Programming Basics for Processing - Variables

What is a program?

Programs are composed of comments, programming statements and code blocks.

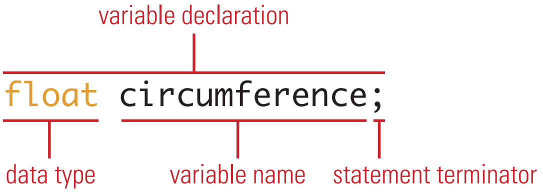
Comments

Comments are used to add notes to your code. They are not executed by the program, but are a useful way of leaving notes for yourself.

```
/* this is a comment */
// this is another type of comment
```

Variables and data types

Variables are used to store information. They are identified by a name that you specify. Each variable has a data type which specifies what kind of information they can store. Variables must be declared with a variable declaration before they can be used.



variable declaration

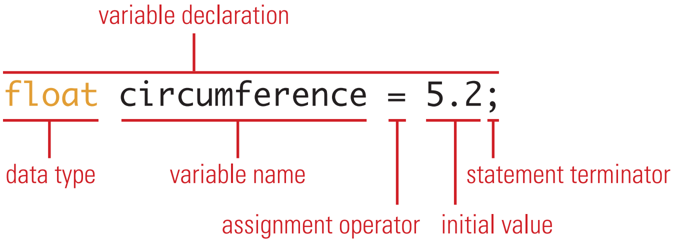
`float circumference;`

data type

variable name

statement terminator

It is also possible to assign a value to a variable when declaring a variable by including an assignment operator with the initial value.



variable declaration

`float circumference = 5.2;`

data type

variable name

statement terminator

assignment operator

initial value

Variable names are case sensitive (capitalisation matters) and can use letters, numbers and the underscore character.

The following is an example of the basic variable data types used in Processing:

`boolean is_active = true;`	only true or false
`int counter = 4;`	integers (whole numbers)
`float radius = 7.52;`	floating point numbers
`char surname_initial = “T”;`	single characters
`String my_name = “Tim”;`	character strings

Reserved words

Some words are reserved by Processing as they are used for specific purposes. This means that you should avoid using these as variable or function names. Generally these words will be displayed in a different colour when written into Processing.

boolean, break, byte, case, catch, class, char, color, continue, defaul, do, double, else, extends, false, final, float, focused, for, if, implements, import, int, long, new, null, private, public, return, static, super, this, true, void, while.

Constants, Environment and State Variables.

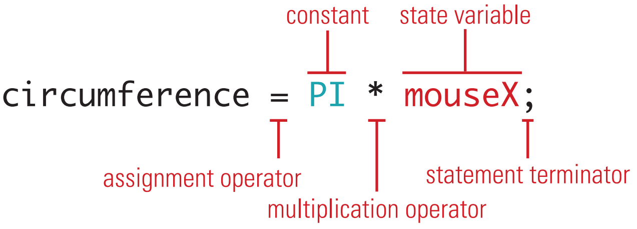
Processing also has some variables which are built into the program without having to be defined. Constants are variables with an unchangable value which are useful in mathematical equations. Some examples of these are: PI, HALF_PI and TWO_PI, which are useful in trigonometric equations.

Environment and state variables are read-only (they might change, but you cannot assign them a value directly) variables which provide information about the state of the program. Some examples are:

`width`	the width of the window
`height`	the height of the window
`mousePressed`	boolean, true if mouse button is pressed
`mouseX`	the x position of the mouse
`mouseY`	the y position of the mouse
`keyPressed`	boolean, true if a keyboard key is pressed
`key`	the current alphanumeric key being pressed
`keyCode`	used for other keys (eg. arrows, ctrl, alt)
`frameCount`	the current frame number

Expressions and Operators

An expression is a calculation that changes a variable value. Expressions are usually composed of variables, operators, constants and/or functions (sometimes just a few, sometimes all of them).



constant

state variable

`circumference = PI * mouseX;`

assignment operator

multiplication operator

statement terminator

In the above example, the value of the variable circumference will be set to pi multiplied by the current x position of the mouse.

+	addition
+	concatenate strings
-	subtraction
*	multiplication
/	division
%	modulo
+=	addition assignment
-=	subtraction assignment
*=	multiplication assignment
/=	division assignment
++	increment by one
--	decrement by one
=	assignment

Processing’s rules of precedence determine the order in which calculations are performed. This is sometimes also referred to as the order of operations.

* and / operators have higher precedence than + and - so are therefore calculated first. Brackets can be used to manipulate the order in which calculations are performed.

```
x = 4 + 2 * 3;          // x = 4 + 6 = 10;
y = (4 + 2) * 3;        // y = 6 * 3 = 12;
```