ispisati novu riječ u kojoj su sva slova „a” zamijenjena slovom „o”.



ZADACI ZA VJEŽBU S STRINGOVIMA

ZADACI

1. Napišite program koji traži od korisnika da unese string. Program zatim treba da ispisati sljedeće:

- a) Ukupan broj znakova u stringu
- b) Sting ponovljen 5 puta
- c) Prvi znak u stringu
- d) Prva tri znaka iz stringa
- e) Zadnja tri znaka iz stringa
- f) Cijeli string unazad (palindrom)
- g) Deseti znak u stringu ako string ima dovoljnu dužinu, a ako ne treba ispisati poruku da „Nema toliko znakova”
- h) Novi string kod koga su odstranjeni prvi i zadnji znak
- i) Novi string sa svim velikim slovima
- j) Novi string u kome je svako a zamijenjeno sa o

RJEŠENJA 1. ZADATKA:

```
a = input("Unesi neki string:")
print(len(a))
print(5*(a+" "))
print(a[0])
print(a[0:3])
print(a[-3:])
print(a[::-1])
if len(a)>=10:
    print(a[9])
else:
    print("Nema 10 znakova")
h = a[1:len(a)-1]
print(h)
i = a.upper()
print(i)
j = a.replace("a", "o")
print(j)
```

ZADACI

2. Napišite program koji traži od korisnika da unese string i jedan znak te ispisuje da li se taj znak pojavljuje u zadanom stringu.
3. Napišite program koji traži od korisnika da unese string i jedan znak te da prebroji koliko se puta znak pojavljuje u zadanom stringu.
4. Napišite program koji traži od korisnika da unese string i jedan znak i da ispisuje indeks prvog pojavljivanja znaka u stringu. Ako se znak ne pojavljuje u stringu program treba ispisati odgovarajuću poruku.
5. Jedan jednostavan način da se procijeni broj riječi u stringu je da se prebroji broj razmaka (space) u stringu. Napišite program koji traži od korisnika da unese tekst, a onda mu vraća broj riječi u tom stringu.

RJEŠENJA 2. ZADATKA:

```
a = input("Unesi neki string:")
z = input("Unesi jedan znak")
if z in a:
    print("Znak se nalazi u zadanom stringu")
else:
    print("Nema tog znaka u zadanom stringu")
```

RJEŠENJA 3. ZADATKA:

```
a = input("Unesi neki string:")  
z = input("Unesi jedan znak")  
print(a.count(z))
```

RJEŠENJA 4. ZADATKA:

```
a = input("Unesi neki string:")
z = input("Unesi jedan znak")
if z in a:
    print(a.find(z))      #1. način
    print(a.index(z))    #2. način
else:
    print("Nema tog znaka u zadanom stringu")
```

RJEŠENJA 5. ZADATKA:

```
a = input("Unesi neki tekst:")  
print("Broj riječi u tekstu je", a.count(" ") + 1)
```

ZADACI

6. Ljudi u programiranju često zaborave zatvoriti zagradu u formulama. Napišite program koji traži od korisnika da unese formulu i ispisuje poruku da li formula ima dobar broj zagrada.
7. Napišite program koji traži od korisnika da unese neki tekst i ispisuje da li taj tekst sadrži samoglasnike ili ne.
8. Napišite program koji traži od korisnika da unese jednu riječ pa određuje je li unijeta riječ palindrom ili ne. (na primjer: potop).
9. U svim školama službene email adrese učenika i nastavnika završavaju sa @skole.hr. U učeničkoj referadi od učenika se traži da unesu svoju službenu adresu u računalo. Napišite program koji dobiva neku email adresu učenika i provjerava je li on unio službenu adresu ili nije.

RJEŠENJA 6. ZADATKA:

```
a = input("Unesi neku formulu:")
if a.count("(")==a.count(")"):
    print("Broj zagrada je dobar")
else:
    print("Broj zagrada nije dobar")
```

RJEŠENJA 7. ZADATKA:

```
a = input("Unesi neku formulu:")
if ("a" in a) or ("e" in a) or ("i" in a) or ("o" in a) or ("u" in a):
    print("Tekst sadrži samoglasnike")
else:
    print("Tekst nema samoglasnika")
```

RJEŠENJA 8. ZADATKA:

```
a = input("Unesi neku riječ:")
if a == a[::-1]:
    print("Riječ je palindrom")
else:
    print("Riječ nije palindrom")
```


RJEŠENJA 9. ZADATKA:

```
a = input("Unesi email adresu:")
if a.endswith("@skole.hr"):
    print("Službena adresa")
else:
    print("Mail adresa nije dobra")
```

ŠIFRIRANJE PORUKA

10. Jedan jednostavan način za šifriranje poruka je premještanje slova u stringu. Jedan način premještanja je da se najprije uzmu sva slova na parnim pozicijama, a zatim sva slova na neparnim pozicijama. (npr. TEKST: poruka, ISPIS: prkoua). Napišite program koji traži od korisnika da unese neku riječ duljine 6 znakova pa premještanjem kreira novi string i ispisuje šifriranu poruku.

11. Napravite program koji će dešifrirati poruku od šest znakova šifriranu premještanjem.

RJEŠENJA 10. ZADATKA:

```
a = input("Unesi riječ sa 6 znakova")  
b = a[0]+a[2]+a[4]+a[1]+a[3]+a[5]  
print(b)
```