In [1]:

```python
# Step 1

def compute_ratio(var):
    ratios = []
    i = 0
    j = 1
    while j < len(var):
        new_ratio = var[i]/var[j]
        ratios.append(new_ratio)
        i += 1
        j += 1
    return ratios
```

In [2]:

```python
# Step 2

def compute_ratio2(var):
    k = 0
    numeric_values = []
    while k < len(var):
        cond1, cond2 = False, False
        try:
            var[k] / 1
        except TypeError:
            pass
        else:
            cond1 = True
            cond2 = var[k] != 0
        if cond1 and cond2:
            numeric_values.append(var[k])
        k += 1
    i = 0
    j = 1
```

Loading [MathJax]/extensions/Safe.js

```python
        os = []
```

```
    while j < len(numeric_values):
        new_ratio = numeric_values[i]/numeric_values[j]
        ratios.append(new_ratio)
        i += 1
        j += 1
    return ratios
```

In [3]:
```
# Step 3

import logging

def get_logger():
    logger = logging.Logger('basic_logger')
    logger.setLevel(logging.INFO)
    handler = logging.StreamHandler()
    logger.addHandler(handler)
    return logger

def compute_ratio3(var):
    logger = get_logger()
    k = 0
    numeric_values = []
    while k < len(var):
        cond1, cond2 = False, False
        try:
            var[k] / 1
        except TypeError:
            logger.warning(f'Value {var[k]} at index {k} is non-numeric.')
        else:
            cond1 = True
            cond2 = var[k] != 0
            if ~cond2:
                logger.warning(f'Value at index {k} is 0.')

        if cond1 and cond2:
```

Loading [MathJax]/extensions/Safe.js

```
                numeric_values.append(var[k])
            k += 1
        i = 0
        j = 1
        ratios = []
        while j < len(numeric_values):
            new_ratio = numeric_values[i]/numeric_values[j]
            logger.info(f'Ratio {numeric_values[i]}/{numeric_values[j]} successfully calc
            ratios.append(new_ratio)
            i += 1
            j += 1
        return ratios
```

Loading [MathJax]/extensions/Safe.js

Loading [MathJax]/extensions/Safe.js