COMPUTER PROGRAMMING I
Introduction To Python

BIL2205

Dokuz Eylul University, Faculty of Science,
Department of Statistics

Logical Operators

- AND
- OR
- NOT

A	B	A and B
0	0	0
0	1	0
1	0	0
1	1	1

AND

A	B	A or B
0	0	0
0	1	1
1	0	1
1	1	1

OR

A	not A
0	1
1	0

NOT

- Logical Operators
  - NOT

```
ders = input("Bu dersin kodu?: ")
if not (ders == "BİL2205"):
    print ("Bilemedin...")
else:
    print ("Evet doğru...")
```

Python Programming Language

- Logical Operators
  - AND

```
True and True => True
True and False => False
False and True => False
False and False => False
```

```
if saat > 9 and saat < 17:
    print ("Mesaideyiz...")
else:
    print ("Dükkan kapalı...")
```

Python Programming Language

- Logical Operators
  - OR

```
True or True => True
True or False => True
False or True => True
False or False => False
```

```
if gün == "Cumartesi" or gün == "Pazar":
    print ("Yaşasın Tatil...")
else:
    print ("Bugün iş günü...")
```

Python Programming Language

Logical Operators

```
a = 5
b = 10
print (a > b and a > 1) ➔ False
print (a > 1 and b > a) ➔ True
print (a == 5 and b < 100) ➔ True
print (a > 1 and b < 1 and b > a)
                          ➔ False
print (a > 1 and b > 1 and b > a)
                          ➔ True
```

Python Programming Language

Logical Operators

```
a = 5
b = 10
print (a > b or a > 1)
print (a > 1 or b > a)
print (a == 5 or b < 100)
print (a > 1 or b < 1 or b > a)
print (a > 1 or b > 1 or b > a)
```

➔ True

➔ True

➔ True

➔ True

➔ True

Python Programming Language

- Modules
  - Modules are containers for additional functions and code.
  - Modules are also called Libraries and Packages.
  - Generally, modules are used to wrap related functionality together, and to make it available optionally – so you don't load it (and take up memory) if you don't need it.
  - Python has many modules that provide standardized solutions for many problems that occur in everyday or specialized programming

Python Programming Language

- Using Modules
  - You "load" a module (tell Python you are going to use it) with the `import` statement.
  - `import` looks for a file with that name and the .py extension. First it looks in the "local" directory (the same directory where your program file lives). If it can't find a file with that name there, it looks in the global modules directory for your Python installation.

Python Programming Language

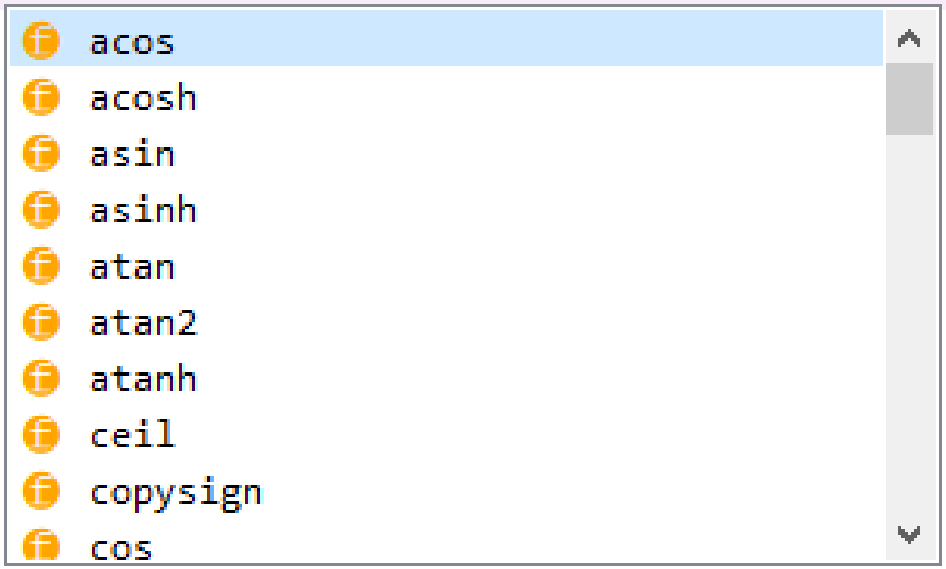
Using Modules

- Once your module is loaded, you use functions or values from it using the "dot" syntax:
  - `math.sqrt(49)`
  - `random.randint(1,10)`
- Example:

```
import math
print (math.sqrt(49))
```

Using Modules

```
import math
print (math.sqrt(49))
math.
```



Python Programming Language

- "random" Module
- contains functions for generating random values.
  - `random.random()`
  - `random.randint(a, b)`
  - `random.choice([ a list of options ])`

Python Programming Language

- `random.random()`
  - generates a float number in the interval [0, 1).
  - `print (random.random())`
- `random.randint(a, b)`
  - returns random integer in range [a, b], including both end points.
  - `print (random.randint(1, 6))`
- `random.choise([ a list of options ])`
  - choose a random element from a non-empty sequence.
  - `print (random.choice([3,1,6,7,8,2]))`

Python Programming Language

- `random.seed(a)`
  - Determine the right seed value to generate the deterministic random data you want.
  - `random.seed(12)`
  - `print (random.randint(1, 6))`

Exercise

- Write a program to guess the number randomly chosen by the computer.
- Roll 2 dice for 2 players. Display the winner in the following rules:
  - Greater dice wins
  - Less dice wins
  - Odd dice wins
  - Even dice wins
  - "1", "2" or "6" wins

Programming Challange

- Write a program to ask the user to select one of three options - Taş (t), Kağıt (k) or Makas (m).
- Use the `random.choice()` function to select an option for the computer.
- Determine the winner and print the result:
  - Taş Makası yener
  - Makas Kağıdı yener
  - Kağıt Taşı yener