بانک فارابی

مشخصات پروژه

عنوان : بانک فارابی

موضوع : پیاده سازی سیستم بانکی

زبان برنامه نویسی : جاوا

پروژه پایان ترم برنامه نویسی پیشرفته

مشخصات تیم پیاده ساز پروژه

نام تیم :DEVROID

اعضاع تیم: مهران آدووند- دانیال فرهنگی – حسین شعله رسا

استاد :

سرکارخانم دکتر موحدی

تاریخ:

تیر و مرداد ماه99

دانشکده مهندسی
پردیس فارابی
دانشگاه تهران

بسمه تعالی

نام پروژه : بانک فارابی
نام تیم : DEVROID

پروژه پایان ترم برنامه نویسی پیشرفته

❖ پیشگفتار:

ما درتیم Devroid درابتدا پیاده سازی پروژه را به سه قسمت پنل مدیریت،پنل کارمند وپنل مشتری تقسیم کردیم و سپس دیاگرام های UML نظیرهرکلاس را طراحی کردیم و بعد شروع به پیاده سازی کلاس ها و طراحی گرافیک کردیم و در نهایت کلاس ها را به گرافیک متصل کردیم.

لازم به ذکر است که تمام مراحله پیاده سازی پروژه به صورت دورکاری و بااستفاده از ابزار گیت وگیت لب انجام شده است و جلسات هماهنگی به صورت مجازی و در بستر اسکایپ انجام شد.

و درپایان ازدستیاران آموزشی درس برنامه نویسی پیشرفته کمال تشکر و قدردانی را داریم زیرا به تمام سوالات ما درطول پیاده سازی پروژه به خوبی پاسخ دادند.

بسمه تعالی

نام پروژه : بانک فارابی
نام تیم : DEVROID

پروژه پایان ترم برنامه نویسی پیشرفته

❖ روش حل مسئله:

1) در این پروژه سعی شد تا پکیج بندی وکلاس بندی به صورت مرتب پیاده سازی شود تا مشکل زدایی و رفع باگ های کلاس را راحت تر انجام شود و از طرف دیگر خوانایی کدها بهبود یابد.
2) ما در طول این پروژه تلاش کردیم تاجایی که امکان دارد از نوشتن کدهای تکراری پرهیز کنیم وباتوجه به مفاهیم ارث بری،چندریختی و اینترفیس تکرار کدها را به حداقل رساندیم.
3) برای کامنت گذاری در طول پروژه ازالگوی javadoc استفاده شده است تا در پایان کارخروجی مطلوب و مناسبی داشته باشد.
4) این پروژه چون برپایه برنامه نویسی شی گرا پیاده سازی شده است درتمامی برنامه از اشیای مختلف استفاده شده است وباتوجه به نیازبه تغییراین اشیا در قسمت های مختلف برنامه تصمیم گرفتیم که اشیا را درفایل های مناسب ذخیره سازی کنیم تا هر زمان که به آنها نیاز پیدا کردیم از فایل شی های مورد نظررا فراخوانی کنیم واین کار سبب شد تا عملکرد برنامه بالابرود و نیازی به ساخت اشیا تکراری نباشد.

5) اطلاعات کارمندان،مشتریان،حساب ها وتراکنش ها به صورت فایل **csv**. ذخیره سازی می شوند و باهرتغییر مجددا بروز رسانی می شوند.
6) برای انتقال وجه از حسابی به حساب دیگرازمسئله نخ ها کمک گرفته شده است و بامدیریت مناسب نخ ها تلاش کرده ایم از بروز خطا درهنگام انتقال پول بین حساب ها جلوگیری کنیم.

بسمه تعالی

نام پروژه : بانک فارابی
نام تیم : DEVROID

پروژه پایان ترم برنامه نویسی پیشرفته

❖ توضیحاتی درباره پیاده سازی کلی برنامه:

1) درابتدا دوکلاس User و Account تعریف شد که هرکدام یک سری پارامترها و متود های بنیادی را دارا هستند. کلاس های Client که همان مشتری بانک وEmployee از کلاس User ارث بری کرده اند و دارای ویژگی های خاصی هستند و از طرف دیگر انواع حساب های جاری ، بلندمدت و کوتاه مدت هم از کلاس Account ارث برده اند که هرکدام مکانیزیم خاص خود را دارند.
2) نحوی پیاده سازی مشتری به این ترتیب هست که ابتدا در بانک ثبت نام می کند و بعد می تواند به تعداد دلخواه حساب برای خود افتتاح کند یا از طریق مدیر ثبت نام می شود.
3) هر حساب و تراکنش یک آیدی منحصربه فرد دارند که مدیریت آنها را آسان ترمی کند این آیدی در زمان افتتاح حساب و یا انجام تراکنش به مشتری نشان داده می شود.

4) به دلیل اینکه در سرتاسر این پروژه نیاز به ذخیره اطلاعات ویا خواندن اطلاعات از فایل ها بود یک کلاس برای این منظور تهیه شده است که ازنوشتن کدهای تکراری جلو گیری می کند و از طرف دیگر چون درهرسه بخش مدیریت، کارمند و مشتری نیاز به جست و جو، نمایش لیست اطلاعات و تغییر ویا حذف آنها بود یک اینترفیس در کنار کلاس قبلی گذاشته شد تا باتوجه به انواع اطلاعات در پکیج های مناسب بازنویسی شود و این دو کلاس تشکیل پکیجی به نام Files را داده اند.

5) پنل مخصوص مدیریت در پکیجی به نام manager پیاده سازی شده است که شامل چندین کلاس ازجمله مدیریت حساب ها،کارمندان ومشتریان است که البته با ایجاد کلاسی به نام Management در همین پکیج از بازنویسی کدهای تکراری که دربخش های مدیریت مشتریان و کارمندان نیاز بود جلوگیری شد.این کلاس شامل متود هایی جهت احرازهویت مدیر،نمایش لیست اطلاعات به ترتیب حرف الفبا،امکان افزودن ویا حذف اطلاعات است.

6) در بخش های مختلف برنامه نیازبه sign up, log in است که این کار توسط کلاس Register که در پکیج user قرارگرفته است انجام می شود که این کار هم در راستای بهینه سازی برنامه انجام شد.

❖ توضیحاتی درباره پکیج Files :

این پکیج از اصلی ترین پکیج های برنامه میباشد که مدیریت فایل ها را برعهده دارد.

1) کلاس IO Files:

این کلاس از دو بخش writeTOFile که عمل نوشتن و ذخیره سازی در فایل ها را بر عهده دارد به این ترتیب که هر نوع شی در پوشه منحصر به خودش یک فایل csv. و در پوشه آبجکت ها یم فایل txt. که شامل آن شی می باشد،را می سازد.البته قبل از ذخیره سازی اطلاعات در فایل اگر فایل مورد نظر وجود داشت ابتدا چک می شود که اطلاعات تکراری نباشند و سپس ذخیره سازی صورت می گیرد.

2) Interface ChangeInfo :

این اینترفیس با توجه به اینکه کلاس های ما از جمله کارمند ، مدیر ، مشتری ، حساب از سه متد edit , search ,delete استفاده شده است ، در این اینترفیس امده است. در این کلاس های نام برده این سه متد override می شوند.

دلیل استفاده از این اینترفیس این بود که در اکثر قسمت های برنامه نیاز به جست و جو،حذف و یا ویرایش اطلاعات بود و از تکرار کد در موارد بسیاری جلو گیری کرد.

❖ توضیحاتی درباره پکیج user :

این پکیج شامل دو کلاس Register , User می باشد که به شرح زیر هستند:

3) کلاس User :

این کلاس اطلاعات کاربری هر یوزر رو دارا میباشد. با توجه به اینکه هر سه کلاس مشتری و مدیر و کارمند این ویژگی های مشترک از جمله اسم ، فامیل ، جنسیت ، تاریخ تولد ، تاریخ عضویت ، کد ملی، رمز عبور را دارا بودند به همین خاطر این کلاس ساخته شد تا با توجه به مبحث ارث بری و پلی مورفیسم ، مشتری و مدیر و کارمند ازین کلاس ارث ببرند.

4) کلاس Register :

این کلاس برای قسمت authentication برنامه میباشد که با متد signup ان کاربر مورد نظر را در برنامه ثبت نام می کند و login برای ورود کاربر به برنامه استفاده می شود. با توجه به اینکه مدیر و کارمند و مشتری هر کدام فایل جدایی برای اهراز هویت دارند به همین دلیل ویژگی filename را دارند.

همینطور متد check برای این منظور به کار می رود که بررسی بکند ک ایا این فرد قبلا وجود دارد یا نه که در موقع ثبت نام از ایجاد کاربران یکسان جلوگیری کند.

بسمه تعالی

نام پروژه : بانک فارابی
نام تیم : DEVROID

پروژه پایان ترم برنامه نویسی پیشرفته

❖ توضیحاتی درباره پکیج مدیر :

مدیر با توجه به فایل ManagerAuth.csv اهراز هویت کرده و وارد برنامه می شود با توجه به اینکه مدیریت تمامی جزء به جزء فرایند های مشتری و کارمندان و حساب ها و تراکنش ها با مدیر می باشد این پکیج درست شده است تا این موارد مدیریت شوند.

❖ کلاس AccountManager :

1) برای مدیریت حساب ها استفاده می شود به این صورت که اسم فایل ذخیره شده برای خواندن حساب از در ان دخیره می شود و همچنین ویژگی یک لیستی از حساب ها برای ذخیره تمامی حساب ها از فایل مورد نظر می باشد .
2) متد newAccount با گرفتن یک حساب ، حساب مورد نظر را در فایل ان write میکند.
3) متد getAccounts تمامی حساب ها رو به صورت سورت شده بر اساس اسم صاحب حساب نمایش می دهد . که برای نمایش لیست حساب ها به مدیر و کارمند استفاده می شود.
4) متد search که با implement کردن از interface ChangeInfo با گرفتن شماره حساب ، حساب مورد نظر را بر می گرداند.
5) متد delete نیز به صورت بالا کار میکند با این تفاوت که ابتدا اکانت مورد نظر را از لیست حساب ها حذف کرده ، فایل را خالی کرده و لیست جدید را درون فایل ذخیره می کند.
6) متد edit ابتدا فایل قبلی را حذف کرده و سپس حساب تغییر یافته جدید را درون فایل با بقیه حساب ها درون فایل ذخیره می کند.

❖ کلاس Management :

این کلاس به منظور مدیریت بر کارمندان و مشتری ها استفاده می شود.

1) Authentication
2) متد newUser با گرفتن یک شی کاربر ،کاربر مورد نظر را با کمک کلاس Register ، کاربر را ثبت نام می کند.
3) متد getUsers تمامی کاربرها رو به صورت سورت شده بر اساس فامیل نمایش می دهد . که برای نمایش لیست کاربر ها در مدیر و کارمند و مشتری استفاده می شود.
4) متد search که با implement کردن از interface ChangeInfo با گرفتن کد ملی، کاربر مورد نظر را بر می گرداند.
5) متد delete نیز به صورت بالا کار میکند با این تفاوت که ابتدا کاربر مورد نظر را از لیست کاربر ها حذف کرده ، فایل را خالی کرده و لیست جدید را درون فایل ذخیره می کند.
6) متد edit ابتدا فایل قبلی را حذف کرده و سپس اطلاعات کاربر تغییر یافته جدید را درون فایل با بقیه کاربرها درون فایل ذخیره می کند.

❖ کلاس Time :

این کلاس برای کنترل تایم زمانی برنامه ساخته شده است. برای ازمایش کردن ویژگی هایی که به گدر زمان نیاز دارند مانند انتقال وجه از حساب های بلند مدت مجبور به تغییر زمان برنامه هستیم. با اجرای برنامه به طور خودکار تایم زمان دستگاه را به عنوان زمان برنامه تنظیم میکند. در قسمت مدیریت با استفاده از setter این کلاس تاریخ برنامه را می تواند عوض بکند.

بسمه تعالی

نام پروژه : بانک فارابی
نام تیم : DEVROID

پروژه پایان ترم برنامه نویسی پیشرفته

❖ توضیحاتی درباره کلاس مشتری :

1) کلاس مشتری با ارث بری از یوزر تمام ویژگی های ان را به ارث می برد. اطلاعات مشتری ها در فایل Clients.txt ذخیره می شوند ک همان اطلاعات کاربری آن ها میباشد.
2) مشتری ها توانایی مشاهده حساب های خود و تراکنش های انجامی اشان را دارند به همین دلیل با استفاده از متد showAllAccounts تمامی حساب های خود را به صورت لیست از حساب برمیگردانیم که شامل تمامی اطلاعات حساب اشان میباشد. این متد با خواندن اطلاعات تمام حساب ها از فایل حساب های فرد مورد نظر را برمیگرداند.
3) متد addAccount متدی استیتک میباشد برای اضافه کردن حساب برای هر مشتری که میخاهیم با توجه به نوع حسابی که مشتری می خواهد حساب مورد نظر با مقدار پول برای شخص ساخته می شود و سپس حساب مورد نیاز در فایل ها ذخیره می شوند. همینطور که حساب با توجه به حروف الفبا صاحب حساب سورت می شوند.

❖ توضیحاتی درباره کلاس کارمند :

کلاس کارمند با ارث بری از یوزر تمام ویژگی های ان را به ارث می برد. همچنین ویژگی های منحصر به فرد خود از جمله حقوق و رزومه خود را دارا میباشد. اطلاعات کارمند ها در فایل Employes.txt ذخیره می شوند ک همان اطلاعات کاربری آن ها میباشد و برای نمایش کارمندان و ورود از ان استفاده می شود.

متد های getter setter کلاس کارمند زده شده اند ک برای تغییر حقوق کارمندان توسط مدیر از انها استفاده شده است.

بسمه تعالی

نام پروژه : بانک فارابی
نام تیم : DEVROID

پروژه پایان ترم برنامه نویسی پیشرفته

❖ توضیحاتی درباره پکیج حساب :

کلاس حساب و انواع حساب و تراکنش و انتقال دوره ای در پکیج Accounts وجود دارند.

❖ کلاس حساب :

1) کلاس حساب (Account) دارای ویژگی های اسم صاحب حساب ، شماره ملی صاحب حساب ، مقدار پول ، نوع حساب و شماره حساب می باشد. شماره حساب ها به صورت منحصر به فرد برای هر حساب به صورت خودکار در constructor کلاس ساخته می شود.
2) متد allowTransaction متدی می باشد که اجازه انتقال یا برداشت وجه رو برای حساب باز میکند. با توجه به اینکه در انواع حساب ، حساب ها محدودیت های خاص خودشون رو دارند به طور مثال حساب بلند مدت ، مدتی را نمیتواند از حساب خود وجه جابجا کند این متد این مسئله را مدیریت میکند.
3) متد depositFund متدی میباشد برای واریز پول از حساب. به این صورت کار میکند که اگر حساب مجاز به واریز پول بود مقدار پول درخواستی را گرفته و به پول حساب اضافه میکند و دوباره در فایل مقدار پول حساب تغییر پیدا میکند و تراکنش مورد نظر ثبت می شود.
4) متد withdrawFund همانند متد depositFund عمل میکند و هر دو از ترد برای انتقال پول استفاده می کنند ک برنامه توانایی انتقال پول به صورت همزمان را هم داشته باشد.
5) متد transfer_money با گرفتن شماره حساب ها و مقدار پول انتقالی با پیدا کردن دو حساب مورد نظر دو متد واریز پول و برداشت پول رو برای حساب ها صدا میزند این متد برای کارمند هم قابل استفاده هست ک بتواند برای مشتریان انتقال پول انجام دهد.
6) این متد ها تمام چک های مورد نظر از جمله ک حساب در لیست حساب های بانک وجود داشته باشد را دارا می باشند.
7) متود singleTransaction به صورت جداگانه عمل برداشت و یا واریز به یک حساب را انجام می دهد.
8) متود addPeriodicTransaction برای اضافه کردن شماره حساب به لیست تراکنش های دوره ای استفاده می شود.
9) متد showInvestory متدی برای نشان دادن موجودی حساب می باشد .

❖ **کلاس تراکنش های دوره ای(PeriodicTransaction):**

این کلاس اطلاعات حساب هایی را که قصد تراکنش های دوره ای را دارند در خود ثبت و ذخیره می کنه به این ترتیب که شماره حساب مبدا و مقصد و زمان ورود به لیست تراکنش های دوره ای را دریافت می کند و توسط متود synchronized شده checkDate اگر زمان یک ماهه انتقال وجه رسیده بود به متود synchronized شده periodicTransfer اعلام می کند که عمل واریز رو انجام بدهد بااین دو متود تردهای بخش واریز دوره ای مدیریت می شود.

❖ توضیحاتی درباره انواع حساب های بانکی :

1) حساب جاری (CurrentDepositAccount) :

حسابی جاری حسابی هست که به ان سود هایی تعلق نمی گیرد و محدودیتی از نظر برداشت پول ندارد و هر لحظه و هر مقدار میتواند پول جابجا کند و مناسب برای نگه داری پول ها میباشند و از کلاس حساب ها ارث برده است.

2) حساب ماه شمار (DemandDepositAccount) :

حساب ماه شمار با ارث بردن از کلاس حساب ها ویژگی های ان را به ارث می برد با توجه به اینکه یک ماه محدودیت انتقال وجه دارد ویژگی expired_at دارد که تاریخ پایان محدودیت را ذخیره میکند ک یک ماه می باشد این محدودیت. و ویژگی InterestRate مقدار سود ماهانه ک عدد 15 درصد میباشد را ذخیره می کند.

این حساب دارای متد محاسبه ماهانه سود میباشد که سود هر ماه را به پول اصلی اضافه میکند.

3) حساب سال شمار (TimeDepositAccount) :

حساب سال شمار با ارث بردن از کلاس حساب ها ویژگی های ان را به ارث می برد با توجه به اینکه یک ماه محدودیت انتقال وجه دارد ویژگی expired_at دارد که تاریخ پایان محدودیت را ذخیره میکند ک یک سال می باشد مقدار سود سالانه عدد 20 درصد میباشد.

این حساب دارای متد محاسبه سالانه سود میباشد که سود هر سال را به پول اصلی اضافه میکند.

❖ توضیحاتی درباره کلاس تراکنش حساب ها :

این کلاس با ارث بری از حساب ها ویژگی شماره پیگیری و شماره حساب واریز کننده و دریافت کننده و نوع تراکنش که به صورت واریز یا برداشت می باشد را دارا هست. شماره پیگیری به صورت منحصر به فرد ساخته می شود و برای هر تراکنش به صورت جدا ساخته می شود که بتوانیم شماره پیگیری مورد نظر را سرچ بکنیم تا اطلاعات لازم تراکنش را بدست بیاوریم.

تراکنش ها همانند حساب ها در فایل مخصوص خود به اسم transactions.txt ذخیره می شوند که برای تراکنش ها ازش استفاده می شود.

بسمه تعالی

نام پروژه : بانک فارابی
نام تیم : DEVROID

پروژه پایان ترم برنامه نویسی پیشرفته

❖ توضیحاتی درباره گرافیک برنامه :

این برنامه دارای menu می باشد که شامل ورود ، درباره ما و خروج است و هر صفحه شامل دکمه هایی است که با کلیک بر روی هرکدام وارد صفحه جدیدی می شود و همچنین دکمه هایی برای جلو و عقب رفتن و خروج از برنامه طراحی شده است.

در گرافیک برنامه صفحه هایی مخصوص مدیر و کارمند و مشتری طراحی شده است که با ورود به این صفحات اطلاعات مربوطه قابل مشاهده است.

- **مدیر :**

1) دارای صفحه لاگین می باشد که در صورت صحت نام کاربری و رمز عبور که از فایل ManagerAuth.csv می خواند، وارد می شود.
2) دارای سه دکمه برای مدیریت مشتری ها و مدیریت حساب ها و مدیریت کارمند ها است که با کلیک بر روی این دکمه ها وارد صفحه مخصوص میشود که دارای جدول هایی برای نمایش اطلاعات است.
3) این جدول ها دارای قابلیت هایی همچون حذف و ویرایش و اضافه کردن و نمایش تراکنش هر حساب می باشد.
4) برای نمایش اطلاعات بر جدول از متد getUsers در کلاس Management و متد getAccounts در کلاس AccountManager استفاده شده است.
5) برای حذف از متد delete استفاده شده است.
6) برای ویرایش از متد edit استفاده شده است.
7) برای اضافه کردن از متد newUser و newAccount استفاده شده است.

بسمه تعالی

نام پروژه : بانک فارابی
نام تیم : DEVROID

پروژه پایان ترم برنامه نویسی پیشرفته

- **کارمند :**

1) دارای صفحه لاگین می باشد که در صورت صحت نام کاربری و رمز عبور که مدیر ثبت نام کرده است، وارد می شود.
2) تمام اطلاعات مربوط به کارمند در فایل Employees.csv ذخیره می شود.
3) دارای سه دکمه برای مدیریت مشتری ها و مدیریت حساب ها و تراکنش ها است که با کلیک بر روی این دکمه ها وارد صفحه مخصوص میشود که دارای جدول هایی برای نمایش اطلاعات است.
4) این جدول ها دارای قابلیت هایی همچون اضافه کردن و انجام تراکنش و خروجی گرفتن می باشد.
5) برای اضافه کردن از متد newUser استفاده شده است.
6) برای انجام تراکنش از متد transfer_money درون کلاس Account استفاده شده است.

- *مشتری :*

1) دارای صفحه ثبت نام می باشد.
2) دارای صفحه لاگین می باشد که در صورت صحت نام کاربری و رمز عبور که مشتری ثبت نام کرده است، وارد می شود.
3) تمام اطلاعات مربوط به مشتری در فایل Clients.csv ذخیره می شود.
4) دارای سه دکمه برای مشاهده حساب ها و برداشت وجه و انتقال وجه است که با کلیک بر روی این دکمه ها وارد صفحه مخصوص میشود.
5) برای برداشت وجه از متد withdrawFund درون کلاس Account استفاده شده است.
6) برای انتقال وجه از متد transfer_money درون کلاس Account استفاده شده است.