

---

# **snekBOL Documentation**

***Release 0.1***

**Thomas Craig**

**Jan 07, 2021**



---

## Contents

---

<b>1</b>	<b>Introduction</b>	<b>3</b>
<b>2</b>	<b>Installation</b>	<b>5</b>
2.1	Using pip . . . . .	5
2.2	Using setup.py . . . . .	5
<b>3</b>	<b>API</b>	<b>7</b>
3.1	Annotation . . . . .	7
3.2	Collection . . . . .	7
3.3	Component defintion . . . . .	7
3.4	Components . . . . .	7
3.5	Document . . . . .	7
3.6	Identified . . . . .	7
3.7	Location . . . . .	8
3.8	Model . . . . .	8
3.9	Sequence . . . . .	8
3.10	Namespaces . . . . .	8
3.11	Types . . . . .	8
<b>4</b>	<b>Indices and tables</b>	<b>9</b>
	<b>Python Module Index</b>	<b>11</b>
	<b>Index</b>	<b>13</b>



snekBOL is a python library for the reading and writing of Synthetic Biology Open Language (SBOL) 2.1.0 files. This is a Python 3 implementation of the standard and does not use a wrapper like pySBOL.

For more details on the SBOL standard see [the official site](#).



# CHAPTER 1

---

## Introduction

---

snekBOL is a python implementation of the [Synthetic Biology Open Language \(SBOL\)](#) standard. The current version of snekBOL is a very early version that aims to eventually fully support the 2.1.0 standard.





snekBOL supports Python 3 (tested on python 3.4+). It is almost pure python so should work on any OS that supports the libxml2 and libxslt libraries (required for lxml). It has been tested on MacOS and Linux; Windows may or may not be supported, I haven't tested it.

### 2.1 Using pip

snekBOL can easily be installed from PyPi using pip. You must ensure that libxml2 and libxslt are installed first otherwise it won't install.

```
pip install snekbol
```

### 2.2 Using setup.py

If you require the latest development version you can download from Github and install manually. Again, like with pip you will need to have libxml2 and libxslt installed first.

1. Get the code from github *git clone <https://github.com/tjomas/snekbol>*
2. Change to the directory and install using *python setup.py install*



## 3.1 Annotation

## 3.2 Collection

```
class snekbol.collection.Collection(identity, members=[], **kwargs)
```

Bases: *snekbol.identified.TopLevel*

Groups together a set of TopLevel objects that have something in common

## 3.3 Component definition

## 3.4 Components

## 3.5 Document

## 3.6 Identified

```
class snekbol.identified.GenericTopLevel(identity, rdf_type, **kwargs)
```

Bases: *snekbol.identified.TopLevel*

```
class snekbol.identified.Identified(identity, name=None, was_derived_from=None, version=None, description=None, display_id=None, annotations=[])
```

Bases: object

Mixin to provide identity support to SBOL objects

```
class snekbol.identified.TopLevel (identity, **kwargs)  
    Bases: snekbol.identified.Identified  
    Mixin to indicate SBOL object is top level and should not be nested
```

## 3.7 Location

## 3.8 Model

## 3.9 Sequence

## 3.10 Namespaces

```
snekbol.namespaces.NS (namespace, tag)  
    Generate a namespaced tag for use in creation of an XML file
```

## 3.11 Types

## CHAPTER 4

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



### S

`snekbol.annotation`, [7](#)  
`snekbol.collection`, [7](#)  
`snekbol.identified`, [7](#)  
`snekbol.namespaces`, [8](#)





## C

`Collection` (*class in snekbol.collection*), 7

## G

`GenericTopLevel` (*class in snekbol.identified*), 7

## I

`Identified` (*class in snekbol.identified*), 7

## N

`NS()` (*in module snekbol.namespaces*), 8

## S

`snekbol.annotation` (*module*), 7

`snekbol.collection` (*module*), 7

`snekbol.identified` (*module*), 7

`snekbol.namespaces` (*module*), 8

## T

`TopLevel` (*class in snekbol.identified*), 7