

به نام خدا

پروژه اول درس برنامه سازی پیشرفته

# مدیریت حساب بانکی



## مقدمه

در پروژه اول درس برنامه سازی پیشرفته ، به سراغ ساخت یک برنامه می رویم که بتواند روند یک سامانه مدیریت حساب بانکی را شبیه سازی کند و امکاناتی مانند افتتاح حساب ، انتقال وجه و موارد دیگر را در اختیار کاربران قرار دهد.

## توضیح بخش های برنامه

به طور کلی برنامه مدیریت حساب بانکی دو صفحه اصلی دارد :

### 1. صفحه ایجاد حساب بانکی :

در این صفحه کاربر می تواند با وارد کردن نام کامل صاحب حساب و رمز عبور حساب ، یک حساب بانکی ایجاد کند. سپس برنامه باید یک عدد چهاررقمی به عنوان شماره حساب تولید کرده و به کاربر نیز نمایش دهد. پس از ایجاد حساب ، باید یک پیغام موفقیت آمیز به کاربر نشان دهید و او را به صفحه اول هدایت کنید.

### 2. صفحه مدیریت حساب بانکی :

کاربران پس از وارد کردن شماره حساب و رمز عبور خود باید بتوانند امکان انجام امور زیر را داشته باشند :

1. افزایش موجودی: کاربر بتواند در این بخش موجودی حساب خود را افزایش دهد.

2. انتقال وجه: کاربر بتواند بخشی از موجودی خود را به حساب بانکی دیگری منتقل نماید.

3. حذف حساب بانکی: کاربر بتواند حساب بانکی خود را حذف نماید.

4. خرید شارژ : کاربر باید بتواند شارژ موبایل خریداری نماید.

5. خروج از حساب: از حساب خارج و به صفحه اصلی برمی گردد.

## ساخت حساب بانکی

در این مرحله نام کامل صاحب حساب و رمز عبور حساب از کاربر گرفته می شود و حساب موردنظر ساخته می شود.

**نکته:** شماره حساب برای راحتی کار یک عدد 4 رقمی در نظر گرفته شود و بصورت رندوم تولید شود و به کاربر نیز نمایش داده شود.

**نکته:** رمز عبور حساب باید ترکیبی از حروف و اعداد انگلیسی باشد. در صورت نقض این مورد به کاربر تذکر داده می شود و رمزعبور جدیدی از کاربر گرفته می شود.

**نکته:** در هر مرحله باید یک گزینه برای بازگشت به صفحه قبل یا صفحه اول وجود داشته باشد.

**نکته:** نام کامل صاحب حساب باید شامل نام و نام خانوادگی باشد و این دو باید با فاصله (Space) از هم جدا شده باشند. ( استفاده از علامت هایی نظیر Underline و ... مجاز نیست.)

**نکته:** پس از وارد کردن شماره حساب ، باید بررسی شود که قبلا حسابی با شماره حساب وارد شده ساخته نشده باشد و متناسب با نتیجه به کاربر خطا نمایش دهد یا به مرحله بعد برود.

## ورود به حساب بانکی

در این مرحله شماره حساب به همراه رمز عبور حساب از کاربر گرفته می شود و وارد حساب موردنظر می شود.

**نکته:** پس از وارد کردن شماره حساب ، باید بررسی شود که حسابی با شماره حساب وارد شده وجود دارد یا خیر و متناسب با نتیجه از کاربر مجدد شماره حساب گرفته شود یا به مرحله بعد برود.

**نکته:** پس از وارد کردن رمز عبور اشتباه ، برنامه متوقف نمی شود و کاربر باید رمز عبور دیگری وارد کند.

**نکته:** در هر مرحله باید یک گزینه برای بازگشت به صفحه قبل یا صفحه اول وجود داشته باشد.

## صفحه مدیریت حساب

در بالای صفحه مقدار موجودی حساب نمایش داده می شود و کاربر می تواند یکی از گزینه های زیر را انتخاب کند.

### 1. افزایش موجودی

در این بخش کاربر می تواند با وارد کردن مقداری عددی ، آن مقدار را به موجودی حساب اضافه کند. پس از اتمام عملیات ، برنامه به صفحه مدیریت حساب برمی گردد. همچنین میتوانید مبالغ مشخصی را از پیش تعیین کنید تا کاربر بتواند آن را به موجودی اضافه نماید.

**نکته:** حساب به طور پیش فرض پس از ساخته شدن مقدار 10000 موجودی دارد.

**نکته:** مقدار وارد شده برای افزایش موجودی نمی تواند صفر یا منفی باشد.

**نکته:** باید یک گزینه برای بازگشت به صفحه قبل یا صفحه مدیریت حساب وجود داشته باشد.

### 2. انتقال وجه

در این بخش کاربر می تواند مقداری از موجودی خود را به حسابی دیگر انتقال دهد.

به این صورت که کاربر اول شماره حسابی که قصد انتقال به آن حساب را دارد وارد می کند. سپس در صورت وجود داشتن حساب مورد نظر ، نام کامل صاحب حساب برای کاربر نمایش داده می شود. در مرحله بعد از کاربر مقدار موجودی ای که قصد انتقال آن را دارد گرفته می شود و در آخر برای تایید، رمز عبور حساب (رمز عبور حساب انتقال دهنده) گرفته می شود و مقدار موجودی وارد شده به حساب موردنظر انتقال داده می شود. پس از اتمام عملیات برنامه به صفحه مدیریت حساب برمی گردد.

**نکته:** پس از وارد کردن شماره حساب ، باید بررسی شود که حسابی با شماره حساب وارد شده وجود دارد یا خیر و متناسب با نتیجه از کاربر مجدد شماره حساب گرفته شود یا به مرحله بعد برود.

**نکته:** پس از وارد کردن مقدار موجودی برای انتقال باید بررسی شود که مقدار وارد شده صفر ، منفی و یا کمتر از مقدار موجودی انتقال دهنده نباشد و متناسب با نتیجه از کاربر مجدد مقدار موجودی انتقالی گرفته شود یا به مرحله بعد برود.

**نکته:** پس از وارد کردن رمز عبور برای انتقال باید بررسی شود که رمز عبور وارد شده با رمز عبور حساب انتقال دهنده مطابقت دارد یا خیر و متناسب با نتیجه از کاربر مجدد رمز عبور گرفته شود یا به مرحله بعد برود.

**نکته:** در تمامی مراحل باید یک گزینه برای بازگشت به صفحه قبل یا صفحه مدیریت حساب وجود داشته باشد.

### 3. حذف حساب بانکی

در این بخش کاربر با وارد کردن رمز عبور حساب ، می تواند حساب را حذف کند. پس از اتمام عملیات ، برنامه به صفحه ساخت حساب و ورود به حساب بانکی بر می گردد.

**نکته:** پس از وارد کردن رمز عبور برای حذف حساب باید بررسی شود که رمز عبور وارد شده با رمز عبور حساب مطابقت دارد یا خیر و متناسب با نتیجه از کاربر مجدد رمز عبور گرفته شود یا به مرحله بعد برود.

### 4. خرید شارژ

پس از ورود کاربر به این بخش ، ابتدا اپراتورهای ارائه دهنده (همراه اول ، ایرانسل و ...) به کاربر نمایش داده شود . پس از انتخاب اپراتور ، مبلغ از پیش تعیین شده به کاربر نمایش داده شود . بعد از انتخاب این موارد ، مبلغ مورد نظر از حساب کاربر کسر شده و رمز شارژ به او نمایش داده شود. رمز شارژ یک عدد 20 رقمی بوده و باید در آن لحظه بصورت تصادفی تولید شود.

### 5. خروج از حساب

در این بخش کاربر می تواند با انتخاب این گزینه مستقیم از حساب خارج شده و به صفحه ساخت حساب و ورود به حساب بانکی برگردد.

## کلاس های اصلی پروژه

### 1. کلاس اصلی BankAccountManagement (Main)

- متدی برای ساخت حساب
- متدی برای ورود به حساب
- متدی برای افزایش موجودی
- متدی برای انتقال وجه
- متدی برای خرید شارژ
- متدی برای حذف حساب

### 2. حساب بانکی BankAccount

- شماره حساب (accountNumber)
- موجودی حساب (Balance)
- نام کامل صاحب حساب (Owner)
- رمز عبور (Password)

انتخاب متد های این کلاس برعهده خودتان.

**تذکر :** نیاز های اصلی پروژه حتما باید رعایت شوند اما جزئیات بر عهده خودتان می باشد . در مواردی که احساس می کنید روش یا پیاده سازیتان ابهامی دارد ، حتما با دستیاران آموزشی هماهنگ کنید.

## نمرات امتیازی:

مشاهده گزارش حساب (20) : کاربر بتواند با انتخاب این گزینه اطلاعات کاملی از حساب شامل تراکنش های اخیر(انتقال وجه ، واریز وجه ، خرید شارژ و ...) و همچنین زمان های ورود به حساب و خروج از حساب را مشاهده کند . مواردی مانند انتقال وجه باید در گزارش هر دو کاربر (فردی که انتقال می دهد و فردی که وجه به حساب او واریز می شود) نمایش داده شود. همچنین گزارش حساب باید در یک فایل جداگانه با نام مناسبی نوشته شود و در محل مناسبی قرار گیرد.

طراحی روند مناسبی جهت دریافت وام (20) : کاربر بتواند وام همراه با شرایط آن را مشاهده کند . سود وام ها به مبلغ اصلی آن اضافه شده و آن مبلغ قسط بندی شود . پس از دریافت وام ، مبلغ آن به موجودی اضافه شده و امکان پرداخت اقساط آن نیز برای کاربر فراهم باشد و پس از پرداخت قسط ، از موجودی کاربر به همان میزان کاسته شود . تعداد اقساط پرداخت شده و مانده برای کاربر مشخص باشد. سایر جزئیات مورد نیاز در این بخش را با نظر خودتان تکمیل کنید اما باید منطق درستی برای آن داشته باشید.

رعایت اصل کپسوله سازی شئی گرایی (10) : دسترسی به ویژگی های کلاس را به طور مستقیم برای کلاس های دیگر محدود کنید و متد هایی برای دسترسی و تغییر ویژگی های کلاس تعریف کنید.

مدیریت خطا و ورودی های غیرمعتبر (15) : خطاها به نحوی مدیریت شود که با وارد کردن ورودی های غیر معتبر (برای مثال وارد کردن String در بخشی که انتظار Integer دارید یا وارد کردن شماره 5 رقمی بجای شماره 4 رقمی شماره حساب) برنامه متوقف نشود ، پیغام متناسب برای کاربر نمایش داده شود و مجدد ورودی بگیرد.

ذخیره ، بروزرسانی و بارگذاری اطلاعات حساب از فایل (25) : برنامه هنگام شروع در صورت وجود اطلاعات را از فایل بخواند و در صورت ایجاد هرگونه تغییرات (ساخت حساب ، تغییر موجودی ، حذف حساب ... ) فایل را بروزرسانی کند.

عدم دسترسی متد main به رمزعبور حساب (10) : برای Login و هنگام بررسی رمزعبور حساب از روشی استفاده شود که رمزعبور حساب return نشود و متد main شما به طور مستقیم به رمزعبور دسترسی نداشته باشد. پیش نیاز این مورد ، نمره امتیازی رعایت اصل کپسوله سازی شئی گرایی میباشد.

برخی نکات لازم :

تمام فایل های مربوط به پروژه را در قالب یک فایل zip. در بخش مشخص شده در سامانه کوئرا آپلود کنید. توجه داشته باشید که نام فایل فشرده باید با فرمت project1-student id ارسال شود.

پیاده سازی پروژه تنها با زبان برنامه نویسی جاوا مورد قبول می باشد و استفاده از سایر زبان ها پذیرفته نیست.

مسئولیت اشکال در فایل ارسالی و یا عدم اجرای صحیح پروژه به هر شکلی به عهده دانشجو خواهد بود.

زمان تحویل پروژه پس از پایان مهلت ارسال آن اطلاع رسانی خواهد شد . دقت کنید که در زمان تحویل آخرین فایل ارسالی شما در کوئرا مورد بررسی قرار خواهد گرفت.

هم فکری و همکاری در پروژه اشکالی ندارد و حتی توصیه نیز می شود ولی پروژه ارسالی شما باید حتما توسط خود شما نوشته شده باشد . این نکته را در نظر بگیرید که شباهت بیش از حد دو پروژه ، سپردن یک بخش یا کل پروژه به دیگران و استفاده از کد دیگر دانشجویان برای انجام آن ، غیرمجاز بوده و در صورت محرز شدن تقلب با فرد خاطی برخورد خواهد شد.

کامنت نویسی و رعایت دندانه گذاری الزامی است.

موفق و پیروز باشید.

استاد سحر مکرمی

تیم دستیاران درس برنامه سازی پیشرفته – بهار 1403