Chapter 6 – Inner Classes

Question 1:

Which of the following statements are true ?

[] A) Construction of an inner class object might require an instance of the outer class

[] B) An inner class should always be anonymous

[] C) An inner class should always be declared private

[] D) An inner class defined in a method can access all the method local variables

[] E) An inner class can be declared static

Question 2:

Is the following statement true or false ?

If Inner, the inner class, is a non-static class declared inside a public class which its name is Outer than an instance of Inner can be constructed in the following way:

new Outer().new Inner();

() A) True

() B) False

Question 3:

Consider the following class definition,

```
1. public class Exterial
2.{
3.      public int var1 = 100;
4.      private int var2 = 22;
5.      public void method(final int var3)
6.      {
7.              int var4 = 30;
8.              public class Interior
9.              {
10.                     private void Method(int var5)
11.                     {
12.
13.                     }
14.             }
15.     }
```

Which variables may be referenced correctly at line 12?

[] A) var1

[] B) var2

[] C) var3

[] D) var5

[] E) var4

Question 4:

True or False:

BambaInterface is an interface. The code below can be successfully compiled.

```
1. class Bisly
2. {
3.      BambaInterface getBamaba()
4.      {
5.              return new BambaInterface(3.4)
6.              {
7.                      int numberOfCalories = 1200;
8.              };
9.      }
10.}
```

() A) true

() B) false

Question 5:

True or False:

It is possible to declare a constructor inside an anonymous inner class as it is in any other class.

() A) True

() B) False

Question 6:

True or False:

The inner class can be defined as abstract.

() A) True

() B) False

Question 7:

True or False:

An inner class can't extend another class.

() A) True

() B) False

Question 8:

True or False:

The JVM doesn't know the difference between an inner class and a normal class.

() A) True

() B) False

Question 9:

True or False:

An object that belongs to an inner class, which isn't a static class, can access all the instance variables that the outer object has.

() A) True

() B) False

Question 10:

True or False:

An object that belongs to an inner class that was declared inside a method can access all the methods local variables.

() A) True

() B) False

Question 11:

Given the code below:

```
1.  public class Popy
2.  {
3.      int luckyNumber;

4.      Popy()
5.      {
6.          luckyNumber = 72;
7.      }

8.      public int getLuckyNumber()
9.      {
10.         return 8;
11.     }

12.     class InnerPopy1 extends Popy
13.     {
14.         public InnerPopy1()
15.         {
16.             luckyNumber = 65;
17.         }
```

```
18.                 public int getLuckyNumber()
19.             {
20.                     return luckyNumber;
21.             }
22.         }
23.  }

24.         public static void main(String args[])
25.         {
26.                  Popy popy = new Popy().new InnerPopy1()
27.                 {
28.                         public int getLuckyNumber()
29.                         {
30.                                 return 10*luckyNumber;
31.                         }
32.                 };
33.                 System.out.println(popy.getLuckyNumber());
34.         }
35. }
```

The output is:

() A) 720

() B) 72

() C) 650

() D) 8

() E) 80

() F) 0

Question 12:

Given the code below:

```
1.  public class Oliv
2.  {
3.      int luckyNumber;
4.      Oliv inner;

5.      public Oliv()
6.      {
7.          luckyNumber = 14;
8.      }

9.      public void setLucky(int num)
10.     {
11.         luckyNumber = 6;
12.     }
13.     public void setInner()
14.     {
```

```
15.             inner = new Oliv()
16.             {
17.                     public void setLucky(int num)
18.                     {
19.                             luckyNumber = num;
20.                     }
21.             };
22.         }
23.         public static void main(String args[])
24.         {
25.             Oliv oliv = new Oliv();
26.             oliv.setInner();
27.             oliv.inner.setLucky(12);
28.             System.out.println(oliv.luckyNumber);
29.         }
30.  }
```

The output is:

() A) 14

() B) 12

() C) 10

() D) 8

() E) 0

() F) 6

Question 13:

Given the code below:

```
public class Oliv
{
        int luckyNumber;
        Oliv inner;
        public Oliv()
        {
            luckyNumber = 14;
        }
        public Oliv(Oliv oliv)
        {
            inner = oliv;
        }
        public Oliv(int num)
        {
            luckyNumber = num;
        }
        public void setLucky(int num)
        {
            luckyNumber = 6;
        }
```

```
        public void setInner()
        {
            inner = new Oliv()
            {
            public void setLucky(int num)
                    {
                            luckyNumber = num;
                    }
            };
        }
        public static void main(String args[])
        {
            Oliv oliv = new Oliv(new Oliv(new Oliv(34)));
            oliv.inner.inner.setInner();
            oliv.inner.inner.setLucky(12);
            System.out.println(oliv.inner.inner.luckyNumber);
        }
}
```

The output is:

() A) 14

() B) 12

() C) 10

() D) 8

() E) 0

() F) 6

Question 14:

Given the code below:

```
public class Oliv
{
    int luckyNumber;
    Oliv inner;
    public Oliv()
    {
        luckyNumber = 14;
    }
    public Oliv(Oliv oliv)
    {
        inner = oliv;
    }
    public Oliv(int num)
    {
        luckyNumber = num;
```

```
    }
    public void setLucky(int num)
    {
        luckyNumber = 6;
    }

    public void setInner()
    {
        inner = new Oliv()
        {
        public void setLucky(int num)
            {
                luckyNumber = num;
            }
        };
    }
    public static void main(String args[])
    {
        Oliv oliv = new Oliv(new Oliv(new Oliv(new Oliv(35))));
        oliv.inner.inner.setInner();
        oliv.inner.inner.inner.setLucky(12);
        System.out.println(oliv.inner.inner.inner.luckyNumber);
    }
}
```

The output is:

() A) 14

() B) 12

() C) 10

() D) 8

() E) 0

() F) 6

Question 15:

Given the code below:

```
public class InnerParty
{
        String name;
        int val;
        InnerParty inner;

        InnerParty()
        {
        }

        InnerParty(String name, int val)
```

```
{
        this.name = name;
        inner = new InnerParty()
        {
                void doSomething()
                {
                        this.name = InnerParty.this.name + 98;
                        this.val = this.val * 2;
                }
        };
}

void doSomething()
{
        this.name = "MALKISHUA";
        inner.name = "BATSHEVA";
}

String getInner()
{
        return inner.name;
}

public static void main(String args[])
{
```

```
        String strs[] = {"Dany","Haim","Liat","Ronit","David"};
        InnerParty vec[] = new InnerParty[strs.length];

        for(int index=0; index<vec.length; index++)
        {
            vec[index] = new InnerParty(strs[index],index);
            vec[index].inner.doSomething();
        }

        System.out.println(vec[1].getInner());
    }

}
```

The output is:

() A) Haim98

() B) MALKISHUA

() C) BATSHEVA

() D) The doesn't compile