作业要求：[不用抄题，共10个作业，每个作业手写（考试也是手写，为了让同学们加深印象）在一张白纸上，一共10张纸，拍照做成1个PDF文件（10页），命名为“2021秋-软件20-\*班-面向对象程序设计(JAVA)（双语）-作业--姓名-学号.pdf”,11月25日前发送给班长，麻烦班长收齐后发送到邮箱2028001455@qq.com@全体成员](mailto:不用抄题，共10个作业，每个作业手写（考试也是手写，为了让同学们加深印象）在一张白纸上，一共10张纸，拍照做成1个PDF文件（10页），命名为\“2021秋-软件20-*班-面向对象程序设计(JAVA)（双语）-作业--姓名-学号.pdf\”,11月25日前发送给班长，麻烦班长收齐后发送到邮箱2028001455@qq.com@全体成员)

**作业1： 选择题**

(1). which one are valid declaraction of a float?

A. float foo=3.03d; B. float foo=1.0; C. float foo=42e1; D. float foo=2.02f;

**D.**  Abc是double类型的，所以错了

(2). You want subclasses in any package to have access to members of a superclass. Which is the most restrictive access modifier that will accomplish this objective?

（*你希望子类在任何包里都能访问父类，为完成这个目的，下列哪个是最严格的访问权限）*

A. Public B. Private C. Protected D. Transient

(3). which one declaretions prevent the overriding of a method?

A. final void methoda(){}

B. void final methoda(){} final void

C. static void methoda(){}

D. final abstract void methoda(){}

(4). Which declaration prevents creating a subclass of an outer class?

A.static class FooBar{}

B.pivate class Foobar{}

C.abstract class FooBar{}

D.final public class FooBar{}

(5). which one are true to describe an entire encapsulation class?

A. member data have no access modifiers

B. member data can be modified directly

C. the access modifier for methods is protected

D. the access modifier to member data is private

**作业2：简答题**

Simulate the program of bank transactions, find the objects that may be used, and list the possible attributes and behaviors for each of them.

**May need three objects: account, customer and bank。**

**account**

**a private object attribute: balance**

**a public constructor with one parameter int\_balance**

**a public method getBalance**

**a public method deposit**

**a public method withdraw**

**Customer**

**Declare three private object attributes: firstName, lastName, and account.**

**setFirstName**

**setLastName**

**getFirstName**

**getLastName**

**setAccount**

**getAccount**

**Bank**

**customers (an array of Customer objects)**

**numberOfCustomers**

**setCustomersSize**

**addCustomer**

**getNumOfCustomers**

**getCustomer**

**作业3：编程题**

Use the nesting of conditional operators to complete this question: students with academic performance >=90 points are represented by A, those with a score of 60-89 are represented by B, and those with a score of less than 60 are represented by C.

Program analysis: (a>b)? a:b is the basic example of conditional operators.

import java.util.Scanner;

public class ScoreLevel {

public static void main(String[] args)

{

System.out.println("请输入学生的成绩:");

Scanner in = new Scanner(System.in);

int score = in.nextInt();

grade(score);

}

private static void grade(int score) {

if((0 > score)||(100 < score)){

System.out.print("输入学生成绩无效!!");

}

else {

String level = ((score >= 90)? "分，为A等级。" :((score >= 60)? "分，为B等级。" :"分，为C等级。"));

System.out.println("该学生的分数是:" + score + level);

}

}

}

**作业4：编程题**

To define an electric fan class Fan, the requirements are as follows:

(a) Attributes include: electric fan model (String type), price (double type) and switch status running (boolean type), and all attributes are private.

(b) Provide at least one parameterized construction method (requires that the model can be initialized to any value, the price cannot be less than 0, and the switch state must be false).

(c) Provide accessor methods for private attributes.

public class Fan {

private String model;

private double price;

private boolean running;

public String getModel(){

return model;

}

public double getPrice() {

return price;

}

public boolean isRunning() {

return running;

}

public Fan(String model, double price, boolean running){

this.model = model;

if(price <= 0)

{

System.out.println("价格不能小于零");

}

else this.price = price;

if(running) this.running = false;

}

}

**作业5：简答题**

Please write down the relationship between the class and the object?

**A class is an abstraction of an object, and an object is a concrete instance of a class. The relationship between class and object: Abstraction and class instantiation are objects.The common characteristics of objects are abstracted out as classes.**

**作业6：编程题**

The existing interfaces CanFly and CanJump describe the ability of flying and jumping respectively. Requirements: Declare the Locust (locust) class and the Balloon (balloon) class. The Locust class has the ability to fly and jump, and the Balloon class has the ability to fly. The method of flying and jumping can be realized by outputting the corresponding string. Then, in the main class Prog1, add a static method testFly() whose function is to allow flying objects to perform flying actions. Create objects of the Locust class and Balloon class in the main method, and call the testFly() method.

public class test {

public interface CanFly{

void Fly();

}

public interface CanJump {

void Jump();

}

public static class Locust implements CanFly,CanJump{

@Override

public void Fly() {

System.out.println("蝗虫可以飞");

}

@Override

public void Jump() {

System.out.println("蝗虫能跳");

}

}

public static class Balloon implements CanFly{

@Override

public void Fly() {

System.out.println("气球能飞起来");

}

}

public class Prog1 {

public static void testFly(CanFly fly) {

fly.Fly();

}

public static void main(String[] args) {

Locust Locust1 = new Locust();

Balloon Balloon1 = new Balloon();

Prog1.testFly(Balloon1);

Prog1.testFly(Locust1);

}

}

}

**作业7：程序运行结果**

class MyException extends Exception {

public String toString() {

return "negative";

}

}

public class ExceptionDemo {

public static void mySqrt(int a) throws MyException {

if (a < 0)

throw new MyException();

System.out.println(Math.sqrt(a));

}

public static void main(String args[]) {

try {

mySqrt(25);

mySqrt(-5);

} catch (MyException e) {

System.out.println("Caught " + e);

}

}

}

5.0

Caught negative

**作业8： 编程题**

Create two threads. Thread 1 calculates the sum of 1 to 99 as follows. Thread 2 prints out the name of thread I and reads the calculation result of thread 1 at regular intervals. It is required to use two ways of inheriting from Thread and implementing Runnable interface.

**作业9：程序运行结果**

public class GeneralSample {

String str = new String("good");

char[] ch = { 'e', 'd', 'c' };

public static void main(String args[]) {

GeneralSample ex = new GeneralSample();

ex.change(ex.str, ex.ch);

System.out.print(ex.str + "and");

System.out.print(ex.ch);

}

public void change(String str, char ch[]) {

str = "test ok";

ch[0] = 'f';

}

}

goodandfdc

**作业10：编程题**

Write data of several JAVA data types to a file and read it out.

import java.io.\*;

public class input {

public static void main(String[] args)

{

File file=new File("C:/Users/90337/Desktop/qqq.txt"); //根据路径创建File类对象--这里路径即使错误也不会报错，因为只是产生File对象，还并未与计算机文件读写有关联

try

{

FileInputStream fileInputStream=new FileInputStream(file);

//与根据File类对象的所代表的实际文件建立链接创建fileInputStream对象

System.out.println(fileInputStream.read());

}

catch (FileNotFoundException e)

{

System.out.println("文件不存在或者文件不可读或者文件是目录");

} catch (IOException e) {

e.printStackTrace();

}

}

}