Assignment Number 8

100 Points

Prerequisites: Completion of Assignment 4

References: Handouts and Websites

[Junit Tutorial using Eclipse](http://courses.cs.washington.edu/courses/cse143/11wi/eclipse-tutorial/junit.shtml)

[Junit for Beginners](http://java.dzone.com/articles/junit-tutorial-beginners)

Skills Required:

* Code from Assignment 4
* Java exceptions
* JUnit test concepts and framework

Task Specifications:

1. Modify your Assignment 4:
   1. Update the PayRoll class
   2. Create one JUnit class called RunPayRollTest
2. **The PayRoll class:**
   1. Delete the public static main method in the class
   2. Add the public static readFromFile(String fileName) method that will read/extract the pay roll system data from a given file “payrollData.txt” and create employees and payRecords arrays as you have done in Assignment 4, but this time without user inputs. You need to pass the proper file name “payrollData.txt” and use try catch statement to open and read file data as following:

try

{

// code to read a file and create employees and payRecords arrays

}catch (IOException e) {

{

//Catch block to produce a proper message

}

* 1. Add public static boolean addEmplyee() method with all employee parameters such as (4, “Joe”, “Former”, Status.*HOURLY*) that will return *false* if the Employee ID (i.e., 4) does exist in the employees array otherwise it adds a new Employee object to employees array and returns *true*
  2. Add public static boolean addPayRecord() method with all required PayRecord parameters such as PayRecord ID = 2, etc. that will return *false* if the PayRecord ID (i.e., 4) does exist in the payRecord array otherwise it adds a new PayRecord object to payRecord array and returns *true*
  3. Add public static Employee addEmplyee() method with the Emplyee ID (i.e., 1) parameter that will search in the employees array using passed parameter Employee ID and it returns *null* if the Employee does not exist otherwise it returns the Employee object
  4. Add public static PayRecord getPayRecord() method with the PayRecord ID (i.e., 1) parameter that will look up in the payRecords array using passed parameter PayRecord ID and it returns *null* if the PayRecord does not exist otherwise it returns the PayRecord object

1. **The RunPayRollTest class:**
   1. add JUnit (e.g., JUnit 4) to your Assignment 8 project: Open eclipse -> right click on project and click on properties > Build Path > Add Library and highlight JUnit4 then hit next, and click OK, it will be added to the project and you can create test cases class and methods. **OR:** simply, right click on project and  Build Path > Add Library highlight JUnit 4 then hit next, and click OK, it will be added to JRE System Library.
   2. Use the provided template class RunJUnitTest in the JUnit CalculateCreditScore Example folder, use the import packages and the class signature to create this class
   3. Add the following test cases (methods) to the RunPayRollTest class:
      1. testAddEmployee() method: uses assertFalse() or assertTrue() to test PayRoll.addEmplyee()
      2. testAddPayRecord() method: uses assertFalse() or assertTrue() to test PayRoll.addPayRecord()
      3. testGetEmployee() method: uses assertNull(e) or assertNotNull(e) to test PayRoll.getEmployee()
      4. testGetPayRecord() method: uses assertNull(e) or assertNotNull(e) to test PayRoll.getPayRecord()
      5. testGrossPay() method: uses assertEquals() to test a given gross pay value with the grossPaye() return in the first pay record in the payRecord array, with index 0
   4. In all of the above test methods, you need to call readFromFile() method in the PayRoll class to create the arrays, for example the code for testGetEmployee() method:

@Test

public void testGetEmployee()

{

PayRoll.readFromFile("payrollData.txt "); // this will create

employees and payRecords arrays

assertNull(PayRoll.getEmplyee(4));

} // end method

* 1. Make sure all the attributes and methods have static keyword

Evaluation Criteria

1. All tasks must be completed to receive credit for this assignment
2. Program should report the correct values