

# Example Typst Document

## 1. Text and headings

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri.

### 1.1. H2

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua quaerat.

#### 1.1.1. H3

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore.

## 2. Links and other references

### 2.1. Links

Typst <https://typst.app>

### 2.2. References

[Section 2](#)

### 2.3. Footnotes

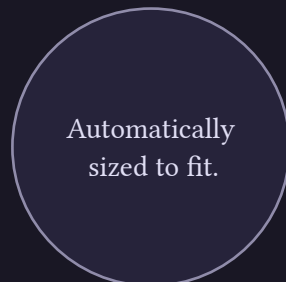
Some text<sup>1</sup>

## 3. Tables

Equation	Area	Parameters
$\pi h \frac{D^2 - d^2}{4}$	$h$ : height $D$ : outer radius $d$ : inner radius	$\frac{\sqrt{2}}{12}a^3$

## 4. Visuals

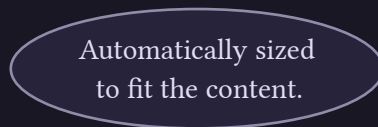
### 4.1. Circles



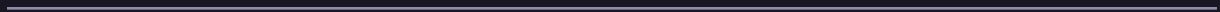
---

<sup>1</sup>footnote test

## 4.2. Ellipses



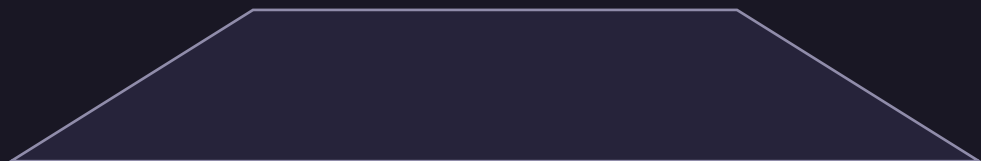
## 4.3. Lines



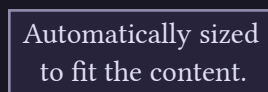
## 4.4. Paths



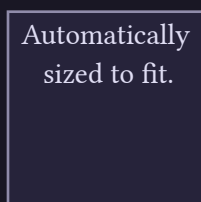
## 4.5. Polygons



## 4.6. Rectangles



## 4.7. Squares



## 4.8. Highlights

This is [important](#).

This [Link](#) is important too.

So is this reference [Section 2](#).

## 4.9. Code

Python example:

```
import numpy as np

def incmatrix(genl1,genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m,1), int) #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2),1)

    for i in range(m-1):
        for j in range(i+1, m):
            [r,c] = np.where(M2 == M1[i,j])
            for k in range(len(r)):
                VT[(i)*n + r[k]] = 1;
                VT[(i)*n + c[k]] = 1;
                VT[(j)*n + r[k]] = 1;
                VT[(j)*n + c[k]] = 1;

            if M is None:
                M = np.copy(VT)
            else:
                M = np.concatenate((M, VT), 1)

        VT = np.zeros((n*m,1), int)

    return M

from functools import wraps

from scht_lab.models.stream import Priorities
from scht_lab.topo import Link, Topology

def delay_calc(link: Link, priority: float = 1.0) -> float:
    """Calculate delay for a link."""
    return 1/(link.delay_calc() * priority)
```