Java实验报告

@*Author* : *liyajun*_208111545116

实验一

搭建环境并输出*Hello World*

```
public static void main(String[] args){
    // write your code here
    System.out.println("Hello World");
}
```

```
 .jar=55238:C:\Users\liyaj\AppData\Local\Jet
 demo.demo01
Hello World

Process finished with exit code 0
```

实验二

回文数

H5 代码实现

```
 boolean valid(int i) {
    String s = Integer.toString(i);
    int l = 0,r = s.length() - 1;
    while(l < r) {
        if(s.toCharArray()[l++] != s.toCharArray()[r--])
return false;
    }
    return true;
}
```

H5 结果

```
12121 -> true;
114514 -> false;
```

猜数字

H5 代码实现

```
public void guess(){
    Random ra = new Random();
    int cur = ra.nextInt(100) + 1;
    System.out.println("please enter a num");
    Scanner sc = new Scanner(System.in);
    int get = sc.nextInt();
    while(get != cur){
        if(get > cur) System.out.println("Try a smaller
number");
        else if(get < cur) System.out.println("Try a larger
number");
        get = sc.nextInt();
    }
    System.out.println("congratulations");
}
```

H5 结果

```
 .jar=56111:C:\Users\liyaj\AppData\Local\JetBrains\Toolbox
 demo.demo01
please enter a num
50
Try a larger number
85
Try a smaller number
65
Try a larger number
75
congratulations

Process finished with exit code 0
```

完数

H5 代码实现

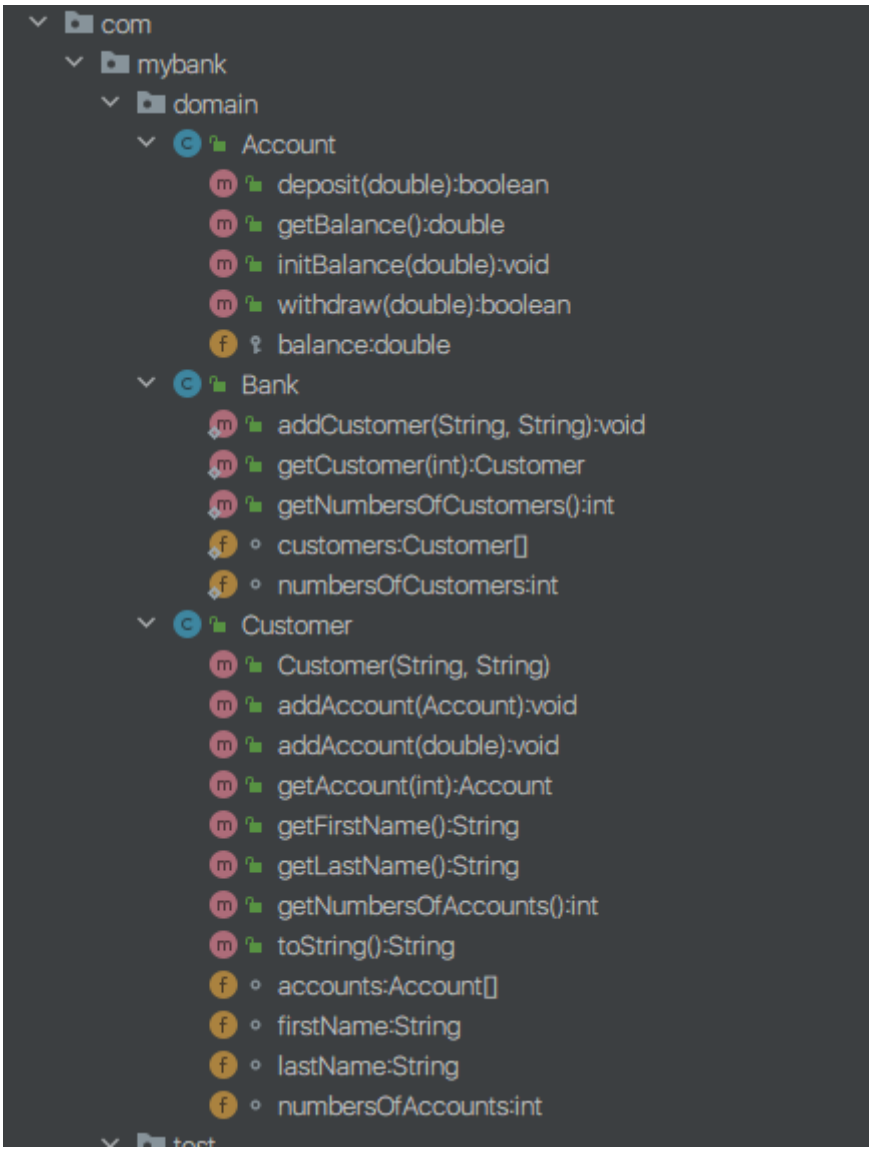
```java
public static void main(String[] args){
    // write your code here
    for(int i = 1;i <= 1000; i++){
        if(i % 2 == 0 && valid(i)) {
            System.out.println(i);
        }
    }
}
boolean valid(int x) {
    int cur = x;
    for(int i = 1; i <= x/2; i++) {
        if(x % i == 0) cur -= i;
    }
    return cur == 0;
}
```

H5 运行结果

```
6
28
496
```

实验三

项目结构



实现代码

H5 Account.java

```java
package com.mybank.domain;

public class Account {
    protected double balance;
    //初始化余额
    public void initBalance(double balance) {
```

```
        this.balance = balance;
    }
    //存钱
    public boolean deposit(double amt){
        if(amt < 0) return false;
        balance += amt;
        return true;
    }
    //取钱
    public boolean withdraw(double amt){
        if(balance < amt) return false;
        balance -= amt;
        return true;
    }

    //获取余额
    public double getBalance(){
        return balance;
    }
}

```

Bank.java

```
package com.mybank.domain;

public class Bank {
    static Customer [] customers;
    static int numbersOfCustomers;
    static {
        customers = new Customer[10];
        numbersOfCustomers = 0;
    }
    public static void addCustomer(String firstName, String
lastName){
        Customer cus = new Customer(firstName,lastName);
        try {
            customers[numbersOfCustomers++] = cus;
        }catch (ArrayIndexOutOfBoundsException e){
            System.out.println(e);
        }
    }

```

```
    public static int getNumbersOfCustomers(){
        return numbersOfCustomers;
    }
    public static Customer getCustomer(int id){
        try {
            return customers[id];
        } catch (ArrayIndexOutOfBoundsException e){
            System.out.println(e);
        }
        return null;
    }
}

```

H5 Customer.java

```
package com.mybank.domain;

public class Customer {
    String firstName;
    String lastName;
    Account [] accounts;
    int numbersOfAccounts;

    public Customer(String firstName, String lastName) {
        this.firstName = firstName;
        this.lastName = lastName;
        numbersOfAccounts = 0;
        accounts = new Account[10];
    }
    public void addAccount(double amount){
           Account acc = new Account();
           acc.initBalance(amount);
           addAccount(acc);
    }
    public void addAccount(Account acc){
        try {
            accounts[numbersOfAccounts++] = acc;
        }catch (ArrayIndexOutOfBoundsException e){
            System.out.println(e);
        }
    }

```

```
27      public Account getAccount(int id){
28          try {
29              return accounts[id];
30          }catch (ArrayIndexOutOfBoundsException e){
31              System.out.println(e);
32          }
33          return  null;
34      }
35      public int getNumbersOfAccounts() {
36          return numbersOfAccounts;
37      }
38      public String getFirstName() {
39          return firstName;
40      }
41
42      public String getLastName() {
43          return lastName;
44      }
45
46      @Override
47      public String toString() {
48          return  firstName + "-" +  lastName ;
49      }
50  }
51
```

H5 TestBanking.java

```
1   package com.mybank.test;
2
3   import com.mybank.domain.Bank;
4
5   import java.util.Random;
6
7   public class TestBanking {
8       public static void main(String[] args) {
9           //初始化成员
10          for(int i = 0; i < 10; i++){
11              Random ra = new Random();
12              Bank.addCustomer(Integer.toString(i), "Jinx");
13              for(int j = 0;j < 10; j++){
```

```
14
 Bank.getCustomer(i).addAccount(ra.nextDouble(1145.14));
15
 Bank.getCustomer(i).getAccount(j).withdraw(ra.nextDouble(1145
.14));
16              }
17          }
18          for(int i = 0;i < 10; i++){
19
 System.out.println(Bank.getCustomer(i).toString());
20              for(int j = 0;j < 10; j++){
21                  System.out.println("第"+(j + 1 )+"个账户的余额
为");
22
 System.out.println(String.format("%.2f",Bank.getCustomer(i).
getAccount(j).getBalance()));
23              }
24          }
25
26      }
27  }
28
```

测试结果

```
C:\Users\liyaj\.jdks\openjdk-17.0.2\bin\java.exe -javaagent:C:\Users\liyaj\AppData\Local\JetBrains\Toolbox\apps\IDEA-U\ch
  .jar=57769:C:\Users\liyaj\AppData\Local\JetBrains\Toolbox\apps\IDEA-U\ch-0\213.7172.25\bin -Dfile.encoding=UTF-8 -classp
  com.mybank.test.TestBanking
0-Jinx
第1个账户的余额为
72.67
第2个账户的余额为
184.98
第3个账户的余额为
30.48
第4个账户的余额为
57.66
第5个账户的余额为
721.69
第6个账户的余额为
191.60
第7个账户的余额为
764.83
第8个账户的余额为
454.07
第9个账户的余额为
223.04
第10个账户的余额为
22.55
1-Jinx
第1个账户的余额为
320.37
第2个账户的余额为
625.96
第3个账户的余额为
23.68
第4个账户的余额为
187.50
第5个账户的余额为
123.28
第6个账户的余额为
700.84
第7个账户的余额为
42.48
```

实验四

带有存款和贷款的银行系统

H5 **CheakAccount.java**

```java
package experiment.four.com.mybank.domain;
//能够透支的账户
public class CheckingAccount extends Account{
    double overdraftAmount;

    public CheckingAccount(double initBalance, double
overdraftAmount){
        super(initBalance);
        this.overdraftAmount = overdraftAmount;
    }
    CheckingAccount(double initBalance){
        super(initBalance);
```

```
        this.overdraftAmount = 0.0;
    }

    public double getOverdraftAmount() {
        return overdraftAmount;
    }

    @Override
    public int withdraw(double amount) {
        if(balance < amount){
            //使用贷款
            double overdraftNeeded = amount - balance;
            if(overdraftNeeded > overdraftAmount){
                return 1;
            }
            else{
                balance = 0.0;
                overdraftAmount -= overdraftNeeded;
                return 2;
            }
        } // 能够付清
        else{
            balance -= amount;
        }
        return 3;
    }
}

```

SavingAccount.java

```
package experiment.four.com.mybank.domain;
//能计算利息的账户
public class SavingsAccount extends Account{
    private double interestRate;
    public SavingsAccount(double initBalance,double
interestRate){
        super(initBalance);
        this.interestRate = interestRate;
    }

```

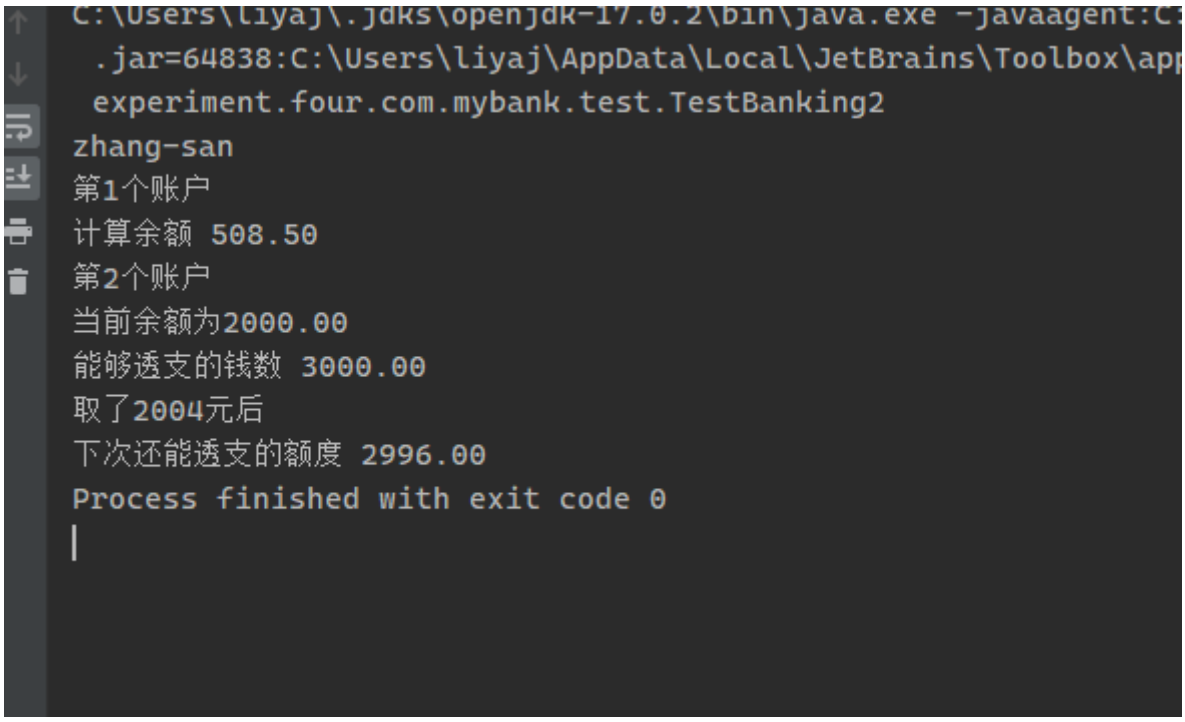
```
9
10
11      //计算利息
12      public double accumulateInterest(){
13          return balance + balance * interestRate;
14      }
15  }
16
```

H5 Test

```
1   package experiment.four.com.mybank.test;
2
3
4   import experiment.four.com.mybank.domain.*;
5
6
7   public class TestBanking2 {
8       public static void main(String[] args) {
9           Bank.addCustomer("zhang", "san");
10          Customer c1 = Bank.getCustomer(0);
11          c1.addAccount(new SavingsAccount(500,0.017));
12          c1.addAccount(new CheckingAccount(2000, 3000));
13          for(int i = 0;i < Bank.getNumbersOfCustomers();i++) {
14              Customer cus = Bank.getCustomer(i);
15              System.out.println(cus);
16              for(int j = 0;j < cus.getNumbersOfAccounts();j++)
    {
17                  Account acc = cus.getAccount(j);
18                  System.out.println("第"+(j+1)+"个账户");
19                  if(acc instanceof SavingsAccount sa){
20                      System.out.printf("计算余额
    %.2f\n",sa.accumulateInterest());
21                  }
22                  else if(acc instanceof CheckingAccount ck){
23                      System.out.printf("当前余额
    为%.2f\n",ck.getBalance());
24                      System.out.printf("能够透支的钱数
    %.2f\n",ck.getOverdraftAmount());
25                      ck.withdraw(2004);
26                      System.out.println("取了2004元后");
```

```
27                      System.out.printf("下次还能透支的额度
%.2f",ck.getOverdraftAmount());
28                  }
29                  else {
30                      //普通账户
31                      System.out.printf("余额为
%.2f",acc.getBalance());
32                  }
33              }
34          }
35      }
36  }
37
```

测试



实验五

动物宠物程序

实现代码

H5 Animal.java

```java
package experiment.five.InterfaceProject;

public abstract class Animal {
    protected int legs;
    protected  Animal(int legs){
        this.legs = legs;
    }
    abstract void eat();
    public void walk(){
        System.out.println("Animal walk with"+legs+"legs.");
    }
}

```

H5 Cat.java

```java
package experiment.five.InterfaceProject;

public class Cat extends Animal implements Pet{
    private String name;
    public Cat(String name){
        super(4);
        this.name = name;
    }
    public Cat(){
        this("");
    }
    @Override
    void eat() {
        System.out.println("Cat likes eating mice.");
    }

    @Override
    public String getName() {

```

```
        return this.name;
    }

    @Override
    public void setName(String n) {
        this.name = name;
    }

    @Override
    public void play() {
        System.out.println("Cat likes playing strings.");
    }
}
```

Duck.java

```
package experiment.five.InterfaceProject;

public class Duck extends Animal implements Pet,Sound{
    private String name;

    public Duck( String name) {
        super(2);
        this.name = name;
    }

    @Override
    public void walk() {
        System.out.println("duck swim and walks
with"+legs+"legs");
    }

    @Override
    void eat() {
        System.out.println("duck like eating waterweeds");
    }

    @Override
    public String getName() {
        return this.name;
    }

```

```
    @Override
    public void setName(String n) {
        this.name = name;
    }

    @Override
    public void play() {
        System.out.println("duck likes playing with water");
    }
    public void shout(String voice){
        Sound.super.shout(voice);
        System.out.println("i am a yellow duck");
    }
}
```

H5 Fish.java

```
package experiment.five.InterfaceProject;

public class Fish extends Animal implements Pet{
    private String name;
    @Override
    void eat() {
        System.out.println("Fish likes eating bugs in the
ponds");
    }

    public Fish() {
        super(0);
    }

    @Override
    public String getName() {
        return this.name;
    }

    @Override
    public void setName(String n) {
        this.name = n;
    }

    @Override
```

```
25      public void play() {
26          System.out.println("fish likes swimming in yje
    ponds");
27      }
28      @Override
29      public void walk(){
30          super.walk();
31          System.out.println("fish are swimming happily in the
    pool.");
32      }
33  }
```

H5 Pet.interface

```
1   package experiment.five.InterfaceProject;
2
3   public interface Pet {
4       public String getName();
5       public void setName(String n);
6       public void play();
7   }
```

H5 Sound.interface

```
1   package experiment.five.InterfaceProject;
2
3   public interface Sound {
4       public default void shout(String voice){
5           System.out.println(voice);
6       }
7   }
```

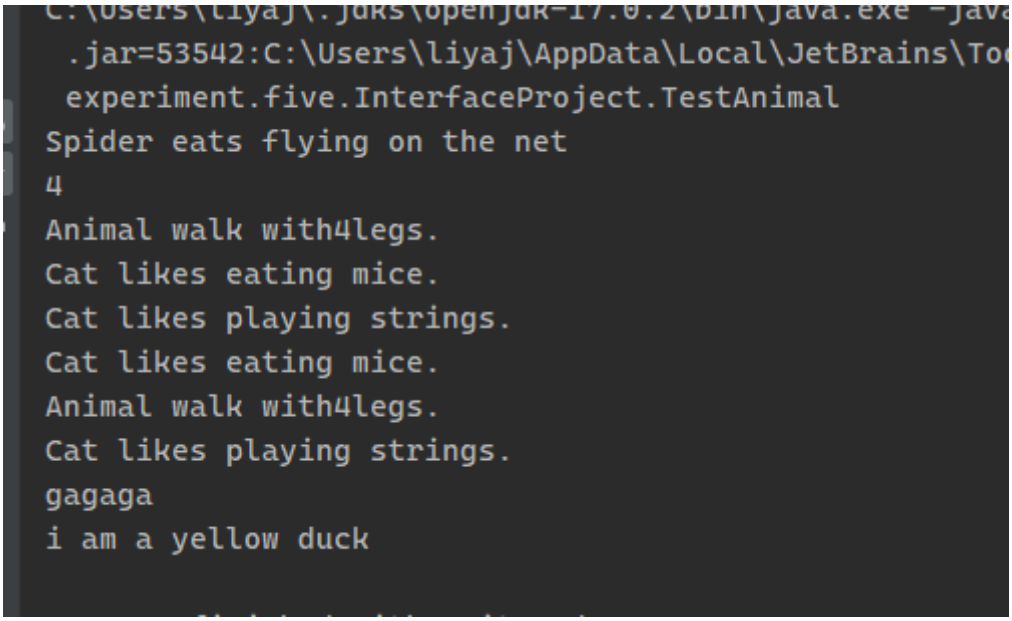
H5 Spider.java

```java
package experiment.five.InterfaceProject;

public class Spider extends Animal{
    public Spider(){
        super(8);
    }

    @Override
    void eat() {
        System.out.println("Spider eats flying on the net");
    }
}
```

H5 Test

```java
package experiment.five.InterfaceProject;

public class TestAnimal {
    public static void main(String[] args) {
        Spider s1 = new Spider();
        s1.eat();
        Animal s2 = new Cat();
        System.out.println(s2.legs);
        s2.walk();
        s2.eat();
        Pet s3 = new Cat("fluffy");
        s3.getName();
        s3.play();
        Cat s4 = new Cat("Grafield");
        s4.getName();
        int legs = s4.legs;
        s4.eat();
        s4.walk();
        s4.play();
        Pet s5 = new Fish();
        s5.getName();
        Duck s6  = new Duck("donald duck");
        s6.shout("gagaga");
    }
}
```

测试



实验六

实验代码

H5 容器*Bank*

```
1  package experiment.six.com.mybank.domain;
2
3  import java.util.ArrayList;
4  import java.util.List;
5
6  public class Bank {
7      static List<Customer> customers;
8      static {
9          customers = new ArrayList<>();
10     }
11     public static void addCustomer(String firstName, String
   lastName){
12         customers.add(new Customer(firstName,lastName));
13     }
14     public static int getNumbersOfCustomers(){
```

```
        return customers.size();
    }
    public static Customer getCustomer(int id){
        try {
            return customers.get(id);
        } catch (ArrayIndexOutOfBoundsException e){
            System.out.println(e);
        }
        return null;
    }
}
```

H5 容器*Customer*

```
package experiment.six.com.mybank.domain;

import java.util.ArrayList;
import java.util.List;

public class Customer {
    String firstName;
    String lastName;
    List<Account> accounts;
    //Account[] accounts;
    int numbersOfAccounts;

    public Customer(String firstName, String lastName) {
        this.firstName = firstName;
        this.lastName = lastName;
        accounts = new ArrayList<>();
    }
    public void addAccount(double amount){
           addAccount(new Account(amount));
    }
    public void addAccount(Account acc){
            accounts.add(acc);
    }
    public Account getAccount(int id){
        try {
            return accounts.get(id);
        }catch (ArrayIndexOutOfBoundsException e){
            System.out.println(e.getMessage());

```

```
            return null;
        }
    }
    public int getNumbersOfAccounts() {
        return accounts.size();
    }
    public String getFirstName() {
        return firstName;
    }

    public String getLastName() {
        return lastName;
    }

    @Override
    public String toString() {
        return  firstName + "-" +  lastName ;
    }
}
```

H5 异常类*OverdraftException*

```
package experiment.six.com.mybank.domain;

/**
 * @author liyajun
 * @date 2022/4/6 14:40
 */
public class OverdraftException extends Exception{
    private double deficit;

    public OverdraftException(String message,double deficit){
        super(message);
        this.deficit = deficit;
    }

    public double getDeficit() {
        return deficit;
    }
}
```

测试

```
package experiment.six.com.mybank.test;


import experiment.six.com.mybank.domain.*;


public class TestBanking2 {
    public static void main(String[] args) {
        Bank.addCustomer("zhang", "san");
        Customer c1 = Bank.getCustomer(0);
        c1.addAccount(new SavingsAccount(500,0.017));
        c1.addAccount(new CheckingAccount(2000, 3000));
        for(int i = 0; i < Bank.getNumbersOfCustomers(); i++)
{
            Customer cus = Bank.getCustomer(i);
            System.out.println(cus);
            for(int j = 0;j < cus.getNumbersOfAccounts();j++)
{
                Account acc = cus.getAccount(j);
                System.out.println("第"+(j+1)+"个账户");
                if(acc instanceof SavingsAccount sa){
                    System.out.printf("计算余额
%.2f\n",sa.accumulateInterest());
                }
                else if(acc instanceof CheckingAccount ck){
                    System.out.printf("当前余额
为%.2f\n",ck.getBalance());
                    System.out.printf("能够透支的钱数
%.2f\n",ck.getOverdraftAmount());
                    try {
                        ck.withdraw(6000);
                    } catch (OverdraftException e) {
                        System.out.println(e.getMessage()+"赤
字为"+e.getDeficit());
                        e.printStackTrace();
                    }
                }
                else {
                    //普通账户
```

```
34                      try {
35                          acc.withdraw(60000);
36                      } catch (OverdraftException e) {
37                          System.out.println(e.getMessage());
38                          e.printStackTrace();
39                      }
40                      System.out.printf("余额为
    %.2f",acc.getBalance());
41                  }
42              }
43          }
44      }
45  }
46
```

```
C:\Users\liyaj\.jdks\openjdk-17.0.2\bin\java.exe -javaagent:C:\Users\liyaj\AppData\Local\JetBrains\Toolbox\apps\IDEA-U\ch-0\213.7172.25\lib\idea_rt
 .jar=64712:C:\Users\liyaj\AppData\Local\JetBrains\Toolbox\apps\IDEA-U\ch-0\213.7172.25\bin -Dfile.encoding=UTF-8 -classpath C:\Users\liyaj\IdeaProjects\2022_Java\out\production\2022_Java
 experiment.six.com.mybank.test.TestBanking2
zhang-san
第1个账户
计算余额 508.50
第2个账户
当前余额为2000.00
能够透支的钱数 3000.00
透支额度不足赤字为4000.0
experiment.six.com.mybank.domain.OverdraftException Create breakpoint : 透支额度不足
    at experiment.six.com.mybank.domain.CheckingAccount.withdraw(CheckingAccount.java:26)
    at experiment.six.com.mybank.test.TestBanking2.main(TestBanking2.java:26)

Process finished with exit code 0
```