**Single Responsibility Prensibi**

Single Responsibility (Tek Sorumluluk) prensibi geliştirilen sınıfların tek bir görevi olması gerektiğini önermektedir. Yani geliştirilen bir sınıfın tek bir amacı olmalı ve o amaç doğrultusunda özellik ve fonksiyonlar barındırmalıdır. Geliştirilen bir sınıfta farklı amaç ve görevler var ise bu prensibe aykırı bir geliştirme süreci olduğu ortaya çıkmaktadır. Bunu engellemek için geliştirilen sınıf uygun şekilde görevlere uygun şekilde parçalanmalıdır.

Bir sınıfın birden fazla sorumluluk alması sınıflar arası bağlılığı artırmaktadır. Sınıflar arası bağlılık arttıkça kod karmaşıklığı artmaktadır. Uygulama geliştirme sürecinde gereksinimler değiştiği zaman bu gereksinimi karşılayacak kod bloğunda değişiklik yapmak yeterli olmalıdır. Ancak sınıflar arası bağlılık fazla olursa gereksinimi karşılamak yeterli olmayacaktır. Ek olarak bu değişimden etkilenen kod bloklarını da değiştirmek gerekecektir. Ancak eğer uygulama tek sorumluluk ilkesine uygun olarak tasarlanırsa gereksinim değişiklikleri diğer bölümleri etkilemeyecektir.

Tek sorumluluk ilkesi test, bağlılık ve anlışabilirlik özelliklerinde kolaylıklar sağlar. Tek sorumluluğu olan bir sınıfın test durumları ve sınıflar arası bağlılığı daha az olacaktır. Sınıfların tek bir görevi olduğu için daha yalın bir yapıda olacak ve bu kodun anlaşılabilirliğini artıracaktır. Özet olarak sorumluluğun azaltılması değişime daha kolay adapte olması anlamına gelmektedir.

**Uygulama Kodları**

**public class** Kullanici {  
 **private** String **isim**;  
 **private** String **mail**;  
 **private** Adres **adres**;  
 **private** String **sifre**;  
 **public void** setIsim(String isim) { **this**.**isim** = isim; }  
 **public** String getIsim() { **return isim**; }  
 **public** String getMail() { **return mail**; }  
 **public void** setMail(String mail) { **this**.**mail** = mail; }  
 **public void** setSifre(String sifre) { **this**.**sifre** = sifre; }  
 **public** String getSifre() { **return sifre**; }  
 **public void** adresGuncelle(Adres adres) { **this**.**adres** = adres; }  
 **public** String getAdres() { **return adres**.toString(); }  
 **public** String toString() {  
 **return** getIsim() + **" "** + getMail() + **" "** + getAdres();  
 }  
}

**public class** Adres {  
 **private** String **ulke**;  
 **private** String **il**;  
 **private** String **ilce**;  
 **private int postKodu**;  
 **public** String getUlke() { **return ulke**; }  
 **public void** setUlke(String ulke) { **this**.**ulke** = ulke; }  
 **public** String getIl() { **return il**; }  
 **public void** setIl(String il) { **this**.**il** = il; }  
 **public** String getIlce() { **return ilce**; }  
 **public void** setIlce(String ilce) { **this**.**ilce** = ilce; }  
 **public int** getPostKodu() { **return postKodu**; }  
 **public void** setPostKodu(**int** postKodu) { **this**.**postKodu** = postKodu; }  
 **public** String toString() {  
 **return** getUlke() + **" "** + getIl() + **" "** + getIlce() + **" "** + getPostKodu();  
 }  
}

**public class** Main {  
 **public static void** main(String[] args) {  
 Kullanici kullanici = **new** Kullanici();  
 kullanici.setIsim(**"Sefa Aras"**);  
 kullanici.setMail(**"sefaaras@ktu.edu.tr"**);  
 Adres adres = **new** Adres();  
 adres.setUlke(**"Turkiye"**);  
 adres.setIl(**"Trabzon"**);  
 adres.setIlce(**"Of"**);  
 kullanici.adresGuncelle(adres);  
 System.***out***.println(kullanici.toString());  
 adres.setPostKodu(61000);  
 kullanici.adresGuncelle(adres);  
 System.***out***.println(kullanici.toString());  
 }  
}