

Основи програмирања

Вежбе 5

Исидора Грујић
isidora@uni.kg.ac.rs

Лазар Илић
lazar@uni.kg.ac.rs

Филип Милић
cofi.milic@gmail.com

Катедра за електротехнику и рачунарство
Факултет инжењерских наука Универзитета у Крагујевцу



Крагујевац, 6. новембар 2024.



- 1 Нови тип објекта - Скупови
- 2 Додатак - Уграђене функције у Пајтону
- 3 Задаци



1 Нови тип објекта - Скупови

2 Додатак - Уграђене функције у Пајтону

3 Задаци



- сложен тип објекта, *променљив*



- сложен тип објекта, *променљив*
- дефинисање скупа:
 - $\{<\text{члан}>\}$
 - $\{<\text{члан1}>, <\text{члан2}>, <\text{члан3}>\}$



- сложен тип објекта, *променљив*
- дефинисање скупа:
 - $\{<\text{члан}>\}$
 - $\{<\text{члан1}>, <\text{члан2}>, <\text{члан3}>\}$
 - $\text{set}(<\text{други објекат}>)$



- сложен тип објекта, *променљив*
- дефинисање скупа:
 - $\{<\text{члан}>\}$
 - $\{<\text{члан1}>, <\text{члан2}>, <\text{члан3}>\}$
 - $\text{set}(<\text{други објекат}>)$
- Чланови скупа су уникатни, односно **нема понављања** чланова (дупликата) унутар скупа!



- сложен тип објекта, *променљив*
- дефинисање скупа:
 - $\{<\text{члан}>\}$
 - $\{<\text{члан1}>, <\text{члан2}>, <\text{члан3}>\}$
 - $\text{set}(<\text{други објекат}>)$
- Чланови скупа су уникатни, односно **нема понављања** чланова (дупликата) унутар скупа!
- Чланови скупа нису у реду, односно **не може** се користити индексирање/одсецање!



- сложен тип објекта, *променљив*
- дефинисање скупа:
 - $\{<\text{члан}>\}$
 - $\{<\text{члан1}>, <\text{члан2}>, <\text{члан3}>\}$
 - $\text{set}(<\text{други објект}>)$
- Чланови скупа су уникатни, односно **нема понављања** чланова (дупликата) унутар скупа!
- Чланови скупа нису у реду, односно **не може** се користити индексирање/одсецање!
- Подржани оператори над скуповима:
 - Унија: $<\text{скуп1}> \mid <\text{скуп2}>$
 - Пресек: $<\text{скуп1}> \& <\text{скуп2}>$
 - Разлика: $<\text{скуп1}> - <\text{скуп2}>$
 - Симетрична разлика: $<\text{скуп1}> \wedge <\text{скуп2}>$



Поређење - Низови, Скупови, Поворке

Низови у пајтону	Скупови у пајтону	Поворке у пајтону
<code>list = [члан1, члан2, члан3]</code>	<code>set = {члан1, члан2, члан3}</code>	<code>tuple = (члан1, члан2, члан3)</code>
подржавају индексирање / слајсовање	не подржавају индексирање / слајсовање	подржавају индексирање / слајсовање
има дупликата	нема дупликата	има дупликата
чланови низа се могу мењати (замена, додавање)	чланови скупа се не могу мењати, могуће је само додавање/уклањање чланова	поворке су непроменљив тип објекта

Пример 1

Разни примери са скуповима.

Пример 2 / Задатак 1

Написати функцију која за унуту ниску, враћа карактере који се понављају.



1 Нови тип објекта - Скупови

2 Додатак - Уграђене функције у Пајтону

3 Задаци



Уграђене функције потенцијално могу олакшати писање кода, као и повећати читљивост истог. Постоји доста таквих функција у Пајтону, ево пар примера:

- `min(x)` / `max(x)` - враћа најмањи/највећи члан итеративног објекта `x`.



Уграђене функције потенцијално могу олакшати писање кода, као и повећати читљивост истог. Постоји доста таквих функција у Пајтону, ево пар примера:

- **min(x) / max(x)** - враћа најмањи/највећи члан итеративног објекта **x**.
- **sum(x)** - враћа сумиране вредности чланова итеративног објекта **x**.



Уграђене функције потенцијално могу олакшати писање кода, као и повећати читљивост истог. Постоји доста таквих функција у Пајтону, ево пар примера:

- **min(x) / max(x)** - враћа најмањи/највећи члан итеративног објекта *x*.
- **sum(x)** - враћа сумиране вредности чланова итеративног објекта *x*.
- **sorted(x)** - враћа сортирани низ чланова итеративног објекта *x*.

Такође, и све класе (типови објеката) имају неке своје уграђене функције (*count*, *index*, ...).

Домаћи задатак: Самостално истражити / провежбати уграђене функције!
:)

Пример 3

Илустрација примене уграђених функција.



1 Нови тип објекта - Скупови

2 Додатак - Уграђене функције у Пајтону

3 Задаци



Задатке са петих вежби можете пронаћи на [страници предмета](#).

