In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
from pprint import pprint
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression

%matplotlib inline
```

In [2]:
```
df = pd.read_csv("colinearity.csv")
```

In [3]:
```
df.head()
```

Out[3]:

	x1	x2	x3	x4	y
0	-1.109823	-1.172554	-0.897949	-6.572526	-158.193913
1	0.288381	0.360526	2.298690	3.884887	198.312926
2	-1.059194	0.833067	0.285517	-1.225931	12.152087
3	0.226017	1.979367	0.744038	5.380823	190.281938
4	0.664165	-1.373739	0.317570	-0.437413	-72.681681

In [4]:
```
X = df.drop(["y"], axis=1)

y = df.y
```

In [5]:
```
linear_coef = []

for i in X:
    x = df[[i]]
    linreg = LinearRegression()
    linreg.fit(x, y)
    linear_coef.append(linreg.coef_)
```

In [6]:
```
multi_linear = LinearRegression()
multi_linear.fit(X, y)
multi_coef = multi_linear.coef_
```

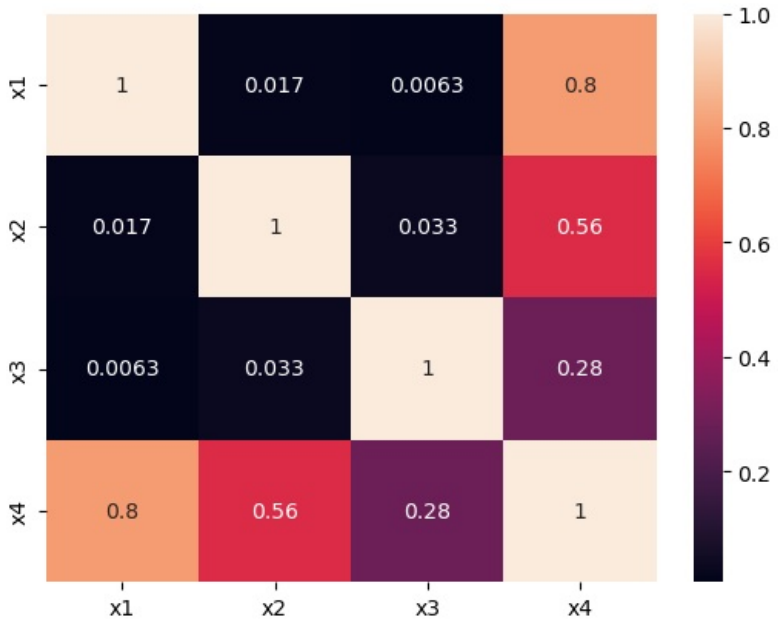
In [7]:
```
print("By simple(one variable) linear regression for each variable:", sep="\n")

for i in range(4):
    pprint(f"Value of beta{i+1} = {linear_coef[i][0]:.2f}")
```

```
By simple(one variable) linear regression for each variable:
'Value of beta1 = 34.73'
'Value of beta2 = 68.63'
'Value of beta3 = 59.40'
'Value of beta4 = 20.92'
```

In [8]:
```
print("By multi-Linear regression on all variables")
for i in range(4):
    pprint(f"Value of beta{i+1} = {round(multi_coef[i],2)}")
```

```
By multi-Linear regression on all variables
'Value of beta1 = -24.61'
'Value of beta2 = 27.72'
'Value of beta3 = 37.67'
'Value of beta4 = 19.27'
```

In [9]:
```
corrMatrix = df[["x1", "x2", "x3", "x4"]].corr()
sns.heatmap(corrMatrix, annot=True)
plt.show()
```



Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js