

27/11/2020

Lab 7

Varsha S 1BM19CS179 3D Batch-1

// A simple generic class with two type parameters T and V.

```
class TwoGen < T, V > {
```

```
    T ob1;
```

```
    V ob2;
```

// Pass the constructor a reference to an object of type T and an object of type V.

```
TwoGen ( T o1 , V o2 ) {
```

```
    ob 1 = o1;
```

```
    ob 2 = o2;
```

```
}
```

// Show types of T and V

```
void Show Types () {
```

```
    System.out.println("Type of T is" + ob 1.getClass().  
                        getName());
```

```
    System.out.println("Type of V is" + ob 2.getClass().  
                        getName());
```

```
}
```

```
T get Ob1 () {
```

```
    return ob1 ;
```

```
}
```

```
V get Ob2 () {
```

```
    return Ob2 ;
```

```
}
```

```
}
```

```
// Demonstrate TwoGen
```

```
class SimpGen {
```

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

```
        TwoGen < Integer, String > tg Obj =
```

```
        new TwoGen < Integer, String > (88, "Generics");
```

```
        // show the types
```

```
        tg Obj.showTypes();
```

Teacher's Signature : \_\_\_\_\_

// obtain and show values.

```
int v = tg.Obj.get ob1();
```

```
System.out.println("value : " + v);
```

```
String str = tg.Obj.get ob2();
```

```
System.out.println("value : " + str);
```

```
}
```

```
}
```