Program

Author

Email

Currently, unsupported in swimDSL

First Name

Fully supported through `set Author "[Author Name]"`

Last Name

Currently, unsupported in swimDSL

Creation Date

Fully supported through `set Date "[Creation Date]"`

Hide Intro

Fully supported through `set HideIntro [True|False]`

Instruction

Breath

Currently, unsupported in swimDSL

Continue

Currently, unsupported in swimDSL

Equipment

Fully supported through `[Distance] [Stroke Name] + [Equipment Name]`

Exclude Align

Currently, unsupported in swimDSL

Instruction Description

Currently, unsupported in swimDSL

Intensity

Start Intensity

Percentage Effort

Fully supported through `[Distance] [Stroke Name] @ [Effort Percentage]%`

Percentage Heart Rate

Currently, unsupported in swimDSL

Zone

Supported via pace definition and intensity specification, e.g.

```
pace [Zone Name] = [Percentage Effort]%
[Distance] [Stroke Name] @ [Zone Name]
```

> Note: If undesired, this step of defining a zone name as a pace is easily removable. However, it is my personal belief that this is how swiML should function, as it makes the language significantly more flexible.

Stop Intensity

Percentage Effort

Fully supported through

`[Distance] [Stroke Name] @ [Effort Percentage]% -> [Effort Percentage]%`

Percentage Heart Rate

Currently, unsupported in swimDSL

Zone

Supported via pace definition and varying intensity specification, e.g.

```
 pace [Zone Name] = [Percentage Effort]%
pace [Zone Name] = [Percentage Effort]%
[Distance] [Stroke Name] @ [Zone Name] -> [Zone Name]
```

Length

Length As Distance

Fully supported through `[Distance] [Stroke Name]`

Length As Laps

Currently, unsupported in swimDSL

Length As Time

Currently, unsupported in swimDSL

Pyramid

Currently, unsupported in swimDSL

Repetition

Repetition Count

Fully supported through

```
[Repetition Count] x {
    [[As many instructions as desired]]
}
```

Repetition Description

Currently, unsupported in swimDSL

Simplify

Currently, unsupported in swimDSL

Rest

After Stop

Fully supported through `rest [Duration]`

> Note: Currently this doesn't match swiML's model of the rest being attached to a particular instruction, and a swimDSL rest generates its own swiML instruction node. This won't be hard to change.

In Out

Currently, unsupported in swimDSL

Since Last Rest

Currently, unsupported in swimDSL

Since Start

Fully supported through `[Distance] [Stroke Name] on [Duration]`

Segment Name

Full supported through `> [Segment Name]`

Stroke

Drill

Currently, unsupported in swimDSL

> Note: swimDSL does have syntax to specify that an instruction is a drill, however there is currently no ability to specify what drill. This was done because of my personal belief that in its current state, swiML should not attempt to create a list of accepted drills. This makes the standard to rigid, limiting coaches from specifying any drill that they wish. I believe that just like how I swimDSL has pace definitions for intensity zones, it should also have drill definitions. This makes the language far more flexible and useful for coaches.

Kicking

Currently, partially unsupported in swimDSL

> Note: swimDSL does have syntax to specify kicking, however to a mismatch between the mental model of kicking/pulling in swimDSL compared to swiML the XML code generation for such as feature was never implemented. In my mind, pull is not a drill and should be treated no differently to kick. As such, I implemented "stroke modifier" syntax to specify kick/pull/drill. It again is my personal belief that swiML should model pulling in the way it models kicking. I believe this makes the language more simple, straight forward, and flexible for users.

Standard Stroke

Fully supported through `[Distance] [Stroke Name]`

Underwater

Currently, unsupported in swimDSL

Layout Width

Fully supported through `set LayoutWidth [Layout Width]`

Length Unit

Fully supported through `set LengthUnit "[Length Unit]"`

Numeral System

Fully supported through `set NumeralSystem "[Numeral System]"`

Pool Length

Fully supported through `set ProolLength [Pool Length]`

Program Align

Fully supported through `set Align [True|False]`

Program Description

Fully supported through `set Description "[Programme Description]"`

Title

Fully supported through `set Title "[Programme Title]"`