CITY
UNIVERSITY OF LONDON
EST 1894

STEM Digital Academy

School of Science & Technology

www.city.ac.uk

Built-in Functions for Dictionaries and Dictionary Methods

Programming and Algorithms

Lecture by

Dr Daniil Osudin

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

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

What will we Cover?

- Using the dictionary data structure
- Applying built-in functions to dictionaries
- Using dictionary methods

Built-in Functions for Dictionaries I

- `min(dict_name)` – used to find the smallest value in the dictionary
- `max(dict_name)` – used to find the largest value in the dictionary

Example	Result
`dict1 = {"a": 1, "b": 2, "c": 3}` `min(dict1)`	'a'
`dict2 = {"a": 1, "b": 2, "c": 3}` `max(dict2)`	'c'

Built-in Functions for Dictionaries II

- `len(dict_name)` – used to find the length (number of values) of a set

Example	Result
`dict3 = {"a": 1, "b": 2, "c": 3}` `len(dict3)`	3

Examples I

Minimum value

```
scores = {"OOP": 78, "AI": 65, "DA": 80, "Networks": 71, "Project": 90}
print(min(scores))
```

```
'AI'
```

Maximum value

```
scores = {"OOP": 78, "AI": 65, "DA": 80, "Networks": 71, "Project": 90}
print(max(scores))
```

```
'Project'
```

Examples II

Length of a dictionary

```
scores = {"OOP": 78, "AI": 65, "DA": 80, "Networks": 71, "Project": 90}
print(len(scores))
```

```
5
```

Converting into a List

- `values()` – used to extract a list of values from a dictionary
- `keys()` – used to extract a list of keys from a dictionary

Example	Result
`dict4 = {"a": 1, "b": 2, "c": 3}` `list(dict4.values())`	`[1, 2, 3]`
`dict5 = {"a": 1, "b": 2, "c": 3}` `list(dict5.keys())`	`['a', 'b', 'c']`

Itemising and Clearing a Dictionary

- `items()` – used to extract a list of key-value pairs from a dictionary
- `clear()` – used to remove all entries from a dictionary

Example	Result
`dict4 = {"a": 1, "b": 2, "c": 3}` `list(dict4.items())`	`[(1, 'a'), (2, 'b'), (3, 'c')]`
`dict5 = {"a": 1, "b": 2, "c": 3}` `dict5.clear()`	`None`

Examples III

Extracting a list of keys

```
scores = {"OOP": 78, "AI": 65, "DA": 80, "Networks": 71, "Project": 90}
print(scores.keys())
```

```
dict_keys(['OOP', 'AI', 'DA', 'Networks', 'Project'])
```

Extracting a list of value

```
scores = {"OOP": 78, "AI": 65, "DA": 80, "Networks": 71, "Project": 90}
print(scores.values())
```

```
dict_values([78, 65, 80, 71, 90])
```

Examples IV

Extracting a list of key-value pairs

```
scores = {"OOP": 78, "AI": 65, "DA": 80, "Networks": 71, "Project": 90}
print(scores.items())
```

```
dict_items([('OOP', 78), ('AI', 65), ('DA', 80), ('Networks', 71), ('Project', 90)])
```

Clearing a dictionary

```
scores = {"OOP": 78, "AI": 65, "DA": 80, "Networks": 71, "Project": 90}
print(scores.clear())
```

```
None
```

Try It Yourself

Enter and run the following statements in the python environment:

```
d = {"movie": "Forrest Gump", "artist": "Robin Wright", "year": 1994}
print(list(d.values()))
print(tuple(d.keys()))
print("number of items is: ", len(d))
```