1. Fill out the following table

	Definition	Purpose and/or how it is dealt with
Analysis and Design		
documentation		
prototyping		
Iterative software development		
Syntax errors		
Logic errors		
Runtime errors		
exceptions		
Exception handling		
Testing		
Debugging		
Refactoring		

2. What is the output of the following Java program?

```
public class Trick
{
     public static void main(String[] args)
     {
          int i = 10, n = 0;

          for (; n<10; n++)
          {
               i = n/2;
          }

          System.out.print(i + i + "" + n + n);
          System.out.println(i + "" + (n + n));
     }
}
```

3. Convert the for-loop construct in the following snippet of Java code into an equivalent while-loop construct.

```
int j = 0;
for (int i = 100; i > 0; i -= 5)
{
     j += 3 * i;
}
System.out.println(j);
```

4. Write a **complete** Java program that displays the string “Hello World!” to the console. Assume that the filename of this program is HelloWorld.java.

5. List the order of the statements in the following Java program as they would be executed **and** show what the program would output.

```
     abstract class Pet
     {
       protected String name;
       protected int age;

       public Pet(String a, int b)
       {
1:         name = a;
2:         age = b;
       }

       public void grow()
       {
3:         age++;
       }

       public String toString()
       {
4:         return name + ":" + age + " years old.";
       }
     }

     class Cat extends Pet
     {
       public Cat(String name, int age)
       {
5:         super(name, age);
       }
     }

     class Dog extends Pet
     {
       public Dog(String name, int age)
       {
6:         super(name, age);
       }
     }

     class Home
     {
       public static void main(String [] args)
       {
7:         Pet mydog = new Dog("Spike", 5);
8:         Pet mycat = new Cat("Tom", 3);

9:         System.out.println(mycat + "\t" + mydog);

10:        for (int i = 0; i < 2; i++)
           {
11:          mydog.grow();
12:          mycat.grow();
           }
13:        System.out.println(mycat + "\t" + mydog);
       }
     }
```