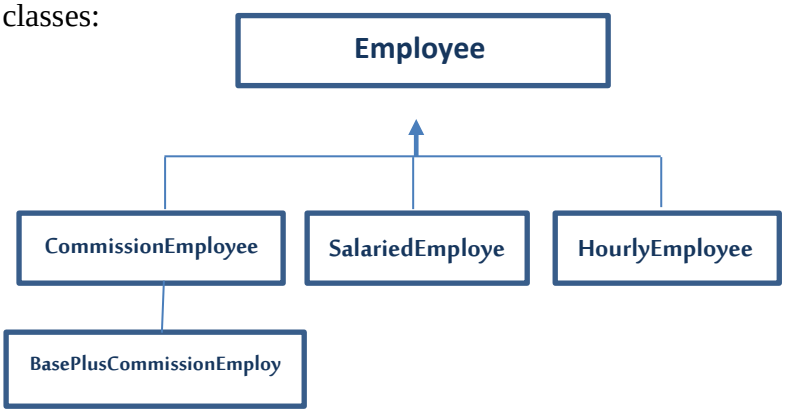
Lab #5 :	Files

Name: _____ **Date:** _____ **Section:** _____

Objectives:
- Opening a file for output.
- Writing objects to a file.

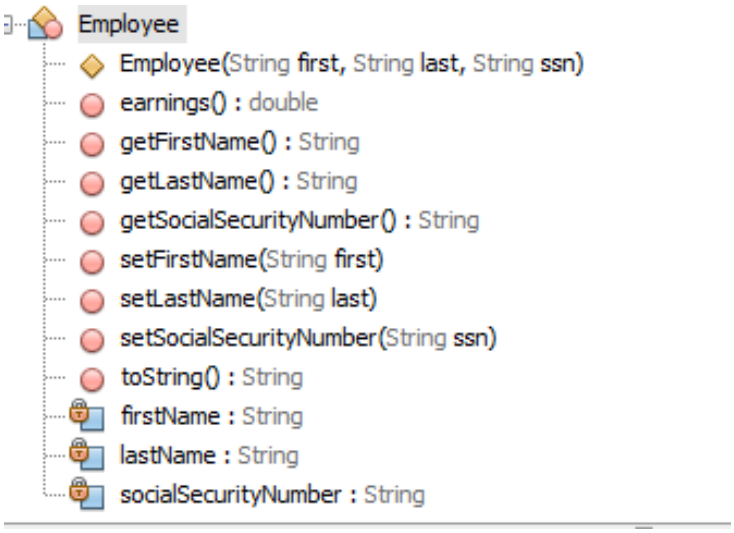
LAB EXERCISES

In this exercise, you will update a program to write data to a sequential text file (**using class: Formatter**). The data that will be written to the file represents records of the following classes:

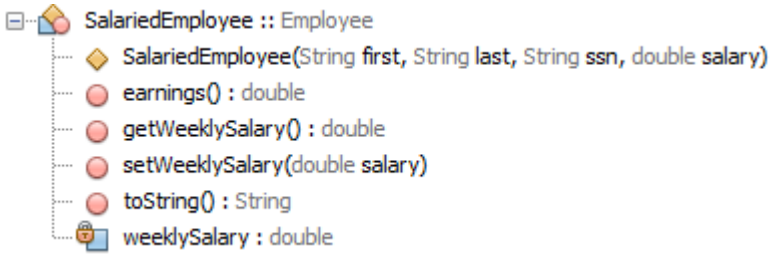


The implementation of the abstract class Employee and the four subclasses are given under ***OutputEmployees*** project folder. As follow:

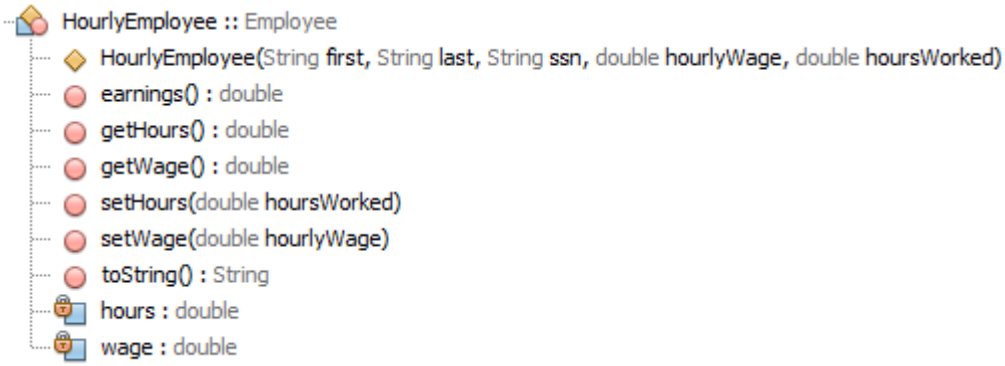
Employee.java



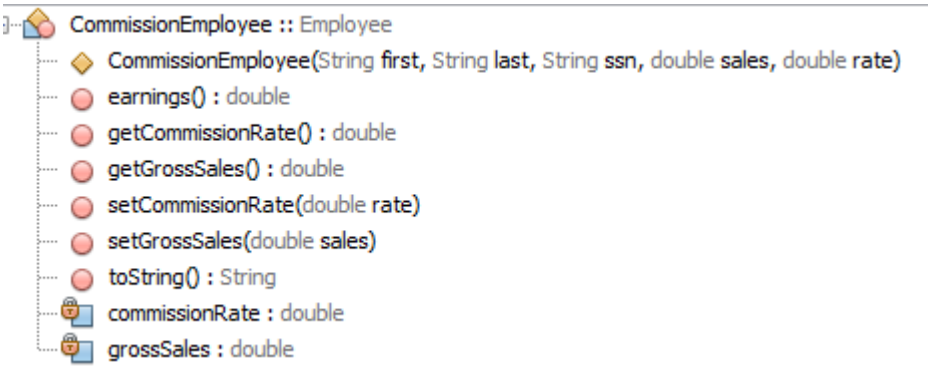
SalariedEmployee.java



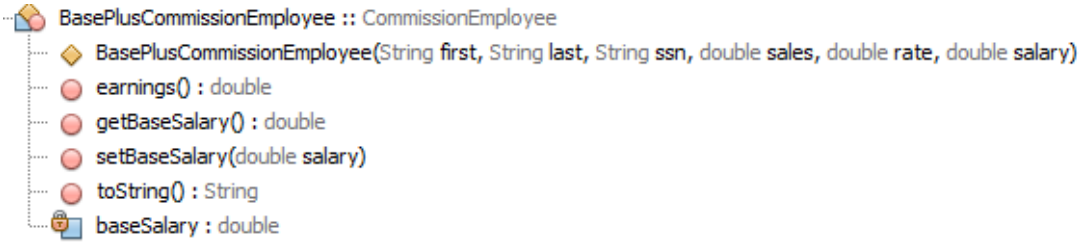
HourlyEmployee.java



CommissionEmployee.java



BasePlusCommissionEmployee.java



OutputEmployees.java (The test class)

This main class preforms the following:

- Create subclass objects
- Print out the employees records in the Console

Lab Exercise:

Update ***OutputEmployees.java*** as follows:

1. Create four-element Employee array
2. Initialize array with Employees
3. Creates a text file the file ***EmployeeData.txt***, then access this file and writes all the elements of the array employees into the file.

```
run:
Employees processed individually:

salaried employee: John Smith
social security number: 111-11-1111
weekly salary: $800.00
earned: $800.00

hourly employee: Karen Price
social security number: 222-22-2222
hourly wage: $16.75; hours worked: 40.00
earned: $670.00

commission employee: Sue Jones
social security number: 333-33-3333
gross sales: $10,000.00; commission rate: 0.06
earned: $600.00

base-salaried commission employee: Bob Lewis
social security number: 444-44-4444
gross sales: $5,000.00; commission rate: 0.04; base salary: $300.00
earned: $500.00

Output the elements of the array into a text file:

Done
BUILD SUCCESSFUL (total time: 0 seconds)
```

```
package outputemployees;

import java.io.FileNotFoundException;
import java.util.Formatter;
import java.util.FormatterClosedException;

public class OutputEmployees {

    public static void main(String[] args) {

       // create subclass objects
      SalariedEmployee salariedEmployee =
         new SalariedEmployee( "John", "Smith", "111-11-1111", 800.00 );
      HourlyEmployee hourlyEmployee =
         new HourlyEmployee( "Karen", "Price", "222-22-2222", 16.75, 40 );
      CommissionEmployee commissionEmployee =
         new CommissionEmployee(
         "Sue", "Jones", "333-33-3333", 10000, .06 );
      BasePlusCommissionEmployee basePlusCommissionEmployee =
         new BasePlusCommissionEmployee(
         "Bob", "Lewis", "444-44-4444", 5000, .04, 300 );

      System.out.println( "Employees processed individually:\n" );

      System.out.printf( "%s\n%s: $%,.2f\n\n",
         salariedEmployee, "earned", salariedEmployee.earnings() );
      System.out.printf( "%s\n%s: $%,.2f\n\n",
         hourlyEmployee, "earned", hourlyEmployee.earnings() );
      System.out.printf( "%s\n%s: $%,.2f\n\n",
         commissionEmployee, "earned", commissionEmployee.earnings() );
      System.out.printf( "%s\n%s: $%,.2f\n\n",
         basePlusCommissionEmployee,
         "earned", basePlusCommissionEmployee.earnings() );

      // create four-element Employee array
      Employee employees[] = new Employee[ 4 ];

     // initialize array with Employees
     employees[ 0 ] = salariedEmployee;
     employees[ 1 ] = hourlyEmployee;
     employees[ 2 ] = commissionEmployee;
     employees[ 3 ] = basePlusCommissionEmployee;

     System.out.println( "Output the elements of the array:\n" );

     /* Write code here that opens the file EmployeeData.ser for object output ther
        writes all the elements of the array employees into the file */
     Formatter output;
     try{
           output=new Formatter("EmployeeData.txt");
           for(int i=0; i<employees.length; i++)
               output.format("%s%n", employees[i].toString());
           if(output!=null)
               output.close();

     }catch(SecurityException securityException)
     {
         System.err.println("You do not have wirte access to this file!");
         System.exit(1);
     }
     catch(FileNotFoundException fileNotFoundException)
     {
         System.err.println("File Not found!");
         System.exit(1);
     }
     catch(FormatterClosedException formatterClosedException)
      {
          System.err.println("Error writing to file!");
          System.exit(1);
      }

   } // end main
} // end class OutputEmployees
```

FOLLOW-UP QUESTIONS AND ACTIVITIES

1. Complete the following tasks, assuming that each applies to the same program:

a) Write a statement that opens file "newmast.txt" for output (and creation)—use formatter variable outNewMaster.

ANS:

```
Formatter outNewMaster = new Formatter( "newmast.txt" );
```

b) Write a statement that outputs a record to the file "newmast.txt". The record is an object of type AccountRecord—use Formatter variable outNewMaster.

ANS:

```
outNewMaster.format( "%d %s %s %.2f\n",
account.getAccount(), account.getFirstName(),
account.getLastName(), account.getBalance() );
```

2. Correct the Code:

The following code segment should open a text file for writing with a Formatter.

```
Formatter output = new Formatter();
```

ANS:

```
Formatter output = new Formatter("FileName.txt");
```