Evaluating Expressions

Code tests always result in a true or false. Expressions with 'OR' (shown by ||) are true if any part inside is true. Expressions with 'AND' are true if ALL parts inside are true. Same with equals (==), NOT equals (!=), greater than (>), less than (<), etc.

Circle T/F for each expression

T	F	3 == 4
T	F	4 > 3
T	F	(3 > 5) || (3 < 5)
T	F	(3 > 5) && (3 < 5)
T	F	4 != 6
T	F	(123 > 0) && (3<5) &&(5 == 6)
T	F	(123 > 0) || (3<5) || (5 == 6)
T	F	(3==4) || (12>10) || (3!= 4)

Programming uses the same tests, you just replace numbers with variables. Try it again with these variables:

X = 3; Y = 4, Z = 5; Test = "start"; check = "start"; finish = "end"

Circle T/F for each expression using variables above

T	F	Test == check
T	F	Y > X
T	F	(X > Z) || (X < Z)
T	F	(X > Z) && (X < Z)
T	F	Test != finish
T	F	(123 > Z) && (X<Z) &&(check == finish)
T	F	(Y > 0) || (X<Z) || (Test == finish)
T	F	(X==Y) || (3*X>10) || (X != Y)

Tricky ones: If you write an expression to check for equals (==) but instead use "assign" (=) then the expression will check the value of the assigned variable. If it is anything but zero, it will be true. But, if the variable gets assigned to 0, then it will be false. Also, you must obey all parentheses to evaluate expressions inside of other expressions.
Circle T/F for each expression using variables above

T	F	Y = 3+Z
T	F	Y = X - X
T	F	(3 != 4) && (((4 == 4) || (53 < 0)) && (2<3))