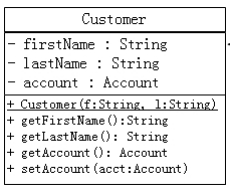


尚硅谷 Java 基础编程 宋红康

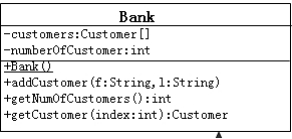
1.按照如下的 UML 类图，创建相应的类，提供必要的结构

在提款方法 withdraw()中，需要判断用户余额是否能够满足提款数额的要求，如果不能， 应给出提示。deposit()方法表示存款。

2. 按照如下的 UML 类图，创建相应的类，提供必要的结构



3. 按照如下的 UML 类图，创建相应的类，提供必要的结构



 addCustomer 方法必须依照参数（姓，名）构造一个新的 Customer 对象，然后把

它放到 customer 数组中。还必须把 numberOfCustomer 属性的值加 1。  getNumOfCustomers 方法返回 numberofCustomers 属性值。

尚硅谷 Java 基础编程 宋红康

 getCustomer 方法返回与给出的 index 参数相关的客户。 4. 创建 BankTest 类，进行测试。

Account:

**package** Test2;

**public** **class** Account {

//属性

**private** **double** balance;

//构造器

**public** Account(**double** init\_balance) {

**this**.balance = init\_balance;

}

//方法

**public** **double** getBalance() {

**return** balance;

}

**public** **void** deposit(**double** amt) {//存钱

**if**(amt > 0) {

**this**.balance += amt;

System.***out***.println("存钱成功");

}

}

**public** **void** wihtdraw(**double** amt) {//取钱

**if**(balance >= amt) {

**this**.balance -= amt;

System.***out***.println("取钱成功");

}**else** {

System.***out***.println("余额不足");

}

}

}

Customer:

**package** Test2;

**public** **class** Customer {

//属性

**private** String firstName;

**private** String lastName;

**private** Account account;

//构造器

**public** Customer(String f, String l) {

**this**.firstName = f;

**this**.lastName = l;

}

//方法

**public** Account getAccount() {

**return** account;

}

**public** **void** setAccount(Account account) {

**this**.account = account;

}

**public** String getFirstName() {

**return** firstName;

}

**public** String getLastName() {

**return** lastName;

}

}

Bank:

**package** Test2;

**public** **class** Bank {

//属性

**private** Customer[] customers;

**private** **int** numberOfCustomer;

//构造器

**public** Bank(){

customers = **new** Customer[10];

}

//方法

**public** **void** addCustomer(String f, String l) {//添加客户

Customer cust = **new** Customer(f, l);

customers[numberOfCustomer] = cust;

numberOfCustomer++;

}

**public** **int** getNumOfCustomers() {//获取客户个数

**return** numberOfCustomer;

}

**public** Customer getCustomer(**int** index) {

**if**(index >= 0 && index < numberOfCustomer) {

**return** customers[index];

}**else** {

**return** **null**;

}

}

}

BankTest:

**package** Test2;

**public** **class** BankTest {

**public** **static** **void** main(String[] args) {

Bank test = **new** Bank();

test.addCustomer("Jane", "Smith");

test.getCustomer(0).setAccount(**new** Account(2000));

test.getCustomer(0).getAccount().wihtdraw(500);

**double** balance = test.getCustomer(0).getAccount().getBalance();

System.***out***.println("客户余额：" + balance);

}

}