Named Constants in Java

- Many of the programs we write include special values that are required for the logic. Examples: sales tax rate, standard sales commission, maximum lives a player can have in a game, maximum number of students in a course.

- Many of these values are constants, by that we mean:

- That they always stay the same during a program run.

- They’re usually the same from one run of the program to the next.

- Changes to the value of these constants are rare.

- There are reasons we should assign names to these values instead of the numerical values representing these variable constants in the code.

- Readability: If I assign

SALES\_TAX\_RATE = 0.075;

Then I know exactly what “SALES\_TAX\_RATE” is. I might have to look back in my code to know its value, but once I do, I know that a statement involving “SALES\_TAX\_RATE” will involve sales tax rates instead of reading “0.075”.

- Conventions for naming constants:

- Use all caps.

- Separate words with underscores.

Which of the following is NOT a reason to use named constants?

1. Named constants make code easier to read.

B. Named constants allow us to change the values of the constants during the program.

C. When we need to change a constant value in a program, we only have to change it in one place.

D. When we have different constants with the same value, it is easier to distinguish between them (and change only one if change is required).

Explanation: Constants usually don’t need to be changed during a program. They’re usually the same. The reason we name them is that in the event of we need to change the value, we can just assign the name the new value.