MYAZ203 – NESNE TABANLI PROGRAMLAMA LAB.

UYGULAMA 1

- *Eğitmen: Doç. Dr. Zafer CÖMERT*

E-posta: zcomert@samsun.edu.tr

- *Yardımcı Öğretim Elemanı: Arş. Gör. Furkancan DEMİRCAN*

E-posta: furkancan.demircan@samsun.edu.tr

Adım 1: OOP'nin Temelleri

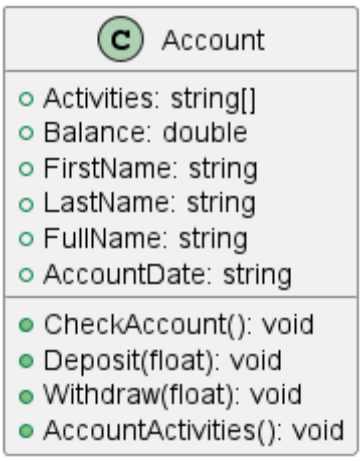
OOP, dört ana prensibe dayanmaktadır: kapsülleme (encapsulation), miras alma (inheritance), çok biçimlilik (polymorphism) ve soyutlama (abstraction).

- **Kapsülleme (Encapsulation):** Kapsülleme, verileri (nitelikler) ve metotları (işlevleri) bir birimde, yani sınıfta birleştirmeyi içerir. Bu, veri gizleme ve korumayı teşvik eder.
- **Miras Alma (Inheritance):** Miras alma, bir sınıfın (alt sınıf) diğer bir sınıftan (üst sınıf) özellikleri ve davranışlarını miras almasına olanak tanır, bu da kodun tekrar kullanımını ve genişletilebilirliği sağlar.
- **Çok Biçimlilik (Polymorphism):** Çok biçimlilik, nesnelerin, özgün davranışlarını korurken ana sınıflarının örnekleri gibi işlem görmesine izin verir.
- **Soyutlama (Abstraction):** Soyutlama, bir nesnenin sadece temel özelliklerini göstermeyi ve uygulama detaylarını gizlemeyi içerir.

Adım 2: Uygulama (Banka Sistemi)

OOP prensiplerini kullanarak basit bir banka sistemi tasarlayalım:

- C:/ sürücüsünde "MYAZ203" isimli bir klasör oluşturunuz.
  (*Hatırlatma*): mkdir …
- Oluşturulan klasör içerisinde yeni bir *konsol* uygulaması oluşturunuz.
- Banka hesabını temsil eden Account adında bir temel sınıfımız olacak.
- Yazmış olduğunuz banka sistemi müşterilere dört temel işlem sunabilmelidir. Bu işlemler; **Deposit** (para yatırma), **Withdraw** (para çekme), **CheckAccount** (banka hesap detaylarını gösterme) ve **AccountActivities** (hesap hareketleri).
- Account sınıfına ait diyagram aşağıda verilmektedir.



Şekil 1. Account sınıfına ait diyagram

Adım 3: Banka Sistemini Kullanma
Oluşturduğunuz banka sistemini Program.cs içinde kullanınız.

- Account sınıfından nesne oluşturunuz.
- Oluşturduğunuz nesneyi kullanarak işlemleri gerçekleştiriniz.

Beklenen İşlemler:

- ✓ Para çekme
- ✓ Para yatırma
- ✓ Hesap bilgilerini gösterme
- ✓ Hesap hareketlerini gösterme
- ✓ Balance değeri 0'dan küçük olamaz
- ✓ Çekilecek para 50'den küçük ve 10000'den büyük olamaz

Kazanımlar
Bu bölüm öğrencimiz tarafından doldurulacaktır.

- Class tanımlayabilirim.
- Field tanımlayabilirim.
- Property tanımlayabilirim.
- Auto-implemented property tanımlayabilirim.
- Expression-bodied accessors tanımı yapabilirim.
- Lambda ifadelerinin kullanımı ile property tanımı arasındaki ilişkiyi kurabilirim.
- Code snippet tanımı yapabilir, örnekler sunabilirim.

Github: https://github.com/FurkancanDemircan