

---

# **simplecalc**

***Release 0.2***

**Nov 26, 2019**



---

## Contents

---

<b>1</b>	<b>simplecalc</b>	<b>1</b>
1.1	Read the Blog Post . . . . .	1
1.2	List of Covered Tools . . . . .	1
1.3	Documentation . . . . .	2
1.4	API . . . . .	2
1.5	Changelog . . . . .	4
1.6	Indices and tables . . . . .	4
	<b>Python Module Index</b>	<b>5</b>
	<b>Index</b>	<b>7</b>



simplecalc is an example project to show how to set up an open source project from scratch.

## 1.1 Read the Blog Post

The blog talks about how this project was set up.

## 1.2 List of Covered Tools

### 1.2.1 Environment

- pyenv
  - pyenv-virtualenv
- poetry

### 1.2.2 Project Styling

- flake8
  - flake8-docstrings
  - darglint

- isort
  - seed-isort-config
- black
- pre-commit

### 1.2.3 Unit Testing

- pytest
  - pytest-cov
  - pytest-mock
  - xdoctest
- coverage
- tox

### 1.2.4 Continuous Integration

- Azure Pipelines

### 1.2.5 Documentation

- sphinx
- readthedocs
- sphinx\_rtd\_theme

### 1.2.6 Release

- towncrier
- poetry (again)

## 1.3 Documentation

Read the docs!

## 1.4 API

### 1.4.1 API Reference

#### Calculator

The most over-engineered calculator.

**exception** `simplecalc.calculator.CalculatorTypeError`  
Custom Type Error for calculation operations.

**exception** `simplecalc.calculator.CalculatorValueError`  
Custom ValueError for calculation operations.

`simplecalc.calculator.difference (nums)`  
Find the difference of a list of numbers.

### Example

```
>>> difference([1, 2, 3, 4])
-8
```

**Parameters** `nums (list)` – A list of numbers

**Returns** The difference

**Return type** int or float

`simplecalc.calculator.power (nums)`  
Find the power of a list of numbers.

$\{1\}^{\{2\}\{3\}}$

**Parameters** `nums (list)` – A list of numbers

**Returns** The power

**Return type** int or float

`simplecalc.calculator.product (nums)`  
Find the product of a list of numbers.

### Example

```
>>> product([1, 2, 3, 4])
24
```

**Parameters** `nums (list)` – A list of numbers

**Returns** The product

**Return type** int or float

`simplecalc.calculator.quotient (nums)`  
Find the quotient of a list of numbers.

### Example

```
>>> quotient([1, 2, 3, 4])
0.041666666666666664
```

**Parameters** `nums (list)` – A list of numbers

**Returns** The quotient

**Return type** int or float

`simplecalc.calculator.sum_(nums)`  
Find the sum of a list of numbers.

### Example

```
>>> sum_([1, 2, 3, 4])
10
```

**Parameters** `nums` (*list*) – A list of numbers

**Returns** The sum

**Return type** int or float

## 1.5 Changelog

### 1.5.1 Changelog

#### Simplecalc 0.2 (2019-11-25)

No significant changes.

#### Simplecalc 0.1.0 (2019-07-25)

### 1.5.2 Features

- Initial release. (#2)

## 1.6 Indices and tables

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



### S

`simplecalc.calculator`, [2](#)



## C

`CalculatorTypeError`, [2](#)

`CalculatorValueError`, [3](#)

## D

`difference()` (*in module `simplecalc.calculator`*), [3](#)

## P

`power()` (*in module `simplecalc.calculator`*), [3](#)

`product()` (*in module `simplecalc.calculator`*), [3](#)

## Q

`quotient()` (*in module `simplecalc.calculator`*), [3](#)

## S

`simplecalc.calculator` (*module*), [2](#)

`sum_()` (*in module `simplecalc.calculator`*), [4](#)