

1. Projenin Amacı

Bu projenin amacı, bir kafenin temel işleyişini (**OOP**) prensiplerini kullanarak modellemektir. Sistem üzerinden ürünler listelenebilir, sipariş oluşturulabilir, siparişe ürün eklenebilir, ödeme alınabilir ve siparişler kaydedilebilir.

2. Kullanılan Teknolojiler

- Java (JDK)** •
- Object Oriented Programming (OOP)** •
- Strategy Design Pattern (Ödeme işlemleri için)** •
- Facade Pattern (CafeService)** •
- Serialization (Siparişlerin kaydedilmesi)** •
- CSV Dosyası (Ürün verileri)** •

3. Proje Yapısı

Proje katmanlı mimari kullanılarak geliştirilmiştir:

- model** •
 - order (Order, OrderItem, OrderStatus) ○
 - product (Product) ○
 - payment (PaymentMethod, CardPayment, CashPayment) ○
 - people (Customer, Person) ○
- service** •
 - CafeService ○
 - OrderService ○
 - InventoryService ○
 - KitchenWorker ○
- repository** •

```
CsvProductStore  o  
SerializationStore  o  
AppPaths  o  
util  •  
Input  o  
IdGenerator  o
```

Bu yapı, kodun okunabilirliğini ve sürdürülebilirliğini artırmak için tercih edilmiştir.

4. Sistem Çalışma Mantığı

Uygulama **konsol tabanlı** çalışmaktadır. Kullanıcıya aşağıdaki seçenekler sunulur:

Ürünleri listele .1

Sipariş oluştur .2

Siparişe ürün ekle .3

Ödeme yap .4

Siparişleri listele .5

Çıkış .6

Ödeme işlemleri **Strategy Pattern** kullanılarak kart veya nakit olarak gerçekleştirilir.

Ödeme tamamlandıktan sonra sipariş mutfak kuyruğuna eklenir.

```
C:\WINDOWS\system32\cmd. x + v
== Cafe Management System ==
1) Ürünleri Listele
2) Sipariş Oluştur
3) Siparişe Ürün Ekle
4) Ödeme Yap
5) Siparişleri Listele
6) Çıkış
Seçiminiz:
```

5. Veri Yönetimi

- Ürünler products.csv dosyasından okunur.
- Siparişler orders.ser dosyasına **serialization** yöntemiyle kaydedilir.
- Dosya yolları merkezi olarak AppPaths sınıfında tanımlanmıştır.

7. Sonuç

Bu proje sayesinde Java dilinde:

- OOP prensipleri
- Design Pattern kullanımı
- Dosya işlemleri
- Katmanlı mimari
- uygulamalı olarak öğrenilmiştir.