

Оператори in и not in

За програм није добро да се прекине његово извршавање ако се покушало добити вредност у речнику помоћу кључа који не постоји.

Да би се то спречило, може се користити оператор in за испитивање да ли кључ уште постоји у речнику пре него се покуша добити вредност повезана са тим кључем.

Пример 100: претражити речник без подизања изузетака

```
telefonski_imenik = {"Marko":'345-9876', 'Ana':"555-6666", "Zoran":"111-0000",\
    "Dragan":"987-6543"}
if "Dragan" in telefonski_imenik:
    print(telefonski_imenik["Dragan"])
if "Dragan" not in telefonski_imenik:
    print("Кључ 'Dragan' не постоји у речнику.")
```

Додавање елемената у постојећи речник

Речници су промењиви објекти.

Може се увек додати нови пар кључ-вредност у речник употребом следеће синтаксе:

```
naziv_recnika[kljuc] = vrednost
```

где је naziv_recnika промењива која указује на речник, а kljuc је кључ из речника.

Ако kljuc већ постоји у речнику, вредност тог кључа постаје vrednost.

Ако kljuc не постоји у речнику, он ће се додати у речник заједно са vrednost која постаје његова придружена вредност у речнику.

Пример 101: додати елемент у речник

```
telefonski_imenik = {"Marko":'345-9876', 'Ana':"555-6666", "Zoran":"111-0000",\
    "Dragan":"987-6543"}
telefonski_imenik["Jovana"] = "012-3210"
print(telefonski_imenik)
telefonski_imenik ["Marko"] = '345-9876'
print(telefonski_imenik)
Дaje telefonski_imenik = {"Marko":'345-9876', 'Ana':"555-6666",
    "Zoran":"111-0000", "Dragan": "987-6543", "Jovana" : "012-3210"}
```

Овакав резултат значи да је немогуће дуплицирати кључ у речнику и да ако се додели вредност постојећем кључу, нова вредност замењује постојећу.

Брисање елемента из речника

Коришћењем исказа del може се обрисати пар кључ-вредност из речника.

```
del naziv_recnika[kljuc]
```

Ако се покуша обрисати непостојећи кључ из речника, појављује се KeyError грешка.