Session 15 : SCALA

Assignment 1

**Problem Statement**

1. Write a simple program to show inheritance in scala.

2. Write a simple program to show multiple inheritance in scala.

**Solution:-**

Project file- 

Code:-

1. Simple program to show inheritance in Scala

Parent class:- Employee

/\*\*

\* author Nihit Rai

\* This is the parent class

\*/

**class** Employee {

**var** salary:Float = 20000 ;

}

Child class:- Programmer

/\*\*

\* author Nihit Rai

\* This is the child class

\*/

**class** Programmer **extends** Employee{

**var** bonus:Int = 2500;

println("Salary = "+salary);

println("Bonus = "+bonus);

}

**object** MainObject{

**def** main(args:Array[String]){

**new** Programmer();

}

}

Output:-



1. Simple program to show multiple inheritance in Scala

//Create the base abstract class

**abstract** **class** Bird{

**def** sound:String;

}

//Create two traits each one overriding sound behaviour with some additional behaviour specific to the subclass

**trait** FlyingBird **extends** Bird{

**override** **def** sound = "Flying Sound";

**def** work () = { "Flying Flying"};

}

//Here we extends the abstract class and overridden sound behaviour and added one work method.

//Create one more subclass:-

**trait** RunningBird **extends** Bird{

**override** **def** sound = "Running Sound"

**def** run() = "Running Running"

}

//Now use mixin concept using with keyword:-

**class** FlyingRunningBird **extends** FlyingBird **with** RunningBird{

}

**class** RunningFlyingBird **extends** RunningBird **with** FlyingBird{

}

**object** MainObj{

**def** main(args:Array[String]){

**var** flyRunBird = **new** FlyingRunningBird();

println(flyRunBird.work()); // takes work method from FlyingBird class

println(flyRunBird.run()); // takes run method from RunningBird class

println(flyRunBird.sound); // As we have written “with RunningBird” so common behaviours(ie sound) is taken from RunningBird class hence we get sound behaviour is taken from RunningBird

println("-----------------------------------");

**var** runFlyBird = **new** RunningFlyingBird();

println(runFlyBird.work()); // takes work method from FlyingBird class

println(runFlyBird.run()); // takes run method from RunningBird class

println(runFlyBird.sound); // As “with FlyingBird” is written, so common behaviours(i.e. sound) is taken from FlyingBird class hence we get sound behaviour is taken from FlyingBird.

}

}

Output:--

