Java实验报告

@*Author : liyajun_208111545116*

实验一

搭建环境并输出 *Hello World*

```java
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

```java
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;
}
```

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

猜数字

```java
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");
}
```
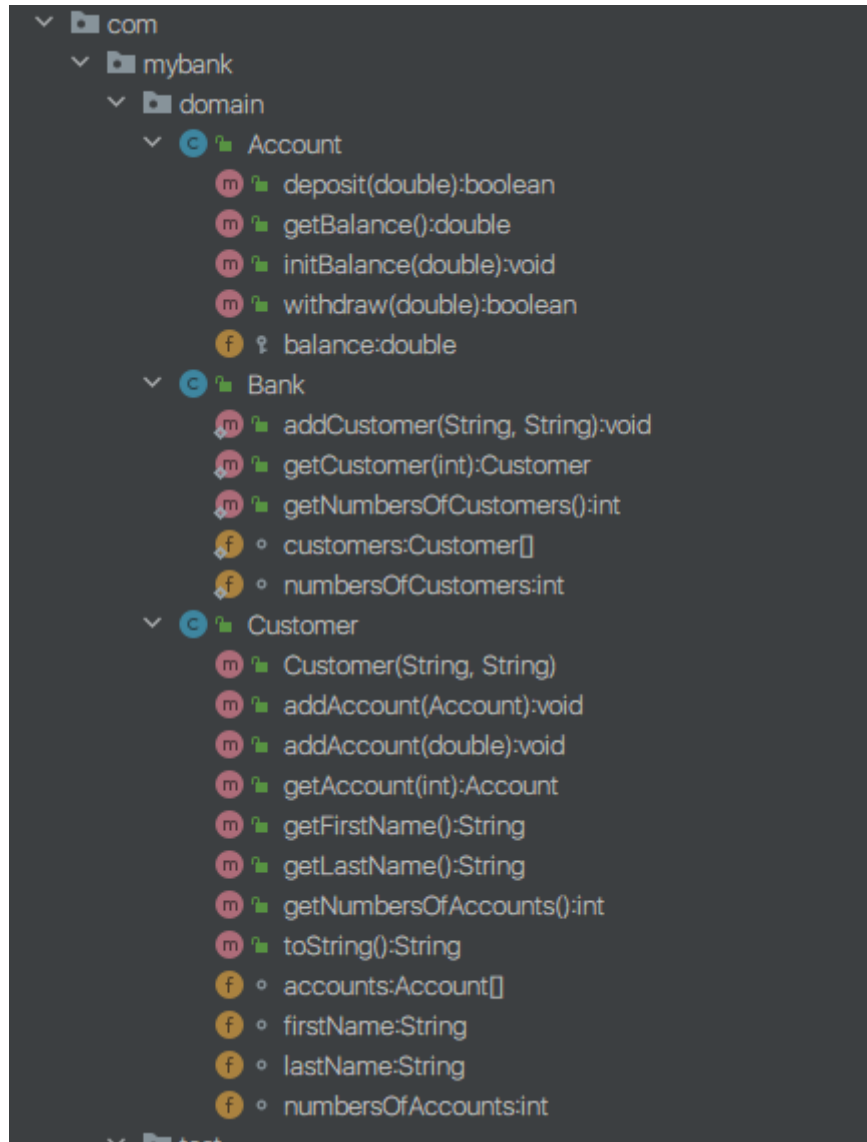
H5 结果

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

实验三

项目结构

实现代码

Account.java

```
1  package com.mybank.domain;
2
3  public class Account {
4      protected double balance;
5      //初始化余额
6      public void initBalance(double balance) {
```

```java
            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

```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);
        }
    }
```

```java
18      public static int getNumbersOfCustomers(){
19          return numbersOfCustomers;
20      }
21      public static Customer getCustomer(int id){
22          try {
23              return customers[id];
24          } catch (ArrayIndexOutOfBoundsException e){
25              System.out.println(e);
26          }
27          return null;
28      }
29  }
30
```

H5 Customer.java

```java
1   package com.mybank.domain;
2
3   public class Customer {
4       String firstName;
5       String lastName;
6       Account [] accounts;
7       int numbersOfAccounts;
8
9       public Customer(String firstName, String lastName) {
10          this.firstName = firstName;
11          this.lastName = lastName;
12          numbersOfAccounts = 0;
13          accounts = new Account[10];
14      }
15      public void addAccount(double amount){
16          Account acc = new Account();
17          acc.initBalance(amount);
18          addAccount(acc);
19      }
20      public void addAccount(Account acc){
21          try {
22              accounts[numbersOfAccounts++] = acc;
23          }catch (ArrayIndexOutOfBoundsException e){
24              System.out.println(e);
25          }
26      }
```

```java
    public Account getAccount(int id){
        try {
            return accounts[id];
        }catch (ArrayIndexOutOfBoundsException e){
            System.out.println(e);
        }
        return  null;
    }
    public int getNumbersOfAccounts() {
        return numbersOfAccounts;
    }
    public String getFirstName() {
        return firstName;
    }

    public String getLastName() {
        return lastName;
    }

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

TestBanking.java

```java
package com.mybank.test;

import com.mybank.domain.Bank;

import java.util.Random;

public class TestBanking {
    public static void main(String[] args) {
        //初始化成员
        for(int i = 0; i < 10; i++){
            Random ra = new Random();
            Bank.addCustomer(Integer.toString(i), "Jinx");
            for(int j = 0;j < 10; j++){
```

```java
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
```

```
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

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){
        this.balance = initBalance;
        this.overdraftAmount = overdraftAmount;
    }
    CheckingAccount(double initBalance){
        this.balance = initBalance;
```

```java
            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

```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){
        this.balance = initBalance;
        this.interestRate = interestRate;
    }
```

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

```
1    package experiment.four.com.mybank.test;
2
3
4    import experiment.four.com.mybank.domain.Account;
5    import experiment.four.com.mybank.domain.Bank;
6    import experiment.four.com.mybank.domain.CheckingAccount;
7    import experiment.four.com.mybank.domain.SavingsAccount;
8
9    import java.util.Random;
10
11   public class TestBanking2 {
12       public static void main(String[] args) {
13           Random ra = new Random();
14           for(int i = 0;i < 10; i++){
15
16               Bank.addCustomer(Integer.toString(i + 1), "Jinx");
17               for(int j = 0;j < 10;j++){
18
 Bank.getCustomer(i).addAccount(ra.nextDouble(114.15));
19               }
20           }
21           //普通账户
22           for(int i = 0;i < 10; i++){
23               if(i == 0)
System.out.println(Bank.getCustomer(0).toString());
24               System.out.print("账户"+(i+1)+"的余额为");
25
 System.out.println(String.format("%.2f",Bank.getCustomer(0).
getAccount(i).getBalance()));
26           }
27           //测试利息账户
```

```java
        for(int i = 0; i < 10; i ++){
            if(i == 0)
System.out.println(Bank.getCustomer(1).toString());
            SavingsAccount sa = new
SavingsAccount(Bank.getCustomer(1).getAccount(i).getBalance(),
ra.nextDouble(114.51));
            System.out.print("账户"+(i+1)+"的余额加上利息为");

 System.out.println(String.format("%.2f",sa.getBalance()));
        }


        //测试贷款账户
        for(int i = 0; i < 10; i ++){
            if( i ==
0)System.out.println(Bank.getCustomer(2).toString());
            System.out.println("账户"+(i+1)+"的取款情况为");
            CheckingAccount ck = new
CheckingAccount(Bank.getCustomer(2).getAccount(i).getBalance()
,ra.nextDouble(114.514));
            if(ck.withdraw(ra.nextDouble(114.51)) == 1){
                System.out.println("余额 + 贷款不足 不能取出");
                System.out.println("当前余额为" +
String.format("%.2f",ck.getBalance()));
                System.out.println("当前可用贷款余额为" +
String.format("%.2f",ck.getOverdraftAmount()));
            }
            else if(ck.withdraw(ra.nextDouble(114.51)) == 2){
                System.out.println("使用贷款可以取出");
                System.out.println("当前余额为" +
String.format("%.2f",ck.getBalance()));
                System.out.println("当前可用贷款余额为" +
String.format("%.2f",(ck.getOverdraftAmount())));
            }
            else{
                System.out.println("当前余额足够取出，不需要贷
款");
                System.out.println("当前余额为" +
String.format("%.2f",ck.getBalance()));
                System.out.println("当前可用贷款余额为" +
String.format("%.2f",ck.getOverdraftAmount()));
            }
        }
    }
```

```
57    }
58
```

H5 正常账户

H5 带利息账户

H5 可以贷款的账户

```
3-Jinx
账户1的取款情况为
余额 + 贷款不足  不能取出
当前余额为71.16
当前可用贷款余额为42.28
账户2的取款情况为
使用贷款可以取出
当前余额为0.00
当前可用贷款余额为56.53
账户3的取款情况为
当前余额足够取出，不需要贷款
当前余额为0.00
当前可用贷款余额为7.64
账户4的取款情况为
余额 + 贷款不足  不能取出
当前余额为30.17
当前可用贷款余额为6.61
账户5的取款情况为
使用贷款可以取出
当前余额为0.00
当前可用贷款余额为102.57
账户6的取款情况为
余额 + 贷款不足  不能取出
当前余额为0.39
当前可用贷款余额为25.07
账户7的取款情况为
当前余额足够取出，不需要贷款
当前余额为11.44
当前可用贷款余额为40.55
账户8的取款情况为
当前余额足够取出，不需要贷款
当前余额为0.00
当前可用贷款余额为48.89
账户9的取款情况为
```

实验五

动物宠物程序

实现代码

Animal.java

```
1   package experiment.five.InterfaceProject;
2
3   public abstract class Animal {
4       protected int legs;
5       protected  Animal(int legs){
6           this.legs = legs;
7       }
8       abstract void eat();
9       public void walk(){
10          System.out.println("Animal walk with"+legs+"legs.");
11      }
12  }
13
```

Cat.java

```
1   package experiment.five.InterfaceProject;
2
3   public class Cat extends Animal implements Pet{
4       private String name;
5       public Cat(String name){
6           super(4);
7           this.name = name;
8       }
9       public Cat(){
10          this("");
11      }
12      @Override
13      void eat() {
14          System.out.println("Cat likes eating mice.");
15      }
16
17      @Override
18      public String getName() {
19          return this.name;
20      }
21
22      @Override
23      public void setName(String n) {
24          this.name = name;
```

```
25        }
26
27        @Override
28        public void play() {
29            System.out.println("Cat likes playing strings.");
30        }
31  }
```

H5 Duck.java

```java
1   package experiment.five.InterfaceProject;
2
3   public class Duck extends Animal implements Pet,Sound{
4       private String name;
5
6       public Duck( String name) {
7           super(2);
8           this.name = name;
9       }
10
11       @Override
12       public void walk() {
13           System.out.println("duck swim and walks
    with"+legs+"legs");
14       }
15
16       @Override
17       void eat() {
18           System.out.println("duck like eating waterweeds");
19       }
20
21       @Override
22       public String getName() {
23           return this.name;
24       }
25
26       @Override
27       public void setName(String n) {
28           this.name = name;
29       }
30
31       @Override
```

```java
32        public void play() {
33            System.out.println("duck likes playing with water");
34        }
35        public void shout(String voice){
36            Sound.super.shout(voice);
37            System.out.println("i am a yellow duck");
38        }
39  }
```

H5 Fish.java

```java
1   package experiment.five.InterfaceProject;
2
3   public class Fish extends Animal implements Pet{
4       private String name;
5       @Override
6       void eat() {
7           System.out.println("Fish likes eating bugs in the
    ponds");
8       }
9
10      public Fish() {
11          super(0);
12      }
13
14      @Override
15      public String getName() {
16          return this.name;
17      }
18
19      @Override
20      public void setName(String n) {
21          this.name = n;
22      }
23
24      @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  }
```

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   }
```

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   }
```

Spider.java

```
1   package experiment.five.InterfaceProject;
2
3   public class Spider extends Animal{
4       public Spider(){
5           super(8);
6       }
7
8       @Override
9       void eat() {
10          System.out.println("Spider eats flying on the net");
11      }
12  }
```

```
1   package experiment.five.InterfaceProject;
2
3   public class TestAnimal {
4       public static void main(String[] args) {
5           Spider s1 = new Spider();
6           s1.eat();
7           Animal s2 = new Cat();
8           System.out.println(s2.legs);
9           s2.walk();
10          s2.eat();
11          Pet s3 = new Cat("fluffy");
12          s3.getName();
13          s3.play();
14          Cat s4 = new Cat("Grafield");
15          s4.getName();
16          int legs = s4.legs;
17          s4.eat();
18          s4.walk();
19          s4.play();
20          Pet s5 = new Fish();
21          s5.getName();
22          Duck s6  = new Duck("donald duck");
23          s6.shout("gagaga");
24      }
25  }
```

测试

C:\Users\liyaj\.jdks\openjdk-17.0.2\bin\java.exe -Java
 .jar=53542:C:\Users\liyaj\AppData\Local\JetBrains\Too
 experiment.five.InterfaceProject.TestAnimal
Spider eats flying on the net
4
Animal walk with4legs.
Cat likes eating mice.
Cat likes playing strings.
Cat likes eating mice.
Animal walk with4legs.
Cat likes playing strings.
gagaga
i am a yellow duck