In [1]:
```
# impoting necessary libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt # visualizing data
%matplotlib inline
import seaborn as sns
```

In [2]:
```
# importing the csv file
df = pd.read_csv(r'C:\Users\hp\Desktop\bio Project\2 pyruvate.csv',encoding= 'unicode_escape')
```

In [3]:
```
# checking for Content loaded in Juypyter notebook
df.head()
```

Out[3]:

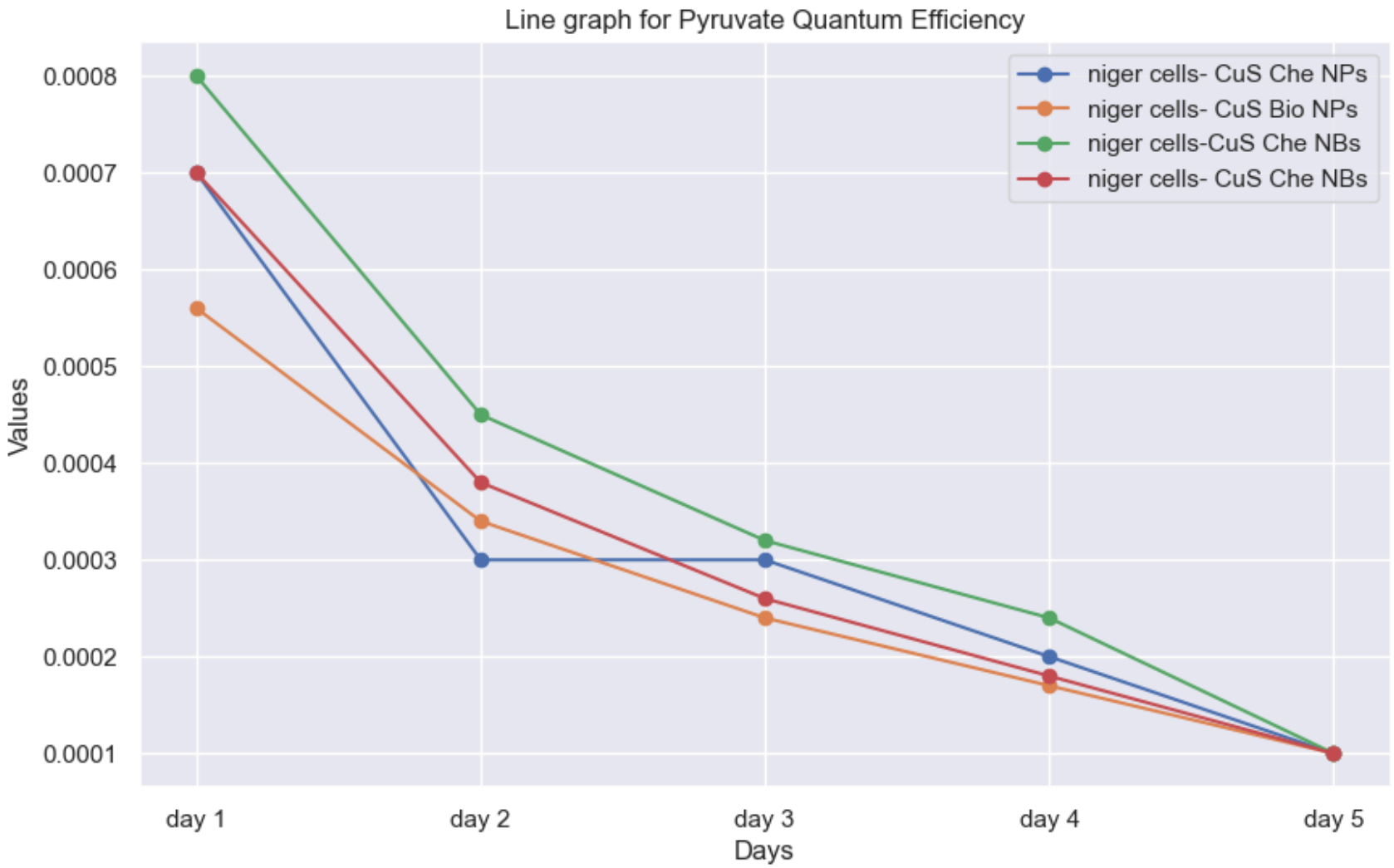
	Days	niger cells- CuS Che NPs	niger cells- CuS Bio NPs	niger cells-CuS Che NBs	niger cells- CuS Che NBs
0	day 1	0.0007	0.00056	0.00080	0.00070
1	day 2	0.0003	0.00034	0.00045	0.00038
2	day 3	0.0003	0.00024	0.00032	0.00026
3	day 4	0.0002	0.00017	0.00024	0.00018
4	day 5	0.0001	0.00010	0.00010	0.00010

In [4]:
```
# Statistics of the loaded data
df.describe()
```

Out[4]:

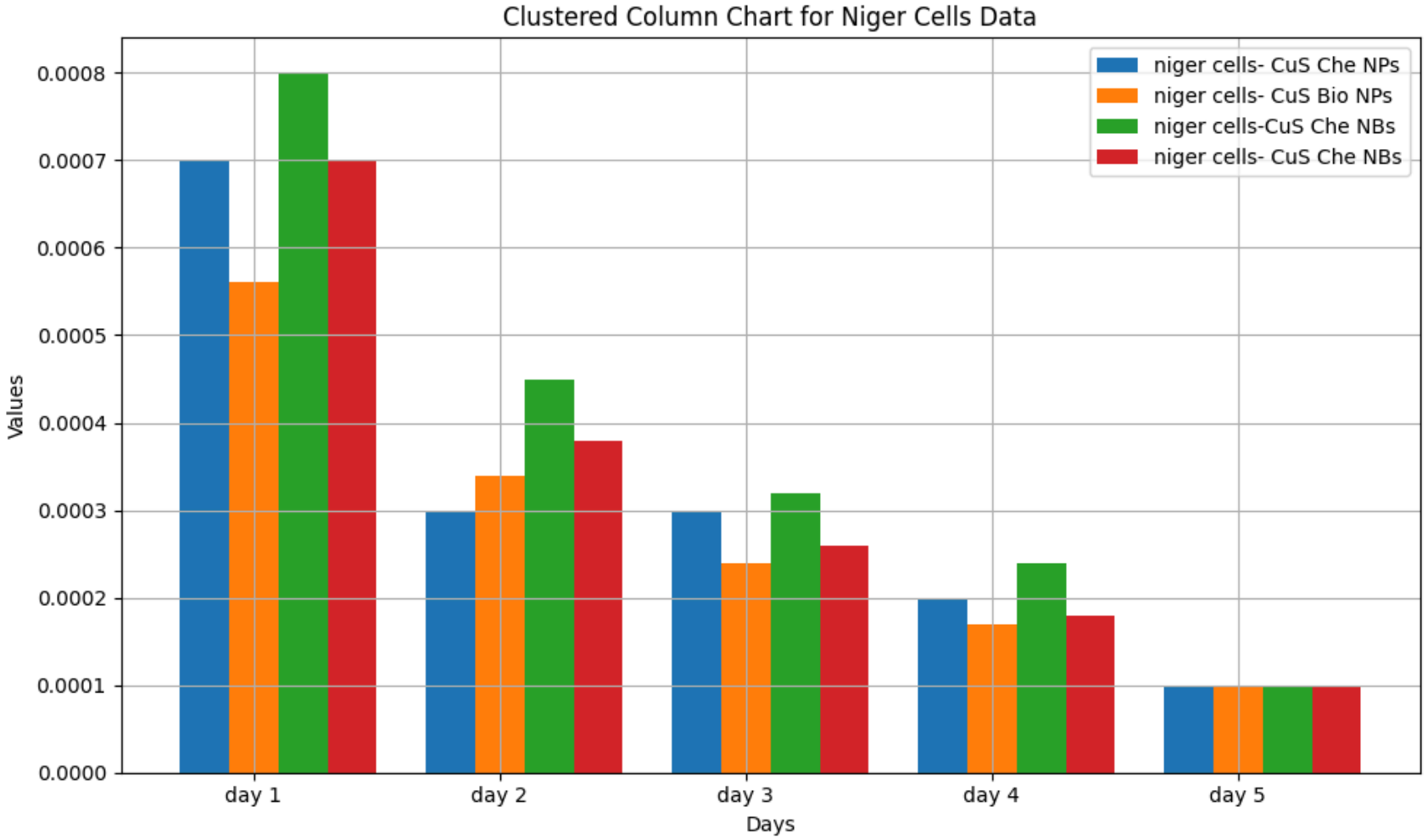
	niger cells- CuS Che NPs	niger cells- CuS Bio NPs	niger cells-CuS Che NBs	niger cells- CuS Che NBs
count	5.000000	5.000000	5.000000	5.000000
mean	0.000320	0.000282	0.000382	0.000324
std	0.000228	0.000179	0.000266	0.000234
min	0.000100	0.000100	0.000100	0.000100
25%	0.000200	0.000170	0.000240	0.000180
50%	0.000300	0.000240	0.000320	0.000260
75%	0.000300	0.000340	0.000450	0.000380
max	0.000700	0.000560	0.000800	0.000700

In [12]:
```
# Drawing linegrpah
plt.figure(figsize=(10, 6))
for column in df.columns[1:]:
    plt.plot(df['Days'], df[column], marker='o', label=column)
    plt.xlabel('Days')
plt.ylabel('Values')
plt.title('Line graph for Pyruvate Quantum Efficiency')
plt.legend()
plt.grid(True)
plt.show()
```



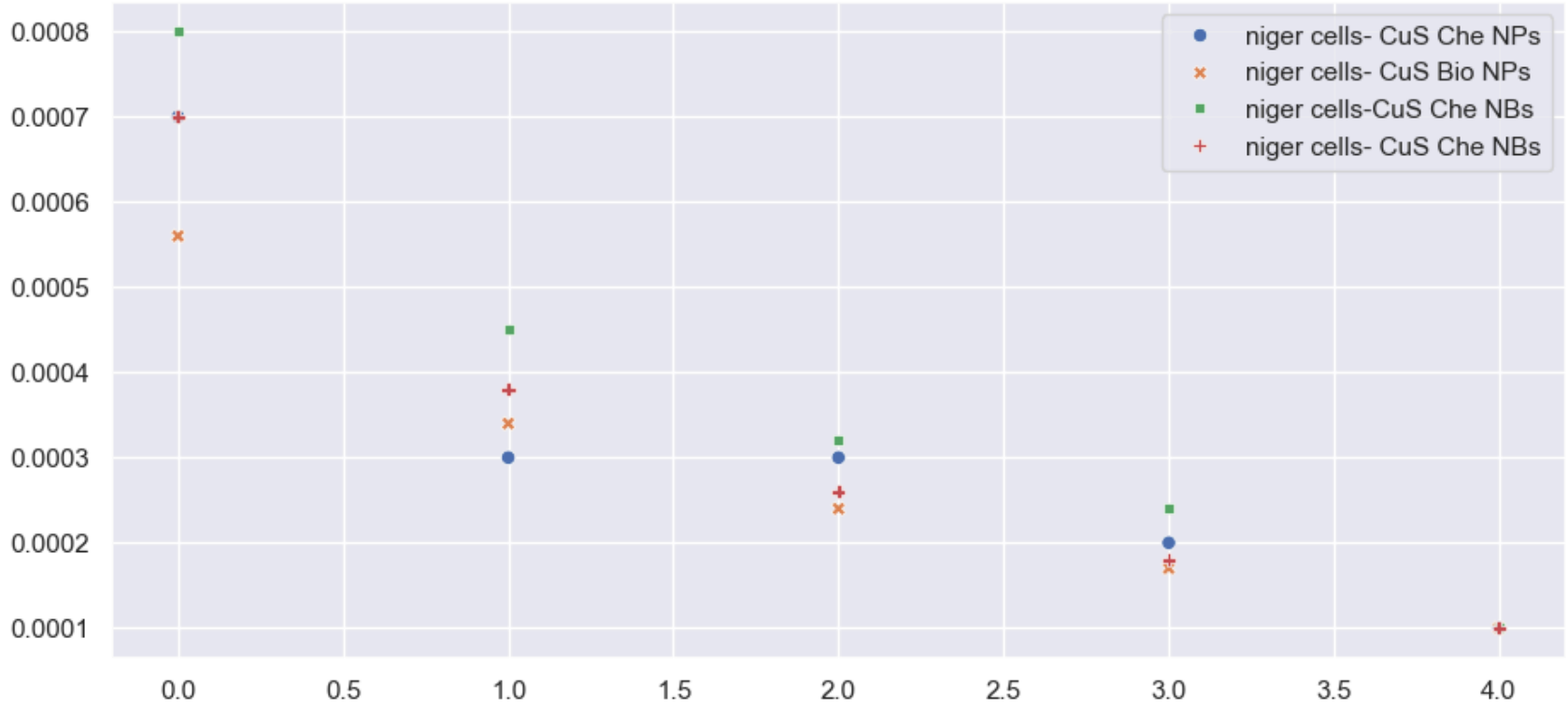
In [8]:
```
#Drawing clustered column chart
plt.figure(figsize=(10, 6))
num_columns = len(df.columns[1:])
bar_width = 0.2
index = np.arange(len(df['Days']))

for i, column in enumerate(df.columns[1:], start=1):
    plt.bar(index + i * bar_width, df[column], bar_width, label=column)
    plt.xlabel('Days')
```

```
plt.ylabel('Values')
plt.title('Clustered Column Chart for Niger Cells Data')
plt.xticks(index + (num_columns / 2) * bar_width, df['Days'])  # Aligning x-ticks with column groups
plt.legend()
plt.grid(True)
plt.tight_layout()  # Adjust layout to prevent clipping of labels
plt.show()
```



In [10]:
```
# Drawing violingraph
sns.violinplot(data=df)
sns.set(rc={'figure.figsize':(11,5)})
```

In [11]:
```
#Drawing scatteredgraph
sns.scatterplot(data=df)
sns.set(rc={'figure.figsize':(3,3)})
```


0.0008
0.0007
0.0006
0.0005
0.0004
0.0003
0.0002
0.0001
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
niger cells- CuS Che NPs
niger cells- CuS Bio NPs
niger cells-CuS Che NBs
niger cells- CuS Che NBs

In [ ]: