

# OTOPARK OTOMASYONU

**AD/SOYAD:** NADİRE NUR SAĞLAM

**NUMARA:** 21100011042

**ÖDEV İÇİNDEKİ DOSYALAR:**

- 21100011042.txt
- 21100011042.py
- 21100011042.pdf

Yapılan proje otopark otomasyonudur. Bu otomasyon;

- Müşteri ekleme ()
- Müşteri arama ()
- Müşteri çıkış saati güncelleme ()
- Müşteri kaydı silme ()
- Müşterileri listeleme ()
- Müşteri borcu hesaplama ()
- Aylık abonman ödeme ()
- Ana fonksiyon ()
- Çıkış ()

Fonksiyonlarından oluşmaktadır.

Ana fonksiyon içinde menü bulunmaktadır. Kullanıcı menüden istediği seçeneği seçerek seçilen fonksiyona gidilerek işlem yapılmaktadır.

‘Müşteri ekle’ fonksiyonunda, kaydolan araçlara random bir sıra numarası atanmıştır. Bu random sıra numaralarının eşsiz olup olmadığı boş küme oluşturularak kontrol ettirilmiştir. Sıra numarası liste içinde bulunmuş ardından boşluklara göre split () metodu ile listeye dönüştürülmüştür. Daha sonra bu listenin 1. Elemanında sıra numarası bulunduğu için 1. Eleman split () ile tekrar başka bir listeye dönüştürülmüştür. Küme eleman sayısı ile en baştaki listenin eleman sayısı farklı olduğunda döngüden çıkılmıştır. Kullanıcıdan araç bilgileri alınmıştır. Kullanıcı kayıt yaptırırken aylık abonman yaptırmak isterse çıkış ve giriş saatleri bilgileri kullanıcıdan alınmamıştır. Eğer abonman yaptırmadıysa çıkış ve giriş saati bilgileri de kullanıcıdan alınmıştır. Alınan tüm bilgiler abonman olup olmama durumuna göre ‘a’ modu ile dosyaya yazdırılmıştır.

‘Müşteri ara’ fonksiyonunda, dosya ‘r’ modu ile okunmak için açılmıştır. Dosya elemanları readlines() metodu ile listeye dönüştürülmüştür. İcfonk() fonksiyonunda müşteriler, kullanıcının sıra numaralarını görebilmeleri için , sıralanmışlardır. Bir üst fonksiyon olan muster\_ara() fonksiyonundaki ‘liste’ kullanılabilmek için nonlocal olarak çağırılmıştır. Daha sonra liste içinde gezilerek aranan müşterinin sıra numarası bulunmuştur.

‘Güncelle’ fonksiyonunda, aranan sıra numarası liste içinde gezilerek bulunmuştur. Ardından bulunan kişinin bilgilerinin olduğu, işlem yapabilmek için satır split() metodu ile ayrı bir liste haline getirildi. Eğer müşterinin abonmanı yoksa sıra numarası liste içinde aranıp bulundu. Güncel liste içeriğinin bozulmaması için kopyalandı. Ve güncel listeden aranan müşteri bilgilerinin

bulunduğu satır silindi. Ardından kullanıcıdan güncellenecek olan çıkış saati alındı. Kopya liste içinde aranan sıra numarasının bulunduğu satır bulundu. Satır split() metodu ile ayrı bir liste haline getirildi. Split () ile oluşturulan listenin son elemanı alınarak güncellendi. Güncellenmiş haliyle split () ile ayrılan liste join () metodu ile tekrar birleştirilip karakter dizisi haline getirildi. Ardından kopya listedeki güncellenmemiş satır silindi. Kopya listeye join () metodu ile birleştirilen karakter dizisi eklendi. Son olarak 'w' modu ile dosyaya yazdırıldı.

'Kayıt sil' fonksiyonunda, tüm müşteriler dosyadan okunduktan sonra silinecek olan sıra numarası listede aranıp bulundu ardından o satır remove () metodu ile silindi. Ve listenin son hali writelines () metodu ile dosyaya yazdırıldı.

'Müşterileri listele' fonksiyonunda, dosyanın içinde for döngüsü ile gezilip liste elemanları yazdırıldı.

'Borç hesapla' fonksiyonunda, müşteriler yazdırıldı. Sözlük içinde kampanyalar tutuldu. Güncelle () fonksiyonunda olduğu gibi aranan satır split () metodu ile yeni bir listeye dönüştürüldü. Listedeki, müşterinin abonmanı bulunup bulunmaması bilgisi kontrol edildi. Borcu hesaplanacak müşteri liste içinde bulunduktan sonra o satır split () ile ayrı bir listeye dönüştürüldü ve çıkış ve giriş saatleri değerleri alınıp kalınan saat bulundu. Kalınan süre kampanyalara dahil olup olmamasına göre borç hesaplandı. Müşterinin abonmanı varsa borç hesaplanması için abonmanode() fonksiyonuna gidildi.

'Abonman öde' fonksiyonunda, abonmaniptal () fonksiyonu kullanıldı. Kullanıcı abonmanını iptal etmek isterse bu fonksiyona gidildi. Müşteri abonman iptal etmek isterse kaldığı gün kadar ücret ödemesi sağlandı. Abonman iptal edildikten sonra diğer fonksiyonlarda hata olmaması için müşteri kaydı silindi.

'Çıkış' fonksiyonunda, kullanıcı isterse break komutu ile otomasyondan çıkış sağlandı.

Otomasyonda koleksiyon veri tiplerinden sözlük, küme ve liste kullanıldı. Müşteri ekle, müşteri ara, güncelle, kayıt sil, borç hesapla, abonman öde fonksiyonlarında iç içe fonksiyon kullanıldı. Gerekli yerlerde while döngüsü ya da try-except yapısı kullanılarak olası yanlış kullanımlar engellenmeye çalışıldı.