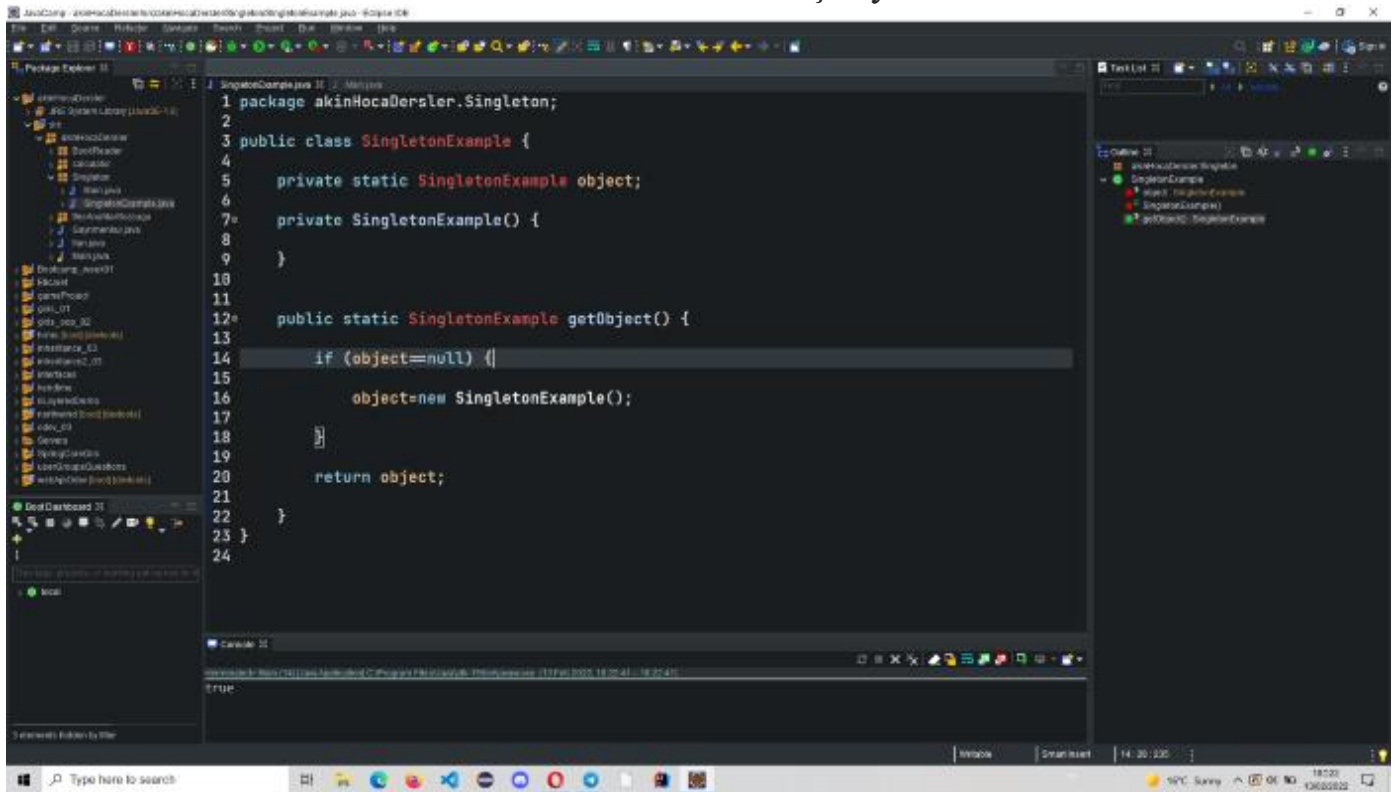


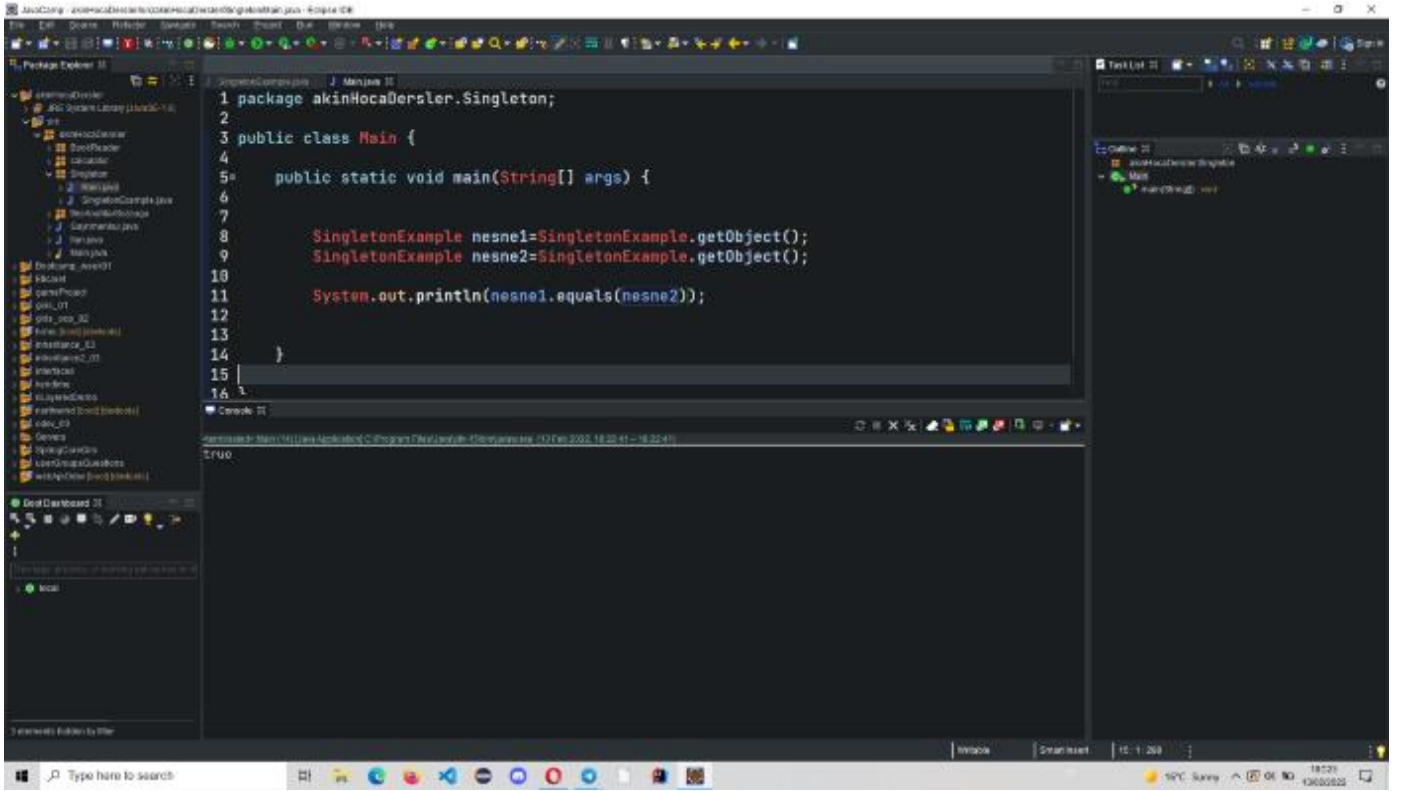
Spring Framework'un kullandığı tasarım desenleri

1-Singleton

Bu tasarım deseni bir classtan sadece bir tane instance yaratılmasını sağlar. Yani tekrar tekrar kullanılmak için oluşturulur. Daha öncesinde bir instance oluşmamışsa yeni bir tane üretilir. Spring bunu IoC container içerisinde gerçekleştiriyor yani bu işi Spring ele alıyor ve kontrol ediyor.

- Yapıcı metodu private yapılmalıdır.
- Static bir metod tanımlanmalıdır ve nesne boşsa yeni bir tane üretilir



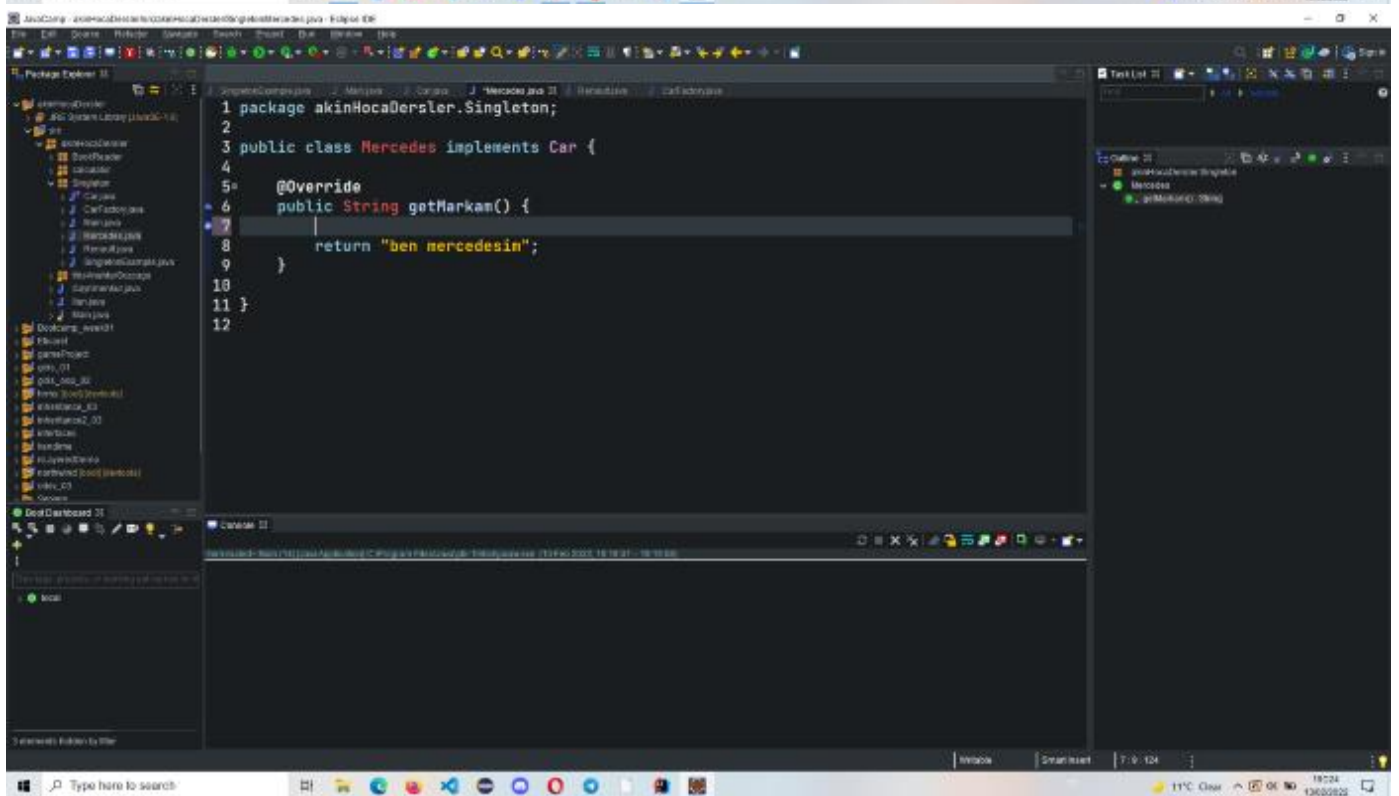
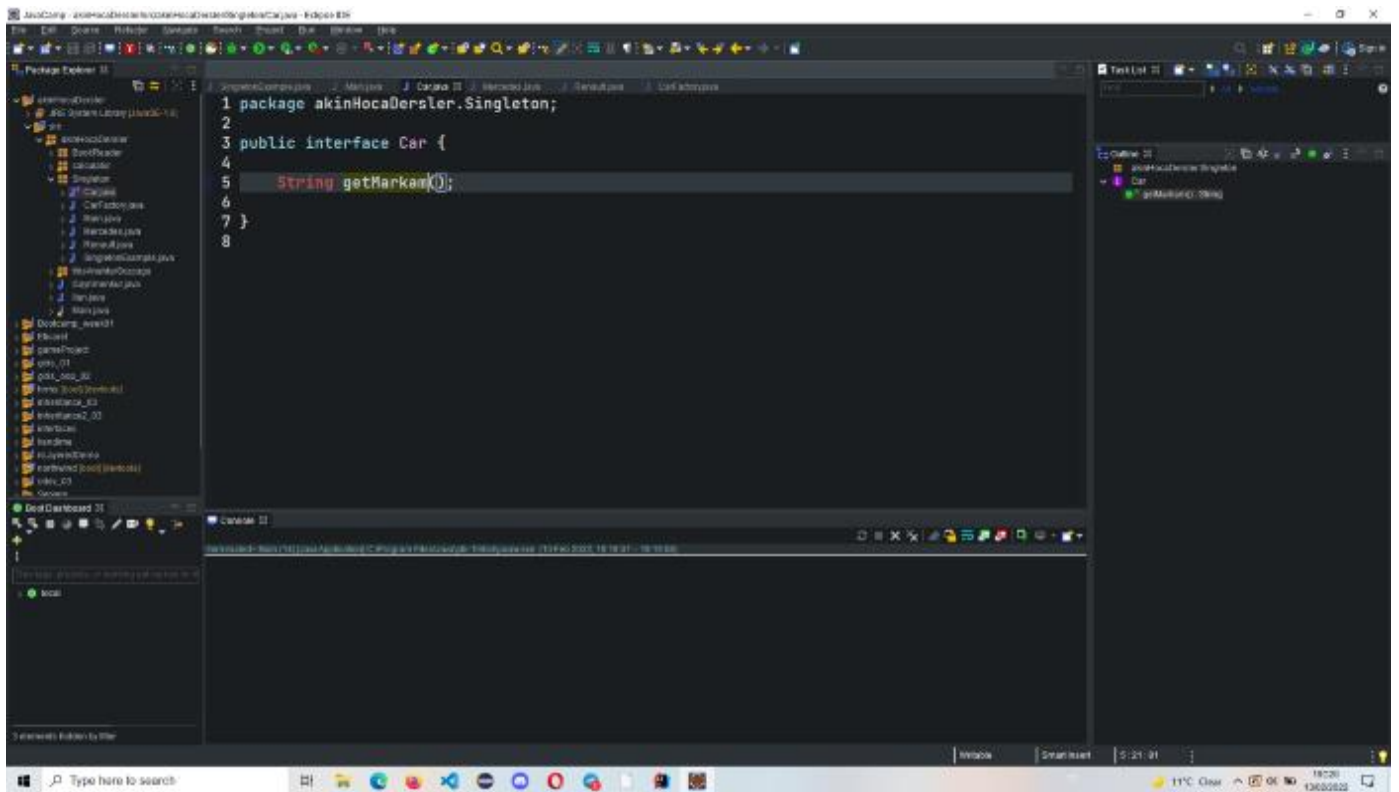


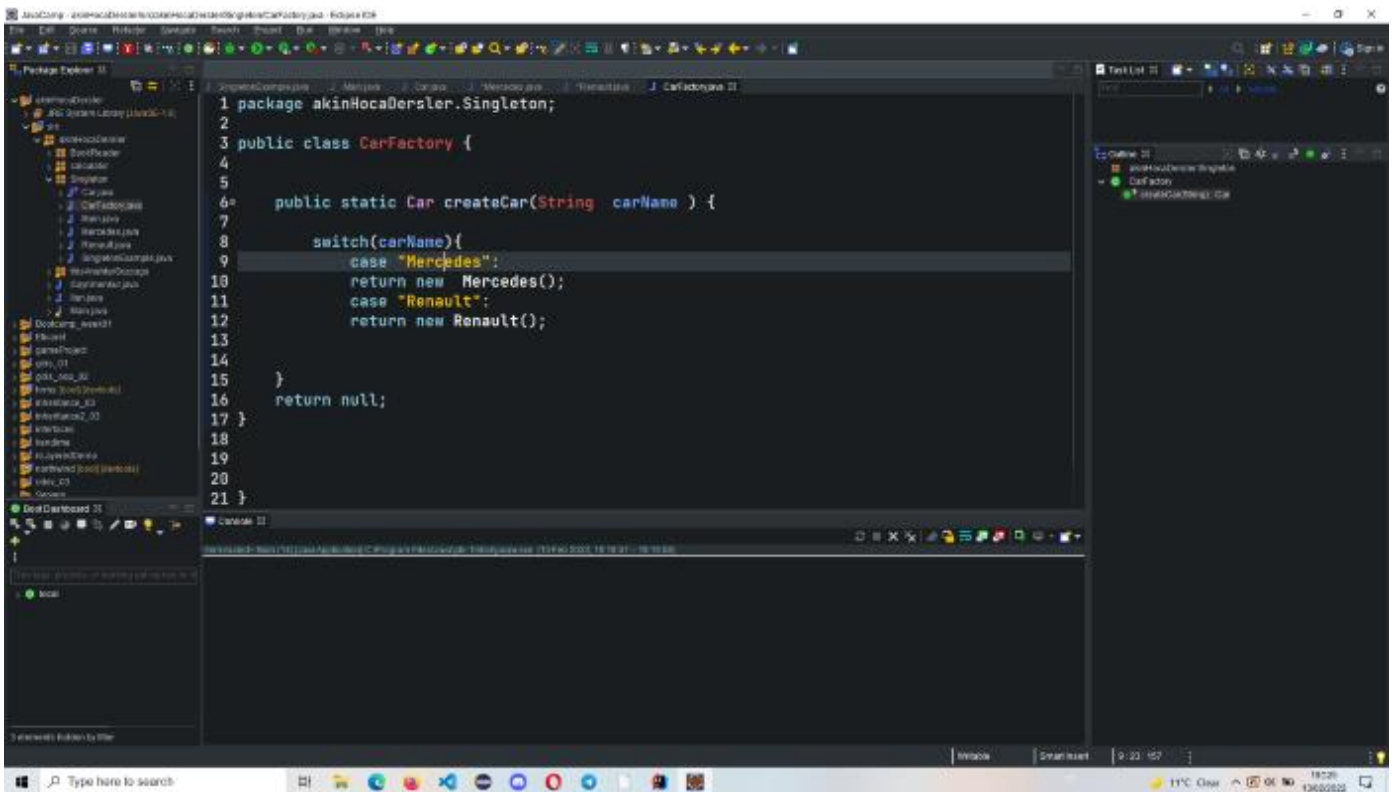
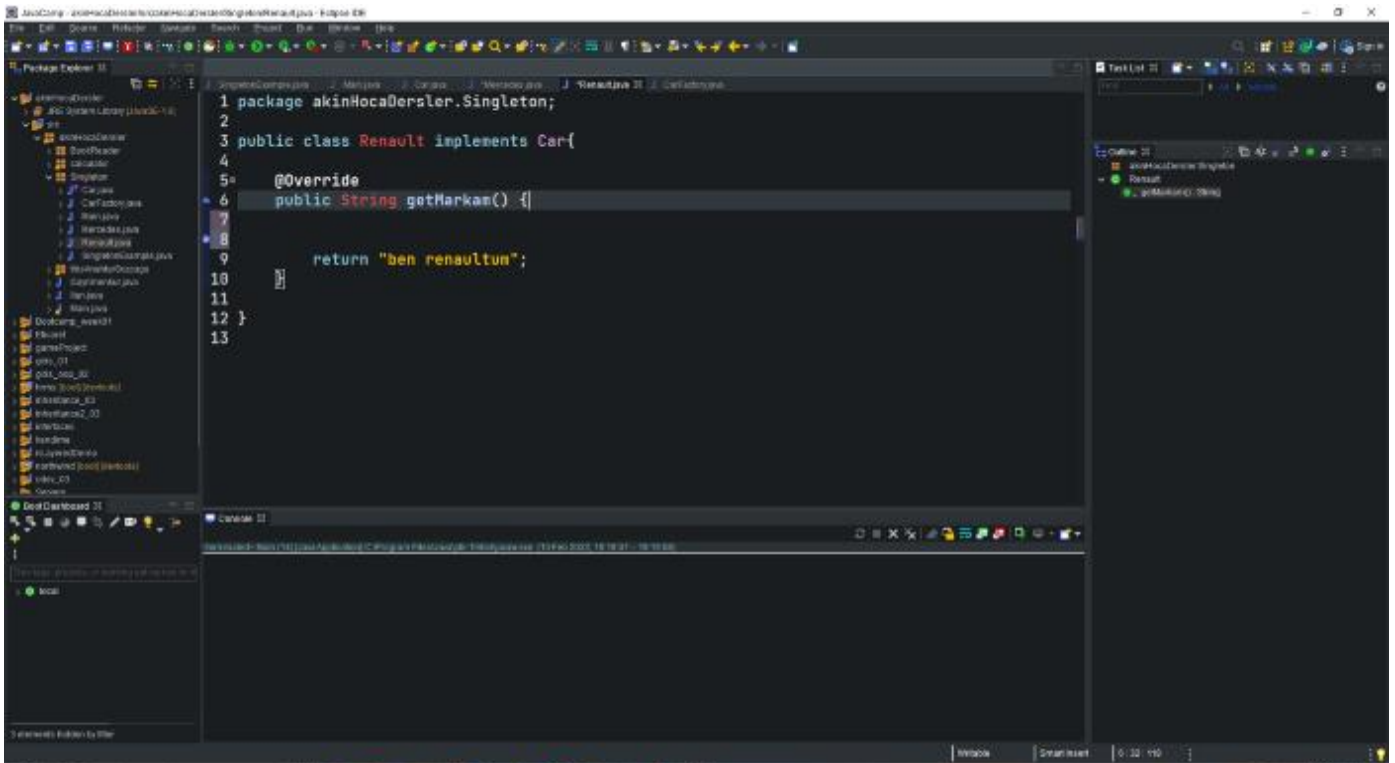
Burada görüldüğü üzere nesneler birbirine eşit olduğu görülmektedir. Çünkü yeni bir nesne üretimi olmadı.

2-Fabrika

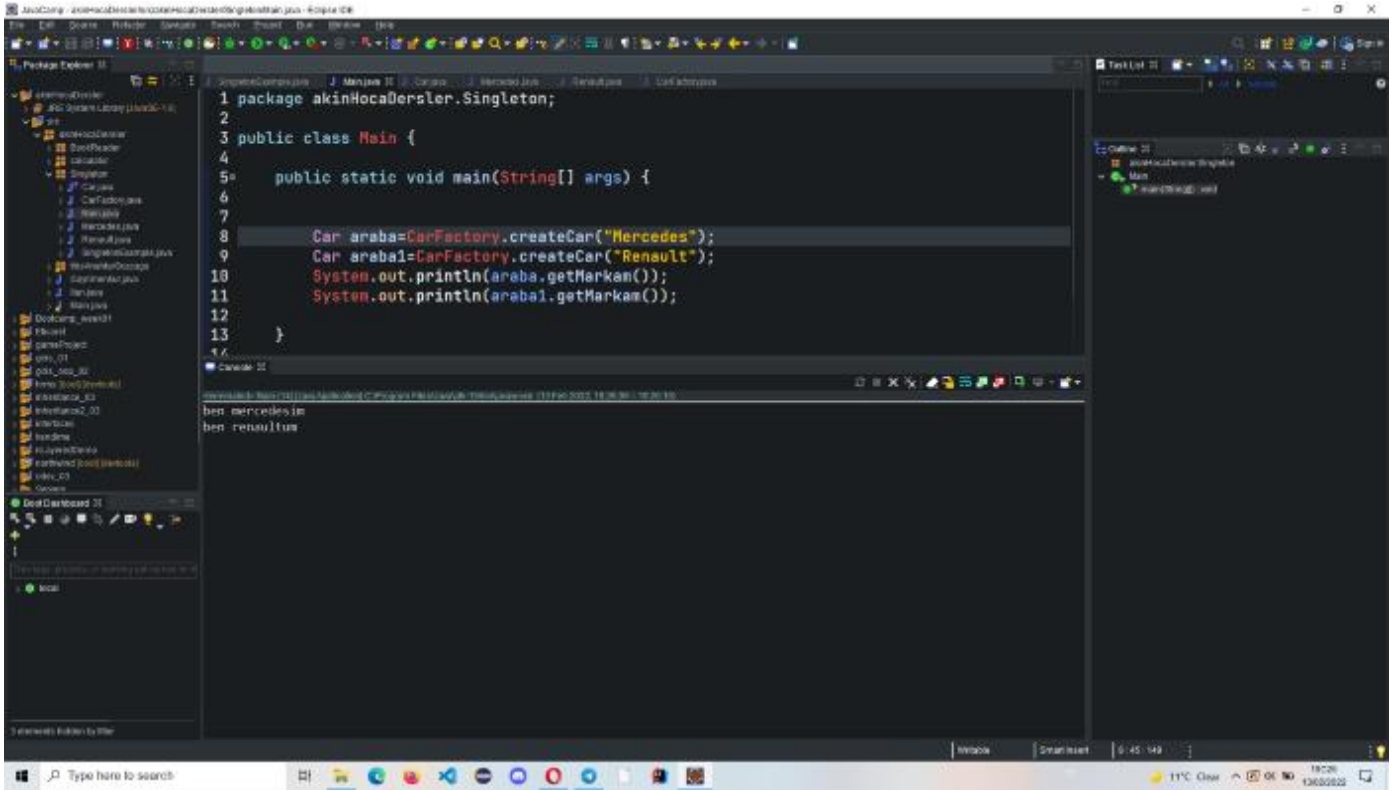
Fabrika tasarım deseninde nesne üretirken bunu soyutlama yöntemiyle bir üretim sınıfındaki üretim metoduna verme işlemidir. Birbirine benzeyen sınıfları her seferinde new anahtar kelimesini kullanmayalım mantığıyla tasarlanmıştır.

Bir Car interface'miz bulunmaktadır.





Yukaradaki sınıfımız nesnelerimizi verilen tipe göre üretme işini devralıyor.



Görüldüğü üzere verilen araba ismine göre arabalar üretilmektedir.

Burada soyutlama işleminden yararlanıldı.

3-Proxy

“Proxy tasarım deseni, oluşturduğumuz bir **Proxy** sınıf üzerinden, başka bir sınıfı temsil etmemizi sağlar. Böylelikle **Proxy** sınıf, esas sınıfa müdahale etmeden istediği işlem veya işlemler bütününe gerçekleştirebilir.”

4-Template

Bu tasarım deseni yaygın olarak kullanılan ve tekrarlanan kodlarla mücadele etmek için gerçekleştirilmektedir.**JdbcTemplate**, **JmsTemplate**, **JpaTemplate**

5-MVC

Bu tasarım kalıbında “model-view-controller” katmanlarımızda farklı görevler üstlenilmektedir. Model katmanında uygulamada kullanılacak somut nesnelerimizin bilgilerinin tutulduğu katmandır. View kısmında Geliştirilen projelerde kullanılan arayüz elemanlarıdır. Controller tarafında veritabanı işlemlerinin,bilgi alış-verişinin yapıldığı katmandır.

