CITY
UNIVERSITY OF LONDON
EST 1894

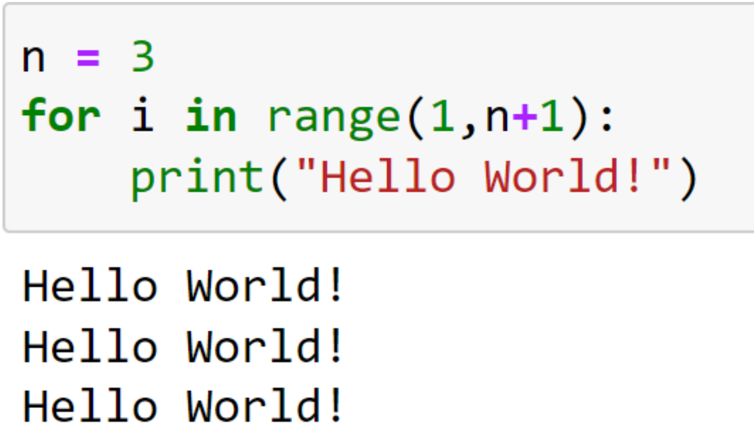
STEM Digital Academy

School of Science & Technology

www.city.ac.uk

Sets

Programming and Algorithms



```
n = 3
for i in range(1,n+1):
    print("Hello World!")
```

```
Hello World!
Hello World!
Hello World!
```

Lecture by

Dr Daniil Osudin

What will we Cover?

- Introduction to the set data structure
- Storing data in sets

What is a Set?

- Unordered sequence or collection of data items
- Immutable, so the values cannot be modified
- Does not contain repeated values or duplicates
- Data cannot be accessed by an index, but by a value instead

What is the Purpose of Sets?

Practical in situations, where storing of unique elements is required

Examples:

- Numbers in the phonebook (or in the list of contacts on the smartphone)
- Collection of usernames on a website

Set Data Type

- Sets are created using curly brackets `{ }`, as opposed to lists which are defined using square brackets `[ ]` around the elements
- Sets will only contain unique values

Example	Result
`my_set1 = {1, 2, 3, 1, 3, 3, 1}`	`{1, 2, 3}`

Repeated values are added only once in the set

Making a Set from a Sequence

- `set()` – is used to convert a sequence into a set
- Works on lists and tuples

Example	Result
`my_set2 = set([10, 20, 30, 30])`	`{10, 20, 30}`
`my_set3 = set(("a", "b", "b", "a"))`	`{"a", "b"}`

Making a Sequence from a Set

- `list(), tuple()` – used to convert a set into a list or tuple respectively

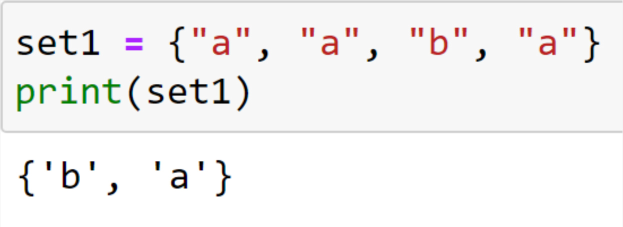
Example	Result

`my_set4 = {10, 20, 30, 10}`
`my_list1 = list(my_set4)`
	`[10, 20, 30]`

`my_set5 = {10, 20, 30, 10}`
`My_tuple1 = tuple(my_set5)`
	`(10, 20, 30)`

Examples

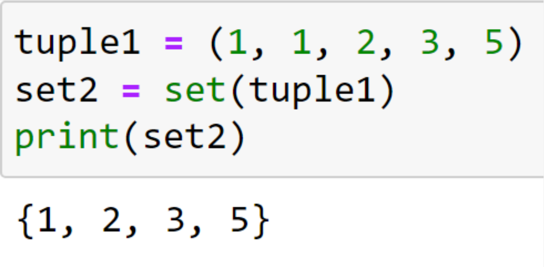
Creating a set



```
set1 = {"a", "a", "b", "a"}
print(set1)
```

```
{'b', 'a'}
```

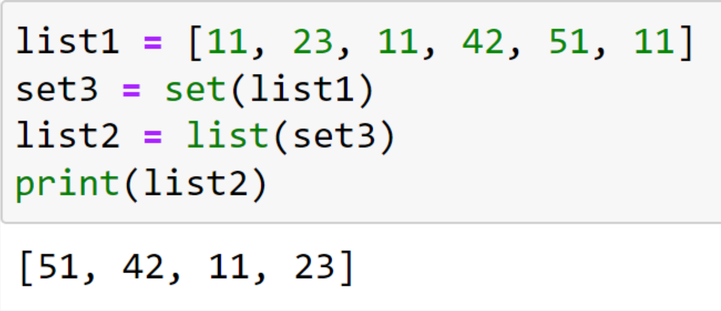
Tuple to a set



```
tuple1 = (1, 1, 2, 3, 5)
set2 = set(tuple1)
print(set2)
```

```
{1, 2, 3, 5}
```

Removing duplicates from a list

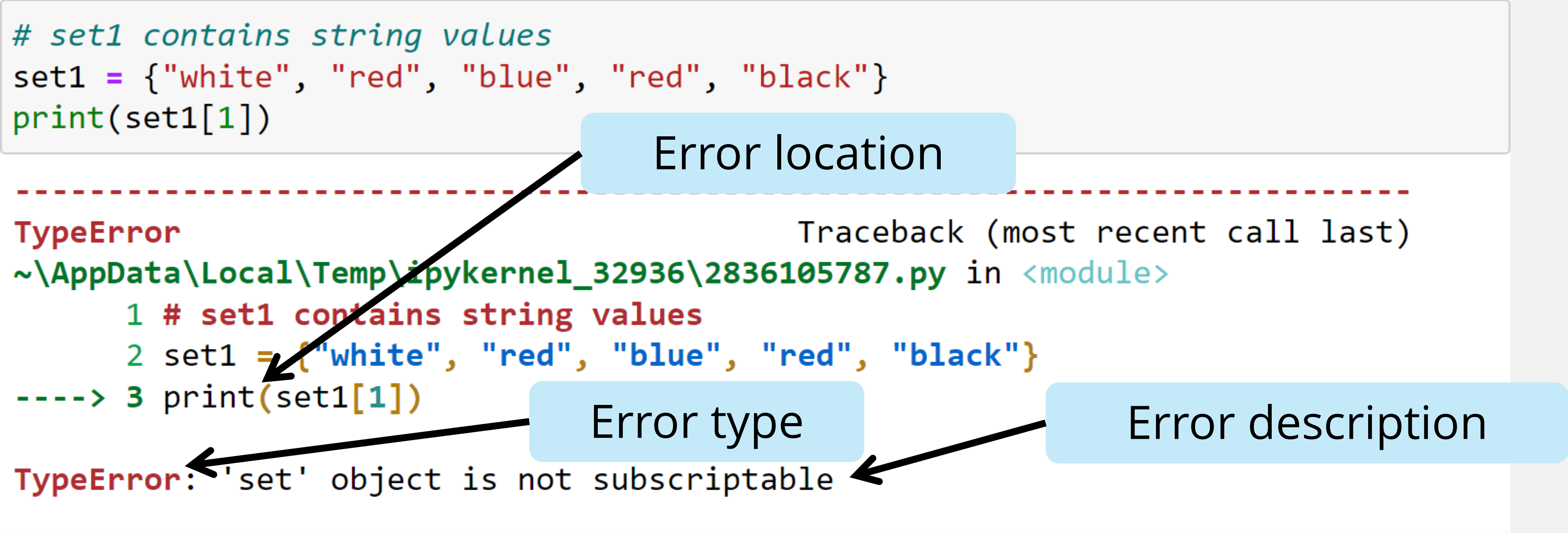


```
list1 = [11, 23, 11, 42, 51, 11]
set3 = set(list1)
list2 = list(set3)
print(list2)
```

```
[51, 42, 11, 23]
```

Common Errors in Python

It is not possible to access a set element using an index



```
# set1 contains string values
set1 = {"white", "red", "blue", "red", "black"}
print(set1[1])
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_32936\2836105787.py in <module>
      1 # set1 contains string values
      2 set1 = {"white", "red", "blue", "red", "black"}
----> 3 print(set1[1])

TypeError: 'set' object is not subscriptable
```

Try It Yourself

Write a program in Python environment that takes 10 inputs, stores them in a list and removes any duplicates.

Hint: use the input() function inside a FOR loop, which repeats 10 times

Append elements into a list, then convert to a set and convert back

CITY
UNIVERSITY OF LONDON
EST 1894