@interface Test{

String test\_case();

}

@Test(test\_case = "1")

class Example{

//@Test is implying only one method "test\_case"//

}

@Test(test\_case = "A2")

class Example2{

//Again @Test is implying only one method "test\_case"//

}

Annotations 2

//Creating annotation//

@interface info

{

}

//Using annotation @info

@info

class Details

{

int AuthorId;

String Author;

String Date;

String Time;

float Version;

public Details(int authorId, String author, String date, String time, double d)

{

AuthorId = authorId;

Author = author;

Date = date;

Time = time;

Version = (float) d;

}

}

public class Question2

{

public static void main(String[] args)

{

Details author1= new Details(101, " Madara Uchiha", " 2/11/2021", " 12:54pm ", 1.0);

Details author2= new Details(102, " Hashirama Senju", " 3/12/2021", " 3.30am ", 1.1);

Details author3= new Details(102, " Kakashi Hatake", " 7/04/2022", " 6.37pm ", 1.2);

Details author4= new Details(102, " Naruto Uzumaki", " 11/09/2022", " 1.17pm ", 1.3);

System.out.println(author1.AuthorId +author1.Author +author1.Date +author1.Time +author1.Version);

System.out.println(author2.AuthorId +author2.Author +author2.Date +author2.Time +author2.Version);

System.out.println(author3.AuthorId +author3.Author +author3.Date +author3.Time +author3.Version);

System.out.println(author4.AuthorId +author4.Author +author4.Date +author4.Time +author4.Version);

}

}

ANNOTAIONS 3

import java.lang.annotation.Annotation;

import java.lang.annotation.ElementType;

import java.lang.annotation.Retention;

import java.lang.annotation.RetentionPolicy;

import java.lang.annotation.Target;

@Target(ElementType.TYPE)

@Retention(RetentionPolicy.RUNTIME)

@interface Execute{

int Sequence();

}

@Execute(Sequence = 1)

class myMethod1{

}

@Execute(Sequence = 2)

class myMethod2{

}

@Execute(Sequence = 3)

class myMethod3{

}

public class Question3{

public static void main(String[] args)

{

myMethod2 b =new myMethod2();

myMethod1 a =new myMethod1();

myMethod3 c =new myMethod3();

Class cb= b.getClass();

Class ca= a.getClass();

Class cc= c.getClass();

Annotation an1 = cb.getAnnotation(Execute.class);

Annotation an2 = ca.getAnnotation(Execute.class);

Annotation an3 = cc.getAnnotation(Execute.class);

Execute e1= (Execute)an1;

Execute e2= (Execute)an2;

Execute e3= (Execute)an3;

System.out.println(e1.Sequence());

System.out.println(e2.Sequence());

System.out.println(e3.Sequence());

}

}