

## Quiz 2 Classes and Objects

**Released:** Friday, 14<sup>th</sup> April 2023

**Due:** 11.59pm , Sunday, 23<sup>rd</sup> April 2023

This quiz test you on creating object class and functions in Java. It is divided into 3 parts, each with assertions to test your answers.

### Instructions

1. Copy the `Quiz2` class and study the objects and functions being called in the class. This is the Java application you need to compile and run. Do not change anything in the `Quiz2` class. However, you should comment off sufficient code in the class so that you could attempt the sections one by one.
2. Write your `PensionPlanner` class with relevant functions to pass all the assertions in the `Quiz2` class. You might need to import Java API classes into your `PensionPlanner.java` file.
3. Copy and paste your `PensionPlanner.java` (including the import statement if any) into the submission form provided in your lecture session. DO NOT SUBMIT THE `Quiz2` class.
4. Your submission code must be able to be compiled into `PensionPlanner.class` file. Failure in compilation will be granted **ZERO** mark.
5. Failure to follow the submission instructions will be granted **ZERO** mark.
6. You can submit multiple submission before the due date. Only the latest submission will be recorded so do not submit again after the quiz is due. (See point below)
7. Submission after the due date will be granted **ZERO** mark.

### Programming with assertions

An assertion is a statement in the Java™ programming language that enables you to test your assumptions about your program. For example, if you write a method that calculates the speed of a particle, you might assert that the calculated speed is less than the speed of light.

Each assertion contains a boolean expression that you believe will be true when the assertion executes. If it is not true, the system will throw an error. By verifying that the boolean expression is indeed true, the assertion confirms your assumptions about the behavior of your program, increasing your confidence that the program is free of errors.

Read more from here.

<https://docs.oracle.com/javase/7/docs/technotes/guides/language/assert.html>

## Part A: Retirement Planning

In January 2023, according to the statistics released by Employees Provident Fund (EPF) in Malaysia, there are more than half of their members under the age of 55 had no more than RM10,000 in their saving. The EPF recommend a sum of RM240,000 as adequate for a poverty level pension by the time they retire at the age of 60. Therefore, we would like to advise our students to start saving as soon as they started working and plan their retirement from day one. In Part A, we are going to determine how long it will take you to save enough money for a poverty level pension, that is RM240,000, given the following assumptions:

1. Call the pension saving target **pensionTarget**. This is set at RM240,000 here.
2. Call the amount you have saved thus far **currentSaving**. You start with a current saving of RM0.
3. Assume your annual salary is **annualSalary**.
4. Assume you are going to dedicate a certain amount of your salary each month to the pension. Call that **monthlySaving**. This variable should be in decimal form (i.e. 0.1 for 10%)
5. Assume that your pension fund invest your savings wisely, with an annual return of  $r$  (in other words, at the end of each month, you receive an additional  $\text{currentSavings} * r / 12$  funds to put into your savings – the 12 is because  $r$  is an annual rate). Assume that your retirement fund earn a return of  $r = 0.04$  (4%). Call that **interestRate**.

Write a program to calculate how many months it will take you to save up enough money for the poverty level pension.

Class `Quiz2` is provided to you for output verification. You need to write and implement the `PensionPlanner` class. The `PensionPlanner` class skeleton is given to you as below:

```
// BEGIN ANSWER Quiz2 <-- Copy From This Line
// Submission code: PensionPlanner class
import java.util.*;
public class PensionPlanner {
    // TODO
}
// END ANSWER Quiz2 <-- Copy Until This Line
```

You should clear all the assertions in Part A in the `Quiz2` class. You are encouraged to test your code with test cases of your own. All submitted code will be tested against other test cases during marking.

Note:

Assertion is disabled by default in Java. To enable it, you need to run your code by `-ea` or `-enableassertions` option.

## Part B: Saving with a raise

In Part A, we assume that your annual salary did not change and that is unrealistic. With a computer science degree, you are going to be worth more to your company over time. So we are going to build your pension fund by factoring a raise every six months. Call that **semiAnnualRaise**.

The semi-annual raise is given as decimal percentage, e.g., 0.03 means 3% raise every six months. After the 6<sup>th</sup> month, increase your salary by the percentage. Do the same for the 12<sup>th</sup> month, 18<sup>th</sup> month, etc.

Clear all assertions in Part B in the [Quiz2](#) class.

Now if you compare the test case 1 to 4, you will notice you can achieve the same pension target 10 years earlier with just 3% semi-annual raise throughout your working years. Similar observations can be found in comparing test case 2 to 5, and 3 to 6.

## Part C: Delaying the withdrawal

Assumed that you are now retired and still working with an income for daily expenses. You would like to delay the withdrawal of your pension till the later years. How much addition saving will you have in your pension after delaying withdrawal for certain number of month?

The **setWorkingMonth** function is used to set the months you work and contribute monthly to the pension. After that you stop contributing to the pension fund and do not withdraw any money for as many month set by **setNoOfMonthDelay** function. However, the pension fund continue to pay interest on monthly basis based on the current saving you have in the account.

The **getAdditionalPensionSaving** function will return the difference in saving at the end of the working month and the end of the delay month. For example, if you have RM300,000 in saving after 30 years of work, and after 3 years without any withdrawal your saving is RM350,000, the **getAdditionalPensionSaving** function will return 50,000.

We will accept your answer between the 100 range from our answer due to possible loss in precision caused by the float during the calculation process.

Clear all assertions in Part C in the [Quiz2](#) class.

Submit your **PensionPlanner.java** file. Do not submit the [Quiz2](#) class.

```

import java.util.*;

public class Quiz2{

    public static void main(String[] a){

        // Part A: Retirement planning
        //test case 1
        double pensionTarget = 240000;
        double currentSaving = 0;
        double annualSalary = 40000;
        double monthlySaving = 0.1;
        double interestRate = 0.04;

        PensionPlanner fp = new PensionPlanner(pensionTarget, currentSaving,
            annualSalary, monthlySaving, interestRate);

        System.out.println("Example: Test case 1 working months: " +
fp.getWorkingMonth());

        assert fp.getWorkingMonth() == 368:"Failed test case 1";

        // test case 2
        fp.setPensionTarget(500000);
        fp.setAnnualSalary(40000);
        fp.setMonthlySaving(0.2);
        assert fp.getWorkingMonth() == 377:"Failed test case 2";

        // test case 3
        fp.setPensionTarget(500000);
        fp.setAnnualSalary(80000);
        fp.setMonthlySaving(0.15);
        assert fp.getWorkingMonth() == 295:"Failed test case 3";

        // Part B: Saving with a raise
        // test case 4
        double semiAnnualRaise = 0.03;
        fp.setPensionTarget(240000);
        fp.setAnnualSalary(40000);
        fp.setMonthlySaving(0.1);
        fp.setSemiAnnualRaise(semiAnnualRaise);
        assert fp.getWorkingMonth() == 255:"Failed test case 4";

        // test case 5
        fp.setPensionTarget(500000);
        fp.setAnnualSalary(40000);
        fp.setMonthlySaving(0.2);
        fp.setSemiAnnualRaise(0.06);
        assert fp.getWorkingMonth() == 199:"Failed test case 5";
    }
}

```

```

// test case 6
fp.setPensionTarget(500000);
fp.setAnnualSalary(80000);
fp.setMonthlySaving(0.15);
fp.setSemiAnnualRaise(0.05);
assert fp.getWorkingMonth() == 179:"Failed test case 6";

// Part C: Delaying withdrawal
// test case 7
fp.setWorkingMonth(360); // work for 30 years
fp.setAnnualSalary(40000);
fp.setMonthlySaving(0.1);
fp.setSemiAnnualRaise(0.03);
fp.setNoOfMonthDelay(36); // 3 years delay in withdrawal
System.out.println("Example: Test case 7 additional pension saving: " +
fp.getAdditionalPensionSaving());

// accept tolerance up to 100 in absolute value
assert Math.abs(fp.getAdditionalPensionSaving()-63454) <= 100: "Failed test
case 7";
// Making about RM21k per year, even without any contribution in the 3 years

// test case 8
fp.setInterestRate(0.06);
assert Math.abs(fp.getAdditionalPensionSaving()-126469) <= 100: "Failed test
case 8";
// Investment return raised from 4% to 6% will double up the additional saving
from RM63k to RM126k

// test case 9
fp.setAnnualSalary(60000);
fp.setMonthlySaving(0.15);
fp.setNoOfMonthDelay(60); // 5 years delay in withdrawal
assert Math.abs(fp.getAdditionalPensionSaving()-474259) <= 100: "Failed test
case 9";
// Almost RM100k per year, it is like having another fulltime job without
having to work on it.

System.out.println("The quiz is done!");
System.out.println("Submission code: PensionPlanner class ONLY.");
System.out.println("\nImportant:");
System.out.println("DO NOT SUBMIT Quiz2 class!");
System.out.println("DO NOT SUBMIT Quiz2 class!");
System.out.println("DO NOT SUBMIT Quiz2 class!");

} //main
} //class

```

```
/* Node: To enable assertion in Java:  
   Compile by javac Quiz2.java  
   Execute by java -ea Quiz2  
   Refer to Lecture01 if you do know how to compile and execute by command lines  
  
   DO NOT SUBMIT THIS FILE  
*/
```

## Conclusion

It is better to keep working for as long as possible even after you past the retirement age and delay in pension withdrawal to maximise the return from the pension fund.

It is also better to choose the pension fund with better interest rate even if is just a tiny difference in terms of the percentage.

By completing this quiz, you have the right tool for personal financial planning for all scenarios. Unfortunately, you have not started working and cannot carry out any plan until after your graduation. Nevertheless, you can help your family, relatives and friends plan for their retirement saving, or financial target in the long run.