**Day 15\_Java Assignment**

**1. Problem Description:**

What is the diamond problem in multiple inheritance and give an example?

**2. My Solution:**

Diamond Problem is a problem faced during Multiple Inheritance in Java. when one class try to inherit properties from more than one class is known as the diamond problem and Java does not allow such this multiple inheritance.

Example:

**package** daily\_assesment;

**class** Family1 {

**void** fun() {

System.***out***.println("Family1");

}

}

**class** Family2 {

**void** fun() {

System.***out***.println("Family2");

}

}

**public** **class** Parents1 **extends** Family1, Family2

{

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

Parents1 p = **new** Parents1();

p.fun();

}

}

Output:

Exception in thread "main" java.lang.Error: Unresolved compilation problems:

Syntax error on token "]", :: expected after this token

Syntax error, insert "SimpleName" to complete Type

at daily\_assesment.Parents1.main(Parents1.java:16)

* There will be ambiguity when calling the fun method from the parent class because both child classes implemented the fun method. As a result, the parent class is confused when selecting which method to choose.
* To avoid this ambiguity problem, multiple inheritance is possible only with the help of interfaces and not by classes in Java. Interfaces provide a way to achieve multiple inheritance by allowing a class to implement multiple interfaces, each defining its own set of methods.

**Example:**

**package** daily\_assesment;

**interface** Parent3 {

**void** fun();

}

**interface** Parent4 {

**void** fun();

}

**class** test **implements** Parent3, Parent4 {

**public** **void** fun()

{

System.***out***.println("fun function");

}

}

**public** **class** Diamond1 {

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

test t = **new** test();

t.fun();

}

}

Output:

fun function