Assignment 10

```java
public class SingletonTest {
    static final SingletonTest t1 = null;
    private SingletonTest() {
        //   to prevent instantion
    }

    public static SingletonTest getSingleTonInstance() {
        if(t1 == null) {
            return new SingletonTest();
        }
        return t1;
    }
}
```

SingletestMain.java

```java
public class SingletonMain {

    public static void main(String[] args) {

        SingletonTest test1 =
SingletonTest.getSingleTonInstance();

        SingletonTest test2 =
SingletonTest.getSingleTonInstance();

        if(test1.equals(test2)) {
            System.out.println("Both Instance are same ");
        }else {
            System.out.println("Both Instance are
different");
        }
        //Having same objects means having same Hashcode
```

```
		System.out.println("test1 
hashCode() :"+test1.hashCode()+" test2 
hashCode() :"+test2.hashCode());
	}

}
```

```
Output :-

Both Instance are different
test1 hashCode() :250075633 test2 hashCode() :358699161
```

SingletonTest.java

```
public class SingletonTest {
	static final SingletonTest t1 = new SingletonTest();
	private SingletonTest() {
		//	to prevent instantion
	}

	public static SingletonTest getSingleTonInstance() {
		return t1;
	}
}
```

SingletonMain.java

```
package Amdocs;

public class SingletonMain {

	public static void main(String[] args) {

		SingletonTest test1 = 
SingletonTest.getSingleTonInstance();

		SingletonTest test2 = 
SingletonTest.getSingleTonInstance();

		if(test1.equals(test2)) {
```

```
              System.out.println("Both Instance are same ");
         }else {
              System.out.println("Both Instance are
different");
         }
         //Having same objects means having same Hashcode
         System.out.println("test1
hashCode() :"+test1.hashCode()+" test2
hashCode() :"+test2.hashCode());
    }

}
```

```
output :-

Both Instance are same
test1 hashCode() :250075633 test2 hashCode() :250075633
```