Annotation Assignment

package LambdaTest;

import java.util.\*;

import org.junit.\*;

public class A1{

@Test

public void test() {

System.out.println("This is an example");

}

}

1. Creating custom annotation @Info and providing the information into the class

package LambdaTest;

import java.lang.\*;

import java.lang.annotation.Annotation;

import java.lang.annotation.ElementType;

import java.lang.annotation.RetentionPolicy;

import java.lang.annotation.Target;

import java.lang.annotation.Retention;

import java.util.\*;

@Target(ElementType.TYPE)

@Retention(RetentionPolicy.RUNTIME)

@interface Info{

int Authorid() default 124;

String Authorname() default "Sravya";

String Date()default "2022-12-22";

String Time() default "09:36 pm";

}

@Info

class info1{

int Authorid;

String Authorname;

String Date;

String Time;

}

public class A2 {

public static void main(String[] args) {

// TODO Auto-generated method stub

info1 obj = new info1();

Class c = obj.getClass();

Annotation an = c.getAnnotation(Info.class);

Info in = (Info)an;

System.out.println(in.Authorid()+" "+in.Authorname()+" "+in.Date()+" "+in.Time());

}

}

1. Creating Custom annotation @Execute that to be applied on methods

package LambdaTest;

import java.lang.\*;

import java.lang.annotation.\*;

import java.lang.annotation.Annotation;

import java.lang.annotation.ElementType;

import java.lang.annotation.RetentionPolicy;

import java.lang.annotation.Target;

import java.lang.reflect.Method;

import java.lang.annotation.Retention;

import java.util.\*;

@Target(ElementType.METHOD)

@Retention(RetentionPolicy.RUNTIME)

@interface Execute{

int sequence();

}

class Myclass{

@Execute(sequence=2)

public void Method1(){

System.out.println("Method 1");

}

@Execute(sequence=1)

public void Method2(){

}

@Execute(sequence=3)

public void Method3(){

}

}

public class A3 {

public static void main(String[] args) {

// TODO Auto-generated method stub

Myclass obj = new Myclass();

//Reflection API

Method c = obj.getClass().getMethod("Method1");

Execute ex = c.getAnnotation(Execute.class);

System.out.println("Sequence"+ex.sequence());

}

}