

Position

AT embed point name

Each line in a script needs to be inside an AT section. This tells the processor where to execute this particular part of the script. A single script can have multiple AT sections. It can also have multiple AT sections for the same embed point. For a full description of all the available embed points see [here](#).

Conditional Construction

IF *expression operator expression*

ELSE

END

An expression is a combination of variables, and functions which ultimately resolve to a value. The operator is one of;

- < Less Than
- > Greater Than
- = Equal To
- <= Less Than or Equal To
- >= Greater Than or Equal To
- <> Not Equal To

If the expression operator expression evaluates True then the lines between the IF and the ELSE are executed. If they evaluate to False then the the lines between the ELSE and END statements are executed. Nested IF statements (ie an IF statement inside another If statement are allowed).

Assignments

variable = expression

To assign the value of an expression to a variable, use the equals sign (=) to the right of the variable, and the left of the expression.

Remarks

REM anything

INFO anything

Use REM when making notations in the script. These notations are for the benefit of people understanding what the script is doing. The Script Processor ignores them completely.

Use INFO when making notations that you want the User Module Operator to see. INFO statements are added to the DETAILS when the processing Details are requested by the user for a specific hours record.

Organizational

CALL script code

INCLUDE script code

These 2 statements allow you to execute code from one macro in another macro.

Use **CALL** when you are inside an embed point, and you only want code in the other macro, inside the same embed point, to execute. **CALLs** inside **IF** statements are also only executed if that part of the code is being executed.

Use **INCLUDE** to include an entire other macro in this macro. This is functionally equivalent to using a Cut & Paste of the other macro at the point of the include. all Embed points in the other macro will be active when this macro is active. **INCLUDE** statements are usually used at the beginning, or end of a macro. If they are used in between then, for easier understanding, they should be placed just before an **AT** statement.

CALLs and **INCLUDEs** can be nested up to 200 levels deep. should you create a recursive call (Macro A includes Macro B, and Macro B includes Macro A) then the script will terminate processing completely when 200 calls are made.

Permalink: <http://tinyurl.com/mjtosg>