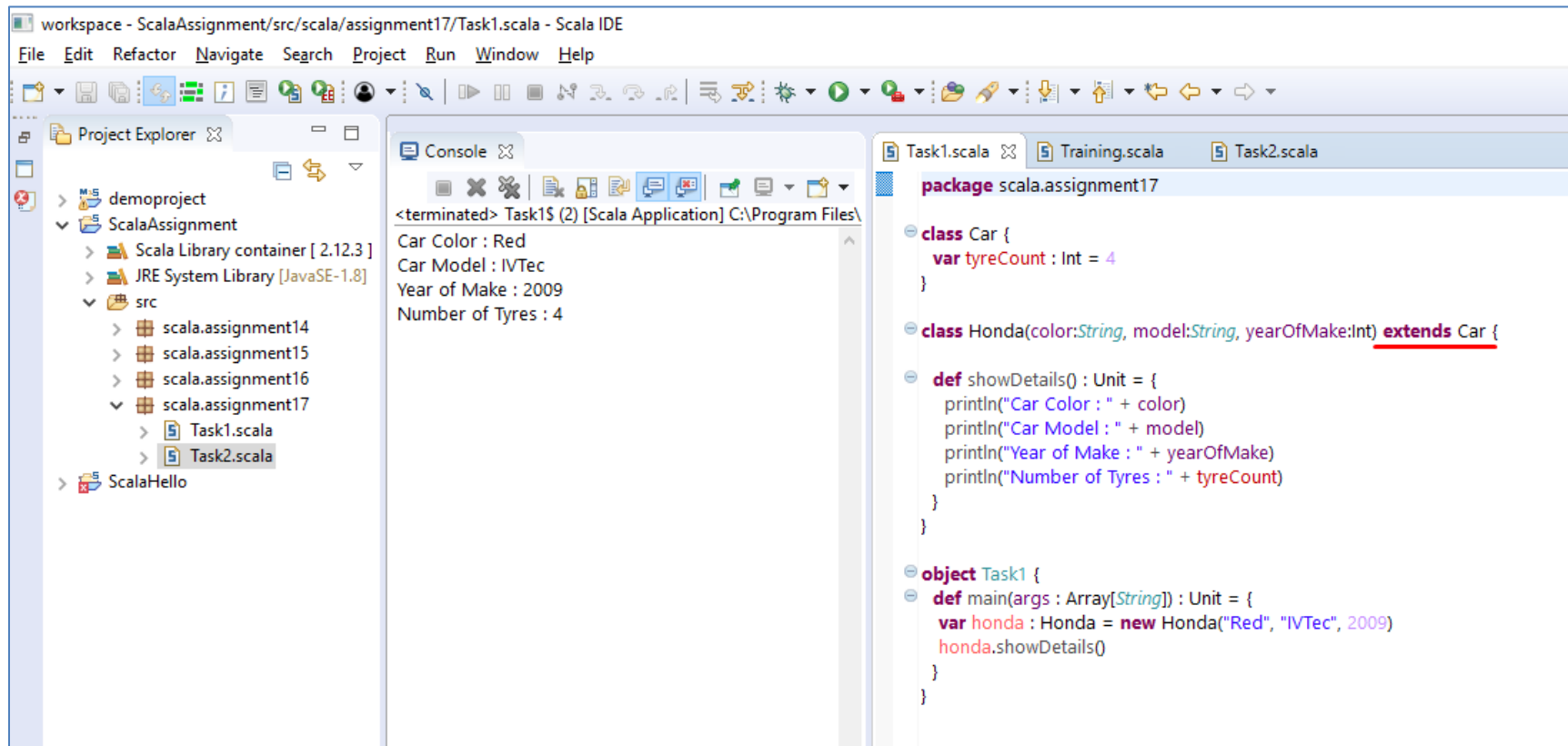


Session 17:
SCALA BASICS 4
Assignment 1

TASK 1:

Write a simple program to show inheritance in scala.

EXECUTION:



The screenshot shows the Scala IDE interface. The Project Explorer on the left displays the project structure, including the 'src' directory and the 'scala.assignment17' package. The Console in the center shows the output of the program, which is the result of running 'Task1\$ (2) [Scala Application]'. The Editor on the right shows the source code for 'Task1.scala'.

```
workspace - ScalaAssignment/src/scala/assignment17/Task1.scala - Scala IDE
File Edit Refactor Navigate Search Project Run Window Help

Project Explorer
  demoproject
    ScalaAssignment
      Scala Library container [ 2.12.3 ]
      JRE System Library [JavaSE-1.8]
      src
        scala.assignment14
        scala.assignment15
        scala.assignment16
        scala.assignment17
          Task1.scala
          Task2.scala
      ScalaHello

Console
  <terminated> Task1$ (2) [Scala Application] C:\Program Files\
  Car Color : Red
  Car Model : IVTec
  Year of Make : 2009
  Number of Tyres : 4

Task1.scala Training.scala Task2.scala
package scala.assignment17

class Car {
  var tyreCount : Int = 4
}

class Honda(color:String, model:String, yearOfMake:Int) extends Car {

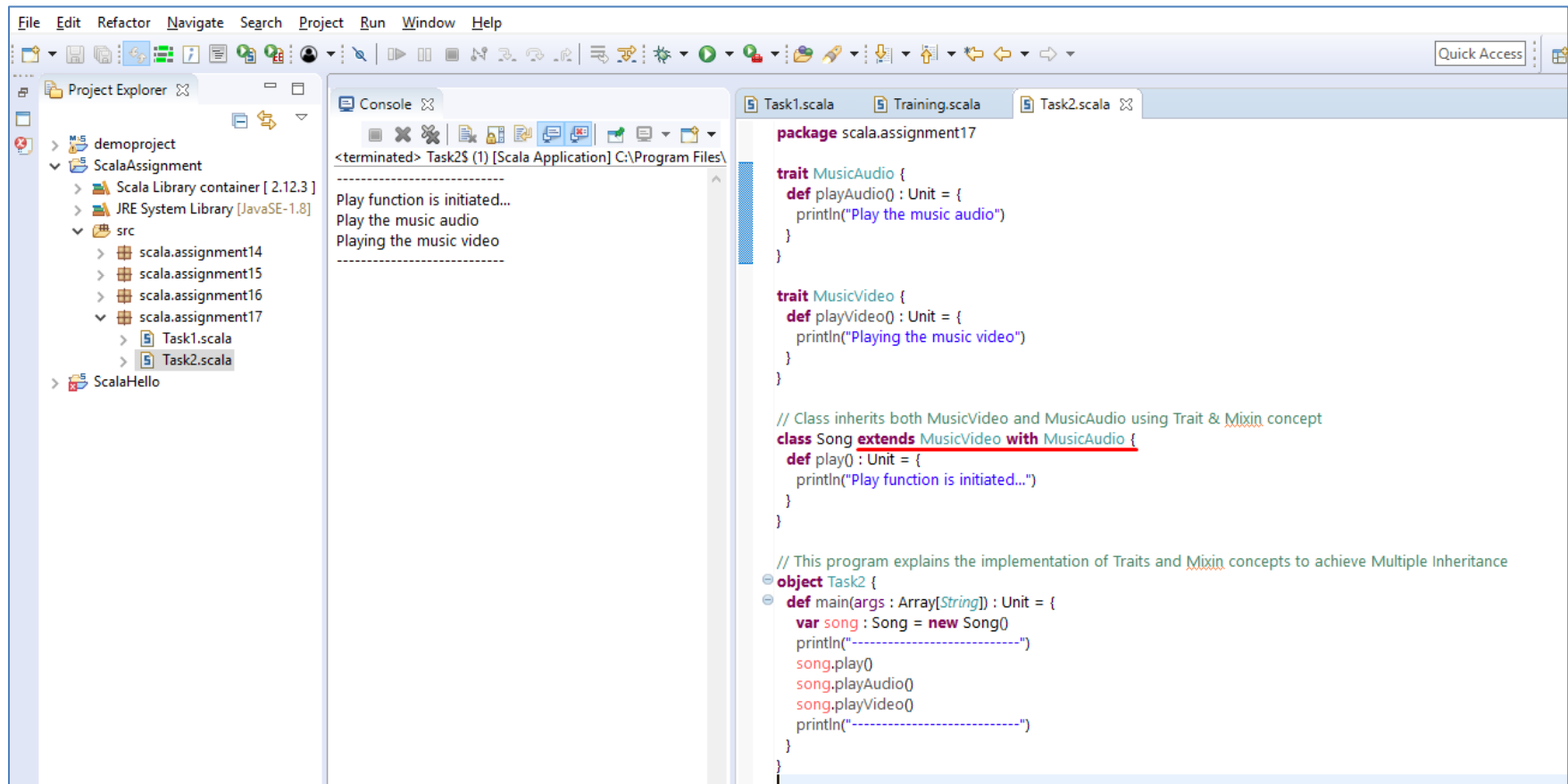
  def showDetails() : Unit = {
    println("Car Color : " + color)
    println("Car Model : " + model)
    println("Year of Make : " + yearOfMake)
    println("Number of Tyres : " + tyreCount)
  }
}

object Task1 {
  def main(args : Array[String]) : Unit = {
    var honda : Honda = new Honda("Red", "IVTec", 2009)
    honda.showDetails()
  }
}
```

TASK 2:

Write a simple program to show multiple inheritance in scala

EXECUTION:



The screenshot shows an IDE with the following components:

- Project Explorer:** Shows a project named 'demoproject' with a sub-project 'ScalaAssignment'. Under 'src', there are several 'scala.assignment' folders and two files: 'Task1.scala' and 'Task2.scala'.
- Console:** Displays the output of the program execution:

```
<terminated> Task2$ (1) [Scala Application] C:\Program Files\
-----
Play function is initiated...
Play the music audio
Playing the music video
-----
```
- Code Editor:** Shows the source code for 'Task2.scala'. The code defines two traits, 'MusicAudio' and 'MusicVideo', and a class 'Song' that inherits from both using the 'extends' and 'with' keywords. It also includes a 'main' method in an 'object Task2' that creates a 'Song' object and calls its 'play', 'playAudio', and 'playVideo' methods.

```
package scala.assignment17

trait MusicAudio {
  def playAudio() : Unit = {
    println("Play the music audio")
  }
}

trait MusicVideo {
  def playVideo() : Unit = {
    println("Playing the music video")
  }
}

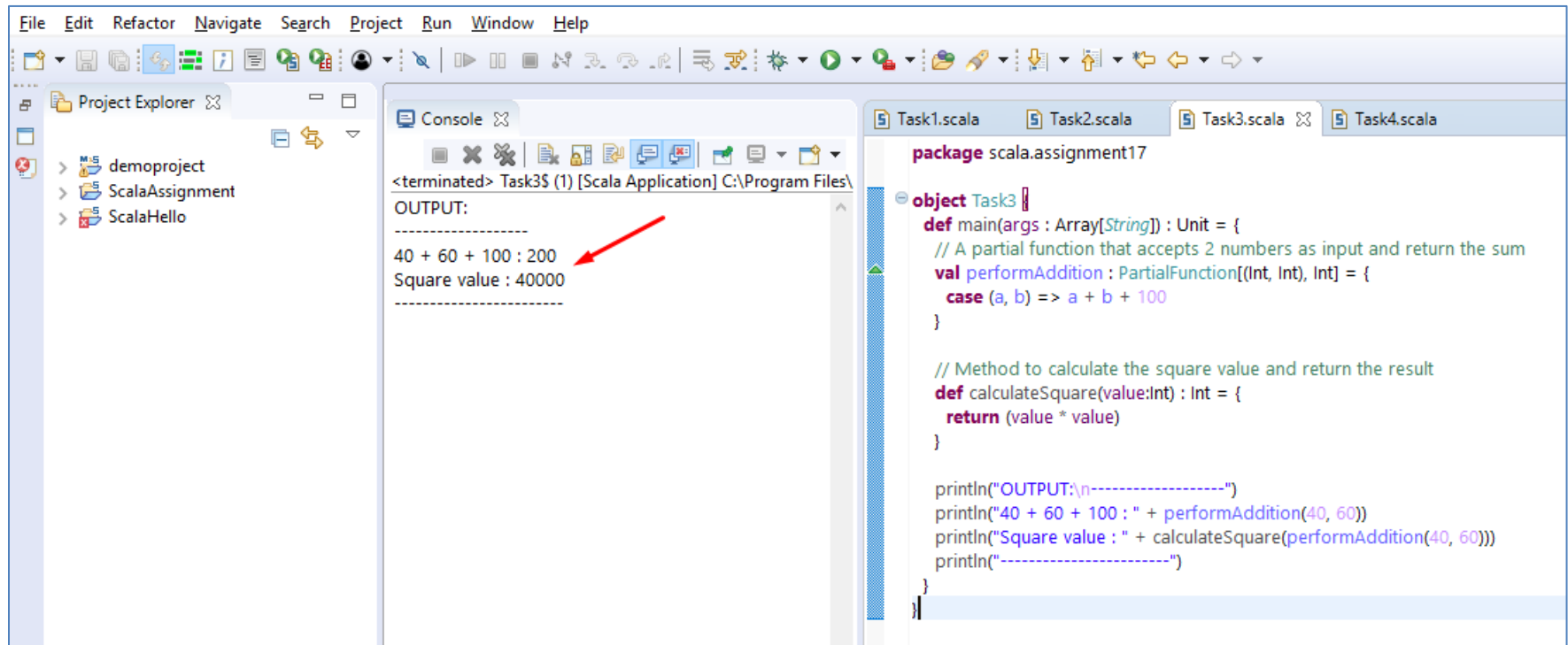
// Class inherits both MusicVideo and MusicAudio using Trait & Mixin concept
class Song extends MusicVideo with MusicAudio {
  def play() : Unit = {
    println("Play function is initiated...")
  }
}

// This program explains the implementation of Traits and Mixin concepts to achieve Multiple Inheritance
object Task2 {
  def main(args : Array[String]) : Unit = {
    var song : Song = new Song()
    println("-----")
    song.play()
    song.playAudio()
    song.playVideo()
    println("-----")
  }
}
```

TASK 3:

Write a partial function to add three numbers in which one number is constant and two numbers can be passed as inputs and define another method which can take the partial function as input and squares the result.

EXECUTION:



The screenshot shows an IDE with the following components:

- Project Explorer:** Shows a project named 'demoproject' containing 'ScalaAssignment' and 'ScalaHello'.
- Console:** Displays the execution output of 'Task3\$. (1) [Scala Application] C:\Program Files\'. The output is:

40 + 60 + 100 : 200
Square value : 40000

A red arrow points to the line '40 + 60 + 100 : 200'.
- Source Editor:** Shows the code for 'Task3.scala' in the 'scala.assignment17' package. The code is as follows:

```
package scala.assignment17

object Task3 {
  def main(args : Array[String]) : Unit = {
    // A partial function that accepts 2 numbers as input and return the sum
    val performAddition : PartialFunction[(Int, Int), Int] = {
      case (a, b) => a + b + 100
    }

    // Method to calculate the square value and return the result
    def calculateSquare(value: Int) : Int = {
      return (value * value)
    }

    println("OUTPUT:\n-----")
    println("40 + 60 + 100 : " + performAddition(40, 60))
    println("Square value : " + calculateSquare(performAddition(40, 60)))
    println("-----")
  }
}
```

TASK 4:

Write a program to print the prices of 4 courses of Acadgild:

Android App Development -14,999 INR

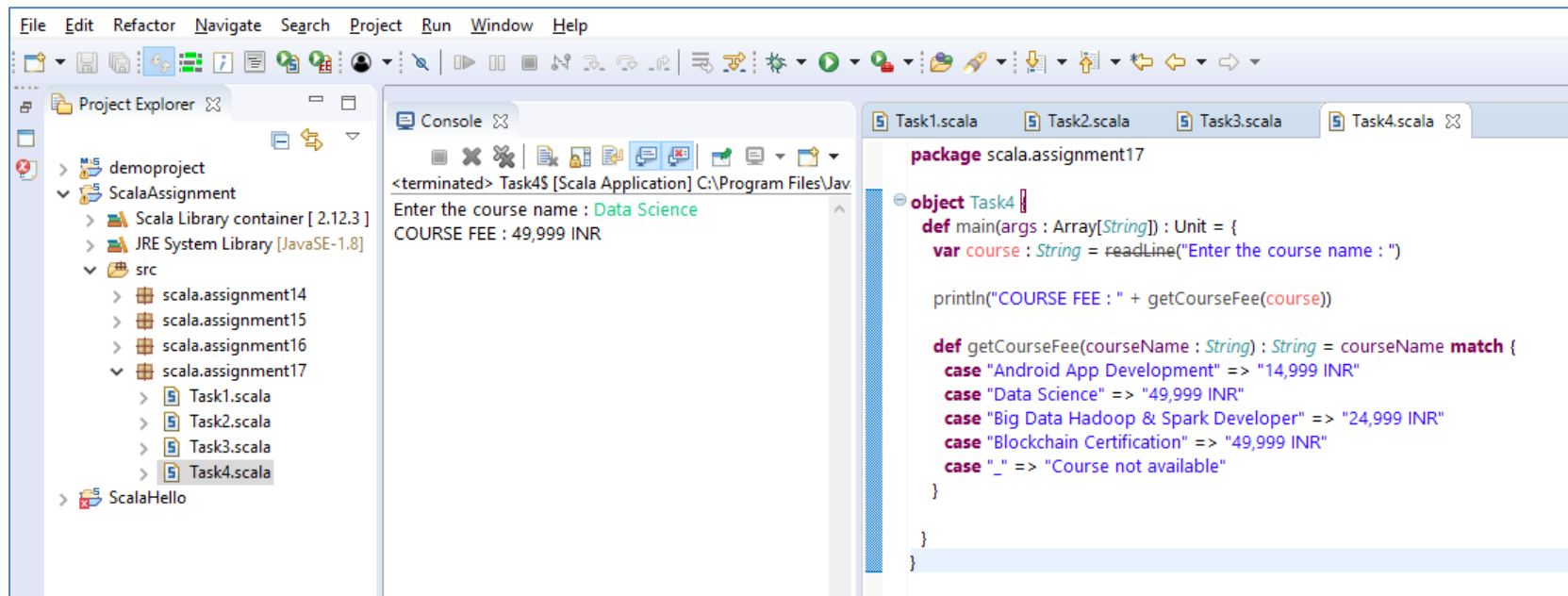
Data Science - 49,999 INR

Big Data Hadoop & Spark Developer – 24,999 INR

Blockchain Certification – 49,999 INR

using match and add a default condition if the user enters any other course.

EXECUTION:



```
File Edit Refactor Navigate Search Project Run Window Help
Project Explorer
demoproject
  ScalaAssignment
    Scala Library container [ 2.12.3 ]
    JRE System Library [JavaSE-1.8]
    src
      scala.assignment14
      scala.assignment15
      scala.assignment16
      scala.assignment17
        Task1.scala
        Task2.scala
        Task3.scala
        Task4.scala
      ScalaHello
Console
<terminated> Task4$ [Scala Application] C:\Program Files\Java
Enter the course name : Data Science
COURSE FEE : 49,999 INR
Task1.scala Task2.scala Task3.scala Task4.scala
package scala.assignment17

object Task4 {
  def main(args : Array[String]) : Unit = {
    var course : String = readLine("Enter the course name : ")

    println("COURSE FEE : " + getCourseFee(course))

    def getCourseFee(courseName : String) : String = courseName match {
      case "Android App Development" => "14,999 INR"
      case "Data Science" => "49,999 INR"
      case "Big Data Hadoop & Spark Developer" => "24,999 INR"
      case "Blockchain Certification" => "49,999 INR"
      case "_" => "Course not available"
    }
  }
}
```

