

Program to demonstrate generics with
multiple 3 type parameters

SUSHMITHA Y.V
1BM19CS165
3D
A

```
System.out.println("Type of 2nd parameter is " + obj2.  
getclass().getName());
```

```
System.out.println("Type of 3rd parameter is "  
+ obj3.getclass().getName());
```

```
}
```

```
/*
```

```
public void print()
```

```
{
```

```
System.out.println(obj1);
```

```
System.out.println(obj2);
```

```
System.out.println(obj3);
```

```
}
```

```
*/
```

```
T getobj1()
```

```
{  
return obj1;
```

```
  
getobj2()
```

```
{  
return obj2;
```

```
class Generics < T, U, S >
```

```
{  
    T obj 1;  
    U obj 2;  
    S obj 3;  
    /* Generics (T obj, U obj 2, S obj 3)
```

```
{  
    this.obj 1 = obj 1;  
    this.obj 2 = obj 2;  
    this.obj 3 = obj 3;  
}
```

```
*/  
Generics (T obja, U objb, S, objc)
```

```
{  
    obj 1 = obja;  
    obj 2 = objb;  
    obj 3 = objc;
```

```
}  
void show Types()
```

```
{  
    system.out.println (" Type of 1st parameter is " + obj 1.  
        getClass(). getName());
```

```
5 getobj3()  
{  
    return obj3;  
}  
}
```

```
class Genericmain  
{  
    public static void main(String[] args)  
    {  
        String a, c;  
        int b;  
        Generic < Strings, Integer,  
String > obj = new Generic < Strings,  
Integer, String > ("sushmita", 2989, "IBM19CS165");  
        obj.showTypes();  
  
        a = obj.getobj1();  
        b = obj.getobj2();  
        c = obj.getobj3();  
        System.out.println(a + " " + b + " " + c);  
        //obj.print();  
    }  
}
```