

پروژه درس برنامه‌سازی پیشرفته

به عنوان پروژه این درس شما باید سیستم یک بانک را پیاده سازی کنید. در صفحه اصلی سیستم، سناریوهای تعریف شده در قالب دکمه و یا گزینه انتخابی نمایش داده می‌شود. با کلیک بر روی هر سناریو و یا انتخاب هر سناریو، به پنجره آن سناریو در قالب فرم نمایش داده می‌شود. سناریوهایی که در این سیستم تعریف شده است، در بخش ۱ آمده است. در بخش ۲ فرم هایی که برای انجام سناریو باید طراحی شود، شرح داده شده است و در بخش ۳ کلاس‌ها، فیلدها و توابعی که باید پیاده‌سازی کنید، به تفکیک شرح داده شده است.

۱. سناریوهای تعریف شده

- اضافه کردن مشتری جدید: با کلیک بر روی این سناریو فرم وارد کردن اطلاعات مشتری نمایش داده می‌شود و در پایان اطلاعات وارد شده در فایل customer.txt ذخیره می‌شود.
- افتتاح سپرده: در صورتی که مشتری از مشتری‌های شعبه باشد، با کلیک بر روی این سناریو، فرم ورود اطلاعات سپرده نمایش داده می‌شود. در غیر این صورت، ابتدا باید بر روی گزینه "اضافه کردن مشتری" کلیک نموده و سناریو اضافه شدن مشتری جدید انجام می‌شود.
- اضافه کردن یک شعبه: فرم ورود اطلاعات شعبه نمایش داده می‌شود.
- اضافه کردن یک کارمند جدید به کارمندان شعبه: فرم ورود اطلاعات کارمند شعبه نمایش داده می‌شود.
- برداشت: فرم برداشت وجه نمایش داده می‌شود.
- واریز: فرم واریز وجه نمایش داده می‌شود.
- انتقال وجه: فرم انتقال وجه نمایش داده می‌شود.
- واریز سود: فرم واریز سود نمایش داده می‌شود.
- نمایش اطلاعات مشتری: فرم نمایش اطلاعات مشتری نمایش داده شود.
- نمایش اطلاعات سپرده: فرم نمایش اطلاعات سپرده نمایش داده شود.
- نمایش اطلاعات کارمند: فرم نمایش اطلاعات کارمند نمایش داده شود.
- نمایش اطلاعات شعبه: فرم نمایش اطلاعات شعبه نمایش داده شود.
- بستن سپرده: فرم بستن سپرده نمایش داده می‌شود.

۲. فرم‌ها

فرم‌های ورود اطلاعات

- فرم ورود اطلاعات مشتری

کاربر اطلاعات مشتری را از طریق فرم وارد کرده و سپس با کلیک بر روی دکمه Save اطلاعات وارد شده برای مشتری را به ترتیب نام، نام خانوادگی، شماره مشتری، کدملی و تاریخ تولد در یک فایل متنی به نام customer.txt ذخیره کند.

- فرم ورود اطلاعات سپرده مشتری

کاربر اطلاعات هر سپرده برای هر مشتری را از طریق فرم وارد کرده و سپس با کلیک بر روی دکمه Save، اطلاعات وارد شده را به ترتیب شماره سپرده، نام نوع سپرده، کد نوع سپرده، شماره سپرده دوم، نوع سپرده دوم، شماره مشتری، تاریخ افتتاح سپرده و موجودی سپرده در یک فایل متنی به نام deposit.txt ذخیره کند.

- فرم ورود اطلاعات کارمند شعبه

کاربر اطلاعات هر کارمند را از طریق فرم وارد کرده و سپس اطلاعات وارد شده را به ترتیب نام، نام خانوادگی، کدملی، تاریخ تولد و شماره کارمندی در یک فایل متنی به نام employee.txt ذخیره کند.

- فرم ورود اطلاعات شعبه‌ها

کاربر اطلاعات هر شعبه را از طریق فرم وارد کرده و سپس این اطلاعات را به ترتیب نام شعبه، کد شعبه، نام و نام خانوادگی مدیر شعبه در یک فایل متنی به نام branch.txt ذخیره کند.

فرم‌های نمایش اطلاعات

- فرم نمایش اطلاعات مشتری

با گرفتن شماره مشتری در فایل customer.txt جستجو نموده و اطلاعات مشتری شامل نام، نام خانوادگی، شماره مشتری، کدملی، تاریخ تولد و سپرده‌های مشتری را نمایش دهد.

○ تابع جستجوی سپرده‌های یک مشتری

```
public Deposit[] getCustDeposits(int custNumber)
```

با گرفتن شماره مشتری آرایه‌ای از سپرده‌های مشتری را بر می‌گرداند. (برای این منظور باید در فایل deposit.txt بر اساس شماره مشتری جستجو نموده و لیست سپرده‌های مشتری را برگرداند.)

- فرم نمایش اطلاعات سپرده

با گرفتن شماره سپرده در فایل deposit.txt جستجو نموده و اطلاعات سپرده شامل شماره سپرده، نام نوع سپرده، شماره سپرده دوم، نوع سپرده دوم، شماره مشتری، تاریخ افتتاح سپرده و موجودی سپرده را نمایش دهد.

- فرم نمایش اطلاعات کارمند یک شعبه

با گرفتن شماره کارمندی در فایل employee.txt جستجو نموده و اطلاعات کارمند شامل نام، نام خانوادگی، کد ملی، تاریخ تولد، شماره کارمندی، نام و کد شعبه استعلامی را نمایش می دهد.

- فرم نمایش اطلاعات شعبه

با گرفتن شماره شعبه، نام شعبه، نام و نام خانوادگی مدیر شعبه و لیست کارمندان شعبه را نمایش می دهد.

○ تابع جستجوی کارمندان شعبه (با گرفتن کد شعبه آرایه ای از کارمندان شعبه را بر می گرداند).

```
public Employee[] getEmployee(int branchCode)
```

- فرم برداشت

با گرفتن شماره مشتری، شماره سپرده و مبلغ برداشتی، مبلغ موردنظر را از موجودی سپرده مشتری کسر کند.

- فرم واریز

با گرفتن شماره سپرده و مبلغ واریزی، مبلغ موردنظر را به موجودی سپرده اضافه کند.

- فرم انتقال وجه

با گرفتن شماره مشتری مبدأ، شماره سپرده مبدأ، مقصد و مبلغ انتقالی، مبلغ موردنظر را از موجودی سپرده مبدأ کسر و به موجودی سپرده مقصد اضافه کند. انتقال وجه در دو گام برداشت از سپرده مبدأ و واریز به سپرده مقصد انجام می گیرد.

- فرم واریز سود

با گرفتن شماره سپرده و کد و نوع سپرده، سود سپرده را محاسبه و به موجودی سپرده اضافه می‌کند. در صورتی که سپرده بلندمدت باشد، سود به سپرده دوم (سپرده پشتیبانی) واریز می‌شود.

سود سپرده‌های کوتاه مدت ۱۰٪ و به صورت روزانه محاسبه می‌شود سپرده‌های بلند مدت به صورت یکساله و با سود ۱۸٪ به طور سالانه هستند. سپرده‌های قرض‌الحسنه جاری و قرض‌الحسنه پس‌انداز نیز سودی ندارند.

فرمول محاسبه سود سپرده‌های کوتاه مدت روزشمار:

$$۳۶۵ / (\text{مدت به روز} \times \text{نرخ سود} \times \text{میانگین موجودی سپرده در طول مدت سرمایه‌گذاری})$$

فرمول محاسبه سود سپرده‌های بلندمدت با پرداخت سود ماهیانه

$$۱۲ / (\text{نرخ سود} \times \text{موجودی سپرده})$$

فرم بستن سپرده

با گرفتن شماره مشتری و شماره سپرده، سپرده مورد نظر را می‌بندد و پیغام موفقیت آمیز بودن عملیات را نمایش دهد.

۳. کلاس‌ها، فیلدها و توابع

در هر کلاس تمامی فیلدها را به صورت private تعریف کنید. کلاس‌ها، فیلدها و توابع مربوط به هر کلاس در زیر لیست شده است.

class Deposit

فیلدهای کلاس

private String depositType; // نوع سپرده

private int depositTypeCode; // کد نوع سپرده

مقادیر ممکن برای فیلدهای depositType و depositeTypeCode در جدول زیر آمده است:

نام سپرده	depositType	depositTypeCode
-----------	-------------	-----------------

1	Short_term_deposit	سپرده کوتاه مدت
2	Long_term_deposit	سپرده بلندمدت
3	Current_deposit	سپرده قرض الحسنه جاری
4	Saving_deposit	سپرده قرض الحسنه پس انداز

private Deposit support; ////

سپرده دوم

private int depositNumber; //

شماره سپرده

private int branchCode; //

کد شعبه

private Customer owner; ////

مشتری صاحب سپرده

private double initialAmount; //

موجودی اولیه

private Date openDate; //

تاریخ افتتاح سپرده

private double balance; //

موجودی سپرده

private double profit_ratio; //// نرخ سود سپرده های کوتاه مدت و بلندمدت

private Date operationDate; //

تاریخ انجام عملیات

private int isOpen; در صورتی که سپرده باز باشد مقدار آن یک و در صورت بسته بودن مقدار آن صفر است.

توابع کلاس :

public Deposit(String depositType, int depositTypeCode, Deposit support, int depositNumber, Customer owner, double initialAmount, Date openDate);

کلاس را با گرفتن نوع سپرده، شماره سپرده، صاحب سپرده، موجودی اولیه و تاریخ افتتاح سپرده می سازد. سپرده دوم برای این است که اگر نوع سپرده از نوع بلند مدت بود، سود مربوط به آن در آن سپرده قرار گیرد. اگر نوع سپرده از نوع بلند مدت نبود، مهم نیست که سپرده دوم چه باشد اما در صورت بلند مدت بودن سپرده باید سپرده دوم متعلق به همین مشتری باشد و از نوع کوتاه مدت یا جاری باشد. در صورت مغایرت ورودی ها با هر یک از موارد بالا، سپرده ساخته شده به هیچ مشتری ای نسبت داده نمی شود.

توابع set و get مربوط به هر کدام از فیلدهای private کلاس.

```
public double getBalance();
```

آخرین موجودی سپرده را بر می گرداند.

```
public double getBalanceInDate(Date date);
```

با گرفتن یک تاریخ موجودی سپرده را در آن تاریخ برمی گرداند.

```
public boolean withdraw(int custNumber, double amount, Date date);
```

با گرفتن شماره مشتری، تاریخ انجام عملیات و مبلغ برداشت آن را از موجودی سپرده کم می کند و true بر می گرداند و در غیر این صورت false برمی گرداند.

شرایطی که در آن برداشت از سپرده ممکن نیست عبارت است از:

- سپرده متعلق به مشتری نباشد. (برای این عملیات، ابتدا در فایل deposit.txt، شماره سپرده جستجو و شماره مشتری این سپرده بازگردانده می شود و با شماره مشتری مقایسه می شود).
- موجودی سپرده از میزان درخواست شده کمتر باشد. (برای این عملیات، ابتدا در فایل deposit.txt، شماره سپرده جستجو و موجودی سپرده بازگردانده می شود).
- سپرده از نوع بلند مدت (long-term) باشد.

```
public boolean settlement(double amount, Date date);
```

با گرفتن تاریخ انجام عملیات و میزان واریز آن را به موجودی سپرده اضافه می کند و true ، بر می گرداند. در صورتی که سپرده از نوع بلند مدت باشد false برمی گرداند (سپرده می تواند متعلق به این مشتری نباشد).

```
public boolean closeDeposit(int custNumber, Date date);
```

با گرفتن شماره مشتری و شماره سپرده و تاریخ انجام عملیات، سپرده را می بندد و فیلد isOpen سپرده را برابر با صفر قرار می دهد و موجودی سپرده را خالی می کند و true بر می گرداند و در صورتی که مشتری صاحب سپرده نباشد، false بر می گرداند.

```
public String toString();
```

اطلاعات مربوط به یک سپرده را در قالب یک رشته برمی گرداند. قالب این رشته به این صورت است که در خط اول شماره سپرده، در خط بعدی نام صاحب سپرده، خط بعدی نوع سپرده و در خط آخر موجودی سپرده آورده می شود.

class Customer

فیلدهای کلاس:

```
private String firstName    ////////// نام مشتری
private String lastName     ////////// نام خانوادگی مشتری
private int  CustNumber     ////////// شماره مشتری
private Date  birthdate     ////////// تاریخ تولد مشتری
private int  nationalID     ////////// کد ملی مشتری
private int  branchCode     ////////// کد شعبه
```

توابع کلاس:

سازنده کلاس

```
public Customer(String firstName, String lastName, int custNumber);
```

کلاس را با گرفتن نام مشتری و شماره مشتری می سازد.

توابع set و get مربوط به هر کدام از فیلدهای private کلاس.

```
public String toString();
```

اطلاعات مربوط به مشتری را در یک رشته برمی گرداند. قالب این رشته به این صورت است که در خط اول شماره مشتری، در خط بعدی نام و نام خانوادگی مشتری می آید.

class Branch

فیلدهای کلاس

```
private String branchName;
```

```
private int branchCode;
```

```
private Employee manager;
```

توابع کلاس

```
public Branch(String branchName, int branchCode);
```

کلاس را با گرفتن نام شعبه و کد شعبه می سازد.

توابع set و get مربوط به هر کدام از فیلدهای private کلاس.

class Employee

فیلدهای کلاس

این کلاس از کلاس Customer به ارث می برد. بنابراین همه فیلدهای تعریف شده برای کلاس Customer برای این کلاس نیز قابل دسترسی خواهد بود (فیلدهای private از طریق متدهای set و get مربوطه قابل دسترسی خواهند بود). علاوه بر آنها فیلدهای زیر نیز تعریف می شود.

```
private int employeeNumber;          // شماره کارمندی
```

```
private int branchCode; // کد شعبه ای که در آن مشغول به کار است.
```

توابع کلاس

```
public Employee(int employeeNumber, int branchCode) ;
```

کلاس را با گرفتن شماره کارمندی و کد شعبه می سازد.

توابع set و get فیلدهای private کلاس.

class Date

فیلدهای کلاس

Private int day;

Private int month;

Private int year;

توابع کلاس

public Date(int day, int month, int year);

از این کلاس برای وارد کردن تاریخ عملیات بانکی و محاسبه سود سپرده‌ها استفاده می‌شود.

توابع set و get مربوط به هر کدام از فیلدهای private کلاس.