SmallTalk
Syntax, structure and the basics
By Scott Franchi
The Basics

- SmallTalk was created in 1971 but was realised in 1980.
- It was influenced by Simula, and influenced Java later on.
- Operates off a virtual machine and is image-based. A popular open source version is Squeak, which is what all of my examples are coded in.
- SmallTalk was among the first object-oriented languages and was the first to use the term object-oriented.
- SmallTalk has only single inheritance.
SmallTalk is dynamically typed, garbage collected and “Just-in-Time compiled”.

This means SmallTalk is compiled to byte code at run time and interpreted from there.

Variables need to be declared statically at the start of each method.

All functions are within classes (ie: everything is a method).
The Syntax

Key Words: True, False, nil, self, super and thisContext.... These are the only key words. Some may classify ^ as a key word as it indicates return

Lines end with periods.

Comments are double quotes ( “comment”).

Strings are single quotes ( x = ‘a string’).

Brackets indicate blocks, often used like curly brackets in C or Java. ( [a>5] ifTrue: [^’greater than 5’])
Methods take arguments like a sentence, rather than being comma separated like many languages.

Methods end with white space, a colon denotes a variable insertion (simpleFunction: withOneArg).

Arrays are denoted by the pound sign (#(Array elements)).

Spaces separate list/arrays (#(1 2 3 4 5)).
**Binary Search**

```plaintext
binarySearch: target over: collection
  "performs a binary search"
  [start end mid] "variables"
  start := 1. "sart at 1!!"
  end := collection size.
  [start <= end] whileTrue: "while loop!"
      [mid := start+end//2.
         Transcript cr; show: (collection at: mid).
         (collection at: mid) == target ifTrue: [Transcript cr; show: 'True'. ^mid].
         target < (collection at: mid) "no period keeps the statement going"
             ifTrue: [end := mid - 1] "-1 since mid has been checked"
             ifFalse: [start := mid +1]]. "end of expression = end of loop"
    Transcript cr; show: 'False'. ^-1 "if not returned, target is not within the collection"
```
SmallTalk uses garbage collection and pointer-counting.

The virtual machine has its own memory and this is limited.

Fully automatic, except the programmer can explicitly trigger garbage collection through a function call (SmallTalkImage garbageCollect) but this is not commonly done.
SmallTalk has few explicit primitives or control structures! All of these have functions to mimic them but they are more like built in functions and can be tampered with.

Since Sqeak is open source this means you could delete or change the code for if statements or other control structures.

Many think of small talk as the ‘purest’ object oriented language.

SmallTalk order of operations works left to right when no parentheses are given.
References

http://www.smalltalk.org/main/

http://www.squeak.org/Documentation/

http://www.smalltalk.org/versions

http://live.exept.de/doc/books/JoyOfST/IntroToST.html