***Dt : 5/9/2022***

***faq:***

***define Access Modifiers?***

***=>Access Modifiers in Java specify the scope of programming***

***components in Project.***

***=>The following are some important Access modifiers in Java:***

***1.public***

***2.private***

***3.protected***

***4.default***

***1.public:***

***=>'public' programming components are accessed within the project.***

***2.private:***

***=>'private' programming components are accessed only inside the***

***class.***

***3.protected:***

***=>'protected' programming components are accessed within the***

***package and these components can also be accessed by the ChildClass***

***outside the package.***

***4.default:***

***=>The programming components which are declared without any***

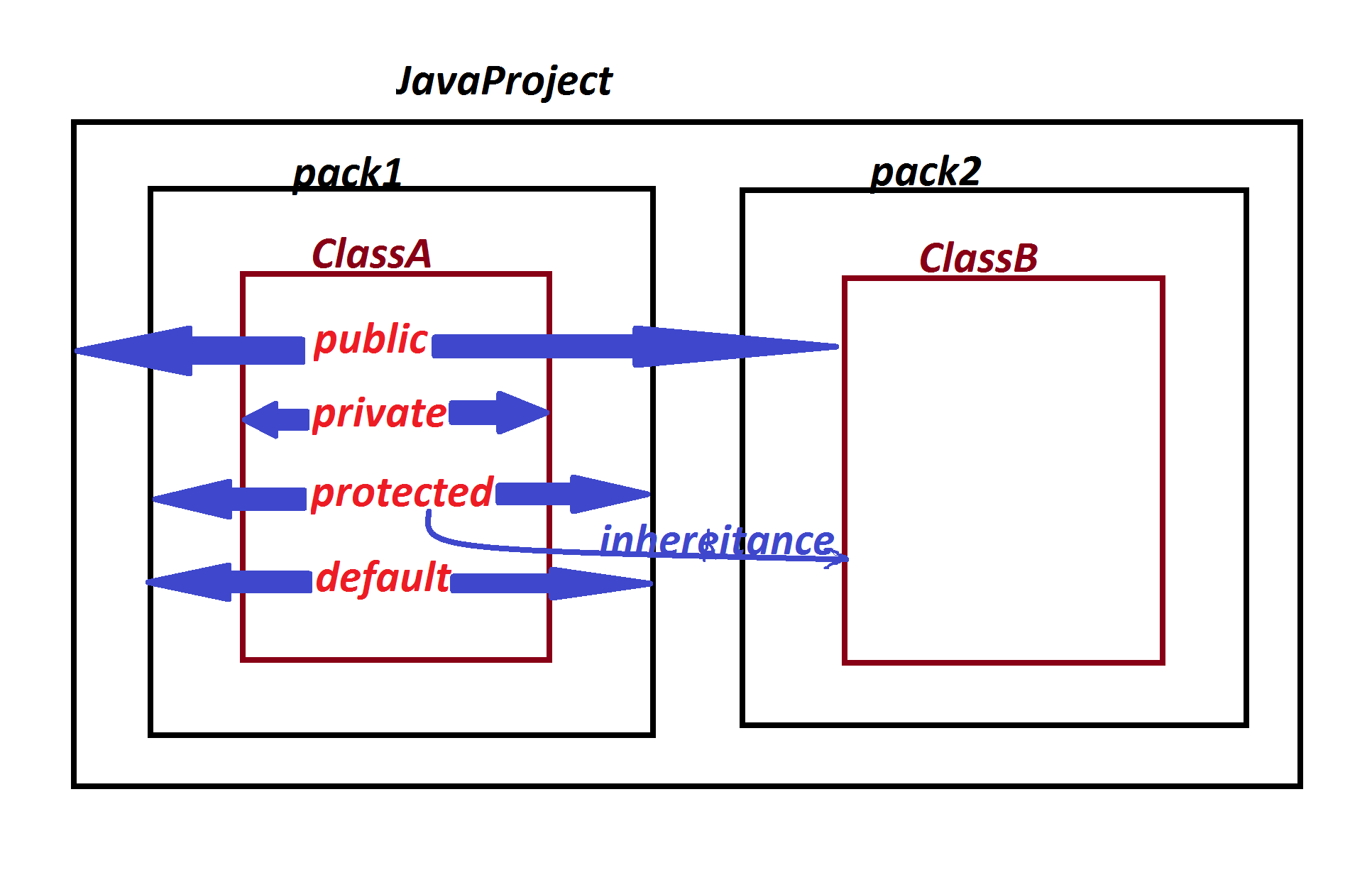
***access modifiers in Class are considered as 'default' programming***

***components.***

***=>These default programming components are accessed only inside***

***the package.***

***Diagram:***

******

***=============================================================***

***faq:***

***define 'import' statement?***

***=>'import statement' in Java will make class or interface***

***available from one package to another package.***

***=>Importing process in Java can be done in three ways:***

***(i)Using 'import package\_name.Class\_name/Interface\_name;'***

***(ii)Using 'import package\_name.\*; '***

***(iii)Using 'Fully Qualified Names'***

***(i)Using 'import package\_name.Class\_name/Interface\_name;'***

***=>In this importing process we specify the class or interface***

***available to current running program.***

***=>This importing process is also known as 'Explicit importing***

***process'.***

***Ex:***

***import java.util.Scanner;***

***import p1.CalculateSalary;***

***(ii)Using 'import package\_name.\*; ':***

***=>In this importing process all the classes and interfaces from***

***the package are available to current running program.***

***=>This importing process is also known as 'Implicit importing***

***process'.***

***Ex:***

***import java.util.\*;***

***import p1.\*;***

***(iii)Using 'Fully Qualified Names'***

***=>The proces of declaring classes and Interfaces with package***

***names part of programming code are known as 'Fully Qualified names'.***

***Ex:***

***java.util.Scanner s = new java.util.Scanner(System.in);***

***p1.CalculateSalary cs = new p1.CalculateSalary();***

***==============================================================***

***faq:***

***define 'static import'?***

***=>The process of declaring import statement with 'static' keyword***

***is known as 'static import' and which is introduced by Java5 version.***

***syntax:***

***import static package\_name.Class\_name/Interface\_name.\*;***

***Advantage:***

***=>when we use 'static import' then all the static members of***

***Class or Interface available to current running program directly***

***and the static programming components canbe accessed without***

***class\_name or Interface\_name.***

***==========================================================***

***faq:***

***define sqrt() method?***

***=>sqrt() is a pre-defined static method from java.lang.Math class***

***Method Signature:***

***public static double sqrt(double);***

***syntax:***

***double var = Math.sqrt(value);***

***DemoImport.java***

***package maccess;***

***import java.util.\*;***

***import static java.lang.Math.\*;***

***public class DemoImport {***

***public static void main(String[] args) {***

***Scanner s = new Scanner(System.in);***

***System.out.println("Enter the value:");***

***double value = s.nextDouble();***

***double v = sqrt(value);//MethodCall***

***System.out.println("Sqrt of "+value+" is "+v);***

***s.close();***

***}***

***}***

***o/p:***

***Enter the value:***

***12345***

***Sqrt of 12345.0 is 111.1080555135405***

***=========================================================***