|  |  |
| --- | --- |
| **ASSESMENT TEST** # **III**  **MAX MARKS : 100 TIME : 120 Minutes** | **Test Objectives**   * *Basic understanding of OOP and its features* * *To understand the control structures ,Inheritance, Interfaces etc* * *Exception Handling, Multithreading etc* |

**NAME: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ MARKS OBTAINED: \_\_\_\_ / 100 GRADE \_\_\_\_\_**

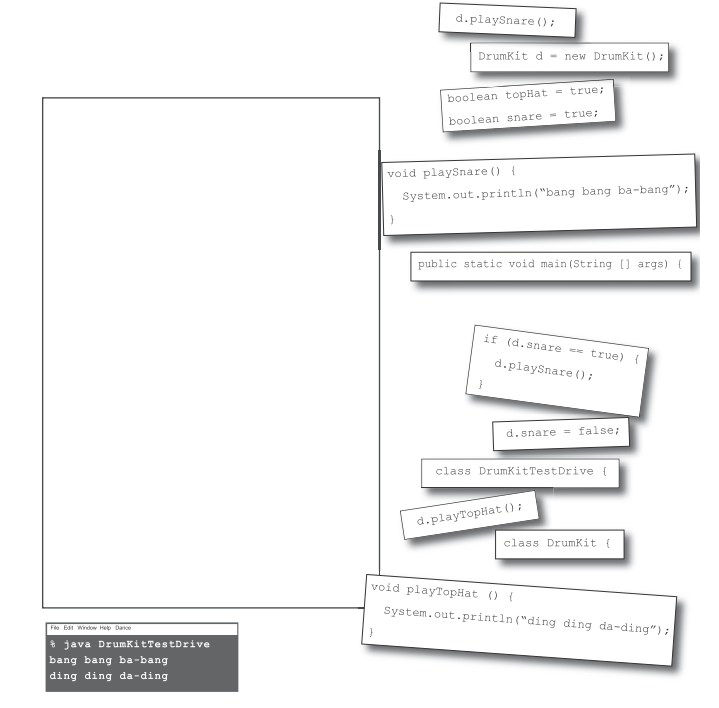
**I. Fill in the Blanks (20 \*1 = 20)**

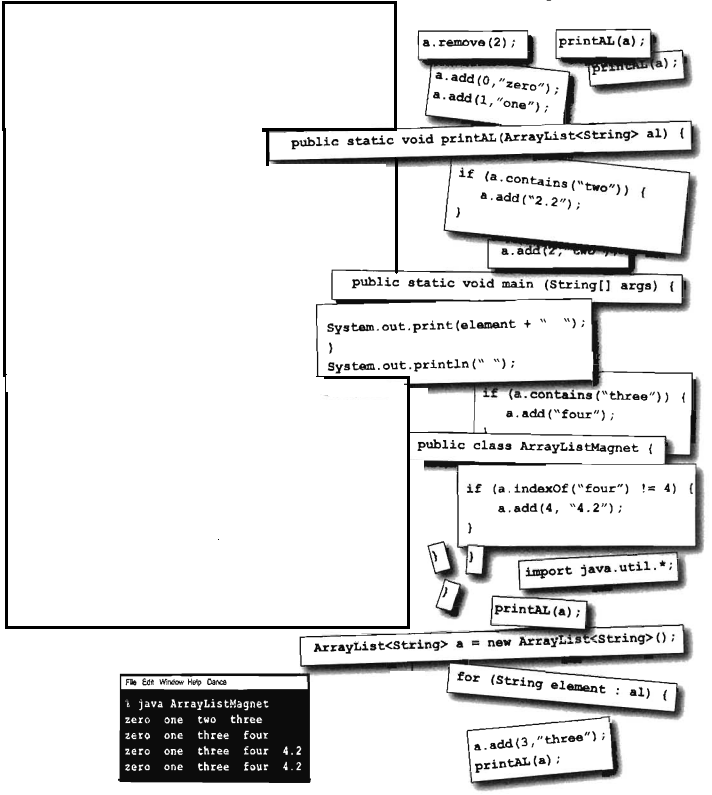
1. Static methods only access \_\_\_\_\_\_\_\_\_\_\_\_\_ and \_\_\_\_\_\_\_\_\_
2. \_\_\_\_\_\_\_\_\_operator is used to create objects and \_\_\_\_\_\_\_operator is used to access the members of the class.
3. A \_\_\_\_\_\_\_\_\_ is a group of objects that have same properties, common behaviour and common relationships.
4. \_\_\_\_\_\_\_\_\_\_\_\_ is a process that allows selective hiding or exposure of properties and methods.
5. All the classes and interfaces which are essential for performing input-output operations are defined in \_\_\_\_\_\_\_\_\_\_\_ package.
6. \_\_\_\_\_\_\_\_\_\_\_\_\_is called as ' pure abstract class '.
7. In call by reference \_\_\_\_\_\_\_\_\_\_ is passed as an argument to a method.
8. Overloading is \_\_\_\_\_\_\_ binding whereas overriding is \_\_\_\_\_ binding
9. \_\_\_\_\_\_\_\_\_\_\_\_\_is the property that allows the reuse of an existing class to build a new class.
10. A method is made to return a reference by writing \_\_\_\_\_\_\_\_\_\_\_\_before its name.
11. The \_\_\_\_\_\_\_ access modifier is the most restrictive access modifier.
12. A class cannot override a method if it is declares as \_\_\_\_\_\_\_\_\_\_\_.
13. \_\_\_\_\_\_\_\_\_ Exception is thrown by the read () method of InputStream class.
14. \_\_\_\_\_\_\_\_\_and \_\_\_\_\_\_\_\_\_\_\_ classes are designed for character streams.
15. The process of writing the state of an object to a file is referred as \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
16. The \_\_\_\_\_\_\_\_method is used to specify the minimum capacity of the StringBuffer object
17. Wrapper classes are found in \_\_\_\_\_ package.
18. Thread class’s \_\_\_\_\_\_\_\_\_\_\_\_\_ method is used to find out the priority given to a thread.
19. The value returned by read() method when the end of file (EOF) is encountered is \_\_\_\_\_\_\_\_\_\_.
20. \_\_\_\_\_\_\_\_ Package includes language utility classes, such as time, date, random number, hash tables etc.

**II. Answer any 5 of the following with an example Program. (5 \* 7 = 35)**

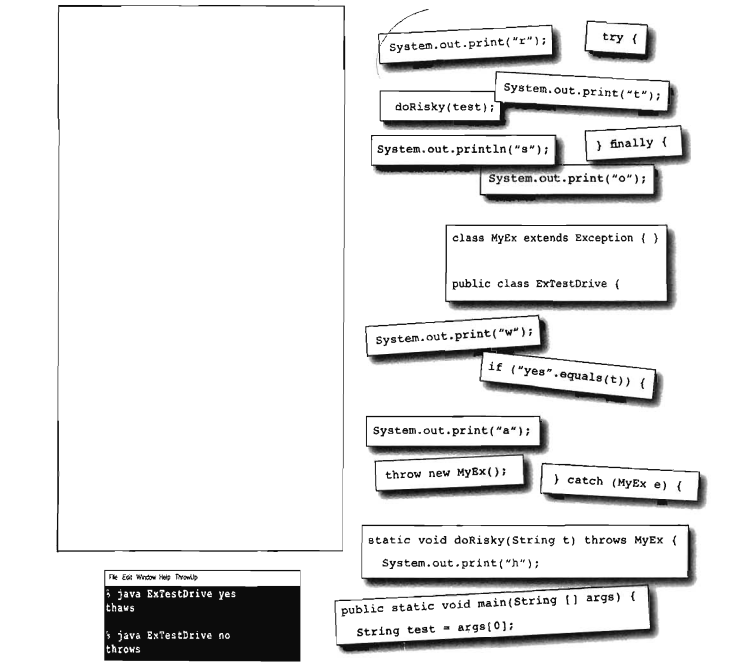
1. Exception Handling
2. Interfaces
3. Abstract classes
4. Creating Threads
5. Method Overriding
6. Types of Constructors
7. Creating User defined Pack

**III. CODING FROM GIVEN SNIPPETS (3 \* 5 = 15)**

1. Can you create a working Java Program from the given code snippets.
2. Create a working program from the following code snippets



1. Reconstruct the program to get the following output from the given code snippets



**IV. DEBUGGING AND TRACING (15\*2=30)**

**1. Identify the correct line**

public class ICFS {

// INSERT CODE HERE

{

System.out.println ("ICFS");

}

}

A. public void main (String[] args)

B. public void main(String args[])

C. static public void main (String[] array)

D. public static void main (String args)

**2. given the following code, select the correct options:**

class Course {

String courseName;

}

class ICFS {

public static void main(String args[]) {

Course c = new Course ();

c.courseName = "Java";

System.out.println (c.courseName);

}

}

Which of the following statements will be true if the variable courseName is defined as a private variable?

A. class ICFS will print Java.

B. class ICFS won’t compile.

C. class ICFS will print null.

D. class ICFS will throw an exception at runtime.

**3. given the following code, select the correct options:**

package com.ICFS.courses;

class Course {

public String courseName;

public void setCourseName(private String name) {

courseName = name;

}

}

A. You can’t define a method argument as a private variable.

B. A method argument should be defined with either public or default accessibility.

C. For overridden methods, method arguments should be defined with protected accessibility.

D. None of the above.

**4. Which of the options are correct for the following code?**

public class Prim { // line 1

public static void main(String[] args) { // line 2

char a = 'a'; // line 3

char b = -10; // line 4

char c = '1'; // line 5

Integer d = 1000; // line 6

System.out.println (++a + b++ \* c - d); // line 7

} // line 8

} // line 9

A. Code at line 4 fails to compile.

B. Code at line 5 fails to compile.

C. Code at line 6 fails to compile.

D. Code at line 7 fails to compile.

**5. Given the following definition of class Person**

class Person {

public String name;

public int height;

}

what is the output of the following code?

class ICFSPassObjects1 {

public static void main(String args[]) {

Person p = new Person();

p.name = "EJava";

anotherMethod(p);

System.out.println(p.name);

someMethod(p);

System.out.println (p.name);

}

static void someMethod(Person p) {

p.name = "someMethod";

System.out.println (p.name);

}

static void anotherMethod(Person p) {

p = new Person ();

p.name = "anotherMethod";

System.out.println (p.name);

}

}

1. anotherMethod

anotherMethod

someMethod

someMethod

1. anotherMethod

EJava someMethod someMethod

1. anotherMethod

EJava someMethod EJava

1. Compilation error.

**6. What is the output of the following code?**

class ICFSPassPrim {

public static void main(String args[]) {

int ejg = 10;

anotherMethod(ejg);

System.out.print (ejg);

someMethod(ejg);

System.out.print(ejg);

}

static void someMethod(int val) {

++val;

System.out.print (val);

}

static void anotherMethod(int val) {

val = 20;

System.out.print(val);

}

}

A. 20 10 11 11

B. 20 20 11 10

C. 20 10 11 10

D. Compilation error

**7. What is the output of the following code?**

class Course {

void enroll(long duration) {

System.out.println ("long");

}

void enroll(int duration) {

System.out.println ("int");

}

void enroll(String s) {

System.out.println ("String");

}

void enroll(Object o) {

System.out.println ("Object");

}

}

class ICFS {

public static void main(String args[]) {

Course course = new Course ();

char c = 10;

course.enroll(c);

course.enroll ("Object");

}

}

A. Compilation error B. Runtime exception

C. int String D. long Object

**8. What is the output of the following code?**

class ICFSArray {

public static void main(String args[]) {

int[] arr = new int[5];

byte b = 4; char c = 'c'; long longVar = 10;

arr[0] = b;

arr[1] = c;

arr[3] = longVar;

System.out.println (arr[0] + arr[1] + arr[2] + arr[3]);

}

}

A. 4c010 B. 4c10

C. 113 D. Compilation error

**9.** class Book {

private int pages = 100;

}

class Magazine extends Book {

private int interviews = 2;

private int totalPages() { /\* INSERT CODE HERE \*/ }

public static void main(String[] args) {

System.out.println (new Magazine().totalPages());

}

}

A. return super.pages + this.interviews\*5;

B. return this.pages + this.interviews\*5;

C. return super.pages + interviews\*5;

D. None of the above

**10. What is the output of the following code?**

class ICFSString {

public static void main(String args[]) {

String ejg1 = new String("E Java");

String ejg2 = new String("E Java");

String ejg3 = "E Java";

String ejg4 = "E Java";

do

System.out.println (ejg1.equals (ejg2));

while (ejg3 == ejg4);

}

}

A. true printed once

B. false printed once

C. true printed in an infinite loop

D. false printed in an infinite loop

**11. What is the output of the following?**

class EBowl {

public static void main(String args[]) {

String eFood = "Corn";

System.out.println (eFood);

mix(eFood);

System.out.println (eFood);

}

static void mix(String foodIn) {

foodIn.concat("A");

foodIn.replace('C', 'B');

}

}

A. Corn BornA

B. Corn Corn

C. Corn CornA

D. Corn Born

**12. What’s the output of the following code?**

class Loop2 {

public static void main(String[] args) {

int i = 10;

do

while (i++ < 15)

i = i + 20;

while (i < 2);

System.out.println (i);

}

}

A.10 B. 30

C.31 D. 32

**13. What is the output of the following**

class ICFS{

public static void main(String args[]) {

int num = 120;

switch (num) {

default : System.out.println("default");

case 0: System.out.println("case1");

case 10\*2-20: System.out.println("case2");

break;

}

}

}

A. default case1 case2

B. case1 case2

C. case2

D. Compilation error

**14. What is the output of the following code?**

class Animal {

void jump() { System.out.println("Animal"); }

}

class Cat extends Animal {

void jump(int a) { System.out.println("Cat"); }

}

class Rabbit extends Animal {

void jump() { System.out.println("Rabbit"); }

}

class Circus {

public static void main(String args[]) {

Animal cat = new Cat();

Rabbit rabbit = new Rabbit();

cat.jump();

rabbit.jump();

}

}

1. Animal

Rabbit

1. Cat

Rabbi

1. Animal

Animal

1. None of the above

**15. What is the output of the following code:**

class Course {

String courseName;

Course() {

Course c = new Course();

c.courseName = "Oracle";

}

}

class ICFSPrivate2 {

public static void main(String args[]) {

Course c = new Course ();

c.courseName = "Java";

System.out.println (c.courseName);

}

}

A. The code will print Java.

B. The code will print Oracle.

C. The code will not compile.

D. The code will throw an exception or an error at runtime.