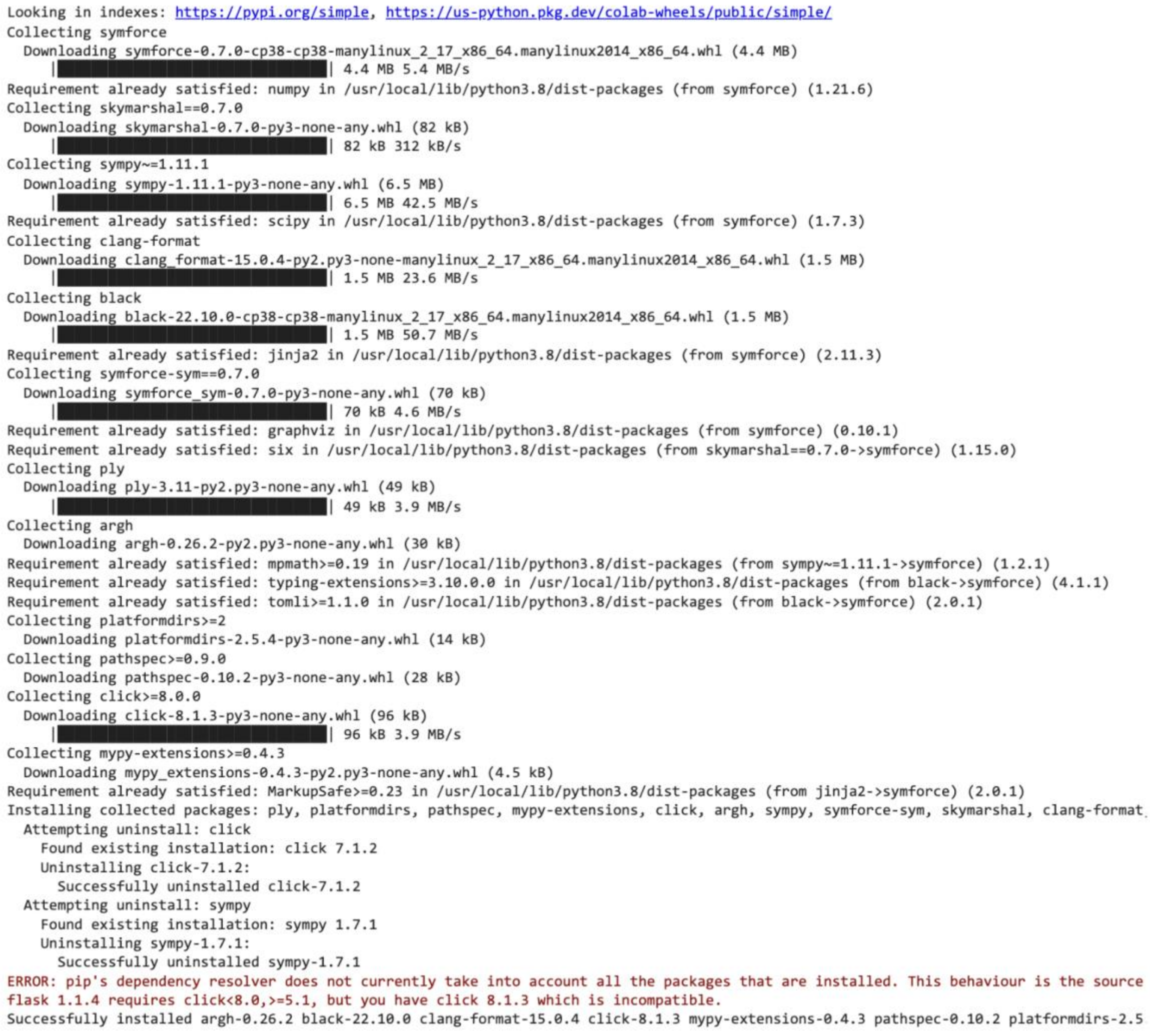
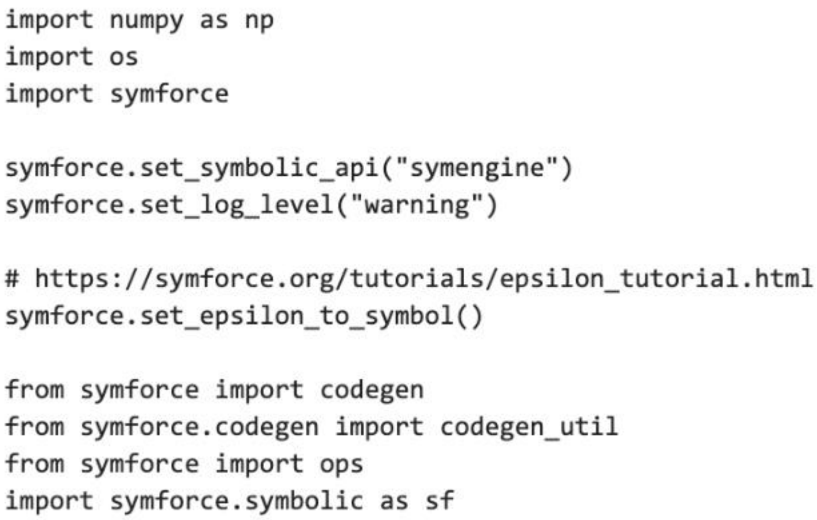
```
pip install symforce
```



```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting symforce
  Downloading symforce-0.7.0-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (4.4 MB)
     |████████████████████████████████| 4.4 MB 5.4 MB/s
Requirement already satisfied: numpy in /usr/local/lib/python3.8/dist-packages (from symforce) (1.21.6)
Collecting skymarshal==0.7.0
  Downloading skymarshal-0.7.0-py3-none-any.whl (82 kB)
     |████████████████████████████████| 82 kB 312 kB/s
Collecting sympy~=1.11.1
  Downloading sympy-1.11.1-py3-none-any.whl (6.5 MB)
     |████████████████████████████████| 6.5 MB 42.5 MB/s
Requirement already satisfied: scipy in /usr/local/lib/python3.8/dist-packages (from symforce) (1.7.3)
Collecting clang-format
  Downloading clang_format-15.0.4-py2.py3-none-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (1.5 MB)
     |████████████████████████████████| 1.5 MB 23.6 MB/s
Collecting black
  Downloading black-22.10.0-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (1.5 MB)
     |████████████████████████████████| 1.5 MB 50.7 MB/s
Requirement already satisfied: jinja2 in /usr/local/lib/python3.8/dist-packages (from symforce) (2.11.3)
Collecting symforce-sym==0.7.0
  Downloading symforce_sym-0.7.0-py3-none-any.whl (70 kB)
     |████████████████████████████████| 70 kB 4.6 MB/s
Requirement already satisfied: graphviz in /usr/local/lib/python3.8/dist-packages (from symforce) (0.10.1)
Requirement already satisfied: six in /usr/local/lib/python3.8/dist-packages (from skymarshal==0.7.0->symforce) (1.15.0)
Collecting ply
  Downloading ply-3.11-py2.py3-none-any.whl (49 kB)
     |████████████████████████████████| 49 kB 3.9 MB/s
Collecting argh
  Downloading argh-0.26.2-py2.py3-none-any.whl (30 kB)
Requirement already satisfied: mpmath>=0.19 in /usr/local/lib/python3.8/dist-packages (from sympy~=1.11.1->symforce) (1.2.1)
Requirement already satisfied: typing-extensions>=3.10.0.0 in /usr/local/lib/python3.8/dist-packages (from black->symforce) (4.1.1)
Requirement already satisfied: tomli>=1.1.0 in /usr/local/lib/python3.8/dist-packages (from black->symforce) (2.0.1)
Collecting platformdirs>=2
  Downloading platformdirs-2.5.4-py3-none-any.whl (14 kB)
Collecting pathspec>=0.9.0
  Downloading pathspec-0.10.2-py3-none-any.whl (28 kB)
Collecting click>=8.0.0
  Downloading click-8.1.3-py3-none-any.whl (96 kB)
     |████████████████████████████████| 96 kB 3.9 MB/s
Collecting mypy-extensions>=0.4.3
  Downloading mypy_extensions-0.4.3-py2.py3-none-any.whl (4.5 kB)
Requirement already satisfied: MarkupSafe>=0.23 in /usr/local/lib/python3.8/dist-packages (from jinja2->symforce) (2.0.1)
Installing collected packages: ply, platformdirs, pathspec, mypy-extensions, click, argh, sympy, symforce-sym, skymarshal, clang-format
  Attempting uninstall: click
    Found existing installation: click 7.1.2
    Uninstalling click-7.1.2:
      Successfully uninstalled click-7.1.2
  Attempting uninstall: sympy
    Found existing installation: sympy 1.7.1
    Uninstalling sympy-1.7.1:
      Successfully uninstalled sympy-1.7.1
ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour is the source
flask 1.1.4 requires click<8.0,>=5.1, but you have click 8.1.3 which is incompatible.
Successfully installed argh-0.26.2 black-22.10.0 clang-format-15.0.4 click-8.1.3 mypy-extensions-0.4.3 pathspec-0.10.2 platformdirs-2.5
```



```python
import numpy as np
import os
import symforce

symforce.set_symbolic_api("symengine")
symforce.set_log_level("warning")

# https://symforce.org/tutorials/epsilon_tutorial.html
symforce.set_epsilon_to_symbol()

from symforce import codegen
from symforce.codegen import codegen_util
from symforce import ops
import symforce.symbolic as sf
```

```
display(StorageOps.storage_dim(float))
```

1

```
display(StorageOps.storage_dim([sf.Pose3, sf.Pose3]))
```

14

```
values = Values(
    pose=sf.Pose3(),
    scalar=sf.Symbol("x"),
)
display(StorageOps.storage_dim(values))
```

8

```
display(StorageOps.to_storage(5))
```

$[5]$

```
display(StorageOps.to_storage(sf.V3(sf.Symbol("x"), 5.2, sf.sqrt(5))))
```

$\left[x,\ 5.2,\ \sqrt{5}\right]$

```
T = sf.Pose3.symbolic("T")
T_serialized = StorageOps.to_storage(T)
T_recovered = StorageOps.from_storage(sf.Pose3, T_serialized)
display(T_serialized)
display(T_recovered)
```

$[T.R_x,\ T.R_y,\ T.R_z,\ T.R_w,\ T.t0,\ T.t1,\ T.t2]$

```
<Pose3 R=<Rot3 <Q xyzw=[T.R_x, T.R_y, T.R_z, T.R_w]>>, t=(T.t0, T.t1, T.t2)>
```

Group Ops

```
display(GroupOps.identity(sf.Pose3))
```

```
<Pose3 R=<Rot3 <Q xyzw=[0, 0, 0, 1]>>, t=(0, 0, 0)>
```

```
display(GroupOps.identity(float))
```

$$0.0$$

```
display(GroupOps.inverse(sf.V3(1.2, -3, 2)).T)
```

$$\begin{bmatrix} -1.2 & 3 & -2 \end{bmatrix}$$

```
display(GroupOps.compose(sf.V2(1, 2), sf.V2(3, -5)))
```

$$\begin{bmatrix} 4 \\ -3 \end{bmatrix}$$

```
R1 = sf.Rot3.from_angle_axis(
    angle=sf.Symbol("theta1"),
    axis=sf.V3(0, 0, 1),
)
display(StorageOps.simplify(GroupOps.compose(R1, R1.inverse()).simplify()))
```

```
<Rot3 <Q xyzw=[0, 0, 0, 1]>>
```

```
R2 = sf.Rot3.from_angle_axis(
    angle=sf.Symbol("theta2"),
    axis=sf.V3(0, 0, 1),
)
R_delta = GroupOps.between(R1, R2)
```

```
display(R2)
display(StorageOps.simplify(GroupOps.compose(R1, R_delta)))
     <Rot3 <Q xyzw=[0, 0, sin(theta2/2), cos(theta2/2)]>>
     <Rot3 <Q xyzw=[0, 0, sin(theta2/2), cos(theta2/2)]>>
```

```
display(LieGroupOps.tangent_dim(sf.Rot3))
```

$$3$$

```
angle = sf.Symbol("theta")
rot2 = LieGroupOps.from_tangent(sf.Rot2, [angle])
display(rot2.to_rotation_matrix())
```

$$\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

```
display(LieGroupOps.to_tangent(rot2))
```

$$\begin{bmatrix} \operatorname{atan}_2(\sin(\theta), \cos(\theta)) \end{bmatrix}$$

```
display(LieGroupOps.from_tangent(sf.V5(), [1, 2, 3, 4, 5]).T)
```

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 \end{bmatrix}$$

```
rot2_perturbed = LieGroupOps.retract(rot2, [sf.Symbol("delta")])
display(rot2_perturbed.to_rotation_matrix())
```

$$\begin{bmatrix} -\sin(\delta)\sin(\theta) + \cos(\delta)\cos(\theta) & -\sin(\delta)\cos(\theta) - \sin(\theta)\cos(\delta) \\ \sin(\delta)\cos(\theta) + \sin(\theta)\cos(\delta) & -\sin(\delta)\sin(\theta) + \cos(\delta)\cos(\theta) \end{bmatrix}$$

```
display(StorageOps.simplify(LieGroupOps.local_coordinates(rot2, rot2_perturbed)))
```

$$\begin{bmatrix} \operatorname{atan}_2(\sin(\delta), \cos(\delta)) \end{bmatrix}$$

```
display(LieGroupOps.storage_D_tangent(rot2))
```

$$\begin{bmatrix} -\sin(\theta) \\ \cos(\theta) \end{bmatrix}$$