Generics

Majority java developer jab bhi ye terminology suntey hai then, sense of insecurity feel hota hai, aaj hum samjhengey ki generic ku chahiyey
Generic ka literal meaning he ye hai ki jo sabkey liyey common ho,
Lets take an example,

```java
public class normalClass {

    public static void main(String args[]){
        Names sample = new Names();
        sample.set("hello");    //note : sirf string pass kar saktey hai
        System.out.println(sample.get());

     }

    public static class Names {
        private String t;
        public void set(String t) { this.t = t; }
        public String get() { return t; }
    }
}
```

output:
hello

In the above example, ye clearly note kar saktey hai ki names mein hum apart string kuch aur nahi bhej saktey, but what if we want to pass other types or an object

Think ….?

Possible solution
1. Make the type as object
        Problem: ki har baar casting karana padega
Best Solution,
Use Generics

```java
public class normalClass {

    public static void main(String args[]){
        Names sample = new Names();
        sample.set("hello");
        System.out.println(sample.get()); // output:  hello

        Names sample1= new Names();
        sample1.set(1);
        System.out.println(sample1.get()); //output : 1
    }

    public static class Names<T> {
        private T t;
        public void set(T t) { this.t = t; }
        public T get() { return t; }
```

```
    }
}
```

Note: now humara problem solve ho gaya, ab aap koi bhi type pass kar saktey ho, and woh use samajh lega kuki ek naya syntax hai **<T> ,** we can use any variable.

But important ye hai ki this makes programming very generic and as a developer this helps in designing good framework.