Indian Institute of Technology Kharagpur
Department of Computer Science and Engineering
Software Engineering Lab (CS29006)
Instructor: Prof. Debasis Samanta

Lab 3: Programming in JAVA **January 22, 2020**

========== **Dealing with Exception handling** ==========

1. Run the following program with the suggested input.

```
/* This program will show run-time errors for certain input. */

class Error {
      public static void main (String args [ ]) {
            int a = Integer.parseInt(args[0]};
            int b = Integer.parseInt(args[1]};
            int c = a/b;
            System.out.println("Value of c =" + c);
       }
}
```

Run this program with the following input:

```
java Error 1 2
java Error 10 20 30
java Error 40
java Error 4.5 5
```

2. Revise the Program 1 with the try-catch block and then run the same with the same input.

3. Rewrite the following program with try-catch-finally blocks

```
/* Multiple errors handling… */
class MultipleErrors   {
    public static int j;
    public static void main (String args[ ] ) {
       for (int i = 0; i < 4; i++ ) {
             switch (i) {
                   case 0 :
                         int zero = 0;
                         j = 999/ zero; // Divide by zero
                         break;
                   case 1:
                         int b[ ] = null;
                         j = b[0] ; // Null pointer error
                         break;
                   case 2:
                         int c[] = new int [2] ;
                         j = c[10]; // Array index is out-of-bound
                         break;
                   case 3:
                         char ch = "Java".charAt(9) ;// String index is out-of-bound
                         break;
                   }
             }
       }
}
```

Indian Institute of Technology Kharagpur
Department of Computer Science and Engineering
Software Engineering Lab (CS29006)
Instructor: Prof. Debasis Samanta

Lab 3: Programming in JAVA **January 22, 2020**

========== **Dealing with Information hiding** ==========

4. Carefully examine the program and then try to run. Check, why some statements are not executable. The statements, which are erroneous are marked in red coloured font for your quick understanding.

```
/*
This program shows all combinations of the access control modifiers. In this
program, we define two packages and five classes.

MyPackage1:
      class X
      class Y  extends X
      class A

MyPackage2:
      class Z extends X
      class B

The packages and the classes in them are shown in the following.
*/

/* Define class X in the package MyPackage1 */

package MyPackage1;

public class X {
      int n = 1;
      private int p = 2;
      protected int q = 3;
      public int r = 4;
      // A constructor of the class protection
      public X() {
            system.out.println("I am constructor from class X:");
            system.out.println("n="+n);
            system.out.println("p="+p);
            system.out.println("q="+q);
            system.out.println("r="+r);
      }
}

//Save this as X.java in Mypackage1 directory

/* Define class Y in the package MyPackage1.   */

package MyPackage1;

class Y extends X {
      Y() {
            system.out.println("I am constructor from class Y:");
```

```
            system.out.println("n="+n);
            system.out.println("p="+p); // Error p is a private
                        // member of X. Not accessible outside X.
            system.out.println("q="+q);//Protected is accessible
            system.out.println("r="+r);   // public is accessible
      }
}

//Save this as Y.java in the Mypackage1 directory


/*Save the following as A.java in the Mypackage1 directory */

class A {           // class with default protection
      A() {             // default constructor with default access
            X x = new X();    // create an object of class X
            System.out.println("Same package constructor ....");
            System.out.println("n from A"+x.n);
            // Default variable is accessible in the same package
            System.out.println("p from A"+x.p);// Error
            System.out.println("q from A"+x.q);//Error rotection
               System.out.println("r from A"+x.r); // OK: public
      }
}

/*Save the class as Z.java in the Mypackage2 directory */

 package MyPackage1;
   class Z extends MyPackage1.X {
      Z() {
          System.out.println("I am constructor from class Z:");
          System.out.println("n from Z"+n); // Error:
          // Default is not accessible outside its package.
            System.out.println("p   from   Z"+p);   //Error:private   of   X
            System.out.println("q from Z"+q);
            // Protected member is accessible by inheritance
          System.out.println("r from Z"+r);
            // public is accessible
        }
    }


/* Save this as B.java in the Mypackage2 directory */

class B {       // class with default protection
       B() {     // default constructor with default access
           MyPackage1.X x = new MyPackage1.X();
           // create an object of class X
           System.out.println("I am constructor from class B of
          MyPackage2");
            System.out.println("n from B of myPackage2"+x.n);
         //default variable but is not accessible in this package
```

```
            System.out.println("p from B of myPackage2"+x.p);
            // Error
            System.out.println("q from B of myPackage2"+x.q);
            // Error protection
            System.out.println("r from A of myPackage2"+x.r);
      }
}


/* Finally, run the above defined classes from any working directory…*/

//This is the demo of MyPackage1

import MyPackage1.*;

public class Demo1{
      public static void main(String args[])
        X x1 = new X();
        Y y1 = new Y();
        A a1 = new A();
}


//This is the demo of MyPackage2

import MyPackage2.*;
public class Demo2{
      public static void main(String args[])
        Z z2 = new Z();
        B b2 = new B();
}
```

========== **Creating persistent objects for your programs** ==========

5. Write a JAVA program to do the following:

   a) Create a file which will store some text reading from the keyboard.

   b) Given one file, make a copy of the same file.

   c) Define a class with three fields and a constructor (to initialize the objects taking values from the keyboard) and a method to print the values in the fields.

   d) Create 10 objects and then store the objects in a file.

   e) Read 10 objects from the file and print the objects on the screen then.

----------****----------