Exam 1

Contents

Section 1

Upload a text file entitled `Section_1.txt` with the answers to these questions.

Question 1: Which of these are valid variable names in Java?

- `1Transaction`
- `User Name`
- `my_password`
- `if`

Question 2: What would be the output of this code?

```
System.out.println("Hi my\nname\n\n is \"Tom\"");
System.out.println("Music Directory C:\\music\nirvana\nevermind");
```

Question 3: What is the difference between a static method and an instance method?

Section 2

There's a Java file attached called `Section2.java`. Make your edits to that.

Task 1

Write a static method with the following signature:

```
public static int sum(int x)
```

The method should take an integer, return the sum of x and every smaller positive integer added together. For example, `sum(5)` should return 15, because $1 + 2 + 3 + 4 + 5 = 15$. `sum(20)` should return 210.

Extra Credit

I will give extra points if you can calculate this in constant-time/closed-form. (HINT: Use the "math formula"/"triangle formula" $\frac{n \times (n+1)}{2}$)

Task 2

Write a static method with the following signature:

```
public static double positiveQuadratic(double a, double b, double c)
```

This method should perform the positive part of the quadratic equation.

For those who need reminding, given $0 = ax^2 + bx + c$, then $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

You may assume that the discriminant (the part under the square root) will always be positive and that the answer won't involve imaginary numbers, and you only need to return back one result instead of two.

Task 3

Write a static method with the following signature:

```
public static void nameAbbreviator()
```

This method should ask the user their first name and their last name. Print out a message using the person's initials in the following template:

```
Hi <first initial>. <last initial>.

or can I call you <first name>?
```

For example: assuming that the user entered the name "Tom" for the first name, and "Gebert" for the last name, it should then print to the screen:

```
Hi T. G.

Or can I call you Tom?
```

Task 4

Write a static method with the following signature:

```
public static boolean areEqual(String s1, String s1)
```

This method should take in two strings `s1` and `s2`, and return `true` if they are equal. *This method should ignore case!*

For example, `areEqual("string", "STRING")` should return `true`, while `areEqual("foo", "bar")` should return `false`.

Task 5

Write a method with the following signature:

```
public static void fizzBuzz(int n)
```

This method should perform the famous "Fizz Buzz" problem.

The Fizz Buzz problem is as follows: take in any positive integer n. Print out the sequence of numbers from 1 to n, except when the number is divisible by 3, print `fizz`. When the number is divisible by 5, print `buzz`. The number is divisible by both 3 *and* 5, print out `fizzbuzz`.

Example output when calling `fizzBuzz(20)`:

```
1
2
fizz
4
buzz
fizz
7
8
fizz
buzz
11
fizz
13
14
fizzbuzz
16
17
fizz
19
buzz
```

Section 3

Dog Class

For this section, create a file called `Dog.java`

In this file, there should be a class `Dog`.

Dog should have the following fields, and they should be `private`.

- `double weight`
- `String name`
- `double height`
- `Date birthday`

There should also be the following getter/setter methods defined:

- `double getWeight()`
  - Returns the value of `weight`.
- `double getHeight()`
  - Returns the value of `height`.
- `String getName()`
  - Returns the value of `name`.
- `Date getBirthday()`
  - Returns the value `birthday`.
- `void setWeight(double w)`
  - Sets the value of `weight` with the argument `w`.
- `void setHeight(double h)`
  - Sets the value of `height` with the argument `h`.
- `void setName(String s)`
  - Sets the value `name` with the argument `s`.
- `void setBirthday(Date d)`
  - Sets the value of `birthday`.

Additionally, there should be the following public instance methods defined:

- `void bark()`
  - Print out a woofing message in the following template: `"Woof woof! I'm <name>! Bow Wow!"`
  - Assuming that the `name` value of the object is `"Spot"`, calling this method should print `"Woof woof! I'm Spot! Bow Wow!"`.
- `double doggieBMI()`
  - Calling this method should return the BMI of the dog. Use the following formula:

$$\frac{weight}{height^2}$$

Last, there should be the following overloaded constructors:

```
public Dog(double h, double w, String n, Date bd)
public Dog(double h, double w, String n)
public Dog()
```

For the first constructor, the argument `h` should be assigned to `height`, the argument `w` should be assigned to `weight`, the argument `n` should be assigned to `name`, and `bd` should be assigned to `birthday`.

For the second constructor, `h` should be assigned to `height`, `w` should be assigned to `weight`, the argument `n` should be assigned to `name`. `birthday` should assigned to the default value of "right now" (see hint below).

For the last constructor, the default value for `height` is 100, the default value for `weight` is 50, the default value of `name` is `"Spot"`, and the default value of `birthday` is "right now" (see hint below).

DATE HINT

To use a Java Date, you will need to `import java.util.Date;` at the top of the file. To do a "right now" date, simply do something like `Date x = new Date()`. `Date` without any arguments initializes to "right now".