In [1]:
```
import numpy as np
import pandas as pd
from sklearn.preprocessing import FunctionTransformer

#sample data
data = np.array([1,2,3,4,5,6], dtype=float)

```

```
C:\Users\CompLab05\anaconda3\lib\site-packages\scipy\__init__.py:138: UserWarning: A NumPy version
is version of SciPy (detected version 1.24.3)
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion} is required for this vers
```

In [2]:
```
#log transformation
transform = FunctionTransformer(func=np.log1p) #using sklearn func
log_transformed_data = transform.fit_transform(data)

```

In [3]:
```
#square trans
square_transformed_data = np.square(data)

```

In [4]:
```
#reciprocal trans
reciprocal_transformed_data = np.square(data)
```

In [5]:
```
#trigo trans
sin_transformed_Data = np.sin(data)
cos_transformed_Data = np.cos(data)
tan_transformed_Data = np.tan(data)

```

In [6]:
```
#square root transfor
sqrt_transformed_data = np.sqrt(data)

```

In [7]:

```
#combine all in df
results = pd.DataFrame({
    'Original' : data,
    'Log(log1p)' : log_transformed_data,
    'Square' : square_transformed_data,
    'Reciprocal': reciprocal_transformed_data,
    'Sin': sin_transformed_Data,
    'Cos' : cos_transformed_Data,
    'Tan' : tan_transformed_Data,
    'Square Root' : sqrt_transformed_data

})
```

In [8]:

```
print(results)
```

```
   Original  Log(log1p)  Square  Reciprocal       Sin       Cos       Tan  \
0       1.0    0.693147     1.0         1.0  0.841471  0.540302  1.557408
1       2.0    1.098612     4.0         4.0  0.909297 -0.416147 -2.185040
2       3.0    1.386294     9.0         9.0  0.141120 -0.989992 -0.142547
3       4.0    1.609438    16.0        16.0 -0.756802 -0.653644  1.157821
4       5.0    1.791759    25.0        25.0 -0.958924  0.283662 -3.380515
5       6.0    1.945910    36.0        36.0 -0.279415  0.960170 -0.291006

   Square Root
0     1.000000
1     1.414214
2     1.732051
3     2.000000
4     2.236068
5     2.449490
```

In [ ]:

```

```