A program in this language is expressed as a series of blocks of the format “SELECTOR { BODY }”. The SELECTOR represents a criterion for selecting notes or sequences of notes to be modified, and the BODY contains the modifications to be made to the selected notes. For each block, beginning with the first and moving down, the program iterates through each note of the piece, testing whether it meets the condition of the SELECTOR for that block (or is the start of a sequence of notes that meets the condition). When it encounters a note or sequence of notes that meets that condition, it modifies those notes by performing the actions that are in the BODY.

A SELECTOR takes the form ATTRIBUTE RELATION CONDITION, or ATTRIBUTE RELATION CONDITION &&/|| SELECTOR.

ATTRIBUTE := “pitch” | “pitches” | “duration” | “durations”

ATTRIBUTE gives the attribute of the note(s) that is being matched with the condition. “pitch” or “pitches” is used if selecting notes by pitch, and “duration” or “durations” is used if selecting notes by duration. The singular forms, “pitch” and “duration,” are used to select single notes, so that the operations inside the BODY are be performed on one note at a time. The plural forms, “pitches” and “durations,” are used to select sequences (of consecutive notes) that meet the specified condition.

RELATION := “==” | “<=” | “>=” | “<” | “>” | “!=”

RELATION gives the operator that is used to compare the selected attribute of the current note(s) with the values stated in the CONDITION. This turns the selector into a conditional statement, and so the notes that are selected are the ones for which this conditional statement is true.

If ATTRIBUTE is “pitch” or “pitches,” then:

* “==” is satisfied when the pitch(es) of the current note(s) match the pitch(es) indicated in the CONDITION. In other words, “==” is used to select notes that are **equal to** a given pitch or sequence of pitches.
* “<=” is satisfied when the pitch of the current note is lower than or the same as the pitch indicated in the CONDITION. If the pitch indicated in the CONDITION does not have a specific octave (e.g. “tonic,” “A”), then this operator cannot be used. In other words, “<=” is used to select notes that are **lower than or equal to** a given pitch.
* “>=” is satisfied when the pitch of the current note is higher than or the same as the pitch indicated in the condition. If the pitch indicated in the CONDITION does not have a specific octave (e.g. “tonic,” “A”), then this operator cannot be used. In other words, “>=” is used to select notes that are **higher than or equal to** a given pitch.
* “<” is satisfied when the pitch of the current note is lower than the pitch indicated in the CONDITION. As with “<=” and “=>,” this can only be used when the pitch indicated in the CONDITION has a specific octave.
* “>” is satisfied when the pitch of the current note is higher than the pitch indicated in the condition. As with “<=,” “=>,” and “<,” this can only be used when the pitch indicated in the CONDITION has a specific octave.
* “!=” is satisfied when the pitch(es) of the current note(s) do not match the pitch indicated in the CONDITION.

If ATTRIBUTE is “duration” or “durations,” then:

* “==” is satisfied when the duration(s) of the current note(s) match the duration(s) indicated in the CONDITION.
* “<=” is satisfied when the duration(s) of the current note(s) is shorter than or the same as the duration(s) indicated in the CONDITION.
* “>=” is satisfied when the duration of the current note is longer than or the same as the duration indicated in the condition.
* “<” is satisfied when the duration of the current note is shorter than the duration indicated in the CONDITION. As with “<=” and “=>,” this can only be used when the pitch indicated in the CONDITION has a specific octave.
* “>” is satisfied when the pitch of the current note is higher than the pitch indicated in the condition. As with “<=,” “=>,” and “<,” this can only be used when the pitch indicated in the CONDITION has a specific octave.
* “!=” is satisfied when the pitch(es) of the current note(s) do not match the pitch indicated in the CONDITION.

CONDITION := <music21.pitch.Pitch object> | <music21.duration.Duration object> | SCALEDEGREE | LETTERNAME | DURATIONNAME | “x” | VARDEF | VAR | CONDITION, CONDITION | CONDITION…CONDITION

CONDITION is a representation of pitches, durations, or a sequence of pitches or durations, which the desired attribute of the selected note(s) is compared to using RELATION.

* If ATTRIBUTE is “pitch” (or “pitches”), then CONDITION can be a music21.pitch.Pitch object; for example, prev.pitch, which gives the pitch of the note previous to the one currently selected.
* Similarly, if ATTRIBUTE is “duration” or “durations,” then CONDITION can be a music21.duration.Duration object, such as prev.duration, which refers to the duration of the previous note.
* If ATTRIBUTE is “pitch” or “pitches,” then, if the key of the input melody is known, CONDITION can contain a representation of a pitch using its scale degree: “tonic,” “supertonic,” “second,” “mediant,” “third,” “subdominant,” “fourth,” “dominant,” “fifth,” “submediant,” “sixth,” “subtonic,” or “seventh.” Each of these represents a note with the corresponding scale degree in any octave. These pitch representations do not have a specific octave, so they can only be used when RELATION is “==” or “!=”.
* If ATTRIBUTE is “pitch” or “pitches,” then CONDITION can represent a pitch using its letter name, with or without a specified octave. These pitches can be represented in the same way they are in music21: “A,” “B,” “C,” “D,” “E,” “F,” “G,” with “+” or “-“ after the letter name to represent sharps and flats, respectively, to represent a pitch with a given letter name that could be in any octave. To specify an octave, the octave number is added after the letter name and any sharps or flats; for example, “G-4”.
* If ATTRIBUTE is “duration” or “durations,” then CONDITION can represent a duration by its name: “double whole,” “half,” “quarter,” “eighth,” “16th,” “32nd,” “64th,” etc. To represent a dotted note, the word “dotted” is added at the beginning (e.g. “dotted eighth”).

When CONDITION is defining a sequence, there are additional forms that can be used to represent pitches and durations.

A sequence of notes of a specified length can be represented as a series of pitch/duration representations above (depending on ATTRIBUTE), separated by commas.

* The keyword “x” can be used in a sequence. It represents the desired attribute (pitch or duration) of any given note in the sequence, and is used to …. For example, two occurrences of “x” in the sequence signify that the corresponding two notes must have the same value of ATTRIBUTE.
* Variables can also be defined in terms of x and used in the sequence. This is done in the format “varname | varname RELATION othervar”; for example, “y|y<x” to specify that another note in the sequence must be lower in pitch or shorter in duration (depending on ATTRIBUTE) than x. A new variable **can** be defined in terms of x or of an existing variable earlier in the sequence than that other variable (or x) first appears. This is because, if there are any variable definitions in the sequence, the program begins by mapping the variables to notes, finding which note x is set to be, checking the conditions for x if there are multiple occurrences of it, then checking the conditions for the variables defined in terms of x, then checking the conditions for the variables defined in terms of those variables, and so on.
* If a variable does not need to be used again in the sequence, it is also valid to simply write “RELATION othervar,” without specifying a new variable name; for example, “<x” to represent that the corresponding note in the sequence is lower in pitch or shorter in duration (depending on ATTRIBUTE) than x.

A sequence of notes of an unspecified/flexible length can be represented given criteria for the starting and ending points. It is also valid to provide criteria for notes in the middle of the sequence. Any of the above representations of pitch or duration can be used to specify the starting and ending criteria (or middle notes) of the sequence.

To specify such a sequence, “CONDITION…CONDITION” is used. The “…” represents a sequence of notes of any length that starts immediately after the note represented by the CONDITION right before it and ends immediately before the **first** occurrence of a note that matches the CONDITION right after it. (The endpoints are included in the sequence already due to the fact that they are being specified.)

To specify two consecutive notes in the sequence, commas may be used as described in the previous section. (For example, “x, <x … tonic”)

The following CONDITIONS are **invalid**:

* pitches <= x, >x, <x
* pitches == tonic…x
* pitches >= G, A, G, C
* pitches < x…tonic…x

The following CONDITIONS are **valid**:

* durations > half, quarter, quarter
  + This returns True for a given sequence of 3 notes if the first note is longer than a half note and the second and third notes are each longer than a quarter note.
* pitches >= G3, A3, G3, C4
* pitches == x…tonic…x