Python training - lab 3

Statements - for

```
s = ['a', 'b', 'c']
for x in s:
    # do something
    if condition:
        break
    if condition:
        continue
else:
    # do something only if there was no break

for x in range(start, stop, step):
    print(s[i])
```

Statements - for

```
s = ['a', 'b', 'c']

for i, e in enumerate(s):
   print(i, e)


for i, e in enumerate(s, start=1):
   print(i, e)
```

string operations

```
# split - splits a string
s = 'ala bala portocala'
s.split()
r = s.split('la')

# join - joins a structure
j = ' # '
l = ['X', 'Y', 'Z']
j.join(r)
j.join(l)
```

string operations

```
# strip - strips characters at the end and
# beginning of a string
s = '  ala bala portocala  \n'
s.strip()
r = s.strip(' al\n')

r = s.rstrip(' al\n')
r = s.lstrip(' al\n')

s.strip().lower().split()
```

string operations

```
# multiline string

s1 = 'one line\n other line'

s2 = '''
one line
other line
'''

s3 = 'one line' \
   'other line'
```

slicing

```
s = 'abcdefghij'
s[3]        # 'd'
s[3:6]      # 'def'
s[3:]       # 'defghij'
s[:6]       # 'abcdef'
s[2:6:2]    # 'ce'
s[6:2:-2]   # 'ge'
s[::-1]     # 'jihgfedcba'

l = ['unu', 'doi', 'trei', 'patru']
l[::-1] # ['patru', 'trei', 'doi', 'unu']
```

slicing

```
s = 'abcdefghij'
l = ['unu', 'doi', 'trei', 'patru']

sl = slice(2, 5, 2)
print(s[sl])
print(l[sl]

sl2 = slice(2, None, 3)
```

Getting help

```
s = 'abcdefghij'

help(s)    # display help for str type
help(str)  # same as above

help(s.split)    # display help for split
help(str.split) # same as above

help(str.split())  # not ok to use parentheses
```

Chaining comparison

```
# the expressions below are equivalent
a < b <= c             # short form
a < b and b <= c       # long form

# more generic example
# a, b, c.. are expressions
# o1, o2, o3.. are comparison operators
a o1 b o2 c o3 d # is equivalent to
a o1 b and b o2 c and c o3 d

# short form only available in Python, not in
other languages
```