Q1.
```
class box {
    int width;
    int height;
    int length;
}
class mainclass {
    public static void main(String args[])
    {
        box obj = new box();
        System.out.println(obj);
    }
}
```

Q2.
```
class overload {
    static  int x;
double y;
    void add(int a , int b) {
        x = x+ a + b;
    }
    void add(double c , double d){
        y = c + d;
    }
    overload() {
        this.x = 20;
        this.y = 10.0;
    }
}
class Output {
    public static void main(String args[])
    {
        overload obj = new overload();
        overload obj1 = new overload();
        int a = 2;
        double b = 3.2;
        obj.add(a, a);
        obj1.add(b, b);
        System.out.println(obj.x + " " + obj.y);
       System.out.println(obj1.x + " " + obj1.y);
        }}
```

Q3.
```
class test {
        int a;
        int b;
        test(int i, int j) {
            a = i;
            b = j;
        }
        void meth(test o) {
            o.a *= 2;
            O.b /= 2;
        }
    }
  class Output
{
 public static void main(String args[])
 {
 test obj = new test(10 , 20);
 obj.meth(obj);
 System.out.println(obj.a + " " + obj.b);
        }
    }
```

Q4.
```
class Main {
  public static void main(String args[]) {
    System.out.println(fun());
  }
  int fun() {
    return 20;
   } }
```

Q5.

```
class Bitwise
{
   public static void main(String [] args)
   {
      int x = 11 & 9;
      int y = x ^ 3;
      System.out.println( y | 12 );
   }
}
```

Q6.

```
class Test
{    public void display(int a, double b)
     {
        System.out.println(" This is Double");
     }
 public void display(int a, float b)
     {
        System.out.println(" This is Float");
     }
     public static void main(Stirng args[])
       {  Test t = new Test();
          t.display(10, 5);       }}
```

Q7. class Overload

```
{
      static  int x;
        double y;
      static { x =10;}
      void add(double c , double d){
         x = (x+c*d);
         y = (c + d)*x;
      }
      Overload() {
              this.y = 10.0;
               }
  }
  class Output {
      public static void main(String args[]){
         Overload obj = new Overload();
         Overload obj1 = new Overload();
         int a = 2; double b = 3.5;
         obj.add(a, a);   obj1.add(b, b);
        System.out.println(obj.x + " " + obj.y);
        System.out.println(obj1.x + " " + obj1.y);
      }
 }
```

Q8. public class A
{
   void A()
   {
      System.out.println("Class A");
   }
   public static void main(String[] args)
   {
      new A();
   }
}

Q9. class A {
   final public int GetResult(int a, int b)
       { return 0; }
}
class B extends A
{ public int GetResult(int a, int b) {return 1; }
}
public class Test
{ public static void main(String args[])
   {
      B b = new B();
      System.out.println("x = " + b.GetResult(0, 1));
   }
}

Q10.

```
class Output {
public static void main(String args[])
{
      short  x = 0x8000;
      byte   y =  010;
      System.out.println(x + " and  "+y);
      x = x >>> 15;
      y = y<<4;
     System.out.println(x + " and  "+y);
}
```

Q10. Needed Type casting

-32768 and 8

-1 and -128

Q1. Answer: Memory address

Q2. Answer: 24 10.0
24 6.4

Q3. Answer: 20 10