Getting Started With Turing – Variables

In order to create useful programs, we need the ability to store information and retrieve that information as needed. Information is stored in the memory of the computer in specially reserved areas known as *variables*.

Identifiers

In order to keep track of these variables, we use *identifier names* (or simply *identifiers*) for each *variable*, *method*, or *class* in our program.

The rules for creating any type of identifier are:

1. Any character used in an identifier must be a letter or the alphabet, a digit (0, 1, ..., 9), or an underscore character (_).

2. The first character cannot be a digit.

3. By convention, *variables* are given identifiers using camelCase, where the first letter is lowercase, and any other words in the identifier are uppercase.

4. You cannot use any *reserved words*, each of which has a predefined meaning in the Turing language. Some examples of reserved words are:

boolean **case** **char** **else** **exit** **false**

for **if** **int** **loop** **real** **true**

Declaring Variables

To reserve space in memory for variables, we need to write a *declaration statement*, where we specify the type of a variable and its identifier (i.e., name). A simple declaration might look like

```
var <identifier> : <type>
```

For example:

```
var count : int
var area : real
var choice : char
```

It is possible to declare multiple variables of the same type using a single declaration statement of the form

var <identifier$_1$>, <identifier$_2$>, ..., <identifier$_k$> : <type>

For example:

```
var radius, circumference, area : real
```

Exercises

1. Identify, with reasons, any identifiers that should not be used for Java variables.
   (a) digitSum
   (b) retail price
   (c) switch
   (d) heightPlusDepth
   (e) this&That
   (f) priceIn$
   (g) number-of-wins
   (h) ageDuGarcon
   (i) average_age
   (j) This