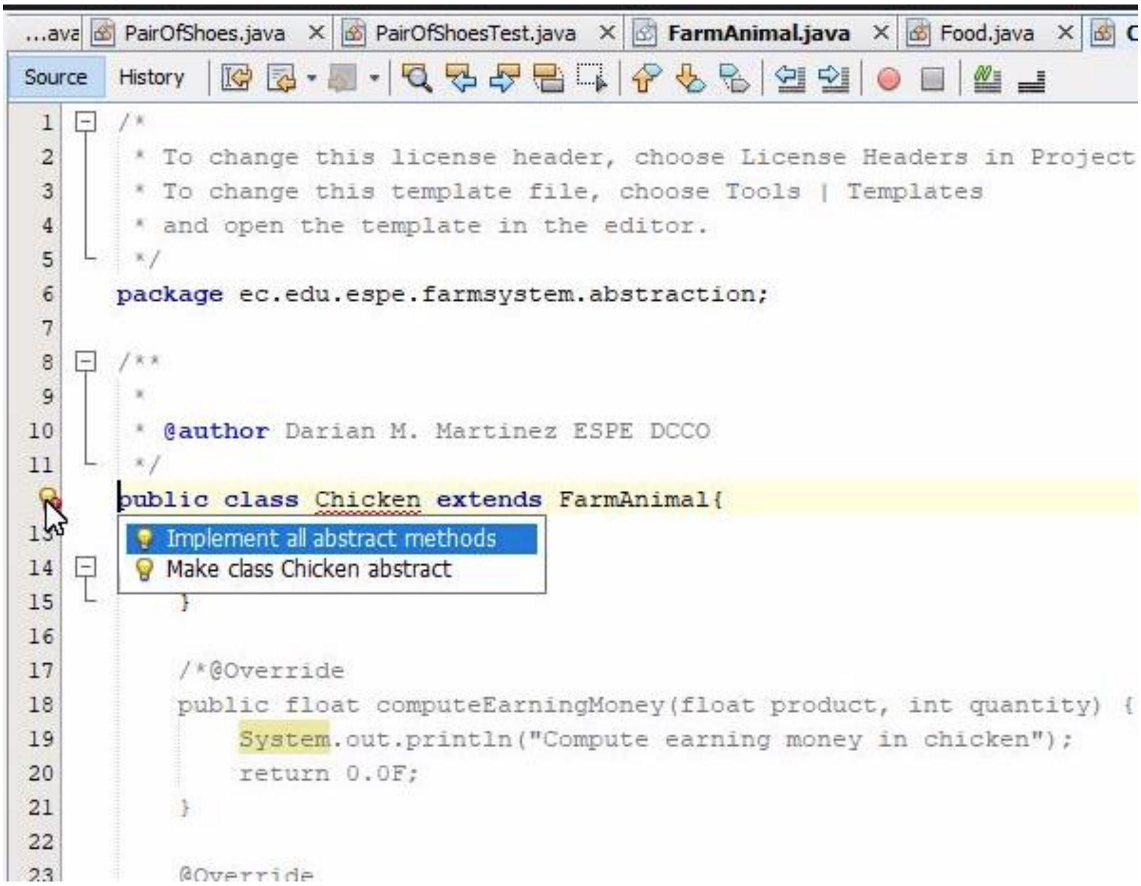
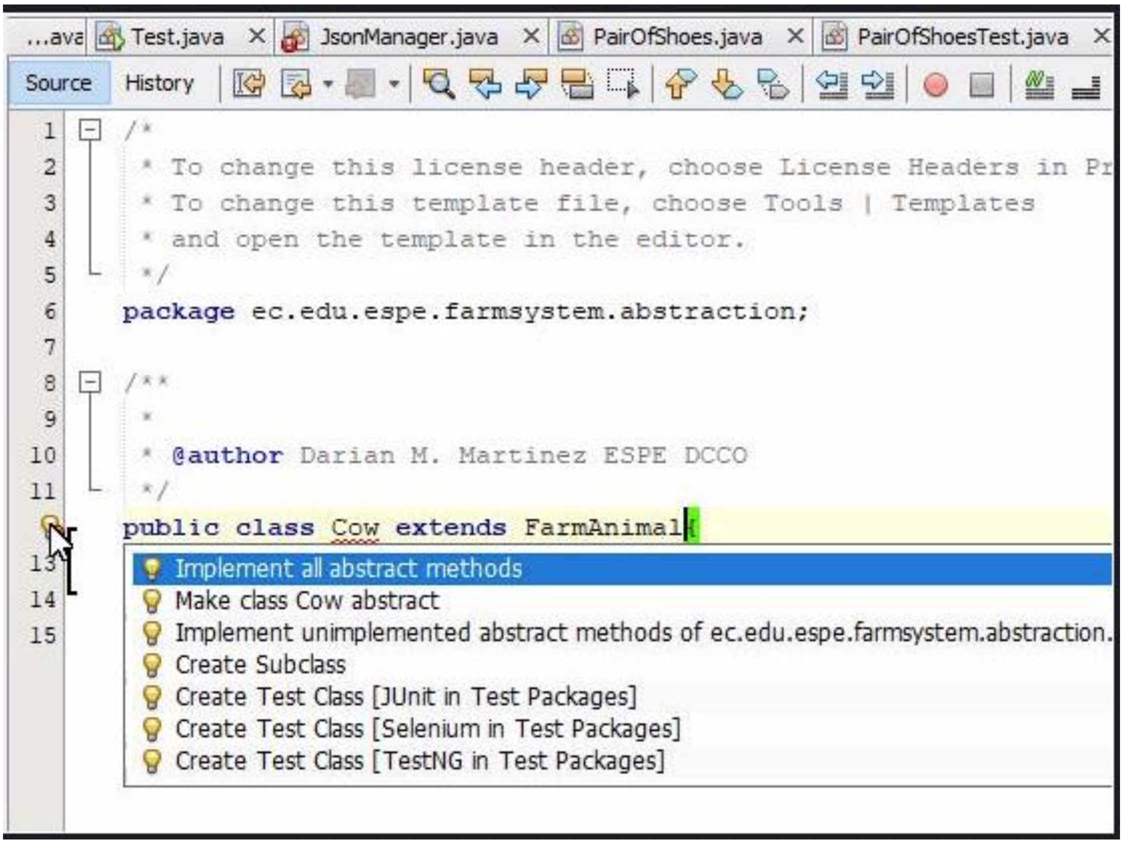
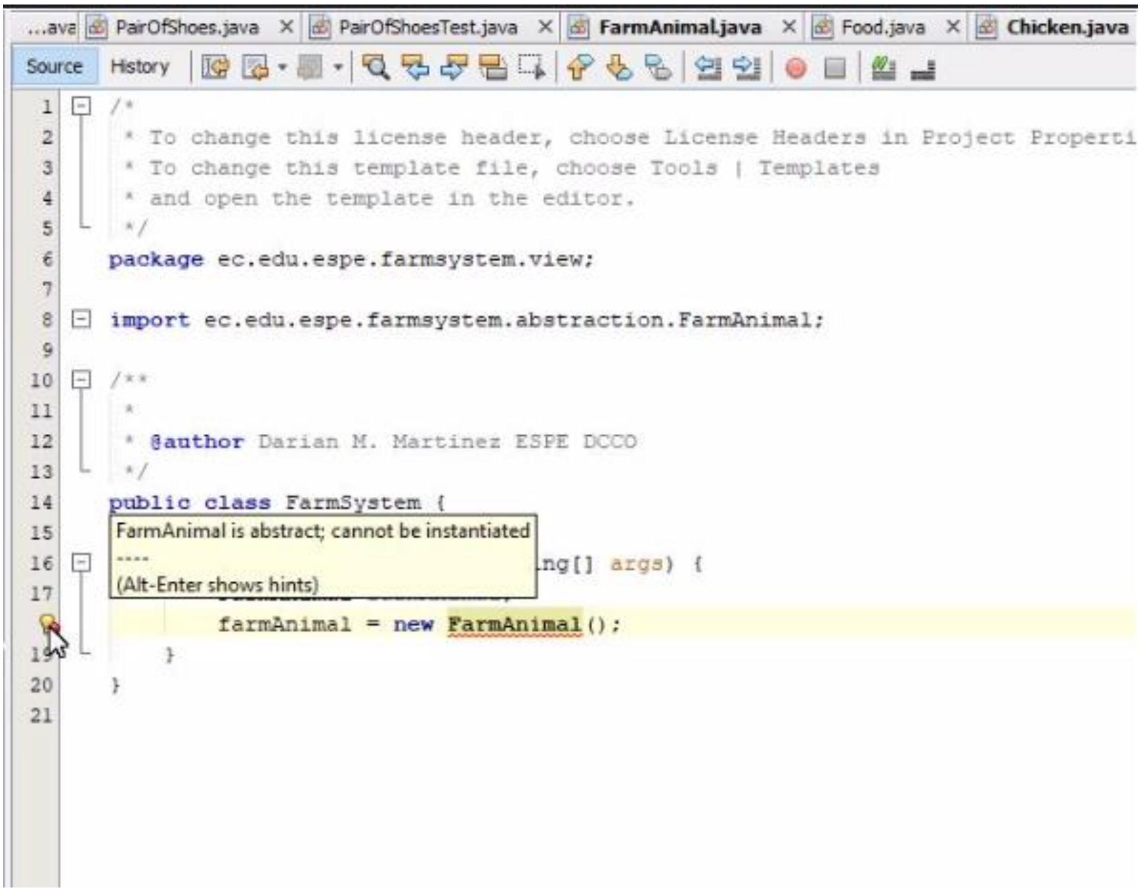


```java
/*
 * To change this license header, choose License Headers in Project
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ec.edu.espe.farmsystem.abstraction;

/**
 *
 * @author Darian M. Martinez ESPE DCCO
 */
public class Chicken extends FarmAnimal{


    }

    /*@Override
    public float computeEarningMoney(float product, int quantity) {
        System.out.println("Compute earning money in chicken");
        return 0.0F;
    }

    @Override
```


...ava
Test.java
JsonManager.java
PairOfShoes.java
PairOfShoesTest.java
Source
History
Implement all abstract methods
Make class Cow abstract
Implement unimplemented abstract methods of ec.edu.espe.farmsystem.abstraction.
Create Subclass
Create Test Class [JUnit in Test Packages]
Create Test Class [Selenium in Test Packages]
Create Test Class [TestNG in Test Packages]

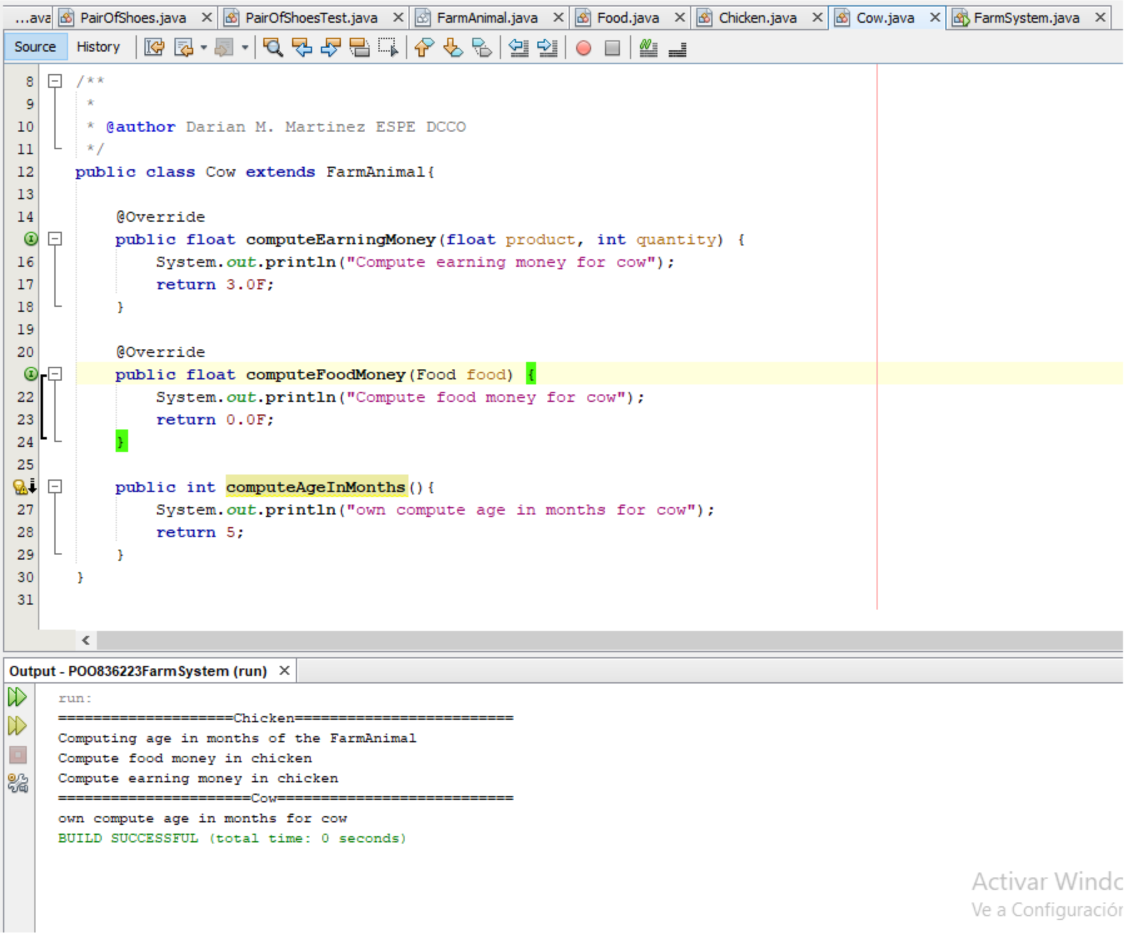
```java
/*
 * To change this license header, choose License Headers in Pr
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ec.edu.espe.farmsystem.abstraction;

/**
 *
 * @author Darian M. Martinez ESPE DCCO
 */
public class Cow extends FarmAnimal{
```



```
/*
 * To change this license header, choose License Headers in Project Properti
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ec.edu.espe.farmsystem.view;

import ec.edu.espe.farmsystem.abstraction.FarmAnimal;

/**
 *
 * @author Darian M. Martinez ESPE DCCO
 */
public class FarmSystem {

                                        ng[] args) {

            farmAnimal = new FarmAnimal();
    }
}
```

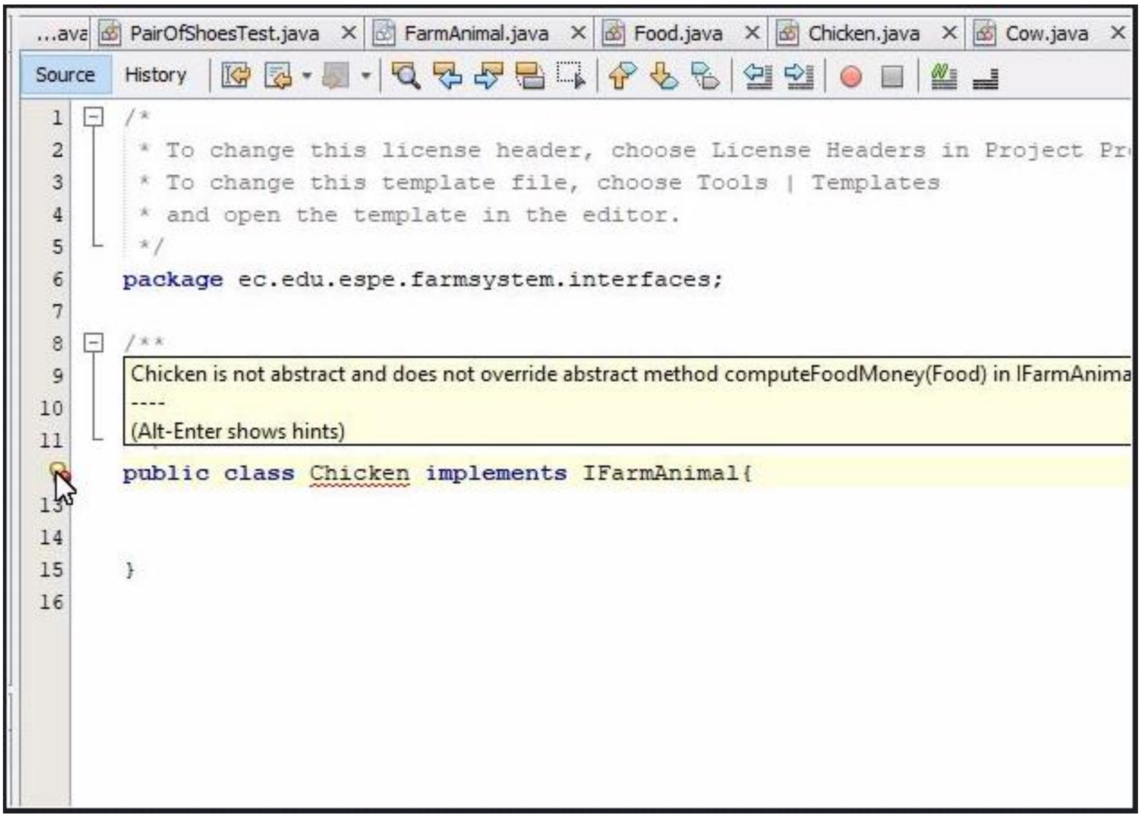
FarmAnimal is abstract; cannot be instantiated

(Alt-Enter shows hints)



```
/**
 *
 * @author Darian M. Martinez ESPE DCCO
 */
public class Cow extends FarmAnimal{

    @Override
    public float computeEarningMoney(float product, int quantity) {
        System.out.println("Compute earning money for cow");
        return 3.0F;
    }

    @Override
    public float computeFoodMoney(Food food) {
        System.out.println("Compute food money for cow");
        return 0.0F;
    }

    public int computeAgeInMonths(){
        System.out.println("own compute age in months for cow");
        return 5;
    }
}
```

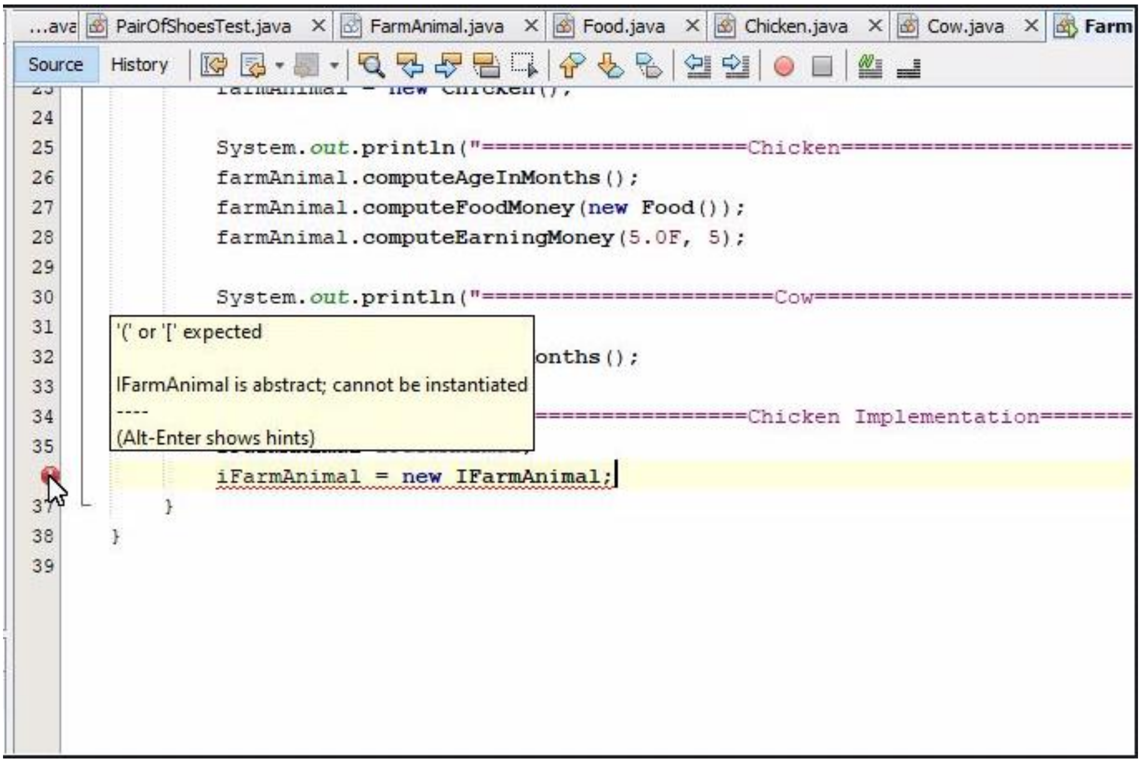
```
run:
===================Chicken========================
Computing age in months of the FarmAnimal
Compute food money in chicken
Compute earning money in chicken
=====================Cow==========================
own compute age in months for cow
BUILD SUCCESSFUL (total time: 0 seconds)
```



```
/*
 * To change this license header, choose License Headers in Project Pr
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package ec.edu.espe.farmsystem.interfaces;

/**
```

Chicken is not abstract and does not override abstract method computeFoodMoney(Food) in IFarmAnima
----
(Alt-Enter shows hints)

```
public class Chicken implements IFarmAnimal{


}
```

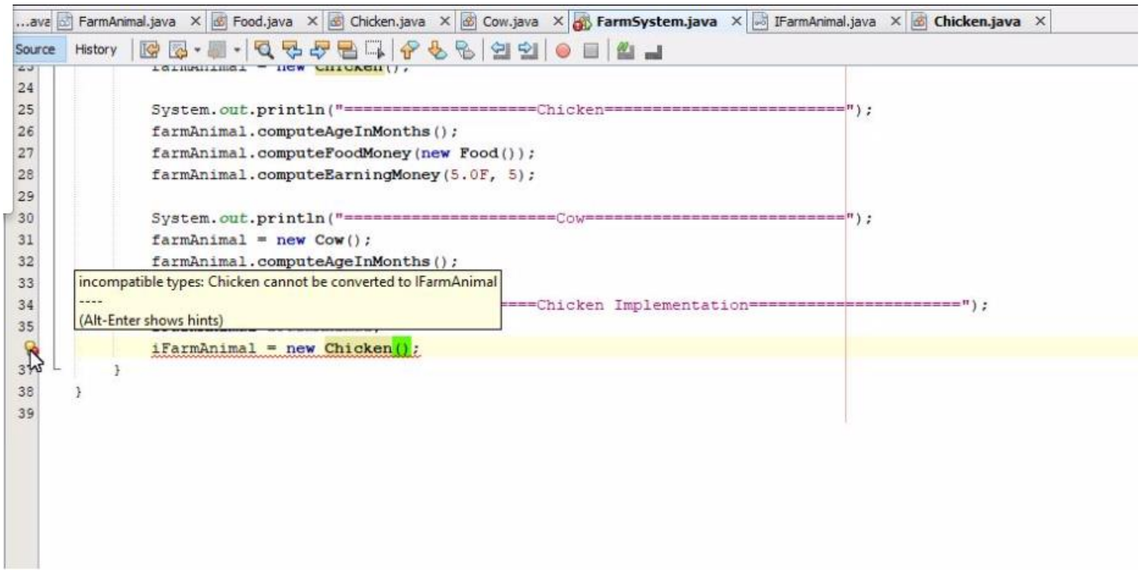


```
            farmAnimal = new Chicken();

            System.out.println("====================Chicken=====================
            farmAnimal.computeAgeInMonths();
            farmAnimal.computeFoodMoney(new Food());
            farmAnimal.computeEarningMoney(5.0F, 5);

            System.out.println("======================Cow=========================
                                        onths();
                                        =================Chicken Implementation=======
            iFarmAnimal = new IFarmAnimal;
        }
    }
```

'(' or '[' expected

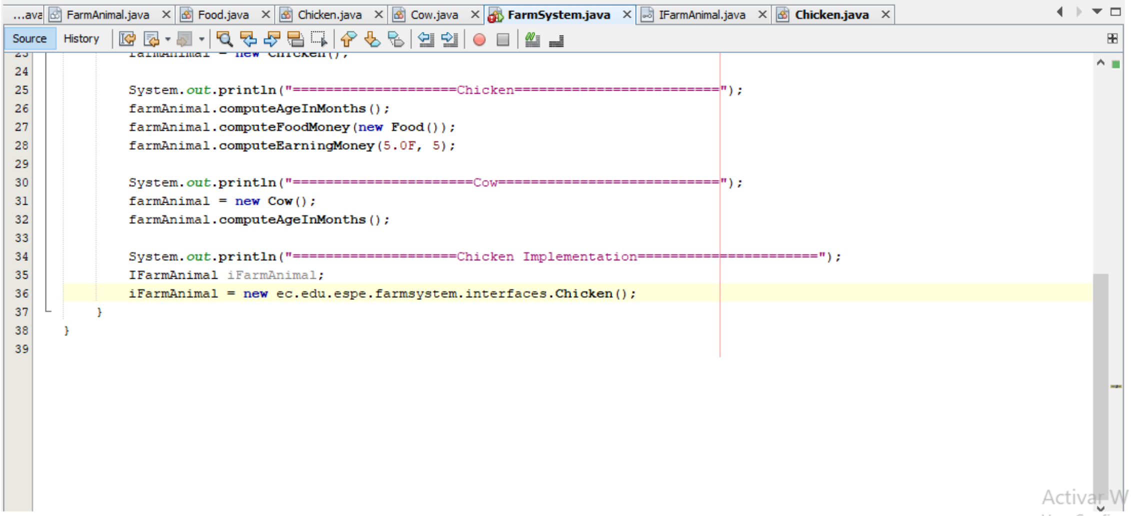
IFarmAnimal is abstract; cannot be instantiated

(Alt-Enter shows hints)

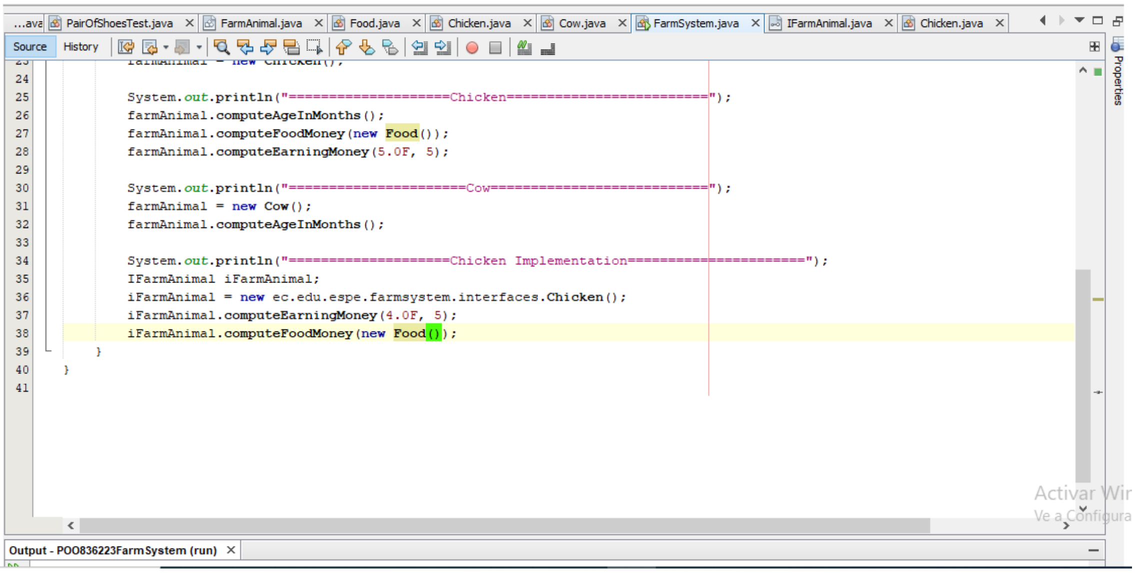


```
        farmAnimal = new Chicken();

        System.out.println("===================Chicken========================");
        farmAnimal.computeAgeInMonths();
        farmAnimal.computeFoodMoney(new Food());
        farmAnimal.computeEarningMoney(5.0F, 5);

        System.out.println("=====================Cow===========================");
        farmAnimal = new Cow();
        farmAnimal.computeAgeInMonths();

                                         ====Chicken Implementation======================");

        iFarmAnimal = new Chicken();
    }
}
```

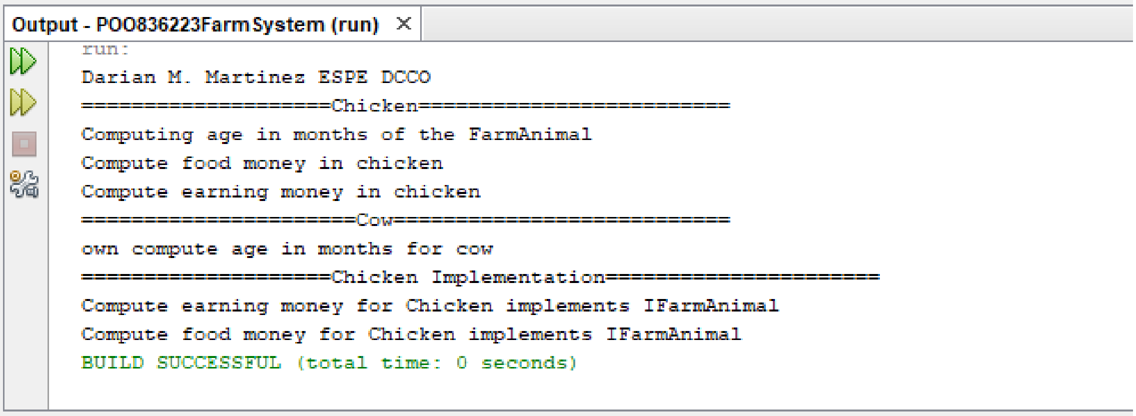
incompatible types: Chicken cannot be converted to IFarmAnimal
----
(Alt-Enter shows hints)



```
        farmAnimal = new Chicken();

        System.out.println("===================Chicken========================");
        farmAnimal.computeAgeInMonths();
        farmAnimal.computeFoodMoney(new Food());
        farmAnimal.computeEarningMoney(5.0F, 5);

        System.out.println("=====================Cow===========================");
        farmAnimal = new Cow();
        farmAnimal.computeAgeInMonths();

        System.out.println("===================Chicken Implementation=====================");
        IFarmAnimal iFarmAnimal;
        iFarmAnimal = new ec.edu.espe.farmsystem.interfaces.Chicken();
    }
}
```



```java
        farmAnimal = new Chicken();

        System.out.println("===================Chicken========================");
        farmAnimal.computeAgeInMonths();
        farmAnimal.computeFoodMoney(new Food());
        farmAnimal.computeEarningMoney(5.0F, 5);

        System.out.println("=====================Cow===========================");
        farmAnimal = new Cow();
        farmAnimal.computeAgeInMonths();

        System.out.println("===================Chicken Implementation=====================");
        IFarmAnimal iFarmAnimal;
        iFarmAnimal = new ec.edu.espe.farmsystem.interfaces.Chicken();
        iFarmAnimal.computeEarningMoney(4.0F, 5);
        iFarmAnimal.computeFoodMoney(new Food());
    }
}
```



```
run:
Darian M. Martinez ESPE DCCO
====================Chicken=========================
Computing age in months of the FarmAnimal
Compute food money in chicken
Compute earning money in chicken
======================Cow===========================
own compute age in months for cow
====================Chicken Implementation======================
Compute earning money for Chicken implements IFarmAnimal
Compute food money for Chicken implements IFarmAnimal
BUILD SUCCESSFUL (total time: 0 seconds)
```