W.01

PROGRAMMING BASICS

What is a program?

Programs are composed of *comments*, *programing statements* and *code blocks*.

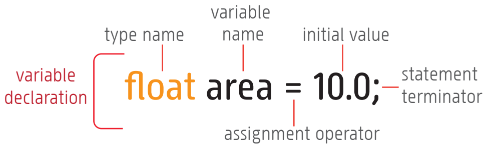
```
/* this is a comment */
// this is a comment that goes to the end of the line

area = PI * radius * radius; // a statement
int counter = 0; // another statement

/*
 * some code blocks
 */
float calculateArea(float radius) {
        return PI * radius * radius;
}
if (area < 10.0) {
        println("Area is too small!");
} else {
        println("Area is ok.");
}
```

Data types and variables

Variables are used to store information. They are identified by a name that you give them. Each variable has a *type*. Variables must be declared before they can be used.



variable declaration

type name

variable name

initial value

statement terminator

assignment operator

```
float area = 10.0;
```

The initial value is optional, but it is good programming practice to initialise variables when you declare them. Variable names can use any letter, number or the '_' character. Here are the basic variable types used by Processing:

boolean isOpen = true;	only true or false
int myCounter = 5;	whole numbers
float radius = 7.654;	floating point numbers
char middleInitial = 'P';	single characters
String yourName = "Abigail";	character strings

Reserved words

Some special words are used by Processing as programming keywords or special values so you can't use these as variable names. These include: *boolean, break, byte, case, catch, class, char, color, continue, default, do, double, else, extends, false, final, float, focused, for, if, implements, import, int, long, new, null, private, public, return, static super, this, true, try, void, while.*

Constants

PI, HALF_PI, TWO_PI, QUARTER_PI – useful for trigonometric functions (sin(), cos(), etc.), rotations.

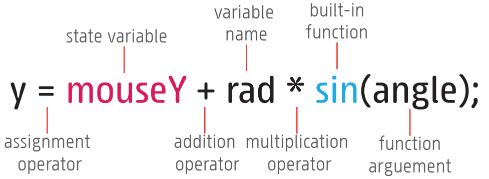
Environment and State variables

These are special, read-only variables that give you information about the mouse, window size, etc. Some examples:

frameCount	the current frame number
width, height	size of the display window
mousePressed	true if the mouse button is pressed
mouseX, mouseY	mouse location in the display
pmouseX, pmouseY	previous frame mouse location
keyPressed	true if a keyboard key is pressed
key	the current key being pressed
keyCode	used for special keys (UP, DOWN)

Operators and Expressions

Expressions typically perform some calculation. They are composed of *variables, operators, constants and functions*. Round brackets '(' and ')' are used to change the order of evaluation (precedence) and to distinguish function arguments.



state variable

variable name

built-in function

```
y = mouseY + rad * sin(angle);
```

assignment operator

addition operator

multiplication operator

function arguement

PRECEDENCE

$$y = 5 + 3 * 4;$$

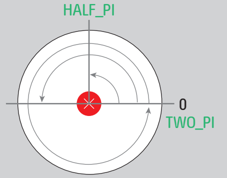
is y = 32 or 17? ie, which of:

$$y = (5 + 3) * 4;$$

$$y = 5 + (3 * 4);$$

the answer is 17, the second expression, because of Processing's rules of *precedence*. The multiplication operator has a higher 'strength' so it is evaluated first, rather than from left to right. * and / operators have higher precedence than + and -

TRIGONOMETRIC FUNCTIONS USE RADIANS



Useful Operators

+	addition	+=	add assign	++	increment
+	join strings	-=	subt. assign	- -	decrement
-	subtraction	*=	mult. assign	=	assignment
*	multiplication	/=	div. assign		
/	division				
%	modulo				

Functions

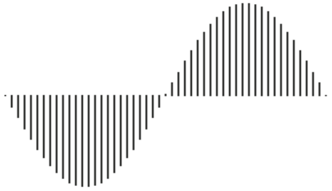
Functions encapsulate a task or calculation. Processing has many useful built-in functions, and you can also define your own. Some useful built-in functions:

random(*n*)	return a random number up to *n*
map(*v,s1,s2,e1,e2*)	remap *v* from one range to another
constrain(*v,l,h*)	constrain *v* to be between *l* and *h*
year()	return the current year
sqrt(*x*)	return the square root of *x*
print(*msg*)	print *msg* to the console area
println(*msg*)	print *msg* followed by a newline
save(*filename*)	save the display window as an image
sin(*angle*)	return the sine of *angle* (in radians)

Code Example

The code below produces the output shown.

```
float a = 0.0;
float inc = TWO_PI/50.0;
for (int i = 0; i < 204; i += 4) {
  line(i, 100, i, 100 + sin(a) * 80.0);
  a = a + inc;
}
```



creative coding

MONASH University