### Java第一阶段—DAY17-JAVA作业答案

1. 在Person类中，年龄的范围是0-120岁，性别只能是男或女。自定义年龄异常类和性别异常类，输出相应异常提示信息。编写测试类，给人物的年龄和性别赋值，如果不符合要求，抛出自定义异常类的异常提示信息。

**class** Person{  
 **private** String **name** ;  
 **private int age** ;  
 **private** String **sex** ;  
  
 **public** String getName() {  
 **return name**;  
 }  
  
 **public void** setName(String name) {  
 **this**.**name** = name;  
 }  
  
 **public int** getAge() {  
 **return age**;  
 }  
  
 **public void** setAge(**int** age) {  
 **if**(age<0 || age>120){  
 **throw new** AgeException(**"年龄必须在0-120之间"**);  
 }  
 **this**.**age** = age;  
 }  
  
 **public** String getSex() {  
 **return sex**;  
 }  
  
 **public void** setSex(String sex) {  
 **if**(!**"男"**.equals(sex) && !**"女"**.equals(sex) ){  
 **throw new** SexException(**"性别只能为男或者女"**);  
 }  
 **this**.**sex** = sex;  
 }  
}  
  
**class** AgeException **extends** RuntimeException{  
 **public** AgeException(){  
 **super**();  
 }  
  
 **public** AgeException(String message) {  
 **super**(message);  
 }  
}  
  
**class** SexException **extends** RuntimeException{  
 **public** SexException() {  
 }  
  
 **public** SexException(String message) {  
 **super**(message);  
 }  
}  
  
**public class** Test {  
  
 **public static void** main(String[] args) {  
 Person p = **new** Person();  
 p.setName(**"blb"**);  
 p.setAge(-200);  
 p.setSex(**"其它"**);  
 }  
  
}

1. 定义一个信用卡类（账户，额度），定义一个People类，属性有：姓名，年龄，信用卡；方法有：刷信用卡。在测试类中实现：杨光，22岁，有一张信用卡，账户：123456789000 额度：2000 。 他现在想刷信用卡给女朋友买一个包包，价值3000. 自定义异常， 当额度超支时，抛出提示“信用额度不足，请理性消费！”。

**class** People{  
 **private** String **name** ;  
 **private int age** ;  
 **private** String **account**;  
 **private int money** ;  
  
 **public void** buy(**int** money){  
 **if**(money > **this**.**money**){  
 **throw new** MoneyException(**"信用额度不足，请理性消费！"**);  
 }  
 **this**.**money** -= money ;  
 System.***out***.println(**"购买成功"**);  
 }  
  
 **public** String getAccount() {  
 **return account**;  
 }  
  
 **public void** setAccount(String account) {  
 **this**.**account** = account;  
 }  
  
 **public** String getName() {  
 **return name**;  
 }  
  
 **public void** setName(String name) {  
 **this**.**name** = name;  
 }  
  
 **public int** getAge() {  
 **return age**;  
 }  
  
 **public void** setAge(**int** age) {  
 **this**.**age** = age;  
 }  
  
 **public int** getMoney() {  
 **return money**;  
 }  
  
 **public void** setMoney(**int** money) {  
 **this**.**money** = money;  
 }  
}  
  
**class** MoneyException **extends** RuntimeException{  
 **public** MoneyException(){  
 **super**();  
 }  
  
 **public** MoneyException(String message) {  
 **super**(message);  
 }  
}  
  
**public class** A {  
  
 **public static void** main(String[] args) {  
 People people = **new** People();  
 people.setName(**"杨光"**);  
 people.setAccount(**"123456789000"**);  
 people.setAge(22);  
 people.setMoney(2000);  
 people.buy(3000);  
  
 }  
  
}

1. 编写一个登录界面，要求帐户名只能由1至10位数字组成，密码只能有6位，任何不符合帐户名和密码要求的情况都视为异常，必须捕获并处理异常。

**class** AccountException **extends** Exception{  
 **public** AccountException(){  
 **super**();  
 }  
  
 **public** AccountException(String message) {  
 **super**(message);  
 }  
}  
  
**public class** Test {  
  
 **public static void** main(String[] args) {  
 Scanner input = **new** Scanner(System.***in***);  
 System.***out***.println(**"请输入用户名："**);  
 String name = input.next();  
 System.***out***.println(**"请输入密码："**);  
 String pwd = input.next();  
 **try** {  
 *checkAccount*(name,pwd);  
 } **catch** (AccountException e) {  
 e.printStackTrace();  
 }  
 }  
  
 **public static void** checkAccount(String name ,String pwd) **throws** AccountException {  
 **if**(name.length()<1 || name.length()>10){  
 **throw new** AccountException(**"用户名长度必须在1-10位之间"**);  
 }  
  
 **if**(pwd.length()!= 6 ){  
 **throw new** AccountException(**"密码长度必须为6位"**);  
 }  
 }  
  
}

1. throw与throws的区别

throw关键字通常用在方法体中，并且抛出一个异常对象。程序在执行到throw语句时立即停止，它后面的语句都不执行。

throws关键字通常被应用在声明方法时，用来指定可能抛出的异常。多个异常可以使用逗号隔开。当在主函数中调用该方法时，如果发生异常，就会将异常对象抛给方法调用处。

1. 利用高效字节输出流往C盘下的blb.txt文件写出一个字节数组数据，如写出：”i love you”

**public class** Test01\_02 {

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

// 创建字节输出流FileOutputStream对象并指定文件路径。

FileOutputStream fos = **new** FileOutputStream("c:\\e.txt");

// 利用字节输出流创建高效字节输出流对象

BufferedOutputStream bos = **new** BufferedOutputStream(fos);

// 调用高效字节输出流对象的write(byte[] buff)方法写出一个字节数据

bos.write("i love java".getBytes());

// 关闭流

bos.close();

}

}