This week's topics were Python.I did not have any information before because I was never interested. So I decided to spend the day learning python and doing examples. Every Linux operating system comes with Python 2.x ready to install.So I started to practice at the terminal without trying to setup. First of all, if you need to talk a bit about python:

Python is an object-oriented, interpretable, modular, and interactive programming language begun in 1990 by Guido van Rossum in Amsterdam. Python language Unlike languages ​​like C - C + +, Interpreter is an interpretive language.   
So you can run it without compiling and you can develop your application very quickly on this site.

• Using Python, desktop programming, game programming, portable device programming, web programming and network programming can be performed easily.

• There are many IDEs (Integrated Development Enviroment) that you can use when programming with Python. Some of those: Eclipse, Pydev, Eric, Komodo IDE, PyCharm.

• Python optimizes your application's memory usage thanks to the Garbage Collector it contains.

• Python is a fully object-oriented programming language. It has an infrastructure on the level to compete with popular OOP languages.

It has the ability to work in an integrated way with Python, Java and .NET platforms.

Bu haftanın konuları arasında Python vardı.Daha önce hiç ilgilenmediğim için bilgi sahibi değildim.O yüzden bugünü python’ı öğrenip , örnekler yaparak geçirmeye karar verdim.Her Linux işletim sisteminde Python 2.x sürümü hazır yüklü olarak gelir.O yüzden kurulumla uğraşmayıp terminalde pratik yapmaya başladım.Öncelikle biraz pythondan bahsetmek gerekirse :

Python 1990 yılında Guido van Rossum tarafından Amsterdam'da geliştirilmeye başlanmış nesne yönelimli, yorumlanabilen, modüler ve etkileşimli bir programlama dilidir. Python dili C – C++ gibi dillerin aksine Interpreter, yani yorumlayıcı bir dildir. Dolayısıyla derlemeye gerek kalmadan çalıştırabilir ve bu sayede çok hızlı bir biçimde uygulama geliştirebilirsiniz.

* Python kullanarak masaüstü programlama, oyun programlama, taşınabilir cihaz programlama, web programlama ve ağ programlama çalışmaları rahatlıkla yürütülebilir.
* Python ile programlama yaparken kullanabileceğiniz pek çok IDE (Integrated Development Enviroment – Tümleşik Geliştirme Ortamı) mevcuttur. Bunlardan bazıları:

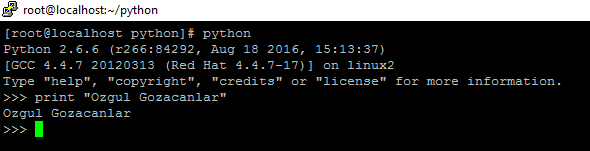
Eclipse, Pydev, Eric, Komodo IDE, PyCharm

* Python, içerisinde barındırdığı Garbage Collector (çöp toplayıcı) sayesinde uygulamanızın bellek kullanımını optimize eder. Bu durum uygulamanızın kararlılığını ve performansını arttıracaktır.
* Python, tamamı ile nesne yönelimli bir programlama dilidir. Popüler OOP dilleri ile yarışacak seviyede bir altyapıya sahiptir.
* Python, Java ve .NET platformları ile entegre biçimde çalışma yeteneğine sahiptir.

**Print Komutu Print Command**

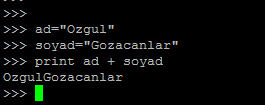
 Print ile ekrana yazı yazdırmak için sadece çift tırnak("yazı") kullanılmak zorunda değildir. Aynı zamanda tek tırnak('yazı') ve üç tırnak("""yazı""") da kullanılabilir.

Print does not have to use double quotes ("text") to print text on the screen. At the same time, single quotes ('text') and three quotes ("" "text" "") can be used.



Değişkenler Variables

It is enough to write the name and value when defining the variable.

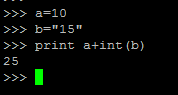


Dönüştürme İşlemleri Conversion Processes

float() => Convert any number or number of character strings to a dotted number

int() => Converts any numeric or numeric character string to an integer.

str() => Converts any number to a character string.



Basic File Operations

Creating a File

A function called open () is used for file operations in the Python programming language.

python = open(file\_name, kip)

bir dosyayı **yazma** kipinde açmak için şöyle bir komut kullanıyoruz:   
We use a command like this to open a file in write mode:

python = open("python", "w")

**Dosyaya Yazmak** Writing to a File

file.write(“…”)

**Dosya Okumak** Read File

python = open("python.txt", "r")

Three different metatags are used to read a file we opened with the "r" mode: read (), readline (), and readlines ().The readlines () method lists when the read () and readline () methods give us a character array as output.

“r” kipiyle açtığımız bir dosyayı okumak için ise read(), readline() ve readlines() adlı üç farklı metottan yararlanılır. read() ve readline() metotları çıktı olarak bize bir karakter dizisi verirken, readlines() metodu liste verir.

If an unexpected error occurs in the program after opening a file, your program may never reach the close () line. In this case, the file will be opened without closing.

bir dosya açıldıktan sonra programda beklenmeyen bir hata gerçekleşirse, programınız asla close() satırına ulaşamayabilir. Bu durumda da açılan dosya kapanmadan öylece bekler.

Bu tür durumlara karşı iki seçeneğiniz var: There are two options for such situations:

1. Using try ... except ... finally ... blocks

1. **try**:
2. dosya = open("dosyaadı", "r")
3. ...burada dosyayla bazı işlemler yapıyoruz...
4. ...ve ansızın bir hata oluşuyor...
5. **except** **IOError**:
6. print("bir hata oluştu!")
7. **finally**:
8. dosya.close()

2. Use an idiom with

**with** open("dosyaadı", "r") **as** dosya:

print(dosya.read())

Veri Giriş - Çıkış İşlemleri Data Input - Output Operations

**raw\_input() Fonksiyonu**

The raw\_input () function allows the user to retrieve data from the keyboard. Raw\_input () takes all values ​​entered by the user as a character sequence.

raw\_input() fonksiyonu klavye aracılığı ile kullanıcıdan veri alınmasını sağlar. raw\_input() kullanıcı tarafından girilen bütün değerleri karakter dizisi olarak alır.

sayi1 = raw\_input("Enter the first number:")

sayi2 = raw\_input("Enter the second number:");

sum = int(sayi1)+int(sayi2)

print sum

Döngüler Loops

While döngüsü : number=1

while number<10:

         number=number+1

          print number

For döngüsü :

|  |
| --- |
| for harf in “Damla-Umut”:      print harf |
|  |

**range():**With this function, we can specify number ranges.

**len():**This is a built-in function. It gives us the length of the character sequences.

Modüller

Every program written in Python is also a module. With this feature, functions and features found in Python programs can be easily used in other Python programs.

Python'da yazılmış her program aynı zamanda birer modüldür. Bu özellik sayesinde Python programlarında bulunan fonksiyon ve özellikler başka Python programlarında da rahatça kullanılabilirler.

1.os Modülü

2.sys Modülü

3.time Modülü

4.random

5.webbrowser

6.pyquery

7.smtqlib

import ilkmodul

print ilkmodul.program

Python

#!/usr/bin/env python

#-\*- coding:utf-8 -\*-

def faktoriyel(sayi):

    fak = 1

    for i in range(sayi):

        fak= fak \* (i+1)

    return fak

modüller sayesinde sadece bu program içinde değil istediğimiz herhangi bir program içerisinde kullanabileceğiz. We can use the codes written by modules in any program we want

import faktoriyel

hesap = faktoriyel.faktoriyel(5)

print hesap

120

**Modülleri İçe Aktarma Yöntemleri** Methods of Transferring Modules

from ilkmodul import \*

print program

Today I learned about the main python commands. In the following hours, I continued to practice with small encodings on the Linux terminal. I realized that Python is not a very difficult language, but it is a widely used language. So I will make more progress on python.

Bugünümü başlıca python komutlarını öğrenerek geçirdim.İlerleyen saatlerde Linux uç biriminde küçük kodlamalar yaparak pratiklerime devam ettim.Python çok zor bir dil olmamakla beraber çokça kullanılan bir dil olduğunu farkettim.Bu yüzden python üzerinde daha fazla ilerleme sağlayacağım.