YZV102E - Introduction to Programming for Data Science (Python)
Lab 2

06.03.2025

Res. Asst. Erhan Biçer (bicer21@itu.edu.tr)
Res. Asst. Uğur Önal (onalug@itu.edu.tr)
Res. Asst. Sümeyye Öztürk (ozturks20@itu.edu.tr)
Res. Asst. Barış Bilen (bilenb20@itu.edu.tr) Res. Asst. Püren TAP (tap23@itu.edu.tr)

1 Part 1: Lists

In this part, we will investigate the lists in Python.

1.1 Creation

In Python, we can create lists like in Code Snippet 1;

Code Snippet 1: List Creation

```
lst1 = []
lst2 = list()
lst3 = [1, 2, 3]
lst4 = ["itu", "1773"]
lst5 = ["itu", 1773]

```

lst1, lst2 are empty lists. You can create empty lists with this way. lst3, lst4 and lst5 are not an empty lists. lst3 consists of three integer values which are 1, 2, 3. lst4 consists of two string values which are "itu", "1773". lst5 has different types of items.

1.2 Indexing

In Python, we can index the list with two ways; positive indexing and negative indexing. To investigate indexing check the Code Snippet 2;

Code Snippet 2: List Indexing

```
lst = [1, 2, 3, "itu", -2, [22, 33, 44], True, 'a']
print(lst[0])
print(lst[-1])
print(lst[-3])
print(lst[-3][2])
print(lst[5][2])
print(lst[6])
print(lst[3])
print(lst[8])
print(lst[2.0])
print(lst[int(2.0)])
print(lst.index("itu")) # returns index of an item

```

Try to guess the output of the above code snippet.

1.3 Slicing

Sometimes, it is necessary to get more than one item from lists. We can do this using slicing. To investigate slicing check the Code Snippet 3;

Code Snippet 3: List Slicing

```
lst = [1, 2, 3, "istanbul", -2, [22, 33, 44]]
print(lst[:])
print(lst[:5])
print(lst[1:3])
print(lst[-4:-1])
print(lst[3][2:5])
print(lst[-1][:2])

```

Try to guess the output of the above code snippet.

1.4 Manipulation

Sometimes, it is necessary to interact with lists in a specific way. Python provides us with some manipulations. To investigate manipulations on the lists check the Code Snippet 4;

Code Snippet 4: List Manipulation

```
lst = [1, 2, 3, 4, 5, 6]
print(lst)
lst.append(0)
print(lst)
lst.insert(0, -1)
print(lst)
last_item = lst.pop()
print(last_item)
print(lst)
item = lst.pop(1)
print(item)
print(lst)
lst.remove(0)
print(lst)
lst.extend([3,4,6,7])
print(lst)
lst.sort()
print(lst)
lst.reverse()
print(lst)
print(len(lst))
lst.clear()
print(len(lst))
print(lst)
```

Try to guess the output of the above code snippet.

2 Part 2: Strings

In this part, we will investigate the strings in Python.

2.1 Creation

In Python, we can create strings like in Code Snippet 5;

Code Snippet 5: String Creation

```
str0 = ""
str1 = "Istanbul Technical University 1773"
str2 = """Introduction to Programming for Data Science (Python)"""
```

str0 is an empty string. str1 and str2 are strings that have characters in it.

2.2 Accessing

Examples for accessing to strings are given in Code Snippet 6;

Code Snippet 6: String Accessing

```
str0 = """
ITU is a large city university with campuses located in the central points of Istanbul.
"""
print(str0[0])
print(str0)
print(str0[10:15])
print(str0[5:-1])

```

Try to guess the output of the above code snippet.

2.3 Operations

Examples for operations on the strings are given in Code Snippet 7;

Code Snippet 7: String Operations

```
str0 = "Istanbul Technical University"
print(str0 + str0)
print(str0 + " " + str0)
```

```
print(str0 + "!!!!")
print(str0 * 3)
print(str0.upper())
print(str0.lower())

```

Try to guess the output of the above code snippet.

2.4 Formatting

Examples for string formatting are given in Code Snippet 8;

Code Snippet 8: String Formatting

```
# printing some characters
# using triple quotes
str1 = '''She said, "What's up?"'''

# escaping single quotes
str2 = 'She said, "What\'s up?"'

# escaping double quotes
str3 = "She said, \"What's up?\""

print(str1)
print(str2)
print(str3)

print('\n')
print('1\t 2')

name = "Ali"
age = 23
city = "Istanbul"

# printing formatted string
print(name + " is " + str(age) + " years old and he lives in " + city + ".")
print("{} is {} years old he lives in {}.".format(name, age, city))
print("{a} is {b} years old he lives in {c}.".format(a=name, b=age, c=city))
print("{a} is {b} years old he lives in {c}.".format(c=city, a=name, b=age))
print("%s is %d years old he lives in %s."%(name, age, city))
print(f"{name} is {age} years old he lives in {city}.")

# printing with precision
```

```
pi = 3.14159265359
print(f"{pi:.1f}")
print(f"{pi:.2f}")
print(f"{pi:.3f}")
print(f"{pi:.4f}")
print(f"{pi:.5f}")
```

Try to guess the output of the above code snippet.

2.5 Methods

Examples for string methods are given in Code Snippet 9;

Code Snippet 9: String Methods

```
str1 = "ITU is a large city university with campuses located in the central points of Istanbul"
str2 = "itu"

print(str2.capitalize())
print(str1.count("a"))
print(str1.startswith("i"))
print(str1.startswith("I"))
print(str1.startswith("It"))
print(str1.startswith("IT"))
print(str1.endswith("istanbul"))
print(str1.endswith("Istanbul"))
print(str1.find("itu"))
print(str1.find("IT"))
print(str1.find("TU"))
print(str2.isupper())
print(str2.islower())
print(str1.replace("ITU", "İTÜ"))
print(str1.replace("itu", "İTÜ"))
print(str1.split())
print(str1.split(" "))
print(str1.split("is"))

```

Try to guess the output of the above code snippet.
You can check other string methods in Python in the URL following;
https://docs.Python.org/3/library/stdtypes.html#string-methods

3 Part 3: Dictionaries

In this part, we will investigate the dictionaries in Python.

3.1 Creation

In Python, we can create dictionaries like in Code Snippet 10;

Code Snippet 10: Dictionary Creation

```
dict0 = {}
dict1 = {'name': 'Ali', 'age': 23, 'city': 'Istanbul'}
dict2 = {'name': 'Ali', 'age': 23, 'city': 'Istanbul', 'age': 24}
dict3 = dict({'name':'Ali', 'age':'23', 1:1})
print(dict0)
print(dict1)
print(dict2)
print(dict3)

dict0["age"] = 23
dict1["name"] = "Aylin"
print(dict0)
print(dict1)

item = dict3.pop('name')
print(item)
print(dict3)

dict3.clear()
print(dict3)
```

Try to guess the output of the above code snippet.

3.2 Accessing

Examples for accessing to values of the dictionaries are given in Code Snippet 11;

Code Snippet 11: Dictionary Accessing

```
dict1 = {'name': 'Ali', 'age': 23, 'city': 'Istanbul'}
print(dict1["name"])
print(dict1.get("name"))
print(dict1["born"])
```

```
5 print(dict1.get("born"))
6
```

Try to guess the output of the above code snippet.