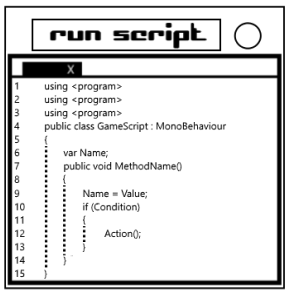
On the Subject of Scripting

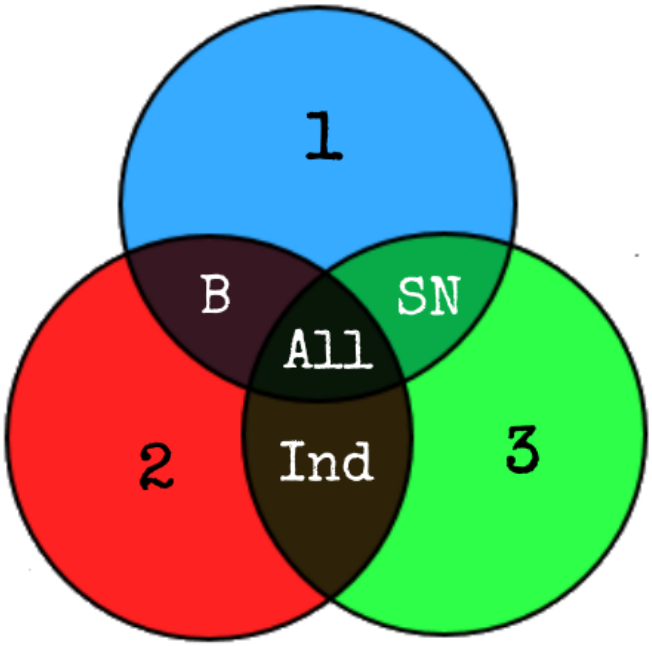


As a scripter, I can say none of this makes sense.

- On this module, a version of "Microsoft Visual Studio" can be found.
- The script has errors and needs to be fixed.
- Press the parts of the script that need to be changed.
- Running a faulty script will strike you.

Section A: Using directives.

Find out what Using directives are unnecessary/ The number indicates the unnecessary using directive.



Rules:

- **If multiple are correct, go to exceptions.**
- **None are unnecessary if none of the rules apply.**
- Blue (Top) is true if the amount of lit indicators is greater than all unlit.
- Red (Bottom-left) is true if the indicator "SND", "TRN" or "CLR" is present.
- Green (Bottom-right) is true if the indicator "FRQ", "SIG" or "BOB" is lit.

Exceptions:

- **If the exception doesn't apply, no Using directive is unnecessary**
- B: Do the amount of batteries modulo 2. If 0, both 1 and 2 are unnecessary.
- SN: If the Serial Number's last digit is 5 or higher, 3 is unnecessary.
- Ind: If the amount of indicators is more than the last Serial digit, 2 and 3 are unnecessary

Section B: The variable.

Check both the variable type and name and cycle through all the different options (Int, Float, Bool and Char). Use **only the first option** you find in this list.

- The variable type is "Int" when the Int value is lower than the last digit of the serial number.
- The variable type is "Float" when the Float value is lower than the amount of batteries.
- The variable is "Bool" when the condition (The If-statement) matches the value of the bool value.
- If none of the following apply, use Char.

Section C: The method type.

- If the amount of solved modules is even, then the type is "Void".
- Else the type is "Bool".

Section D: The action.

Check the action and the using directives, and look up the pair in the final table.

Using directives (Right) **Serial Number (Below)**	**Any using directive contains "KTaNE"**	**Else**
Contains any letter in the word "KTaNE"	HandleSolve();	HandleStrike();
Else has a vowel	Solve();	Strike();
Else	OnSolve();	OnStrike();