

1.

X & Y variables are required to be strings. This is due the function foo specifies the required types that must be inputted and returned. Setting X to foo when foo's return type is String requires X to be a string. Foo's requirement of the first parameter to be a String requires y to be a String. S is required to be an int because of Foo's requirement of the second parameter to be an integer. Although this assumes that the user input will always be an integer.

2.

Given two sequential statements, for example, "s1;s2", s1 and s2 have axioms to describe basic truths, such as variables that are used in a method to ensure that they are initialized properly. The axioms s1 and s2 are in fact program statements, therefore the new axioms for each statement after execution of each statement (such as new variables initializing a value). Essentially for s1 and s2, if the initial axiom, the precondition, is true and the program successfully terminates, thus the ending axiom, the postcondition, must also be true.

3.

This should be sufficient to allow type checking to be performed, as Java does similar type checking during compile time and looks the same as the above initialized variables. The issue may just be syntactically confusing, however, the compiler should be fine in detecting the type and is able to internally separate the type declaration from the rest of the characters to the physical variable name.