

Variables and Data Types

For this exercise, you will need to provide a response to each of the fifty word problems defined in the code. Your goal will be to create the appropriate variables with the correct data types, assign values to the variables, and also specify the correct operators necessary to provide the expected answer.

Learning Objectives

After completion of this exercise, students will understand:

- Variable declaration, to include
 - Good variable names
 - Appropriate data types
- Variable assignment
- How to use variables, literals, and operators within arithmetic expressions

Evaluation Criteria & Functional Requirements

Your work will be evaluated based on the following criteria:

- Project compiles without build errors.
- Variables are named appropriately, with good, meaningful names.
- Data types for variables are appropriate based on the value that will be assigned to the variable.
- The expected result is correct for each task.

Getting Started

- Import the variables-and-datatypes-exercises maven project into Eclipse.
- Open the Exercises.java file that is found in the variables-and-datatypes-exercises project.
- Make sure the project builds.
- Begin working on the exercise by adding the necessary code below each question. An example is specified below.

Example

Problem:

There are four squirrels, and sixteen acorns. Assuming squirrels have a sense of fairness, how many acorns will each squirrel receive?

Solution:

```
int acorns = 16;
int squirrels = 4;
int sharePerSquirrel = acorns / squirrels;
```

Tips and Tricks

- One of the [hardest things about programming is choosing good names](#). **Spend time to find good, meaningful, and descriptive names for your variables.** Clarity and expressiveness are more important than brevity. Err on the side of longer, more descriptive names over short, cryptic ones.
 - When you are naming your variables, read them out loud. Do they read like a sentence? If you were to read them to another person, would they be able to make sense of what you are saying? If not, keep working on the names of your variables.
 - Remember, variables should be camelCase in Java.
 - **Be consistent with your choice of data types**, especially when it comes to some values like money. There are many different ways to express money. *Pick one, and stick with it throughout these exercises.*
 - **Keep your code consistent and well formatted.** When code is poorly indented, or lost in a flood of blank lines, it can be difficult to read.
 - **If you find yourself stuck on a problem for longer than fifteen minutes, move onto the next, and try again later.**
 - As you are working through problems, it is a good idea to **commit early and often**. This means that once you have written enough code to complete a task, you should add your changes and commit them.
 - Work on getting into the habit of writing [good commit messages](#).
-