

NAME – RAJDEEP JAISWAL

DATE – 29 SEPT 2021

BRANCH – BTECH CSE

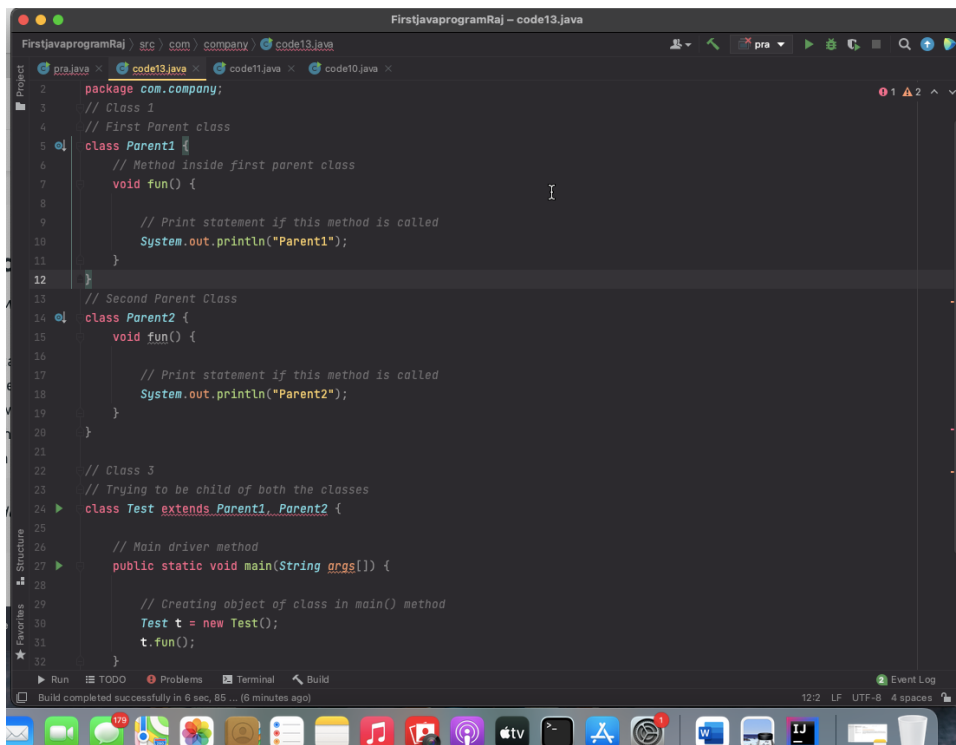
SEC = 13 A

UID -20BCS2761

SUB- JAVA LAB MST

Q - 6 Why Java Does not Support Multiple Inheritance?

Multiple Inheritance is a feature of an object-oriented concept, where a class can inherit properties of more than one parent class. The problem occurs when there exist methods with the same signature in both the superclasses and subclass. On calling the method, the compiler cannot determine which class method to be called and even on calling which class method gets the priority.



```
FirstjavaprogramRaj - code13.java
package com.company;
// Class 1
// First Parent class
class Parent1 {
    // Method inside first parent class
    void fun() {
        // Print statement if this method is called
        System.out.println("Parent1");
    }
}
// Second Parent Class
class Parent2 {
    void fun() {
        // Print statement if this method is called
        System.out.println("Parent2");
    }
}
// Class 3
// Trying to be child of both the classes
class Test extends Parent1, Parent2 {
    // Main driver method
    public static void main(String args[]) {
        // Creating object of class in main() method
        Test t = new Test();
        t.fun();
    }
}
```

The screenshot shows an IDE window titled "FirstjavaprogramRaj - code13.java". The code defines three classes: Parent1, Parent2, and Test. Parent1 and Parent2 both have a method named fun(). The Test class extends both Parent1 and Parent2. The main method in Test creates an instance of Test and calls the fun() method. The IDE shows a compilation error at line 24, indicating that the compiler cannot determine which parent class's fun() method to call because both have the same signature.

Output: Compilation error is thrown