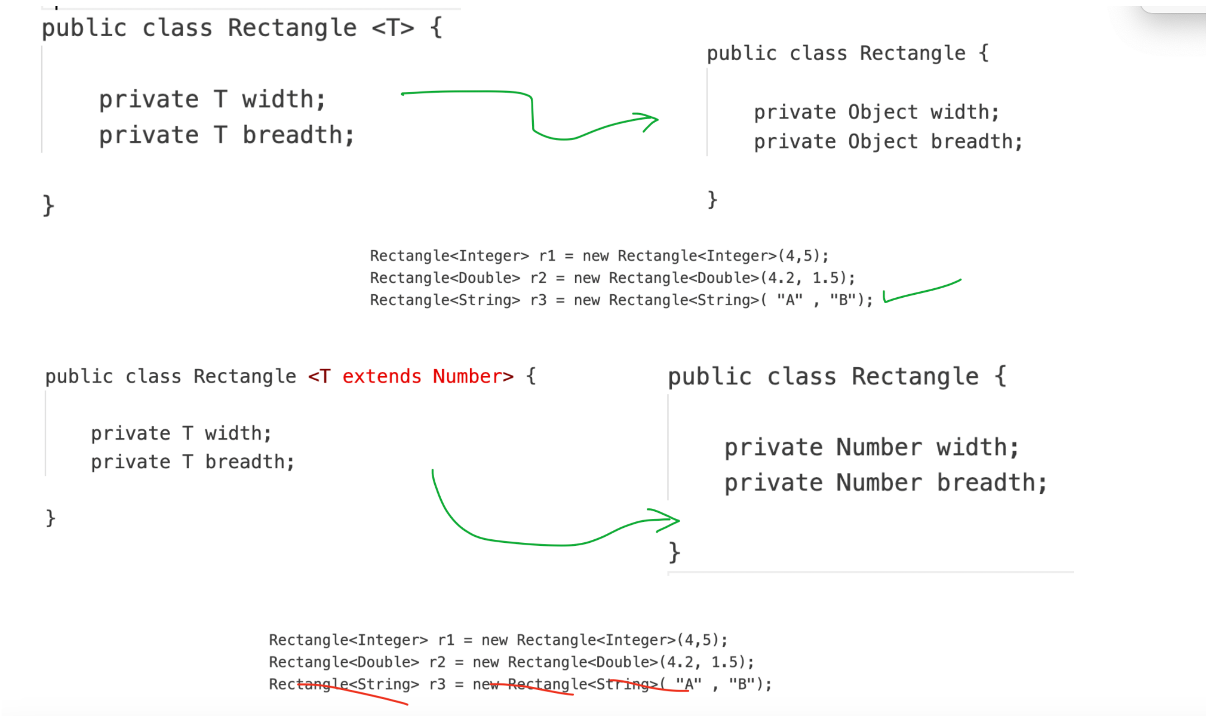


```
public class Rectangle <T> {

    private T width;
    private T breadth;

}
```

```
public class Rectangle {

    private Object width;
    private Object breadth;

}
```

```
Rectangle<Integer> r1 = new Rectangle<Integer>(4,5);
Rectangle<Double> r2 = new Rectangle<Double>(4.2, 1.5);
Rectangle<String> r3 = new Rectangle<String>( "A" , "B");
```

```
public class Rectangle <T extends Number> {

    private T width;
    private T breadth;

}
```

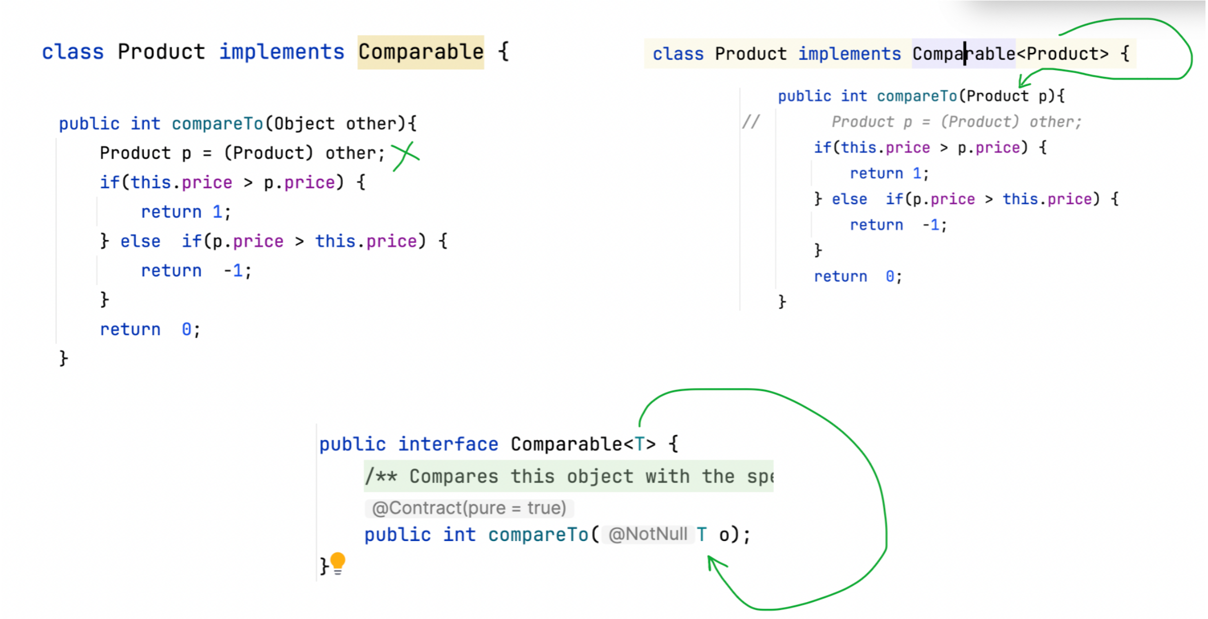
```
public class Rectangle {

    private Number width;
    private Number breadth;

}
```

```
Rectangle<Integer> r1 = new Rectangle<Integer>(4,5);
Rectangle<Double> r2 = new Rectangle<Double>(4.2, 1.5);
```

~~Rectangle<String> r3 = new Rectangle<String>("A"~~ , "B");



```
class Product implements Comparable {

 public int compareTo(Object other){
     Product p = (Product) other;
     if(this.price > p.price) {
         return 1;
     } else  if(p.price > this.price) {
         return  -1;
     }
     return  0;
 }
```

```
class Product implements Comparable<Product> {

    public int compareTo(Product p){
//        Product p = (Product) other;
        if(this.price > p.price) {
            return 1;
        } else  if(p.price > this.price) {
            return  -1;
        }
        return  0;
    }
```

```
public interface Comparable<T> {
    /** Compares this object with the spe
    @Contract(pure = true)
    public int compareTo( @NotNull T o);
}
```