

**Behavioral - Strategy Design Pattern** - Bir işlemi gerçekleştirmek için birden fazla yöntem(algoritma) mevcut olabilir bunları if else bloklarıyla çözmek yerine strategy design pattern kullanılır. Duruma göre bir yöntem seçip, uygulamak için strategy tasarım şablonu kullanılır.String tipinde bir değişkenimiz olduğunu düşünelim. String tipindeki değişkenimizi XML yada JSON şeklinde serialize ve deserialize yapmamız gereken bir senaryomuz olsun.Oluşturulması gerekenler;

1. BaseSerialize adlı bir interface tanımlanır ve içerisine void tipinde geri dönüşü olan ve içerisine String tipinde değer alan Serialize ve Deserialize adlı 2 method imzasını koyalım

```
interface SerializeStrategy{
    void Serialize(string value);
    void Deserialize (string value);
}
```

2. Context adlı bir class create edelim ve BaseSerialize interface'ini private olarak enjekte edelim. ExecuteSerialize-ExecuteDeserialize adlı Context sınıfına özel 2 adet method tanımlıyorum. Bu methodlar kullanılarak SerializeStrategy interface'i encapsule edilmiş olacak. Serializer işlemini yapacak olan asıl methodlarım Context sınıfı ile çalışacaklar

```
class Context{
    private SerializeStrategy _serializeStrategy;
    public Context(SerializeStrategy serializeStrategy)
    {
        _serializeStrategy = serializeStrategy;
    }

    public void executeSerialize(string value){
        _serializeStrategy.Serialize(value);
    }

    public void executeDeserialize(string value){
        _serializeStrategy.Deserialize(value);
    }
}
```

3. Bu algorithmada XmlSerializer yapması için ilgili class'ı ekliyorum

```
class XmlSerializer : SerializeStrategy
{
    public void Deserialize(string value)
    {
        Console.WriteLine("Value deserialized : {0}",value);
    }

    public void Serialize(string value)
    {
        Console.WriteLine("Value serialized : {0}",value);
    }
}
```

4. Artık main method içerisinden Context'i call edip String value'muzu Serialize - Deserialize edebiliriz

```
void Main()
{
    Context myContext = new Context(new XmlSerializer());
    myContext.executeSerialize("Hello from Serialize");
    myContext.executeDeserialize("Hello from Deserialize");
}
```

Not : Dikkat edildiği üzere tek interface üzerinden istediğimiz kadar algoritmik işlem yaptırabiliriz.JSON Serializer ihtiyacımız olduğunda sınıfı yazıp algoritmasını yazmamız yeterli olacaktır.Strategy pattern buralar da kullanılır.Bir diğer örnek ise bir text'in tamamen büyütülmesi tamamen küçültülmesi gibi iki farklı işlemimiz olduğunu düşünelim.Burada da strategy design pattern kullanılabilir