

ÖMER BAYRAMÇAVUŞ

191180763

BM402

ÖDEV RAPORU

FTP PROGRAMI

İçindekiler

1. Giriş	1
2. Server Uygulaması	1
3. Client Uygulaması	2
3.1 Sunucu Bağlantısı ve Giriş	3
3.2 Ana Menü	4
3.3 Yerel ve Uzak Sunucu İşlemleri	4
3.3.1 Dizin Oluştur	5
3.3.2 Dizin Sil	6
3.3.3 Dosya Adı Değiştir	6
3.3.4 Dosya Sil	6
3.3.5 Dizin Değiştir	7
3.3.6 Dosya Listele	7
3.4 Dosya Yükle	8
3.5 Dosya İndir	8
4. Sonuç	9
Kaynakça	10
EKLER	11

Şekiller Listesi

Şekil 1 FTP Server Uygulama Arayüzü.....	1
Şekil 2 Sunucu Bağlantısı ve Girişi	4
Şekil 3 Ana Menü	4
Şekil 4 Yerel İşlemler Menü	5
Şekil 5 Uzak Sunucu İşlemler Menü	5
Şekil 6 Dizin Oluşturma.....	5
Şekil 7 Dizin Silme	6
Şekil 8 Dosya Adı Değiştirme	6
Şekil 9 Dosya Silme.....	6
Şekil 10 Dizin Değiştirme.....	7
Şekil 11 Yerel Dosya Listeleme	7
Şekil 12 Uzak Sunucu Dosya Listeleme Oluşturma	7
Şekil 13 Dosya Yükle	8
Şekil 14 Dosya İndir	8

1. Giriş

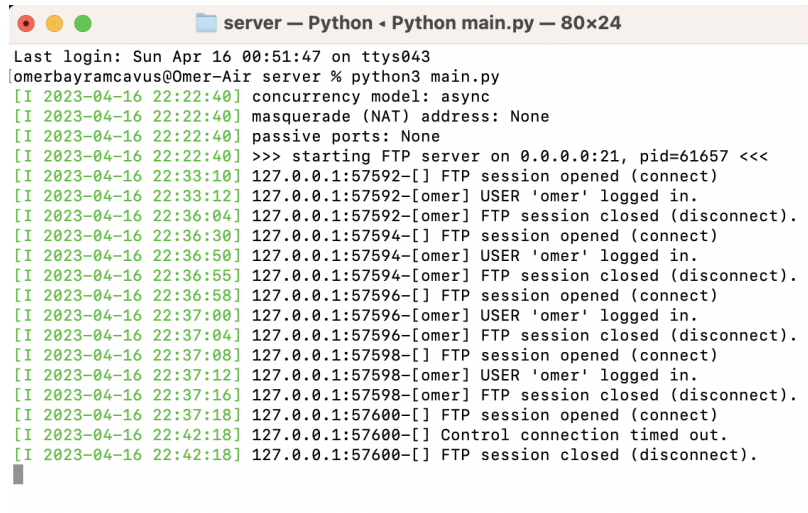
FTP uygulamasını yaparken iki taraflı olarak tüm işlemleri gerçekleştirdim. Bunlar client ve server tarafı olarak ikiye ayrıldı. İsterleri çoğunlukla client tarafını kodlarken gerçekleştirdim.

Server uygulaması çalıştırıldığında 21 portuna gelen ftp isteklerini karşılamak üzere hazır halde beklemektedir. Client uygulamasını bir console uygulaması olarak tasarladım ve isterleri console arayüzü üzerinden gerçekleştiriliyor.

FTP uygulaması ile ilgili tüm kodlar ekler kısmına eklenmiştir.

2. Server Uygulaması

Server uygulaması çalıştırıldığında 21 portunda hazır hale gelmektedir ve ftp isteklerini karşılamak üzere hazır bir şekilde çalışmayı sürdürür. Şekil 1 de server uygulaması console arayüzünü göstermektedir. Gelen tüm istekler burada loglanmaktadır.



```
Last login: Sun Apr 16 00:51:47 on ttys043
omerbayramcavus@Omer-Air server % python3 main.py
[I 2023-04-16 22:22:40] concurrency model: async
[I 2023-04-16 22:22:40] masquerade (NAT) address: None
[I 2023-04-16 22:22:40] passive ports: None
[I 2023-04-16 22:22:40] >>> starting FTP server on 0.0.0.0:21, pid=61657 <<<
[I 2023-04-16 22:33:10] 127.0.0.1:57592-[] FTP session opened (connect)
[I 2023-04-16 22:33:12] 127.0.0.1:57592-[omer] USER 'omer' logged in.
[I 2023-04-16 22:36:04] 127.0.0.1:57592-[omer] FTP session closed (disconnect).
[I 2023-04-16 22:36:30] 127.0.0.1:57594-[] FTP session opened (connect)
[I 2023-04-16 22:36:50] 127.0.0.1:57594-[omer] USER 'omer' logged in.
[I 2023-04-16 22:36:55] 127.0.0.1:57594-[omer] FTP session closed (disconnect).
[I 2023-04-16 22:36:58] 127.0.0.1:57596-[] FTP session opened (connect)
[I 2023-04-16 22:37:00] 127.0.0.1:57596-[omer] USER 'omer' logged in.
[I 2023-04-16 22:37:04] 127.0.0.1:57596-[omer] FTP session closed (disconnect).
[I 2023-04-16 22:37:08] 127.0.0.1:57598-[] FTP session opened (connect)
[I 2023-04-16 22:37:12] 127.0.0.1:57598-[omer] USER 'omer' logged in.
[I 2023-04-16 22:37:16] 127.0.0.1:57598-[omer] FTP session closed (disconnect).
[I 2023-04-16 22:37:18] 127.0.0.1:57600-[] FTP session opened (connect)
[I 2023-04-16 22:42:18] 127.0.0.1:57600-[] Control connection timed out.
[I 2023-04-16 22:42:18] 127.0.0.1:57600-[] FTP session closed (disconnect).
```

Şekil 1 FTP Server Uygulama Arayüzü

Ek-1'de server uygulaması ile ilgili tüm kodlar bulunmaktadır. Burda kullanılan kodları aşağıda açıkladım.

- pyftplib.authorizers modülünden DummyAuthorizer sınıfı dahil edilir.
- pyftplib.handlers modülünden FTPHandler sınıfı dahil edilir.
- pyftplib.servers modülünden FTPServer sınıfı dahil edilir.
- DummyAuthorizer sınıfından bir nesne oluşturulur.
- add_user() yöntemi kullanılarak, kullanıcı adı "omer", şifre "123" ve izinler "elradfmw" olan bir kullanıcı oluşturulur ve authorizer nesnesine eklenir.
- FTPHandler sınıfından bir nesne oluşturulur.
- FTPHandler nesnesinin authorizer özelliği, yukarıda oluşturulan authorizer nesnesine ayarlanır.

- FTPServer sınıfından bir nesne oluşturulur ve ("0.0.0.0", 21) adresinde dinleme yapacak şekilde ayarlanır.
- max_cons özelliği, eşzamanlı bağlantıların maksimum sayısını belirtmek için 256 olarak ayarlanır.
- max_cons_per_ip özelliği, her IP adresi için eşzamanlı bağlantıların maksimum sayısını belirtmek için 5 olarak ayarlanır.
- serve_forever() yöntemi kullanılarak FTP sunucusu başlatılır ve sürekli olarak çalışır.

3. Client Uygulaması

Client uygulamasını bir console uygulaması olarak tasarladım ve herhangi bir ftp sunucusuna bağlanabilecek ve giriş yapılabilir bir şekilde tasarlanmıştır.

Client uygulamasında 4 ana işlem vardır. Bunlar yerel işlemler, uzak sunucu işlemleri, dosya yükleme ve dosya indirmedir. Yerel işlemler ve uzak sunucu işlemleri altında yapılabilecek alt işlemler vardır. Her iki ana işlemin altında da izin oluşturma, izin silme, dosya adı değiştirme, dosya silme, izin değiştirme ve dosya listeleme işlemleri yapılabilmektedir.

Client uygulaması ile ilgili tüm kodlar EK-2’de verilmiştir. Nesne tabanlı olarak kodlamayı gerçekleştirmiş olup Yerel, Uzak ve Common olarak 3 class oluşturdum. Yerel classı tüm yerel işlemleri gerçekleştirmektedir. Uzak classı uzak sunucu ile ilgili işlemleri gerçekleştirmektedir. Uzak ve Yerel classları aynı methodları sahiptir fakat bu methodların işlevleri farklıdır. Uzak classı uzak sunucu ile ilgili işlemleri gerçekleştirirken Yerel classı yerel dosyalar ile ilgili işlemleri gerçekleştirir. Common classı dosya indirme ve dosya yükleme işlemlerini gerçekleştirir. Kodlamayı yaparken, console uygulamasında kullanılan genel komutlar dışında bazı kütüphaneler ve bu kütüphaneler içerisinde bazı nesneleri kullandım. Aşağıda bu kütüphaneler ve nesneler ile ilgili açıklamalar maddeler halinde ekledim.

- ftplib: FTP sunucularıyla etkileşim kurmak için kullanılır.
- time: Zamanla ilgili işlemler yapmak için kullanılır.
- termcolor: Terminallerde renkli metin yazmasını sağlar.
- getpass: Girdi alırken kullanıcıdan gizli metin almak için kullanılır.
- os: İşletim sistemi fonksiyonlarına erişim sağlamak için kullanılır. os modülü, dosya yönetimi, işletim sistemi çağrıları, izin yapısı ve diğer çeşitli işletim sistemi özellikleri için bir dizi fonksiyon sağlar.
- colored("Dosya Yüklendi", "green"): "Dosya Yüklendi" metnini yeşil renkte yazdırmak için kullanılır. Bu kod termcolor kütüphanesi içindeki colored fonksiyonunu kullanır.
- time.sleep(1): Programın 1 saniye beklemesini sağlar.
- os.system("clear"): Terminal ekranını temizlemek için kullanılır.
- os.mkdir(dizinadi): Yeni bir izin oluşturmak için kullanılır. dizinadi değişkeni oluşturulacak dizinin adını içerir.
- os.rmdir(dizinadi): Belirtilen isimdeki bir dizini silmek için kullanılır. dizinadi değişkeni silinecek dizinin adını içerir.

- `os.rename(eskiDosyaAdi, yeniDosyaAdi)`: Bir dosyanın adını değiştirmek için kullanılır. `eskiDosyaAdi` değişkeni eski dosyanın adını içerir, `yeniDosyaAdi` değişkeni ise yeni dosya adını içerir.
- `os.remove(dosyaadi)`: Belirtilen isimdeki bir dosyayı silmek için kullanılır. `dosyaadi` değişkeni silinecek dosyanın adını içerir.
- `os.chdir(dizin)`: Çalışma dizinini değiştirmek için kullanılır. `dizin` değişkeni yeni dizin adını içerir.
- `ftp.connect(url, int(port))`: FTP sunucusuna bağlanmak için kullanılır. `url` değişkeni sunucu adresini içerir, `port` değişkeni ise FTP bağlantı noktasını belirtir.
- `ftp.login(name,password)`: FTP sunucusuna giriş yapmak için kullanılır. `name` değişkeni kullanıcı adını, `password` değişkeni ise şifreyi içerir.
- `ftp.mkd(dizinadi)`: FTP sunucusunda yeni bir dizin oluşturmak için kullanılır. `dizinadi` değişkeni oluşturulacak dizinin adını içerir.
- `ftp.rmd(dizinadi)`: FTP sunucusundan belirtilen isimdeki bir dizini silmek için kullanılır. `dizinadi` değişkeni silinecek dizinin adını içerir.
- `ftp.rename(eskiDosyaAdi, yeniDosyaAdi)`: FTP sunucusunda bir dosyanın adını değiştirmek için kullanılır. `eskiDosyaAdi` değişkeni eski dosyanın adını içerir, `yeniDosyaAdi` değişkeni ise yeni dosya adını içerir.
- `ftp.delete(dosyaadi)`: FTP sunucusundan belirtilen isimdeki bir dosyayı silmek için kullanılır. `dosyaadi` değişkeni silinecek dosyanın adını içerir.
- `ftp.dir()`: FTP sunucusunda bulunan dosya ve dizinlerin listesini alır.
- `ftp.cwd(dizin)`: Çalışma dizinini FTP sunucusunda belirtilen dizine değiştirir.

3.1 Sunucu Bağlantısı ve Giriş

Sunucu bağlantısı yapılırken sunucu adresi ve sunucu portu kullanıcı tarafından girilir ve bu işlem başarıyla devam ederse kullanıcıdan giriş yapılması istenir. Eğer hatalı bir işlem yapılırsa işlemin tekrarlanması istenir.

Bağlantı başarıyla gerçekleştirdikten sonra kullanıcının giriş yapması istenmektedir. Bu işlemde başarıyla gerçekleşirse ana menüye yönlendirilir. İşlem hatalı olursa tekrar giriş işlemini yapmasını istenir.

```
client — Python main.py — 80x24
Yapımcı: Omer Bayramcavus
Numara: 191180763
FTP Client
Sunucuya Bağlan
Sunucu Adresi: localhost
Sunucu Portu: 21
Bağlantı Başarılı
Sunucuya Giriş Yap
Kullanıcı Adı: omer
Şifre: ?
```

Şekil 2 Sunucu Bağlantısı ve Girişi

3.2 Ana Menü

Ana menü üzerinden yerel işlemler, uzak sunucu işlemleri, dosya yükleme ve dosya indirme işlemleri yapılmaktadır. Yerel işlemler ve Uzak sunucu işlemleri seçimi yapılırsa alt seçimleri yapabilecekleri menüye yönlendirilir.

```
client — Python main.py — 80x24
FTP Client
1) Yerel İşlemler
2) Uzak Sunucu İşlemler
3) Dosya Yükle
4) Dosya İndir
5) Çıkış Yap
Seçim: █
```

Şekil 3 Ana Menü

3.3 Yerel ve Uzak Sunucu İşlemleri

Yerel ve uzak sunucu işlemlerinin alt menüleri birebir aynıdır. Fakat yerel işlemler menüsünde yerel bilgisayar üzerinde işlemler yapılır. Uzak sunucu işlemlerinde uzak sunucu tarafında işlemler yapılmaktadır.

```
client — Python main.py — 80x24
Yerel İşlemler
1) Dizin Oluştur
2) Dizin Sil
3) Dosya Adı Değiştir
4) Dosya Sil
5) Dizin Değiştir
6) Dosya Listele
7) Ana Menüye Dön
Seçim: █
```

Şekil 4 Yerel İşlemler Menü

```
client — Python main.py — 80x24
Uzak Sunucu İşlemler
1) Dizin Oluştur
2) Dizin Sil
3) Dosya Adı Değiştir
4) Dosya Sil
5) Dizin Değiştir
6) Dosya Listele
7) Ana Menüye Dön
Seçim: █
```

Şekil 5 Uzak Sunucu İşlemler Menü

3.3.1 Dizin Oluştur

Dizin oluşturma işlemi hem yerel hemde uzak sunucu işlemi için aynı şekilde gerçekleştirilir. Kullanıcıdan yeni dizin ismi alınır ve bu dizin oluşturulur.

```
Seçim: 1
Dizin Adı: deneme
Dizin Oluşturuldu
█
```

Şekil 6 Dizin Oluşturma

3.3.2 Dizin Sil

Dizin silme işlemi hem yerel hemde uzak sunucu işlemi için aynı şekilde gerçekleştirilir. Kullanıcıdan silinecek dizin ismi alınır ve bu dizin silinir.

```
Seçim: 2
Dizin Adı: deneme
Dizin Silindi
█
```

Şekil 7 Dizin Silme

3.3.3 Dosya Adı Değiştir

Dizin adı değiştirme işlemi hem yerel hemde uzak sunucu işlemi için aynı şekilde gerçekleştirilir. Kullanıcıdan ismi değiştirilecek dizin ismi ve yeni dizin ismi alınarak işlem gerçekleştirilir.

```
Seçim: 3
Eski Dosya Adı: test.txt
Yeni Dosya Adı: test1.txt
Dosya Adı Değiştirildi
█
```

Şekil 8 Dosya Adı Değiştirme

3.3.4 Dosya Sil

Dosya silme işlemi hem yerel hemde uzak sunucu işlemi için aynı şekilde gerçekleştirilir. Kullanıcıdan silinecek dosya ismi alınır ve bu dosya silinir.

```
Seçim: 4
Dosya Adı: test1.txt
Dosya Silindi
█
```

Şekil 9 Dosya Silme

3.3.5 Dizin Değiştir

Dizin değiştirme işlemi hem yerel hemde uzak sunucu işlemi için aynı şekilde gerçekleştirilir. Kullanıcıdan hangi dizine gidilecek ise bu dizin ismi alınır. Eğer bir önceki dizine dönmek ister ise ‘..’ ile geri dönebilmektedir. Bu işlemin yapılmasının nedeni kullanıcı hangi dizinde ise dosya yükleme ve indirme işlemleri bu dizinde yapılmaktadır.

```
Seçim: 5
Dizin Adı(Geri Dönmek için .. yazınız): deneme
Dizin Değiştirildi
```

Şekil 10 Dizin Değiştirme

3.3.6 Dosya Listele

Dosya listeleme işlemi hem yerel hem de uzak sunucu işlemi için aynı şekilde gerçekleştirilir. Tüm dosya ve klasörler listelenir. Bu işlem yerel ve uzak sunucu için aynı şekilde gerçekleşse bile işlem çıktısı farklıdır.

```
Seçim: 6
Yerel Dizin:
test1.txt      test2.txt
```

Şekil 11 Yerel Dosya Listeleme

```
Uzak Dizin:
-rw-r--r--  1 omerbayramcavus staff      6148 Apr 16 20:40 .DS_Store
-rw-r--r--  1 omerbayramcavus staff         0 Apr 16 20:56 test3.txt
-rw-r--r--  1 omerbayramcavus staff         0 Apr 16 20:56 test4.txt
```

Şekil 12 Uzak Sunucu Dosya Listeleme Oluşturma

3.4 Dosya Yükle

Yerel dizinden uzak dizine yüklenecek dosya ismi kullanıcıdan alınır ve bu dosya yerel den uzak sunucuya yüklenir.

```
FTP Client
1) Yerel İşlemler
2) Uzak Sunucu İşlemler
3) Dosya Yükle
4) Dosya İndir
5) Çıkış Yap
Seçim: 3
Yerel Dizin:
test1.txt      test2.txt
Uzak Dizin:
-rw-r--r--  1 omerbayramcavus staff      6148 Apr 16 20:40 .DS_Store
-rw-r--r--  1 omerbayramcavus staff         0 Apr 16 20:56 test3.txt
-rw-r--r--  1 omerbayramcavus staff         0 Apr 16 20:56 test4.txt
Yüklenecek Dosya Adı: test1.txt
Dosya Yüklendi
```

Şekil 13 Dosya Yükle

3.5 Dosya İndir

Uzan sunucu dizinden yerel dizine indirilecek dosya ismi kullanıcıdan alınır ve bu dosya uzak sunucudan yerele indirilir.

```
client — Python main.py — 80x24
FTP Client
1) Yerel İşlemler
2) Uzak Sunucu İşlemler
3) Dosya Yükle
4) Dosya İndir
5) Çıkış Yap
Seçim: 4
Yerel Dizin:
test1.txt      test2.txt
Uzak Dizin:
-rw-r--r--  1 omerbayramcavus staff      6148 Apr 16 20:40 .DS_Store
-rw-r--r--  1 omerbayramcavus staff         0 Apr 16 21:06 test1.txt
-rw-r--r--  1 omerbayramcavus staff         0 Apr 16 20:56 test3.txt
-rw-r--r--  1 omerbayramcavus staff         0 Apr 16 20:56 test4.txt
İndirilecek Dosya Adı: test3.txt
Dosya İndirildi
```

Şekil 14 Dosya İndir

4. Sonuç

Bu çalışmada Python programlama dili kullanılarak bir FTP client uygulaması geliştirildi. Uygulama, FTP server'a bağlanarak, sunucu üzerinde yer alan dosya ve dizinleri yönetmeyi sağlamaktadır.

Geliştirilen uygulama sayesinde, kullanıcılar yerel ve uzak sunucu üzerinde dosya ve dizin oluşturma, silme, adını değiştirme, dosya listeleme, uzak sunucudan dosya indirme ve yerel sunucuya dosya yükleme işlemlerini gerçekleştirebilmektedir.

Uygulamanın başarılı bir şekilde çalıştığı görülmüştür. Geliştirilen fonksiyonlar sayesinde kullanıcıların FTP sunucusu üzerinde daha hızlı ve verimli bir şekilde işlem yapmaları sağlanmaktadır.

Bunun yanı sıra, geliştirilen uygulama üzerinde daha fazla geliştirme yapılabilir. Örneğin, kullanıcı arayüzü daha kullanışlı hale getirilebilir, farklı protokolleri de destekleyecek şekilde genişletilebilir veya sunucu üzerinde dosya arama gibi ek özellikler eklenilebilir.

Sonuç olarak, Python programlama dili ve ftplib kütüphanesi kullanılarak geliştirilen bu FTP client uygulaması, kullanıcılara FTP sunucusu yönetiminde yardımcı olan bir araç sağlamaktadır.

Kaynakça

1. python ftp server: <https://serverpilot.io/docs/how-to-run-a-simple-ftp-server/>
2. python ftp client tutorial: <https://www.devdungeon.com/content/python-ftp-client-tutorial>
3. pyftplib: <https://pypi.org/project/pyftplib/>
4. ftplib: <https://docs.python.org/3/library/ftplib.html>
5. termcolor: <https://pypi.org/project/termcolor>
6. getpass: <https://docs.python.org/3/library/getpass>
7. os: <https://docs.python.org/3/library/os>

EKLER

EK-1 FTP Server Kodları

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

# create a user for login
authorizer = DummyAuthorizer()
authorizer.add_user("omer", "123", "files", perm="elradfmw")

# Instantiate FTP handler class
handler = FTPHandler
handler.authorizer = authorizer

# Instantiate FTP server class and listen on 0.0.0.0:21
server = FTPServer(("0.0.0.0", 21), handler)

# set a limit for connections
server.max_cons = 256
server.max_cons_per_ip = 5

# start FTP server
server.serve_forever()
```

EK-2 FTP Client Kodları

```
from ftplib import FTP

import time

from termcolor import colored

import getpass

import os

ftp = FTP('')


def main():

    uzak = Uzak()
    os.system("clear")

    print(colored("Yapımcı: Omer Bayramcavus", "blue"))
    print(colored("Numara: 191180763", "blue"))
    print(colored("FTP Client", "blue"))
    print(colored("Sunucuya Bağlan", "blue"))
    uzak.connect()
    print(colored("Sunucuya Giriş Yap", "blue"))
    uzak.login()
    common = Common()
    yerel = Yerel()
    os.system("clear")
    donguBilgi()
    while True:
        secim1 = input("Seçim: ")
        if secim1 == "1":
            os.system("clear")
            yerelBilgi()
        elif secim1 == "2":
            os.system("clear")
            uzakBilgi()
```

```

elif secim1 == "3":
    yerel.dosyaListeleme()
    uzak.dosyaListeleme()
    common.dosyaYukle()

elif secim1 == "4":
    yerel.dosyaListeleme()
    uzak.dosyaListeleme()
    common.dosyaIndir()

elif secim1 == "5":
    print(colored("Çıkış Yapılıyor", "blue"))
    time.sleep(1)
    os.system("clear")
    break

else:
    print(colored("Hatalı Seçim", "red"))
    time.sleep(1)
    os.system("clear")
    donguBilgi()
    continue

while secim1 == "1":
    secim2 = input("Seçim: ")

    if secim2 == "1":
        yerel.dizinOlusturma()

    elif secim2 == "2":
        yerel.dizinSilme()

    elif secim2 == "3":
        yerel.dosyaAdiDegistirme()

    elif secim2 == "4":
        yerel.dosyaSilme()

    elif secim2 == "5":
        yerel.dizinDegistirmek()

    elif secim2 == "6":
        yerel.dosyaListeleme()

    elif secim2 == "7":
        os.system("clear")

```



```

        donguBilgi()

        break

    else:

        print(colored("Hatalı Seçim", "red"))

        time.sleep(1)

        os.system("clear")

        donguBilgi()

        break

while secim1 == "2":

    secim2 = input("Seçim: ")

    if secim2 == "1":

        uzak.dizinOlusturma()

    elif secim2 == "2":

        uzak.dizinSilme()

    elif secim2 == "3":

        uzak.dosyaAdıDegistirme()

    elif secim2 == "4":

        uzak.dosyaSilme()

    elif secim2 == "5":

        uzak.dizinDegistirmek()

    elif secim2 == "6":

        uzak.dosyaListeleme()

    elif secim2 == "7":

        os.system("clear")

        donguBilgi()

        break

    else:

        print(colored("Hatalı Seçim", "red"))

        time.sleep(1)

        os.system("clear")

        donguBilgi()

        break

class Common:

```

```

def dosyaYukle(self):
    try:
        dosyaadi = input("Yüklenecek Dosya Adı: ")
        ftp.storbinary('STOR '+dosyaadi, open(dosyaadi, 'rb'))
        print(colored("Dosya Yüklendi", "green"))
        time.sleep(1)
    except:
        print(colored("Dosya Yüklenemedi", "red"))
        time.sleep(1)
    finally:
        os.system("clear")
        donguBilgi()

def dosyaIndir(self):
    try:
        dosyaadi = input("İndirilecek Dosya Adı: ")
        dosya = open(dosyaadi, 'wb')
        ftp.retrbinary('RETR ' + dosyaadi, dosya.write, 1014)
        print(colored("Dosya İndirildi", "green"))
        time.sleep(1)
    except:
        print(colored("Dosya İndirilemedi", "red"))
        time.sleep(1)
    finally:
        os.system("clear")
        donguBilgi()

#yerel

class Yerel:
    def dizinOlusturma(self):
        try:
            dizinadi = input("Dizin Adı: ")
            os.mkdir(dizinadi)

```

```

        print(colored("Dizin Oluşturuldu", "green"))

        time.sleep(1)

    except:

        print(colored("Dizin Oluşturulamadı", "red"))

        time.sleep(1)

    finally:

        os.system("clear")

        yerelBilgi()

def dizinSilme(self):

    try:

        dizinadi = input("Dizin Adı: ")

        os.rmdir(dizinadi)

        print(colored("Dizin Silindi", "green"))

        time.sleep(1)

    except:

        print(colored("Dizin Silinemedi", "red"))

        time.sleep(1)

    finally:

        os.system("clear")

        yerelBilgi()

def dosyaAdıDegistirme(self):

    try:

        eskiDosyaAdi = input("Eski Dosya Adı: ")

        yeniDosyaAdi = input("Yeni Dosya Adı: ")

        os.rename(eskiDosyaAdi, yeniDosyaAdi)

        print(colored("Dosya Adı Değiştirildi", "green"))

        time.sleep(1)

    except:

        print(colored("Dosya Adı Değiştirilemedi", "red"))

        time.sleep(1)

    finally:

        os.system("clear")

        yerelBilgi()

```

```

def dosyaSilme(self):
    try:
        dosyaadi = input("Dosya Adı: ")
        os.remove(dosyaadi)
        print(colored("Dosya Silindi", "green"))
        time.sleep(1)
    except:
        print(colored("Dosya Silinemedi", "red"))
        time.sleep(1)
    finally:
        os.system("clear")
        yerelBilgi()

def dizinDegistirmek(self):
    try:
        dizin = input("Dizin Adı (Geri Dönmek için .. yazınız): ")
        os.chdir(dizin)
        print(colored("Dizin Değiştirildi", "green"))
        time.sleep(1)
    except:
        print(colored("Dizin Değiştirilemedi", "red"))
        time.sleep(1)
    finally:
        os.system("clear")
        yerelBilgi()

def dosyaListeleme(self):
    try:
        print(colored("Yerel Dizin:", "blue"))
        os.system("ls")
    except:
        print(colored("Dosya Listelenemedi", "red"))

```

```

        time.sleep(1)

#uzak

class Uzak:

    def connect(self):

        try:

            url = input("Sunucu Adresi: ")

            port = input("Sunucu Portu: ")

            ftp.connect(url, int(port))

            print(colored("Bağlantı Başarılı", "green"))

        except:

            print(colored("Bağlantı Başarısız...\nTekrar Deneyin", "red"))

            time.sleep(1)

            self.connect()

    def login(self):

        try:

            name = input("Kullanıcı Adı: ")

            password = getpass.getpass("Şifre: ")

            ftp.login(name,password)

            print("Giriş Başarılı")

            time.sleep(1)

        except:

            print(colored("Giriş Başarısız...\nTekrar Deneyin", "red"))

            time.sleep(1)

            self.login()

    def dizinOlusturma(self):

        try:

            dizinadi = input("Dizin Adı: ")

            ftp.mkd(dizinadi)

            print(colored("Dizin Oluşturuldu", "green"))

            time.sleep(1)

```

```

except:

    print(colored("Dizin Oluşturulamadı", "red"))

    time.sleep(1)

finally:

    os.system("clear")

    uzakBilgi()

def dizinSilme(self):

    try:

        dizinadi = input("Dizin Adı: ")

        ftp.rmd(dizinadi)

        print(colored("Dizin Silindi", "green"))

        time.sleep(1)

    except:

        print(colored("Dizin Silinemedi", "red"))

        time.sleep(1)

    finally:

        os.system("clear")

        uzakBilgi()

def dosyaAdıDegistirme(self):

    try:

        eskiDosyaAdi = input("Eski Dosya Adı: ")

        yeniDosyaAdi = input("Yeni Dosya Adı: ")

        ftp.rename(eskiDosyaAdi, yeniDosyaAdi)

        print(colored("Dosya Adı Değiştirildi", "green"))

        time.sleep(1)

    except:

        print(colored("Dosya Adı Değiştirilemedi", "red"))

        time.sleep(1)

    finally:

        os.system("clear")

        uzakBilgi()

def dosyaSilme(self):

```

```

try:
    dosyaadi = input("Dosya Adı: ")
    ftp.delete(dosyaadi)
    print(colored("Dosya Silindi", "green"))
    time.sleep(1)

except:
    print(colored("Dosya Silinemedi", "red"))
    time.sleep(1)

finally:
    os.system("clear")
    uzakBilgi()

def dosyaListeleme(self):
    try:
        print(colored("Uzak Dizin:", "blue"))
        ftp.dir()
    except:
        print(colored("Dosya Listelenemedi", "red"))
        time.sleep(1)

def dizinDegistirmek(self):
    try:
        dizin = input("Dizin Adı: ")
        ftp.cwd(dizin)
        print(colored("Dizin Değiştirildi", "green"))
        time.sleep(1)
    except:
        print(colored("Dizin Değiştirilemedi", "red"))
        time.sleep(1)
    finally:
        os.system("clear")
        uzakBilgi()

```

```
def donguBilgi():  
    print(colored("FTP Client", "blue"))  
    print(colored("1) Yerel İşlemler", "blue"))  
    print(colored("2) Uzak Sunucu İşlemler", "blue"))  
    print(colored("3) Dosya Yükle", "blue"))  
    print(colored("4) Dosya İndir", "blue"))  
    print(colored("5) Çıkış Yap", "red"))  
  
def yerelBilgi():  
    print(colored("Yerel İşlemler", "blue"))  
    print(colored("1) Dizin Oluştur", "blue"))  
    print(colored("2) Dizin Sil", "blue"))  
    print(colored("3) Dosya Adı Değiştir", "blue"))  
    print(colored("4) Dosya Sil", "blue"))  
    print(colored("5) Dizin Değiştir", "blue"))  
    print(colored("6) Dosya Listele", "blue"))  
    print(colored("7) Ana Menüye Dön", "blue"))  
  
def uzakBilgi():  
    print(colored("Uzak Sunucu İşlemler", "blue"))  
    print(colored("1) Dizin Oluştur", "blue"))  
    print(colored("2) Dizin Sil", "blue"))  
    print(colored("3) Dosya Adı Değiştir", "blue"))  
    print(colored("4) Dosya Sil", "blue"))  
    print(colored("5) Dizin Değiştir", "blue"))  
    print(colored("6) Dosya Listele", "blue"))  
    print(colored("7) Ana Menüye Dön", "blue"))  
  
main()
```