

1、写一个名为 `Account` 的类模拟账户。该类的属性和方法如下图所示。该类包括的属性：账号 `id`，余额 `balance`，年利率 `annualInterestRate`；包含的方法：访问器方法（getter 和 setter 方法），取款方法 `withdraw()`，存款方法 `deposit()`。

Account
<code>private int id</code> <code>private double balance</code> <code>private double annualInterestRate</code>
<code>public Account (int id, double balance, double annualInterestRate)</code>
<code>public int getId()</code> <code>public double getBalance()</code> <code>public double getAnnualInterestRate()</code> <code>public void setId(int id)</code> <code>public void setBalance(double balance)</code> <code>public void setAnnualInterestRate(double annualInterestRate)</code> <code>public void withdraw (double amount)//取钱</code> <code>public void deposit (double amount)//存钱</code>

提示：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。

2. 创建 `Customer` 类。

Customer
<code>private String firstName</code> <code>private String lastName</code> <code>private Account account</code>
<code>public Customer(String f,String l)</code>
<code>public String getFirstName()</code> <code>public String getLastName()</code> <code>public Account getAccount()</code> <code>public void setAccount(Account account)</code>

- 声明三个私有对象属性：`firstName`、`lastName` 和 `account`。
- 声明一个公有构造器，这个构造器带有两个代表对象属性的参数（`f` 和 `l`）
- 声明两个公有存取器来访问该对象属性，方法 `getFirstName` 和 `getLastName` 返回相应的属性。
- 声明 `setAccount` 方法来对 `account` 属性赋值。
- 声明 `getAccount` 方法以获取 `account` 属性。

3. 写一个测试程序。

- 创建一个 `Customer`，名字叫 Jane Smith，他有一个账号为 1000，余额为 2000 元，年利率为 1.23% 的账户。
- 对 Jane Smith 操作。
存入 100 元，再取出 960 元。再取出 2000 元。
打印出 Jane Smith 的基本信息

成功存入 : 100.0

成功取出: 960.0

余额不足, 取款失败

Customer [Smith, Jane] has a account: id is 1000, annualInterestRate is 1.23%, balance is 1140.0

