

Assignment #5

CSE110 - Arizona State University

Topics

- Implementing classes & testing

Coding Guidelines:

- Give identifiers semantic meaning and make them easy to read (examples numStudents, grossPay, etc).
- Keep identifiers to a reasonably short length.
- Use upper case for constants. Use title case (first letter is upper case) for classes. Use lower case with uppercase word separators for all other identifiers (variables, methods, objects).
- Use tabs or spaces to indent code within blocks (code surrounded by braces). This includes classes, methods, and code associated with ifs, switches and loops. Be consistent with the number of spaces or tabs that you use to indent.
- Use white space to make your program more readable.

Part #1: Written Exercises (0 pts)

None

Part #2 - Programming (20 pts)

Write a class definition (not a program, there is no main method) named **Geek** (saved in a file **Geek.java**) that models a person who is a geek. For our purposes, a geek is someone who delights in answering all sorts of questions, such as “is this string a palindrome?”, “what is the sum of all numbers between two numbers?” among other things. A Geek has a name and also keeps track of how many questions s/he has answered.

Your Geek class must have:

- Only two instance variables - the Geek’s name and number of questions asked so far.
- `public Geek(String name, int numQuestions)` - Constructor - sets the **Geek**’s name and the number of questions asked.
- `public String getName()` - Takes no parameters and returns the **Geek**’s name as a **String** (don’t count this request in the total questions).
- `public int getNumberOfQuestions()` - Takes no parameters and returns as an **int** how many questions has been asked (don’t count this request in the total).
- `public boolean isEven(int num1, int num2)` - It takes two integers and returns a **boolean** value indicating if the sum of the numbers is even or not (this counts as the **Geek** being asked one more question, so update the appropriate instance variable).

- `public int sum(int num1, int num2)` - Takes two integers and returns an `int` which is the sum of all numbers between the two inclusive (include the numbers in the sum) - for full credit this method should work even if the two numbers are the same (the sum is just one of the numbers) or if the first number is larger than the second. Also, you cannot assume which number will be greater.
- `public boolean leapYear(int year)` - It takes an integer and returns a `boolean` value indicating if the year is a leap year. A leap year is one with 366 days. A year is a leap year if:
 - it is divisible by 4 (for example, 1980),
 - except that it is not a leap year if it is divisible by 100 (for example 1900);
 - however, it is a leap year if it is divisible by 400 (for example, 2000).

There were no exceptions before the introduction of the Gregorian calendar on October 15, 1582 (for example, 1500 was a leap year). You may NOT use Java's `GregorianCalendar` class. (This counts as the **Geek** being asked one more question, so update the appropriate instance variable).

Save the **Geek** class in a file called `Geek.java`. Write a test driver called `Assignment5.java` with the main method to create a new **Geek** object and to test the methods in the class **Geek**. A sample output is shown below.

`Assignment5.java` will provide the following menu to the user. Based on the user's choice, the program needs to perform corresponding operation. This will be done by calling (invoking) one of the methods you defined in the **Geek** class. The program will terminate when the user enters 'q'.

Command Options

```
-----
a : get name
b: number of questions asked
c: sum is even
d: sum between two integers
e: leap year
?: display the menu again
q: quit this program
```

Here is the description for each option:

```
a: asks for the Geek's name
b: returns the number of questions asked so far
c: asks for two integers and prints if the sum of the numbers is even
d: asks for two integers and returns the sum of all numbers between them, inclusive
e: asks for an integer and returns if it is a leap year or not
?: displays the menu
q: quits
```

Helpful Hints

- Work on it in steps - write one method, test it with a test driver and make sure it works before going on to the next method.
- Always make sure your code compiles before you add another method.
- Your methods should be able to be called in any order.

Sample Output

(user input is in bold)

Command Options

a : get name
b: number of questions asked
c: sum is even
d: sum between two integers
e: leap year
?: display the menu again
q: quit this program

Please enter a command or type? **a**
Name: Geek

Please enter a command or type? **b**
Number of questions: 0

Please enter a command or type? **c**
Enter a number: 3
Enter the second number: 3
The sum of the numbers is even

Please enter a command or type? **c**
Enter a number: 3
Enter the second number: 4
The sum of the numbers is odd

Please enter a command or type? **b**
Number of questions: 2

Please enter a command or type? **d**
Enter a number: 1
Enter a second number: 10
The sum between 1 and 10 is 55

Please enter a command or type? **d**
Enter the first number: 7
Enter the second number: 4
The sum between 4 and 7 is 22

Please enter a command or type? **b**
Number of questions: 4

Please enter a command or type? **e**
Enter a year: 1900
1900 is not a leap year

Please enter a command or type? **e**
Enter a year: 1996
1996 is a leap year.

Please enter a command or type? **e**
Enter a year: 2000
2000 is a leap year

Please enter a command or type? **b**
Number of questions: 7

Please enter a command or type? **q**
Press any key to continue . . .

Submission

- Go to the course web site (my.asu.edu), and then click on the on-line Submission tab.
- Submit your Assignment5.java and Geek.java file on-line. Make sure to choose Hw5 from drop-down box.

Important Note: You may resubmit as many times as you like until the deadline, but we will only mark your last submission.

NO LATE ASSIGNMENTS WILL BE ACCEPTED.