LAB 7- GENERICS USING MULTIPLE OBJECT PARAMETERS

```
class FourGen<T, V, W, X> {
T ob1;
V ob2;
W ob3;
X ob4;
FourGen(T o1, V o2, W o3, X o4) {
ob1 = o1;
ob2 = o2;
ob3 = o3;
ob4 = o4;
}
void showTypes() {
System.out.println("Type of T is " +ob1.getClass().getName());
System.out.println("Type of V is " +ob2.getClass().getName());
System.out.println("Type of W is " +ob3.getClass().getName());
System.out.println("Type of X is " +ob4.getClass().getName());
}
T getob1() {
return ob1;
}
V getob2() {
return ob2;
}
W getob3() {
return ob3;
}
X getob4() {
return ob4;
}
```

```
}
class SimpGen {
public static void main(String args[]) {
FourGen<Integer, String, Double, Long> tgObj =
new FourGen<Integer, String, Double, Long>(19, "Anitej", 23.8765, 327666L);
tgObj.showTypes();
int v = tgObj.getob1();
System.out.println("value: " + v);
String str = tgObj.getob2();
System.out.println("value: " + str);
Double dbl= tgObj.getob3();
System.out.println("value:" + dbl);
Long lng=  tgObj.getob4();
System.out.println("value:" + lng);


}
}
```