Annotations provice data about a program that is ****not**** part of the program it self.

Annotations have no direct effect on the operation of the code they annotate. That is to say, annotation does not have any functionality utill it is read by some other tools or code.

// the override annotation is to tell the java compiler that the method is a @Override //overrding method, the compiler will double check.public int hashCode(){

return 0;}

Annotations can be put on classes, methods, variables, parameters.

Declear a annotation using keyword @interface.

// java source code: Override.java

@Target(ElementType.METHOD)@Retention(RetentionPolicy.SOURCE)public @interface Override {}

// We can create customized annotation@Target({ElementType.TYPE, ElementType.METHOD})@Retention(RetentionPolicy.RUNTIME)public @interface AntraAnnotation {

String name() default ""; // declear attribute}

// Target is to declear where the annotation can be put on.// Retention is to declear until when this annotation can be read. Indicates how long annotations with the annotated type are to be retained.

// Now we can put this annotation on any class/interface and method. // But it doesn't do anything.@AntraAnnotationclass TestClass{

@AntraAnnotation

public void testMethod(){}}

// Using reflection api in java, we can write code to read the annotation and do something.// Annotation is like marker of our code.