Connecting Intelligence

Institute of Computer Engineering Technology

iCET Certified Developer

ASSIGNMENT

Assignement	Object Oriented Programming
Batch No	iCD 110
Name	Encapsulation
Ass. Date	13th May 2024

01. Briefly explain of following terms with appropriate examples.

a. Class/ Template	b. Object/ Instance
c. Methods/ Functions	d. Attributes/ Properties
e Reference Variables	f. Primitive Variables
g. Method Parameters	h. Local Variables
i. Default Values	j. Declaration Values.
k. Constructor	l. Instance Blocks
m. Static Variables	n. Instance Variables

02. Given Code:

```
//--------------------Test.java-----------------------
class Test{
    int a=8;
    int b=9;
}
//--------------------Demo.java-----------------------
class Demo{
    public static void main(String args[]){
       System.out.print(a);
       System.out.println(b);
    }
}
```

What is the result of attempting to compile and run the program?

A. 89 B. 00 C. 88

D. Compile time error E. None of the above

03. Given Code:

```
//--------------------Test.java-----------------------
class  Test{
   int a=8;
   int b=9;
}
//--------------------Demo.java-----------------------
class Demo{
   public static void main(String args[]){
      Test t1=new Test();
      System.out.print(t1.a);
      System.out.println(t1.b);
   }
}
```

What is the result of attempting to compile & run the program?

04. Briefly explain the difference between “attributes” and “methods” in java Object using real world examples.

05. Which of the following lines are illegal? Explain your answer.

```
class Box{
   int length;
   int width;
   int height;
}
class Demo{
   public static void main(String args[]){
      Box b1=new Box();// 1
      System.out.println("length of box : "+length); //2
      System.out.println("width  of box : "+width);  //3
       System.out.println("height of box : "+height); //4
   }
}
```

06. What is the output of the following program? Explain your answer.

```
//--------------------Box.java-----------------------
class Box{
    int length;
    int width;
    int height;
    void printVolume(){
        int volume;
        volume=length*width*height;
        System.out.println("Volume : "+volume);
    }
}
//--------------------Demo.java-----------------------
class Demo{
    public static void main(String args[]){
        Box b1=new Box();
        b1.printVolume();
        System.out.println("length of box : "+b1.length);
        System.out.println("width  of box : "+b1.width);
        System.out.println("height of box : "+b1.height);
    }
}
```

07. What is the output of the following program? Explain your answer.

```
//--------------------Box.java-----------------------
class Box{
    int length=12;
    int width=5;
    int height=3;
    void printVolume(){
        int volume;
        volume=length*width*height;
        System.out.println("Volume : "+volume);
    }
}
//--------------------Demo.java-----------------------
class Demo{
    public static void main(String args[]){
        Box b1=new Box();
        b1.printVolume();
        System.out.println("length of box : "+b1.length);
        System.out.println("width  of box : "+b1.width);
        System.out.println("height of box : "+b1.height);
    }
}
```

08. What is the output of the following program? Explain your answer.

```
//--------------------Box.java-----------------------
class Box{
    int length;
    int width;
    int height;
    Box(){
        System.out.println("default constructor");
        length=2;
        width=2;
        height=2;
    }
}
//--------------------Demo.java-----------------------
class Demo{
    public static void main(String args[]){
        Box b1=new Box();
        System.out.println("length of box : "+b1.length);
        System.out.println("width  of box : "+b1.width);
        System.out.println("height of box : "+b1.height);
    }
}
```

09. What is the output of the following program? Explain your answer.

```
//--------------------Box.java-----------------------
class Box{
    int length;
    int width;
    int height;
    Box(int l,int w,int h){
        System.out.println("Parameterized constructor");
        length=l;
        width=w;
        height=h;
    }
    void printVolume(){
        int volume;
        volume=length*width*height;
        System.out.println("Volume : "+volume);
    }
}
//--------------------Demo.java-----------------------
class Demo{
    public static void main(String args[]){
        Box b1=new Box(5,4,3);
        b1.printVolume();
        Box b2=new Box(12,5,3);
        b2.printVolume();
    }
}
```

10. What is the output of the following program? Explain your answer.

```
class Box{
   int length;
   private int width;
   int height;
}
//--------------------Demo.java-----------------------
class Demo{
   public static void main(String args[]){
      Box b1=new Box();
      System.out.println("length of box : "+b1.length);
      System.out.println("width  of box : "+b1.width);
      System.out.println("height of box : "+b1.height);
   }
}
```

11. Which of the following lines are illegal? Explain your answer.

```
class Box{
    int length;     //Line 1
    int width;      //Line 2
    int height;     //Line 3

    length=12;      //Line 4
    width=5;        //Line 5
    height=3;       //Line 6
}
```

12. What is the purpose of the constructor in a java Template (Class)? Explain your answer the suitable examples.

13. Briefly explain the difference between “constructor s” and “methods” in java Object using real world examples.

14. Which of the following lines of code could be inserted at line 12 and still allow the code to compile?

```
class Box{
    int length;
    int width;
    int height;
}
class Demo{
    public static void main(String args[]){
       //Insert code here
    }
}
```

A. Box b1; B. Box b1=new Box(); C. new Box();
D. Box b1=new Box(12,5,3); E. Box b1=new Box(10);

15. What is the output of the following program? Explain your

```
class Box{
 int length;
 int width;
}
class Demo{
    public static void main(String args[]){
       Box b1=new Box();
       System.out.print(b1.length);          //Line 1
       System.out.print(b1.WIDTH);   //Line 2
       System.out.print(b1.height);   //Line 3
    }
}
```

16. What is the output of the following program?

```
class  Test{
   int a=3;
   int b=4;
   Test(int i,int j){a=i;b=j;}
   Test(){}
}
class Demo{
   public static void main(String args[]){
      Test t1=new Test(1,2);
```

```
        System.out.print(t1.a+" "+t1.b+" ");
        Test t2=new Test();
        System.out.println(t2.a+" "+t2.b);
    }
}
```

A. 4 4 2 2 B. 4 3 2 1 C. 1 2 3 4 D. 4 3 2 1

17. Which of the following lines of code could be inserted at line 10 to get the following output?

Prints "Values 100 200"

```
class MyClass{
/*  * Insert Code Here Line 10  */
    void printValues(){
        System.out.println("Values : "+x+", "+y);
    }
}
class Demo{
    public static void main (String []args){
        MyClass c=new MyClass(100,200);
        c.printValues();
    }
}
```

A.
```
int x;
int y;
MyClass(int i, int j){   x=i;   y=j;  }
```

B.
```
int x=100;
int y=200;
MyClass(int i, int j){   }
```

C.
```
int x;
int y;
MyClass(int i,int j){   i=x;   j=y;  }
```

D.
```
int x;
int y;
MyClass(int i, int j){   x=100;   y=200;
}
```

18. What will be the output when you compile and run the following program? Explain your answer.

```
class A{
    int a;
    A(int i){     a=i; }
}
class Demo{
    public static void main(String args[]){
        A a1=new A(100);
        A a2=new A(101);
        System.out.println(a1.a+" "+a2.a);
        a1=a2;
        a1.a=0;
        System.out.println(a1.a+" "+a2.a);
    }
}
```

19. What will be the output when you compile and run the following program? Explain your answer.

```
class Item{
    int code;
    Item(int i){     code=i;}
    void printCode(int code){
        System.out.println("Code : "+code);
    }
}
class Demo{
    public static void main(String args[]){
        Item t1=new Item(2001);
        t1.printCode(3001);
    }
}
```

20. What will be the output when you compile and run the following program? Explain your answer.

```
class Item{
    int code;
    Item(int i){    code=i;}
    void printCode(int code){
        System.out.println("Code : "+this.code);
    }
}
class Demo{
    public static void main(String args[]){
        Item t1=new Item(2001);
        t1.printCode(3001);
    }
}
```

21. Which of the following lines of code could be inserted at line 12 and still allow the code to compile? Explain your answers.

```
class A{
   int a;
   //Line 12
}
class Demo{
   public static void main(String args[]){
    A a2=new A();
   }
}
```

A. A(int i){a=i;}
B. void A(int i){a=i;}
C. A(){}
D. Insert nothing

22. Which of the following lines of code could be inserted at line 12 to get output "100" for the following java application? Explain your answers.

```
class A{
    int a;
    //
    void printValue(){
        System.out.println(a);
    }
}
class Demo{
    public static void main(String args[]){
        A a1=new A(100);
        a1.printValue();
    }
}
```

A. A(int i){a=i;}
B. A(int a){a=a;}
C. A(int a){a=this.a;}
D. A(int a){this.a=a;}
E. A(int i){a=100;}
F. A(int i){i=a;}

23. Describe "encapsulation" the key concept in the Object Oriented Programming with appropriate examples.

24. What is difference between "Tightly encapsulated" and "Loosely encapsulated"? Explain your answer with appropriate examples.

25. Given:

```
class Test{
    int a=100;              // Line 1
    System.out.print(a);  //Line 2
}
class DemoTest{
    public static void main(String args[]){
        Test t=new Test();
    }
}
```

What is the result of attempting to compile and run the program?

A. Prints: 100
B. Prints: 0
C. Compiler error at line 1.
D. Compiler error at line 2

26. Create fully encapsulated class "Date" with the following functionalities.

```
//--------------------Date.java-----------------------
class Date{
    int year=1970;
    int month=1;
    int day=1;
}
//--------------------Demo.java-----------------------
class Demo{
    public static void main(String args[]){
        Date d1=new Date();
        d1.printDate(); //1970-1-1
        d1.year=2016;       //Illegal
        d1.month=5;                 //Illegal
        d1.day=30;          //Illegal
        //year, month and day attributes cannot be accessed to another class
        d1.setYear(2016);
        d1.setMonth(5);
        d1.setDay(31);
        System.out.println("Year    : "+d1.getYear());
        System.out.println("Month :"+d1.getMonth());
        System.out.println("Day     : "+d1.getDay());
    }
}
```

27. Explain the following lines of code according to class "Customer" is given bellow.

```
class Customer{
        private String id;
        private String name;
        public Customer(){}
        public Customer(String id, String name){
            this.id=id;
            this.name=name;
        }
        public void printCustomer(){
            System.out.println(id+" - "+name);
        }
        public void setCustomerDetail(String id, String name){
            this.id=id;
            this.name=name;
        }
        public void setId(String id){
            this.id=id;
        }
        public void setName(String name){
            this.name=name;
        }
}
//--------------------DemoCustomer.java--------------------
class DemoCustomer{
        public static void main(String args[]){
        Customer c1;                                    //Line 1
        c1=new Customer("C001", "Danapala");            //Line 2
        c1.printCustomer();                             //Line 3
        Customer c2;                                    //Line 4
        c2=new Customer();                              //Line 5
        c2.setCustomerDetail("C002", "Gunapala");       //Line 6
        c2.printCustomer();                             //Line 7
        Customer c3;                                    //Line 8
        c3=new Customer();                              //Line 9
        c3.setId("C003");                               //Line 10
        c3.setName("Somapala");                         //Line 11
        c3.printCustomer();                             //Line 12
    }
}
```

28. What will be the output when you compile and run the program?

```
class Test{
    int x=10;
    static int y=20;
}
class Demo{
    public static void main(String args[]){
        System.out.println(MyClass.x);  //Line 1
        System.out.println(MyClass.y);  //Line 2
        MyClass c1=new MyClass();
        System.out.println(c1.x);       //Line 3
        System.out.println(c1.y);       //Line 4
    }
}
```

A. Compile Error at line 1 B. Compile Error at line 2
C. Compile Error at line 4 D. Compile Error at line 3

29. What will be the output when you compile and run the program?

```
class MyClass{
    int x;
    static int y;
}
class Demo{
    public static void main(String args[]){
        MyClass c1=new MyClass();
        MyClass c2=new MyClass();
        MyClass c3=new MyClass();
        c1.x=1;  c1.y=2;   c2.x=10;
        c2.y=20;
        c3.x=100;
        c3.y=200;
        System.out.println(c1.x+" "+c1.y+" "+c2.x+" "+c2.y+" "+c3.x+" "+c3.y);
    }
}
```

A. 1 2 10 20 100 200 B. 1 1 10 10 100 100 C. 1 2 10 2 20 2
D. 1 100 10 100 100 100 E. 1 200 10 200 200 200 F. None of the above

30. Create a class Rectangle with attributes length and width, each of which defaults to 1. Provide methods that calculate the rectangle's perimeter and area. It has set and get methods for both length and width. The set methods should verify that length and width are each floating-point numbers larger than 0.0 and less than 20.0. Write a program to test class Rectangle.

31. What will be the output when you compile and run the program?

```
class Demo{
    static void staticMethod(){
      System.out.println("staticMethod");
    }
    void instanceMethod(){
      System.out.println("instanceMethod");
    }
    public static void main(String args[]){
      instanceMethod();         //Line 1
      staticMethod();    //Line 2
      Demo.instanceMethod();          //Line 3
      Demo.staticMethod();    //Line 4
      Demo d1=new Demo();  //Line 5
      d1.instanceMethod();      //Line 6
      d1.staticMethod();    //Line 7
    }
}
```

A. Compile Error at line 1
B. Compile Error at line 2
C. Compile Error at line 3
D. Compile Error at line 4
E. Compile Error at line 5
F. Compile Error at line 6

32. What will be the output when you compile and run the program?

```
class Test{
    int x=10;
    static int y=20;
    Test(int i, int j){   x=i;    y=j; }
    void printXY(){
      System.out.print(x+" "+y+" ");
    }
}
class Demo{
    public static void main(String args[]){
      Test t1=new Test(1,2);
      Test t2=new Test(10,20);
      Test t3=new Test(100,200);
      t1.printXY();
      t2.printXY();
      t3.printXY();
    }
}
```

A. 1 2 10 20 100 200
B. 1 1 10 10 100 100
C. 1 2 10 2 20 2
D. 1 100 10 100 100 100
E. 1 200 10 200 200 200
F. None of the above

33. Which of the following code lines are illegal?

```
class Test{
    int x=10;
    static int y=20;
    static void staticMethod(){}
    void instanceMethod(){}
    void mA(){
      System.out.println(x);              //Line 1
      System.out.println(y);              //Line 2
      staticMethod();       //Line 3
      instanceMethod();         //Line 4
    }
    static void mB(){
      System.out.println(x);              //Line 5
      System.out.println(y);              //Line 6
      staticMethod();       //Line 7
      instanceMethod();         //Line 8
    }
}
```

34. Explain the difference between “static variables” and “instance variables”

35. Given:

```
class Demo{
    int a=10;
    static int b=20;
    public static void main(String args[]){
        int c=20;
        Demo ob=new Demo();
        System.out.println(a);                  //Line 1
        System.out.println(b);                  //Line 2
        System.out.println(c);                  //Line 3
        System.out.println(Demo.a);  //Line 4
        System.out.println(Demo.b);  //Line 5
        System.out.println(Demo.c);  //Line 6
        System.out.println(ob.a);           //Line 7
        System.out.println(ob.b);           //Line 8
        System.out.println(ob.c);           //Line 9
    }
}
```

Which of these lines will cause a compile error?

A. Compile Error at line 1
B. Compile Error at line 2
C. Compile Error at line 3
D. Compile Error at line 4
E. Compile Error at line 5
F. Compile Error at line 6
G. Compile Error at line 7
H. Compile Error at line 8
I. Compile Error at line 9

36. Extends the class "Date" of Question 26 with these given functionalities.

```
//--------------------Demo.java-----------------------
class Demo{
    public static void main(String args[]){
        Date d1=new Date();
        d1.set(Date.YEAR,2016);  //set(int field, int value)
        d1.set(Date.MONTH,5);
        d1.set(Date.DAY,30);
        d1.printDate(); //2016-5-30
    }
}
```

37. Explain the difference between “static methods ” and “instance methods”

38. What is the output, and explain the answer.

```
class Box{
    int length;
    int width;
    int height;
    static{
        System.out.println("Box is loaded into memory");
    }
}
class Demo{
    public static void main(String args[]){
        Box b1=new Box();
    }
}
```

39. What is the output, and explain the answer.

```
class Box{
    int length;
    int width;
    int height;
    {
        System.out.println("A box object is created..");
    }
}
class Demo{
    public static void main(String args[]){
        Box b1=new Box();
        Box b2=new Box();
    }
}
```

40. Write a Java class called “Cylinder” with attributes length (type double) and radius (type double) of the Cylinder.
    a. Create methods that calculate the volume and area of the Cylinder.
    b. Implements two constructors default constructor and parameterized constructor.
    c. Implements setters and getters of each attribute of the class cylinder.

    Write an application to test your class “Cylinder”

41. What is the output, and explain the answer.

```
class Box{
    int length;
    int width;
    int height;
    static{
        System.out.println("Box is loaded into memory");
    }
    {
        System.out.println("A box object is created..");
    }
}
class Demo{
    public static void main(String args[]){
        Box b1=new Box();
        Box b2=new Box();
        Box b3=new Box();
    }
}
```

42. Given:

```
class Test{
    int x=10;
    static int y=20;
    void mA(){
        System.out.println(this.x);      //Line 1
        System.out.println(this.y);      //Line 2
    }
    static void mB(){
        System.out.println(this.x);      //Line 3
        System.out.println(this.y);      //Line 4
    }
}
```

What is the result of attempting to compile and run the above program?

A. Compile error at Line 1
B. Compile error at Line 2
C. Compile error at Line 3
D. Compile error at Line 4
F. Program will run without errors

43. Given:

```
class MyClass{
    int x=10;
    static int y=20;
    MyClass(){
        System.out.print("3  ");
    }
    static{
        System.out.print("1  ");
    }
    {
        System.out.print("2  ");
    }
    static void mB(){
        System.out.print("4  ");
    }
}
class Demo{
    public static void main(String args[]){
        //Insert code here line 12
    }
}
```

Write the output generated by the given program inserting the following code fragments at line 12.

A. new MyClass();

B. new MyClass();
new MyClass();

C. MyClass.mB();

D. new MyClass().mA();

E. MyClass.mB();

F. MyClass c1;

44. Consider this class example:

```
class MyPoint {
    void myMethod() {
        int x, y;
        x = 5;
        y = 3;
        System.out.print( " ( " + x + "," + y + " ) " );
        switchCoords( x, y );
        System.out.print( " ( " + x + "," + y + " ) " );
    }
    void switchCoords( int x, int y ) {
        int temp;
        temp = x;
        x = y;
        y = temp;
        System.out.print( " ( " + x + "," + y + " ) " );
     }
}
```

What is printed to standard output if myMethod() is executed?

A. (5, 3) (5, 3) (5, 3)
B. (5, 3) (3, 5) (3, 5)
C. (5, 3) (3, 5) (5, 3)
D. (3, 3) (5, 5) (5, 3)

45. Describe "Method Call by Values" vs "Method Call by Reference" with appropriate examples

46. Which of the following lines are illegal? Explain your answer.

```
class Cat{
    private String name;
}
class Demo{
    public static void main(String args[]){
        Cat cat;        //Line 1
        Cat[] cats; //Line 2
        cat=new Cat[5];    //Line 3
        cat=new Cat(); //Line 4
        cats=new Cat();    //Line 5
        cats=new Cat[5]; //Line 6
    }
}
```

47. Insert code at line 50 to get the output as follows

[Aldo, Bear, Toby, Teddy, Henry]

```
class Cat{
  private String name;
  Cat(String name){this.name=name;}
  public String getName(){return name;}
  public void setName(String name){this.name=name;}
}
class Demo{
  public static void main(String args[]){
    Cat[] cats={   new Cat("Aldo"),
                   new Cat("Bear"),
                   new Cat("Toby"),
                   new Cat("Teddy"),
                   new Cat("Henry")    };
    //Line 50
  }
}
```

48. Create a class called Employee that includes three instance variables a first name (type String), a last name (type String) and a monthly salary (double). Provide a constructor that initializes the three instance variables. Provide a set and a get method for each instance variable. If the monthly salary is not positive, do not set its value. Write a test application named EmployeeTest that demonstrates class Employee's capabilities. Create two Employee objects and display each object's yearly salary. Then give each Employee a 10% raise and display each Employee's yearly salary again.

49. What is the result of attempting to compile and run the following program?

```
class Item{
    private int code;
    Item(int code){
     this.code=code;
    }
    void setCode(Item item){
     this.code=item.code;
    }
    void printCode(){
     System.out.println("code : "+code);
    }
}
class Demo{
    public static void main(String args[]){
     Item item1=new Item(1001);
     item1.printCode();

     Item item2=new Item(2002);
     item2.setCode(item1);
     item2.printCode();
    }
}
```

50. Extends the class "Date" of Question 36 with this given functionalities.

```
//--------------------DemoDate.java---------------------
class DemoDate{
    public static void main(String args[]){
        Date d1=new Date();
        d1.set(Date.YEAR,2016);  //set(int field, int value)
        d1.set(Date.MONTH,05);
        d1.set(Date.DAY,30);
        d1.printDate();    //2016-5-30
        Date d2=new Date(2016,1,31);
        d2.printDate();   //2016-1-31
        Date d3=new Date(d2);
        d3.printDate();   //2016-1-31
        Date d4=new Date();
        d4.set(d1);
        d4.printDate();
        Date d5=Date.getDateInstance();
        d5.printDate();    //1970-1-1
    }
}
```

51. Given the following code, what will be the output?

```
class ValHold{
   public int i = 10;
}
class ObParm{
    public static void main(String argv[]){
        ObParm o = new ObParm();
     o.amethod();
    }
    public void amethod(){
        int i = 99;
        ValHold v = new ValHold();
        v.i=30;
        another(v,i);
        System.out.println(v.i);
    }//End of amethod
    public void another(ValHold v, int i){
        i=0;
        v.i = 20;
        ValHold vh = new ValHold();
        v =  vh;
        System.out.println(v.i+ " "+i);
    }//End of another
}
```

A. 10,0, 30 B. 20,0,30 C. 20,99,30 D. 10,0,20

52. What will be the result when you compile and the following program?

```
class MyClass{
    MyClass(){
        System.out.println("MyClass()");
    }
    MyClass(int i){
        this(i,i);
        System.out.println("MyClass(int)");
    }
    MyClass(int i,int j){
        this();
        System.out.println("MyClass(int,int)");
    }
}
class Demo{
    public static void main(String args[]){
        MyClass c1=new MyClass(100);
    }
}
```

53. Complete the class Box with the given functionalities.

```
class Box{
    private int length;
    private int width;
    private int height;
    //Insert code to complete the class "Box"
}
class Demo{
    public static void main(String args[]){
        Box b1=new Box();
        b1.setLength(12);
        b1.setWidth(5);
        b1.setHeight(3);
        b1.printVolume();

        b1.setDimension(120,50,30);
        System.out.println("Volume "+b1.getVolume());

        Box b2=new Box(4,2,3);
        b2.printVolume();

        Box b3=new Box(b2);
        //copy dimensions of b2 into b3
        b3.printVolume();

        Box b4=new Box(10);//length for a square cube
        b4.printVolume(); //


        Box b5=new Box();
        b5.setDimension(12); //length for a square cube
        b5.printVolume();

        Box b6=new Box();
        b6.printVolume();
        b6.setDimension(b1);
        //copy dimensions of b1 into b6
        b6.printVolume();

        Box b7=b3.getCopy();
        b7.printVolume();
    }
}
```