In [1]:

**import** pandas **as** pd

dataset**=**pd.read\_csv("C:/Users/Harshini/Downloads/50\_Startups.csv") df**=**pd.DataFrame(dataset)

df

Out[1]:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **R&D Spend** | **Administration** | **Marketing Spend** | **State** | **Profit** |
| **0** | 165349.20 | 136897.80 | 471784.10 | New York | 192261.83 |
| **1** | 162597.70 | 151377.59 | 443898.53 | California | 191792.06 |
| **2** | 153441.51 | 101145.55 | 407934.54 | Florida | 191050.39 |
| **3** | 144372.41 | 118671.85 | 383199.62 | New York | 182901.99 |
| **4** | 142107.34 | 91391.77 | 366168.42 | Florida | 166187.94 |
| **...** | ... | ... | ... | ... | ... |
| **103** | 119943.24 | 156547.42 | 256512.92 | Florida | 132602.65 |
| **104** | 114523.61 | 122616.84 | 261776.23 | New York | 129917.04 |
| **105** | 78013.11 | 121597.55 | 264346.06 | California | 126992.93 |
| **106** | 94657.16 | 145077.58 | 282574.31 | New York | 125370.37 |
| **107** | 91749.16 | 114175.79 | 294919.57 | Florida | 124266.90 |

108 rows × 5 columns

In [11]:

df.info()

<class 'pandas.core.frame.DataFrame'> RangeIndex: 108 entries, 0 to 107

Data columns (total 5 columns):

# Column Non-Null Count Dtype

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 0 |  | R&D Spend | 108 | non-null |  | float64 |
| 1 |  | Administration | 108 | non-null |  | float64 |
| 2 |  | Marketing Spend | 108 | non-null |  | float64 |
| 3 |  | State | 108 | non-null |  | object |
| 4 |  | Profit | 108 | non-null |  | float64 |

dtypes: float64(4), object(1) memory usage: 4.3+ KB

In [12]:

df.describe()

Out[12]:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **R&D Spend** | **Administration** | **Marketing Spend** | **Profit** |
| **count** | 108.000000 | 108.000000 | 108.000000 | 108.000000 |
| **mean** | 75653.105556 | 121750.788889 | 224031.590648 | 113523.760000 |
| **std** | 44348.861595 | 27322.385654 | 113887.603123 | 38991.013654 |
| **min** | 542.050000 | 51283.140000 | 1903.930000 | 14681.400000 |
| **25%** | 42692.090000 | 105077.645000 | 137962.620000 | 90708.190000 |
| **50%** | 75791.365000 | 122699.795000 | 249744.550000 | 109543.120000 |
| **75%** | 101913.080000 | 145077.580000 | 298932.675000 | 141585.520000 |
| **max** | 165349.200000 | 182645.560000 | 475411.300000 | 192261.830000 |

In [13]:

df.shape

Out[13]: (108, 5)

In [14]:

df.columns

Out[14]: Index(['R&D Spend', 'Administration', 'Marketing Spend', 'State', 'Profit'], dtype='object')

In [15]:

df.dtypes

Out[15]: R&D Spend float64 Administration float64 Marketing Spend float64 State object

Profit float64

dtype: object

|  |  |  |
| --- | --- | --- |
| In [16]: | df.isnull().sum() |  |
| Out[16]: | R&D Spend | 0 |
|  | Administration | 0 |
|  | Marketing Spend | 0 |
|  | State | 0 |
|  | Profit | 0 |
|  | dtype: int64 |  |
| In [17]: | df.duplicated() |  |

Out[17]: 0 False

1. False
2. False
3. False
4. False

...

1. True
2. True
3. True
4. True
5. True

Length: 108, dtype: bool

In [26]:

df.drop\_duplicates(keep**=False**,inplace**=True**)

In [27]:

df.duplicated().sum()

Out[27]: 0

In [28]:

df.head()

Out[28]:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **R&D Spend** | **Administration** | **Marketing Spend** | **State** | **Profit** |
| **19** | 86419.70 | 153514.11 | 123452.10 | New York | 122776.86 |
| **30** | 61994.48 | 115641.28 | 91131.24 | Florida | 99937.59 |
| **31** | 61136.38 | 152701.92 | 88218.23 | New York | 97483.56 |
| **34** | 46426.07 | 157693.92 | 210797.67 | California | 96712.80 |
| **35** | 46014.02 | 85047.44 | 205517.64 | New York | 96479.51 |

In [29]:

df.tail()

Out[29]:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **R&D Spend** | **Administration** | **Marketing Spend** | **State** | **Profit** |
| **80** | 27892.92 | 84710.77 | 164470.71 | New York | 77798.83 |
| **82** | 20229.59 | 127382.30 | 35534.17 | New York | 69758.98 |
| **86** | 1234.10 | 135426.92 | 31234.40 | California | 42559.73 |
| **87** | 542.05 | 51743.15 | 25671.30 | New York | 35673.41 |
| **88** | 734.50 | 116983.80 | 45173.06 | California | 14681.40 |

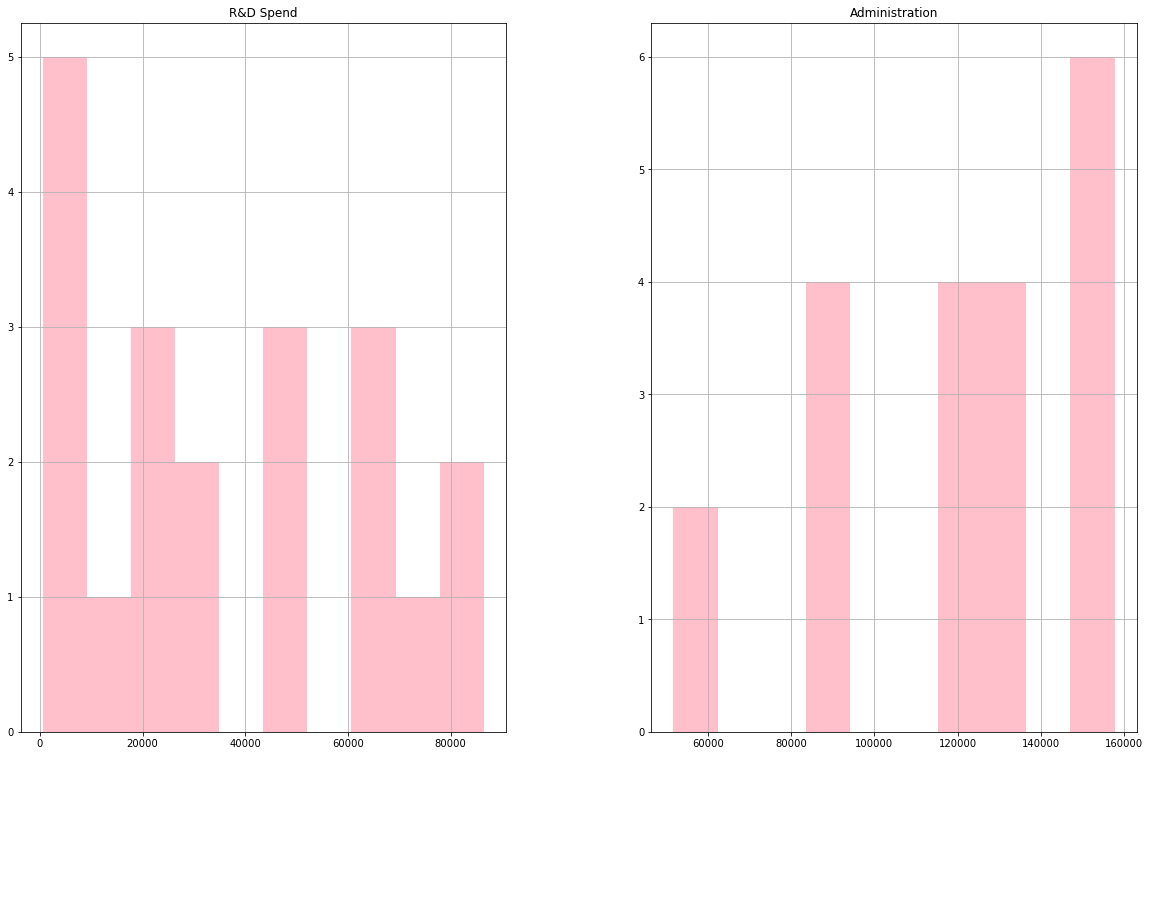
In [3]:

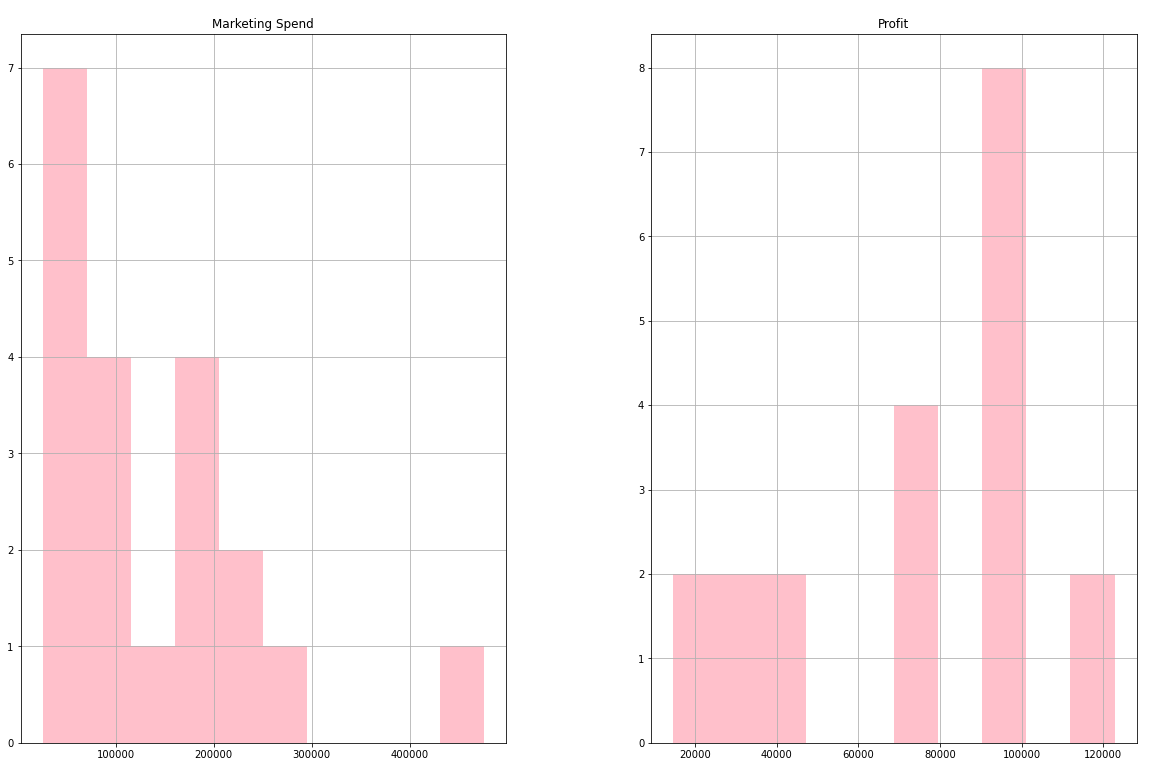
**import** matplotlib.pyplot **as** plt

**import** seaborn **as** sns

In [31]:

df.hist(figsize**=**(20,30),color**=**'pink') plt.show()

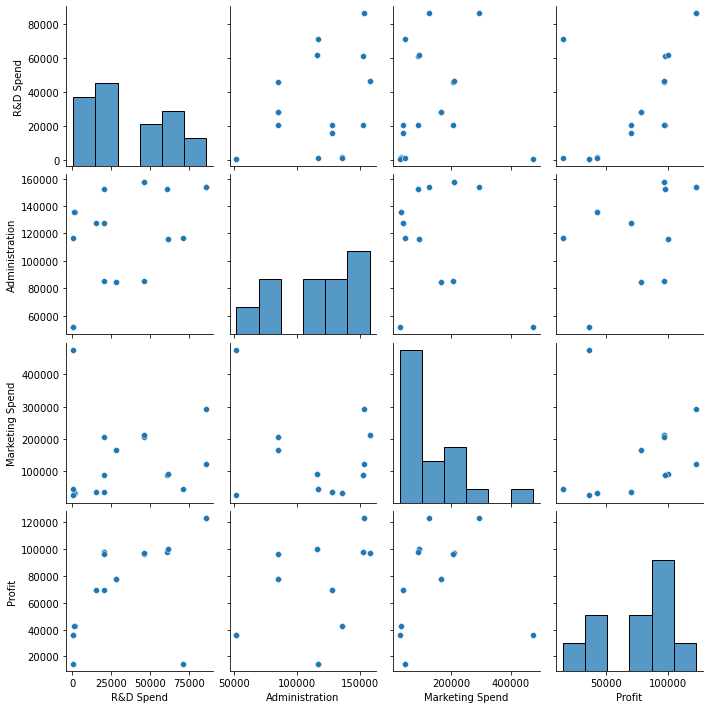






In [32]:

sns.pairplot(df) plt.show()





In [34]:

corr**=**df.corr() corr

Out[34]:

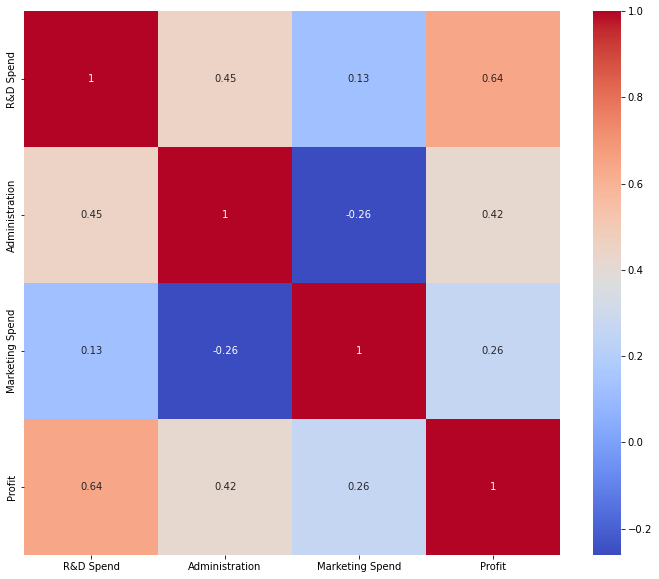
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **R&D Spend** | **Administration** | **Marketing Spend** | **Profit** |
| **R&D Spend** | 1.000000 | 0.451334 | 0.127247 | 0.640613 |
| **Administration** | 0.451334 | 1.000000 | -0.260522 | 0.418904 |
| **Marketing Spend** | 0.127247 | -0.260522 | 1.000000 | 0.261639 |
| **Profit** | 0.640613 | 0.418904 | 0.261639 | 1.000000 |

In [35]:

plt.figure(figsize**=**(12,10))

sns.heatmap(corr,annot**=True**,cmap**=**'coolwarm')

Out[35]: <AxesSubplot:>

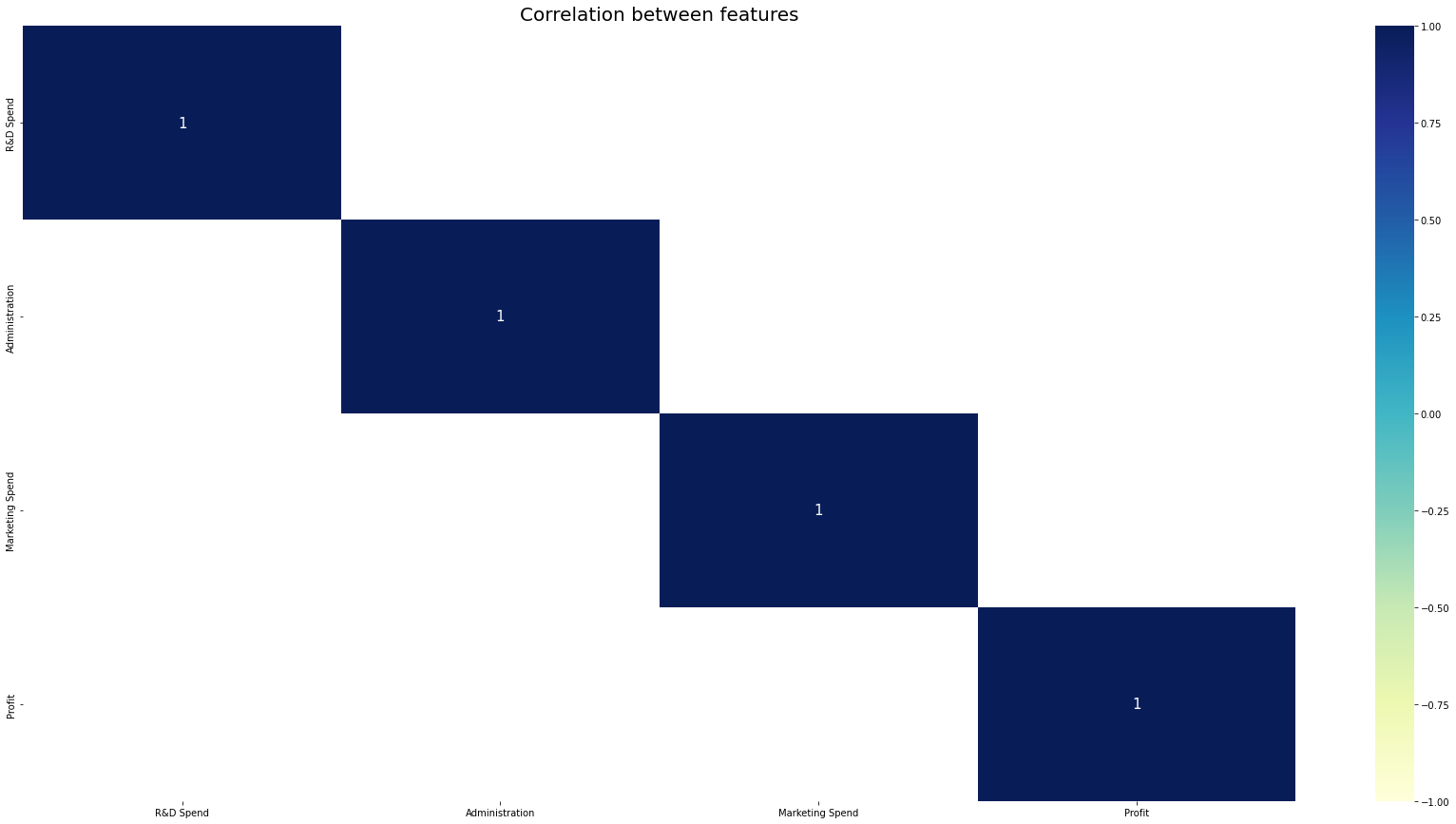


In [36]:

plt.figure(figsize**=**(30, 15))

sns.heatmap(corr[(corr **>=** 0.8) **|** (corr **<= -**0.9)], cmap**=**'YlGnBu', vmax**=**1.0, vmin**=-**1.0, annot**=True**, annot\_kws**=**{"size": 15})

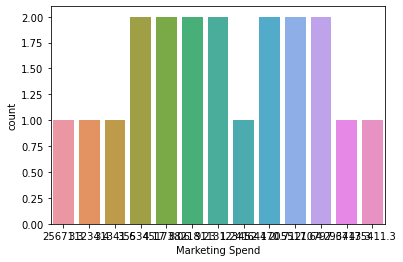
plt.title('Correlation between features', fontsize**=**20) plt.show()



In [41]:

sns.countplot(x**=**"Marketing Spend",data**=**df)

Out[41]: <AxesSubplot:xlabel='Marketing Spend', ylabel='count'>

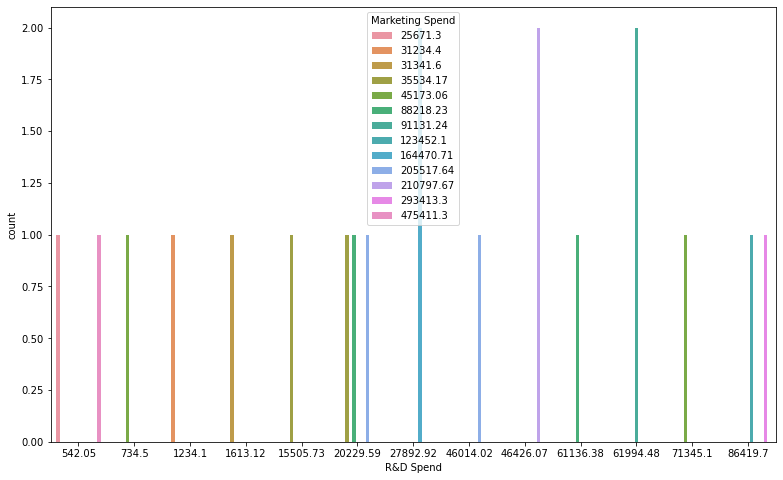


In [48]:

plt.figure(figsize**=**(13,8))

sns.countplot(data**=**df, x**=**'R&D Spend', hue**=**'Marketing Spend')

Out[48]: <AxesSubplot:xlabel='R&D Spend', ylabel='count'>

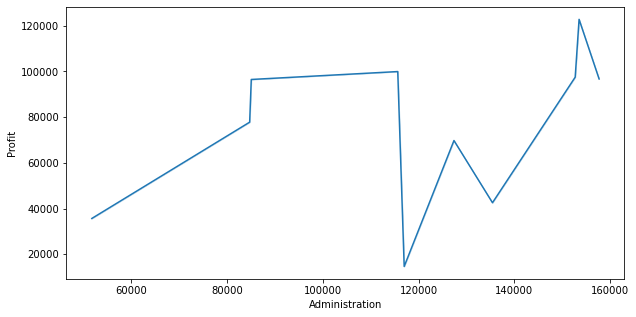


In [42]:

plt.figure(figsize**=**(10,5))

sns.lineplot(data**=**df, x**=**'Administration', y**=**'Profit')

Out[42]: <AxesSubplot:xlabel='Administration', ylabel='Profit'>

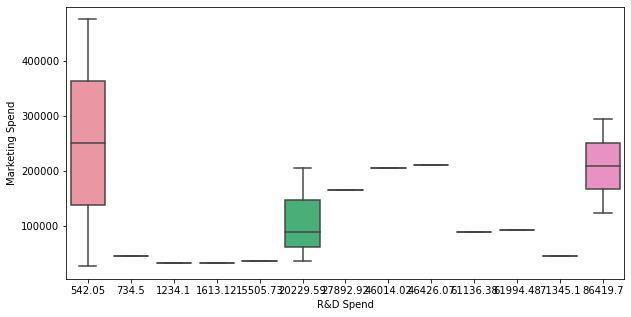


In [46]:

plt.figure(figsize**=**(10,5))

sns.boxplot(data**=**df, x**=**"R&D Spend", y**=**"Marketing Spend")

Out[46]: <AxesSubplot:xlabel='R&D Spend', ylabel='Marketing Spend'>



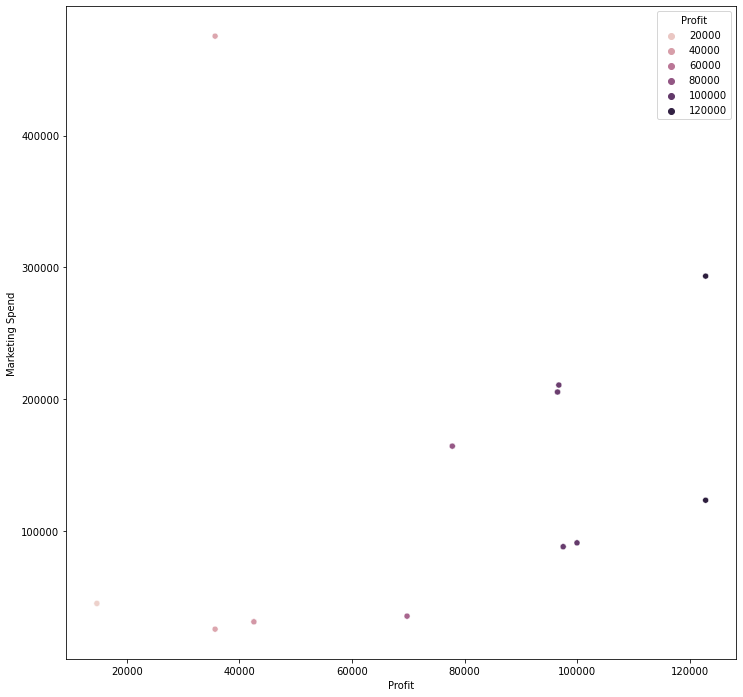
In [47]:

plt.figure(figsize**=**(12,12))

sns.scatterplot(data**=**df, x**=**'Profit', y**=**'Marketing Spend', hue**=**'Profit')

Out[47]: <AxesSubplot:xlabel='Profit', ylabel='Marketing Spend'>



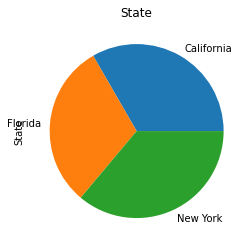




In [4]:

df.groupby('State').State.count().plot(kind**=**'pie') plt.title('State')

plt.show()



In [ ]: