In [1]:
```
import pandas as pd
```

In [2]:
```
df = pd.read_csv("Ecommerce Customers-1.csv")
```

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

Out[3]:

	Email	Address	Avatar	Avg. Session Length	Time on App	Time on Website	Length of Membership	Ye Amo Sp
0	mstephenson@fernandez.com	835 Frank Tunnel\nWrightmouth, MI 82180-9605	Violet	34.497268	12.655651	39.577668	4.082621	587.951
1	hduke@hotmail.com	4547 Archer Common\nDiazchester, CA 06566-8576	DarkGreen	31.926272	11.109461	37.268959	2.664034	392.204
2	pallen@yahoo.com	24645 Valerie Unions Suite 582\nCobbborough, D...	Bisque	33.000915	11.330278	37.110597	4.104543	487.547
3	riverarebecca@gmail.com	1414 David Throughway\nPort Jason, OH 22071-1220	SaddleBrown	34.305557	13.717514	36.721283	3.120179	581.852
4	mstephens@davidson-herman.com	14023 Rodriguez Passage\nPort Jacobville, PR 3...	MediumAquaMarine	33.330673	12.795189	37.536653	4.446308	599.406

In [4]:
```
df.describe()
```

Out[4]:

	Avg. Session Length	Time on App	Time on Website	Length of Membership	Yearly Amount Spent
count	500.000000	500.000000	500.000000	500.000000	500.000000
mean	33.053194	12.052488	37.060445	3.533462	499.314038
std	0.992563	0.994216	1.010489	0.999278	79.314782
min	29.532429	8.508152	33.913847	0.269901	256.670582
25%	32.341822	11.388153	36.349257	2.930450	445.038277
50%	33.082008	11.983231	37.069367	3.533975	498.887875
75%	33.711985	12.753850	37.716432	4.126502	549.313828
max	36.139662	15.126994	40.005182	6.922689	765.518462

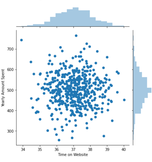
In [5]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 500 entries, 0 to 499
Data columns (total 8 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   Email                 500 non-null    object 
 1   Address               500 non-null    object 
 2   Avatar                500 non-null    object 
 3   Avg. Session Length   500 non-null    float64
 4   Time on App           500 non-null    float64
 5   Time on Website       500 non-null    float64
 6   Length of Membership  500 non-null    float64
 7   Yearly Amount Spent   500 non-null    float64
dtypes: float64(5), object(3)
memory usage: 31.4+ KB
```

In [6]:
```
import seaborn as sns
```

In [7]:
```
# More time on site, more money spent.
sns.jointplot(x='Time on Website',y='Yearly Amount Spent',data=df)
```

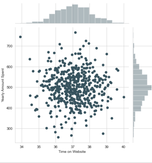
Out[7]:
<seaborn.axisgrid.JointGrid at 0x1b6051f9640>



In [8]:
```
sns.set_palette("GnBu_d")
sns.set_style('whitegrid')
```

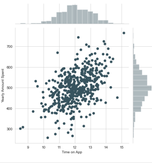
In [9]:
```
# More time on site, more money spent.
sns.jointplot(x='Time on Website',y='Yearly Amount Spent',data=df)
```

Out[9]:
<seaborn.axisgrid.JointGrid at 0x1b603a2b8b0>



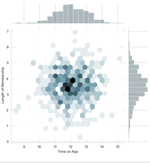
In [10]:
```
sns.jointplot(x='Time on App',y='Yearly Amount Spent',data=df)
```

Out[10]:
<seaborn.axisgrid.JointGrid at 0x1b6804ef2b0>



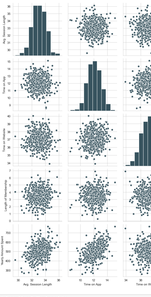
In [11]:
```
sns.jointplot(x='Time on App',y='Length of Membership',kind='hex',data=df)
```

Out[11]:
<seaborn.axisgrid.JointGrid at 0x1b6806d2da30>



In [12]:
```
sns.pairplot(df)
```

Out[12]:
<seaborn.axisgrid.PairGrid at 0x1b608b6ea5e0>



In [13]:
```
# Length Of Membership
```

In [14]:
```
sns.lmplot(x='Length of Membership',y='Yearly Amount Spent',data=df)
```

Out[14]:
<seaborn.axisgrid.FacetGrid at 0x1b608f49490>

In [15]:
```
X = df.iloc[:,3:8]
```

In [16]:
```
y = df['Yearly Amount Spent']
```

In [17]:
```
from sklearn.model_selection import train_test_split
```

In [18]:
```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.4, random_state=101)
```

In [19]:
```
from sklearn.linear_model import LinearRegression
```

In [20]:
```
lm = LinearRegression()
```

In [21]:
```
lm.fit(X_train,y_train)
```

Out[21]:
```
LinearRegression()
```

In [22]:
```
# The coefficients
print('Coefficients: \n', lm.coef_)
```

```
Coefficients:
 [ 1.77633340e-15  4.99608361e-15 -3.08515149e-15  1.58948375e-14
  1.00000000e+00]
```

In [23]:
```
predictions = lm.predict( X_test)
```

In [24]:
```
from matplotlib import pyplot as plt
```

In [25]:
```
import numpy as np
predictions = lm.predict( X_test)
np.savetxt("prediction_LinearRegression_Ecommerce.csv",predictions)
lm.score(X_test,y_test)
```

Out[25]:
```
1.0
```

In [26]:
```
plt.scatter(y_test,predictions)
plt.xlabel('Y Test,')
plt.ylabel('Predicted Y')
```

Out[26]:
```
Text(0, 0.5, 'Predicted Y')
```

from sklearn import metrics import numpy as np

print('MAE:', metrics.mean_absolute_errorr(y_test, predictions)) print('MSE:', metrics.mean_squared_errorr(y_test, predictions)) print('RMSE:', np.sqrt(metrics.mean_squared_errorr(y_test, predictions)))

In [27]:
```
sns.distplot((y_test-predictions),bins=50);
```

```
C:\ProgramData\Anaconda3\lib\site-packages\seaborn\distributions.py:369: UserWarning: Default bandwidth for data is 0; skipping density estimation.
  warnings.warn(msg, UserWarning)
```

In [28]:
```
coeffecients = pd.DataFrame(lm.coef_,X.columns)
coeffecients.columns = ['Coeffecient']
coeffecients
```

Out[28]:

	Coeffecient
Avg. Session Length	1.776333e-15
Time on App	4.996084e-15
Time on Website	-3.085151e-15
Length of Membership	1.589484e-14
Yearly Amount Spent	1.000000e+00

In [29]:
```
lm.score(X_test,y_test)
```

Out[29]:
```
0.9855861240024658
```

In [ ]:
```

```