



**GAZİ ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ**

BM402 BİLGİSAYAR AĞLARI

FTP PROGRAMI

141120008 MİNEL AYDIN

**ANKARA
MART, 2019**

MİNNELA'S WORK -FTP İSTEMCİ TARAFI KODLARI-PYHTON

FTP programı bağlantısı kurmak için, ftplib kütüphanesinden yararlandım. Arayüz tasarımı (GUI) için ise Pyhton'ın Tkinter kütüphanesini kullandım. Öncelikle kütüphanelerimizi import edelim;

```
import io
from io import StringIO
import shutil
from ftplib import FTP
from tkinter import *
import os
import glob
from tkinter import filedialog
```

Os kütüphanesi, istemci dosyaları üzerinde işlem yapabilmemiz için gerekli olan bir kütüphanedir.

```
ftp = FTP('')
ftp.connect('localhost',1200)
ftp.login(user='minnela', passwd='italy')
ftp.cwd('.')

os.chdir("C:/istemci")

root = Tk()
root.geometry("1350x700+300+150")
root.resizable(width=True, height=True)
```

FTP bağlantım için kullanıcı adı ve şifremi belirledikten sonra, Tk kütüphanesi ile kullanacağım GUI'min boyutlarını, uzunluğunu ve genişliğini tayin ettim.

Daha sonra ise sırayla önce sunucu tarafının yapacağı klasör değiştir-dosya indir-dosya yükle vs gibi işlemleri oluşturdum.

```
def klasor_degistir():
    ftp.cwd('..' + e2.get())
    RemoteServerFileFolderList.delete('1.0', END)
    for i in ftp.nlst():
        RemoteServerFileFolderList.insert(END, str(i) + "\n")

def geri_git():
    ftp.cwd('..')
    RemoteServerFileFolderList.delete('1.0', END)
    for i in ftp.nlst():
        RemoteServerFileFolderList.insert(END, str(i) + "\n")
```

Klasör değiştirirken önce var olan klasör silinir ve daha sonra eklenmek isteyen klasör for döngüsü ile eklenir. ftp.cwd komutu dosyaların değiştirilmesini sağlayan bir komuttur. [ftp.cwd\('./'\)](#) ise bir önceki klasöre yönlendirmeyi sağlayan komuttur.

Ardından dosya ve klasör açmak için gereken kodları yazdım:

```
def dosya_ac():
    to_bytes = bytes(acilacakDosyayaVerilecekString.get(), "UTF-8")
    yazilacak_yazi = io.BytesIO(to_bytes)
    ftp.storbinary('STOR '+acilacakDosya.get(), yazilacak_yazi)
    RemoteServerFileFolderList.delete('1.0', END)
    for i in ftp.nlst():
        RemoteServerFileFolderList.insert(END, str(i)+"\n")

def klasor_ac():
    ftp.mkd(uzakSunucudaAcilacakKlasor.get())
    RemoteServerFileFolderList.delete('1.0', END)
    for i in ftp.nlst():
        RemoteServerFileFolderList.insert(END, str(i)+"\n")
```

STOR, dosya açmaya yarayan bir komuttur. Dosya açmak için ftp.storbinary komutunu kullanırız. Dosya ismini ve yazılacak yazıyı da belirleriz. Bunun için önce kullanıcı inputundan gelen stringi byteye çevir. Ardından STOR komutu ile dosya açarız ve içine byte stringleri yazarız.

Ardından dosya ve klasör silmek için aşağıdaki kodları yazdım;

```
def dosya_sil():
    ftp.delete(silinecekDosya.get())
    RemoteServerFileFolderList.delete('1.0', END)
    for i in ftp.nlst():
        RemoteServerFileFolderList.insert(END, str(i)+"\n")

def klasor_sil():
    ftp.rmd(silinecekKlasor.get())
    RemoteServerFileFolderList.delete('1.0', END)
    for i in ftp.nlst():
        RemoteServerFileFolderList.insert(END, str(i)+"\n")
```

[ftp.delete](#)(dosya.get()) komutu silmek istediğimiz dosyayı silmemizi sağlar. [ftp.rmd](#) ise silmek istediğimiz klasörü silmek için kullandığımız bir koddur.

Dosya indirmek için ise RETR kullanarak ardından local adresi belirtiyoruz.

```
def dosya_indir():
    targerfilename = 'C:/istemci/'+str(indirilecekDosyaIsmi.get())
    localfile = open(targerfilename, 'wb')
    ftp.retrbinary('RETR '+str(indirilecekDosyaIsmi.get()), localfile.write, 1200)
    localfile.close()
    T2.delete('1.0', END)
    for file in os.listdir(os.getcwd()):
        T2.insert(END, str(file)+"\n")

def adini_degistir():
    ftp.rename(isimiDegisecekDosya.get(), isimiDegisecekDosyaYeniIsim.get())
    RemoteServerFileFolderList.delete('1.0', END)
    for i in ftp.nlst():
        RemoteServerFileFolderList.insert(END, str(i)+"\n")
```

Dosyanın ismini değiştirmek için kullandığımız komut ise ftp.rename 'dir. Değiştirmek istediğimiz dosyanın ismi virgül yeni isim şeklinde tanımlayarak dosya adını değiştirebilmekteyiz.

Şimdi ise istemci dosyalarımızın üzerinde işlem yapmaya başlayabiliriz:

```
def change_working_directory_local():
    os.chdir('..' + e4.get()) #./ = ŞU ANKI DOSYA DİZİNİNİ İFADE EDİYOR
    T2.delete('1.0', END)
    for file in os.listdir(os.getcwd()):
        T2.insert(END, str(file)+"\n")

def go_back_local():
    if os.getcwd() != ("C:\\istemci"):
        os.chdir('..') # ../ = BİR ÜST DOSYAYA GİT. Bİ ÖNCEKİ DOSYA
    T2.delete('1.0', END)
    for file in os.listdir(os.getcwd()):
        T2.insert(END, str(file)+"\n")
```

İstemci dosyasındaki çalışma klasörünü değiştirmek için os kütüphanesi kullanırız. Şuanki dosya dizinini öncelikle silerek sonra yeni dosya dizinini ekler. Os.chdir şuanki klasörümüz, os.getcwd ise değiştirdiğimiz yeni directorydir.

Bir dosya upload edebilmek için yine STOR komutu kullanmamız gerekiyor ve ardından yükleyeceğimiz yeri belirtiyoruz;

```
def uploadFile():
    filename = download_filename_entry2.get()
    ftp.storbinary('STOR '+filename, open(os.getcwd()+"/"+filename, 'rb'))
    RemoteServerFileFolderList.delete('1.0', END)
    for i in ftp.nlst():
        RemoteServerFileFolderList.insert(END, str(i)+"\n")

def createFileLocal():
    file = open(create_file_local.get(), "w+")
    file.write(create_file_local_input_text.get())
    file.close()
    T2.delete('1.0', END)
    for file in os.listdir(os.getcwd()):
        T2.insert(END, str(file)+"\n")
```

Bir dosya yaratmak için ise dosyayı açıp, oluşturup sonrasında ise kapatarak yeni bir dosya oluştururuz.

Local dosyamızı silmek için os kütüphanesini kullanarak os.remove komutunu kullanırız.

Local directory'i silmek için ise şuan bulunduğumuz os.getcwd yi replace etmemiz gerekmektedir;

```
def deleteFileLocal():
    os.remove(delete_filename_entry_local.get())
    T2.delete('1.0', END)
    for file in os.listdir(os.getcwd()):
        T2.insert(END, str(file)+"\n")

def deleteDirectoryLocal():
    dirname = os.getcwd()+"\\"+delete_directory_entry_local.get()
    dirname=dirname.replace('\\', '/')
    shutil.rmtree(dirname)
    T2.delete('1.0', END)
    for file in os.listdir(os.getcwd()):
        T2.insert(END, str(file)+"\n")
```

Son olarak Local dosyamıza yeniden isim vermek için os.rename, yeni bir dosya yaratmak için ise os.mkdir kodlarını kullanmaktayım;

```
def renameFileLocal():
    os.rename(rename_file_local_first.get(),rename_file_local_last.get())
    T2.delete('1.0', END)
    for file in os.listdir(os.getcwd()):
        T2.insert(END,str(file)+"\n")

def createDirectoryOnLocal():
    os.mkdir(create_directory_on_local_directory_name.get())
    T2.delete('1.0', END)
    for file in os.listdir(os.getcwd()):
        T2.insert(END,str(file)+"\n")
```

Dosya açma,indirme, silme gibi komutları tanımladıktan sonra GUI kısmının tasarımına geçtim.

Oluşturduğum GUI'min genel görüntüsü;



Sol tarafta sunucu bölgesinde yapılacak işlemler, sağda ise istemci bölgesi yer almaktadır. Çalışma klasörünü değiştir komutu ile çalışmak istediğimiz bölgeyi belirtiriz. Dosya ve Klasör silmek için silmek istediğimiz değeri gireriz ve dosyayı sileriz.

Bir dosyanın ismini değiştirmek istediğimizde ise önce dosya adını gireriz ve değiştirmek istediğimiz yeni ismi girerek yeniden adlandır butonuna basarız.

Yeni bir klasör ve dosya yaratmak için ise en aşağıda gördüğümüz satır ve butonları kullanmaktayız.

FTP-SUNUCU KODLARI

```
from pyftplib.authorizers import DummyAuthorizer
from pyftplib.handlers import FTPHandler
from pyftplib.servers import FTPServer

yonetici = DummyAuthorizer()
yonetici.add_anonymous("C:/sunucu", perm="elradfmw")
yonetici.add_user("minnella", "italy", "C:/sunucu", perm="elradfmw")
yonetici.add_user("sixfinger", "Angela Fortunata", "C:/sunucu", perm="elradfmw")
```

Sunucu portu dinlemek için pyftplib kütüphanesini kullanmaktadır. FTP protokolü için kullanıcı ve yöneticileri tanımladım.

```
handler = FTPHandler
handler.authorizer = yonetici
server = FTPServer(("127.0.0.1", 1200), handler)
server.serve_forever()
```

Server programım local hostsu 1200 portu üzerinden dinlemektedir. Gelen istekler 1200 portu üzerinden geçmektedir.