Nesneye Yönelik Programlama
Final Proje Ödevi
(Son Teslim Tarihi: – 10.06.2020 – 23:59)

Bu projede yeni kurulan bir banka için C++ uygulama gerçekleştirmeniz beklenmektedir. Bu bankada Kısa ve Uzun Vadeli, Özel ve Cari hesap diye dört farklı türden hesap olmalıdır, ve her hesap türü aşağıda belirtilen kendine ait özelliklere sahip olmalıdır.

- ***Account*** sınıfı *abstract* olarak tanımlanmalıdır.
- Bu sınıftan üretilen her nesne bir banka hesabıdır.
- Bu sınıf şu fonksiyonlara sahip olmalıdır:
  - ***deposit*** : hesaba para yatırılmasına yarar.
  - ***withdraw*** : hesaptan para çekilmesine yarar.
    Bu işlem yapılırken hesap türüne ve mevcut bakiyeye dikkat edilmelidir.
    (Çekilebilecek miktar belirtilmeli, mevcut bakiyeden fazla çekilmeye çalışılırsa hata vermeli gibi özellikler düşünülmelidir.)
  - ***getBalance*** : hesabın anlık bakiyesini görüntülemeye yarar.
  - ***getID*** : hesap numarasını görüntülemeye yarar. (hesap numarası sabit tam sayıdır).
  - ***Benefit(kâr)*** : hesap açma günü ve işlem tarihine göre kar miktarı hesaplar.
    (yıllık kar r olursa günlük kar r/365 hesaplanır).

    programda 4 tür hesap için şu ***kurallar*** tanımlamalı:
    1. ***ShortTerm***: bu tür hesap yıllık %17 faiz verir ve en az 1000 TL hesapta bakiye olması gerekiyor.
    2. ***LongTerm***: bu tür hesap yıllık %24 faiz verir ve en az 1500 TL hesapta bakiye olmasi gerekiyor.
    3. ***Special***: bu tür hesap yıllık %12 faiz verir ve en az hesap açtığındakı kadar para hesapta bakiye olması gerekiyor.
    4. ***Current***: bu hesap faizsizdir ve hesapta para olma zorunluluğu yoktur.

Tüm sistemi kontrol etmesi için başka bir sınıf (***Bank)*** tanımlanmalıdır.
Bu sınıfta ***Account*** türünden bir pointer olmalı. (Account *accounts). Bu pointer tüm hesapların adresini tutar.
Bu sınıf şu fonksiyonlara sahip olmalıdır:

- ***getAccount*** : mevcut hesapların listesini görüntülemeye yarar.
- ***getDate***: sistemin şu anki tarihini görüntülemeye yarar.
- ***deposit(ID, cash)***: ID'si belirtilen hesaba, belirtilen miktarda para yatırma işlemi yapmaya yarar.
- ***Withdraw(ID, cash):*** eğer mümkün ise, belirtilen miktarda hesaptan para çekme işlemi yapar.
  (transaction işleminin başarılı ya da başarısız olması gibi durumlar kullanıcıya bildirilir.)
- ***sortition:*** Özel hesaplar arasında çekiliş yapmaya yarar. Özel hesaplarda her 2000 tl için bir puan tanımlanmalıdır ve çekilişler bu puanlar üzerinden yapılmalıdır. (Yüksek puana sahip olan hesabın çıkma olasılığı yüksektir.)
  - ***kurada çekilen hesap için 10000 TL ödül, hesabına yatırılır.***

ÖNEMLİ NOT: main içerisinde **tek bir nesne olmalıdır ve bu nesne** "***Bank"*** sınıfından üretilmelidir.

Program kullanıcıya konsoldan aşağıdaki işlemleri yapabilmesini sağlamalıdır:

1. **Create_S_ID_balance**: Kısa vadeli hesap açar. (hesap no=ID, başlangıç parası = balance)
2. **Create_L_ID_balance**: Uzun vadeli hesap açar. (hesap no=ID, başlangıç parası = balance)
3. **Create_O_ID_balance**: Özel hesap açar. (hesap no=ID, başlangıç parası = balance)
4. **Create_C_ID_balance**: Cari hesap açar. (hesap no=ID, başlangıç parası = balance)
5. **Increase_ID_cash**: ID'ye ait hesaba "cash" kadar para ekler.
6. **Decrease_ID_cash**: ID'ye ait hesaptan "cash" kadar para çeker.
7. **Set_dd_mm_yy**: Sistemin şu anki tarihini düzenler.
8. **ShowAccount**: Tüm hesapların ID'sini ve yaptığı son 5 işlemi gösterir.
9. **ShowIDs**: Sistemdeki tüm hesap numaralarını listeler.
10. **Sortition**: Özel hesaplar için kura çeker ve sonucu gösterir.

NOT:

- Sistemin başlangiç tarihi (default) = 01.01.2020
- Sınıflara istediğiniz ek özellikleri(değişkenler ve methodlar) ekleyebilirsiniz.
- Hesap bakiyeleri tam sayı(int) olmak zorundadır.

EK KURALLAR (DİKKATLİCE OKUYUNUZ) !!

- Projenin C++ dilinde yazılması ve OOP(Object-Oriented Programming) kullanılması zorunludur.
- Proje **"Program + Rapor"** 'dan oluşmaktadır.
- Proje erken ya da belirtilen teslim tarihinden sonra teslim edilmemelidir.
- Proje için bir **repl.it hesabı oluşturarak** projeyi bu site üzerinden yazmanız gerekmektedir. Repl.it üzerinden yazılmayan ve çalışmaz halde gönderilen projeler düşük puan almanıza sebep olabilir. Projenizin çalıştığından emin olunuz!
- Proje puanları Final notu olarak değerlendirileceğinizden, kurallar konusunda hassas davranmanız beklenmektedir.
- Yüksek benzerlik tespit edilen projeler **"kopya"** olarak sayılacaktır.
- Proje ile ilgili bütün bilgiler açıkça belirtilmiş olup, burada yazılı olan bilgiler ile ilgili lütfen mail atmayınız.

Ödev Bireysel ya da 2 kişilik gruplar halinde yapılabilir, grup olarak yapacak olan öğrenciler raporlarında gruplarını belirtmelidirler.
(LMS'e ayrı ayrı yüklemelisiniz, fakat maili tek 1 kişi göndermelidir)

Ödev Teslimi ve Sunumu

Ödev 2 farklı şekilde teslim edilecektir, her ikisini de yaptığınızdan emin olunuz!

1) Projeyi teslim edeceğiniz zaman **repl.it** üzerinden paylaş diyerek oluşturduğunuz **"Paylaşma Linki"**'ni ve **"OgrenciNo.pdf"** dosyasını **odev.blm1222@gmail.com** adresine mail olarak göndermeniz gerekmektedir.

Örnek; "https://repl.it/@edemirci/Test" + "190522123.pdf"

2) **Proje Dosyaları** ve **Rapor.pdf** dosyası "**tek bir .rar haline getirilip"** LMS sistemine yüklenmelidir.

Rapor dosyası şu yapıda olmalıdır;

Genel Özet: Projenizin işleyiş mantığını anlattığınız bir kısım olmalıdır.

Tanımlama: Projenizde oluşturduğunuz sınıf/nesne yapılarını, methodları ve değişkenleri açıklayıcı şekilde tanımlamanız gerekmektedir.

Çıktı: Projenizin çalışma yapısıyla ilgili bir kaç örnek ile test edilmiş ekran görüntülerinin raporun son kısmına eklenmesi gerekmektedir.