

SDLang

Simple Declarative Language

SDLang is a simple and concise way to textually represent data. It has an XML-like structure – tags, values and attributes – which makes it a versatile choice for data serialization, configuration files, or declarative languages. Its syntax was inspired by the C family of languages (C/C++, C#, D, Java, ...).

```
// This is a node with a single string value
title "Hello, World"

// Multiple values are supported, too
bookmarks 12 15 188 1234

// Nodes can have attributes
author "Peter Parker" email="peter@example.org" active=true

// Nodes can be arbitrarily nested
contents {
  section "First section" {
    paragraph "This is the first paragraph"
    paragraph "This is the second paragraph"
  }
}

// Anonymous nodes are supported
"This text is the value of an anonymous node!"

// This makes things like matrix definitions very convenient
matrix {
  1 0 0
  0 1 0
  0 0 1
}
```

Implementations



More implementations

- [SingingBush/SDL](#) An up-to-date Java fork of the original SDL library
- [SDLite](#) Lightweight and fast parser/generator implementation for D
- [SDL-dart](#) An SDLang parser written in Dart (also usable in JavaScript with `dart2js`)

Data types

```
// Strings
"String \"with escape support\""
`String "without escape support"`

// Numbers
10 // 32-bit integer
10L // 64-bit integer
10.5 // 64-bit float (double)
10.5d // 64-bit float
10.5f // 32-bit float
10.123BD // 128-bit decimal

// Boolean and null values
true // boolean true
false // boolean false
on // boolean true
off // boolean false
null // a null value

// Date/time formats
2015/12/06 12:00:00.000-UTC // Date/time value (UTC timezone)
2015/12/06 12:00:00.000 // Date/time value (local time)
2015/12/06 // Date value
12:14:34 // Duration: 12 hours, 14 minutes, 34 seconds
12:14:34.123 // 12 h, 14 min, 34 s, 123 ms
2d:12:14:34 // 2 days, 12 h, 14 min, 34 s

// Binary data (Base64 encoded)
[sdf789GSfsb2+3324sf2]
```

Comment styles

```
// C++ style

/*
C style multiline
*/

tag /*foo=true*/ bar=false

# Shell style

-- Lua style
```

Syntax details

```
// Trailing semicolons are optional
title "Some title";

// They can be used to separate multiple nodes
title "Some title"; author "Peter Parker"

// Tags may contain certain non-alphanumeric characters
this-is_a.valid$tag-name

// Namespaces are supported
renderer:options "invisible"
physics:options "nocollide"

// Nodes can be separated into multiple lines
title \
  "Some title"
```

Tools

[SDLang HTML syntax highlighter](#)

[Syntax highlighting plug-in for Sublime Text](#)

[Syntax highlighting plug-in for Atom](#)

[Major mode for Emacs](#)

[Experimental VIM syntax file](#)

Who uses SDLang? ¹

ORACLE®

Bank of America 

IKAYZO 

DJB 

JTest

References

[Language Reference](#)

[Website issue tracker/Contact](#)

[Original homepage \(mirror\)](#)

[Original Reference \(mirror\)](#)

Copyright © 2015, Sönke Ludwig. Distributed under the terms of the MIT license. All trademarks in this document are owned by their respective trademark holders.

¹ Information based in parts on the [SDL homepage \(mirror\)](#)