In [1]:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

In [2]:

```
customers = pd.read_csv('Ecommerce Customers')
```

In [3]:

```
customers.head()
```

Out[3]:

	Email	Address	Avatar	Avg. Session Length	Time on App	Time on Website	Length of Membership	Yearly Amount Spent
0	mstephenson@fernandez.com	835 Frank Tunnel\nWrightmouth, MI 82180-9605	Violet	34.497268	12.655651	39.577668	4.082621	587.951054
1	hduke@hotmail.com	4547 Archer Common\nDiazchester, CA 06566-8576	DarkGreen	31.926272	11.109461	37.268959	2.664034	392.204933
2	pallen@yahoo.com	24645 Valerie Unions Suite 582\nCobbborough, D...	Bisque	33.000915	11.330278	37.110597	4.104543	487.547505
3	riverarebecca@gmail.com	1414 David Throughway\nPort Jason, OH 22070-1220	SaddleBrown	34.305557	13.717514	36.721283	3.120179	581.852344
4	mstephens@davidson-herman.com	14023 Rodriguez Passage\nPort Jacobville, PR 3...	MediumAquaMarine	33.330673	12.795189	37.536653	4.446308	599.406092

In [5]:

```
customers.shape
```

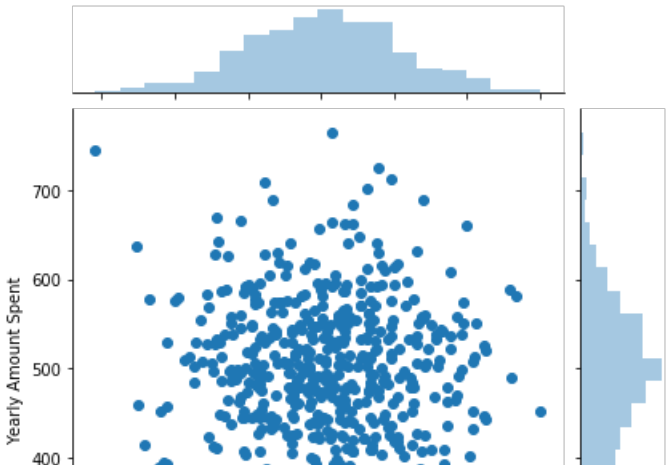
Out[5]:

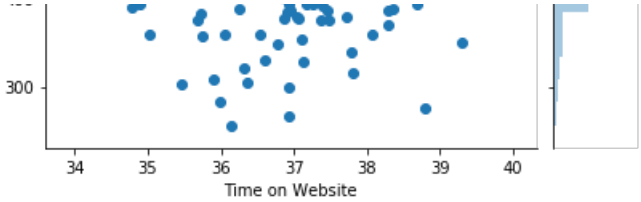
```
(500, 8)
```

In [6]:

```
sns.jointplot(data=customers, x='Time on Website', y='Yearly Amount Spent');
```

```
/Users/sudeng/anaconda3/lib/python3.7/site-packages/scipy/stats/stats.py:1713: FutureWarning: Using a non-tuple sequence for multidimensional indexing is deprecated; use `arr[tuple(seq)]` instead of `arr[seq]`. In the future this will be interpreted as an array index, `arr[np.array(seq)]`, which will result either in an error or a different result.
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```

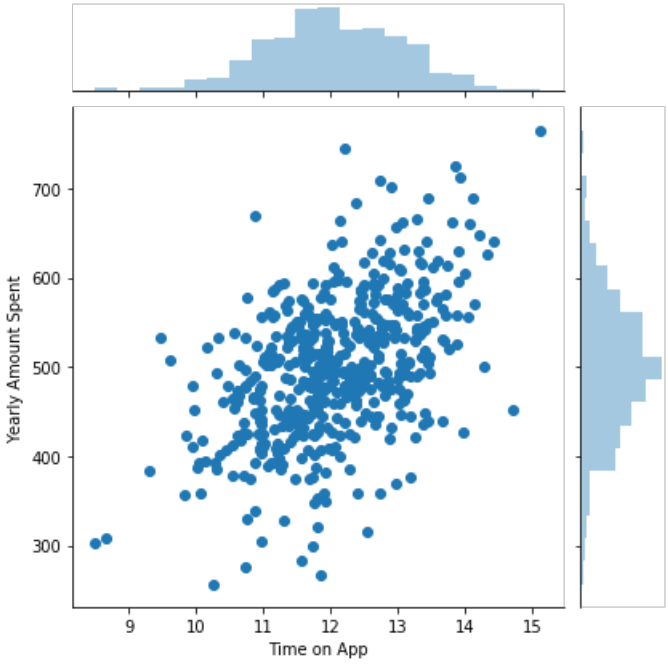




In [7]:

```
sns.jointplot(data=customers, x='Time on App', y='Yearly Amount Spent');
```

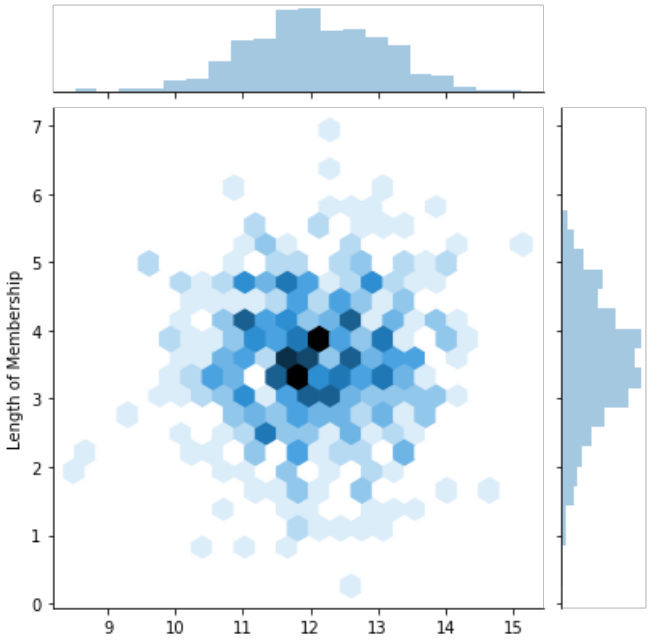
```
/Users/sudeng/anaconda3/lib/python3.7/site-packages/scipy/stats/stats.py:1713: FutureWarning: Using a non-tuple sequence for multidimensional indexing is deprecated; use `arr[tuple(seq)]` instead of `arr[seq]`. In the future this will be interpreted as an array index, `arr[np.array(seq)]`, which will result either in an error or a different result.
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```



In [8]:

```
sns.jointplot(data=customers, x='Time on App', y='Length of Membership', kind='hex');
```

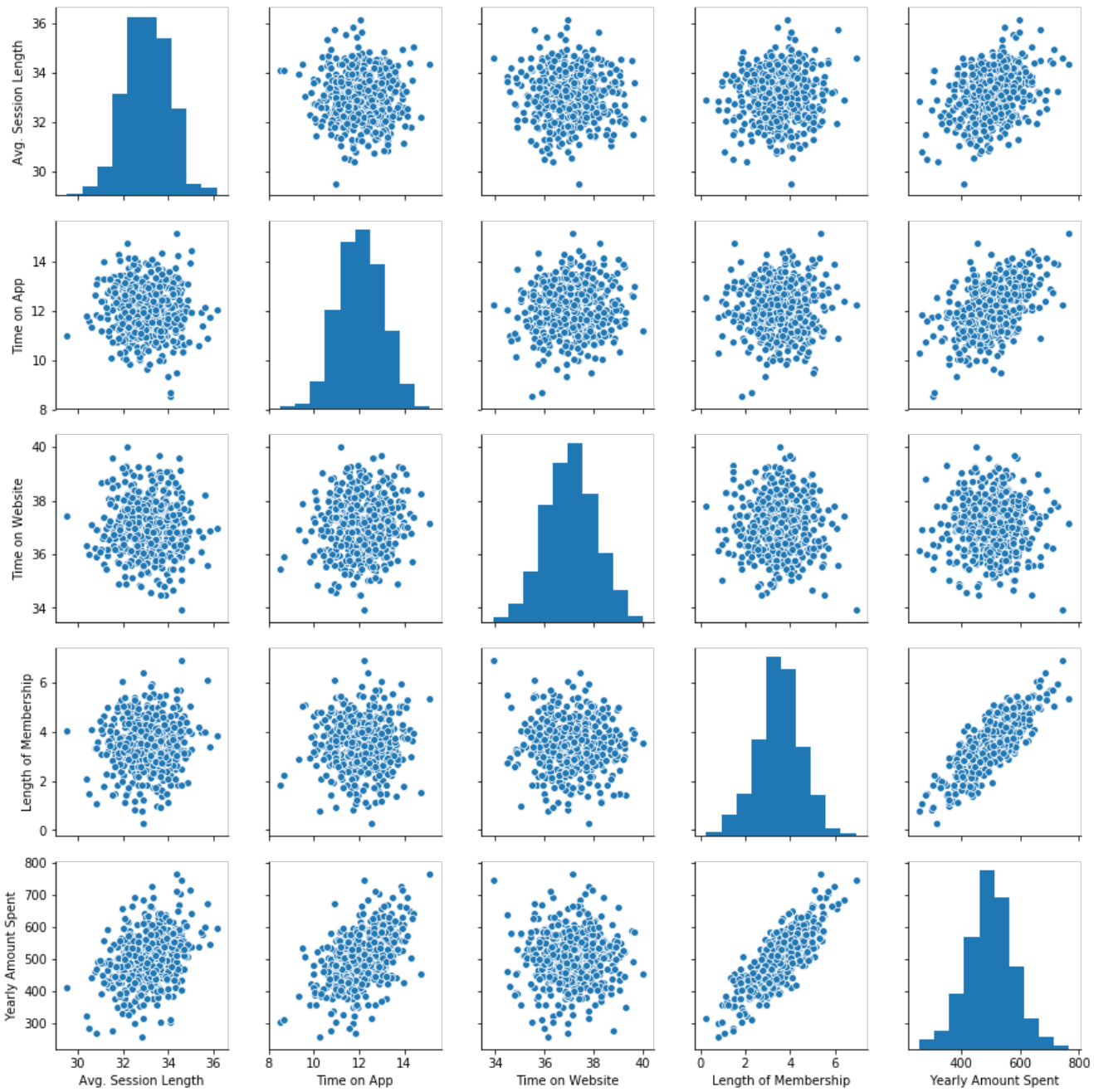
```
/Users/sudeng/anaconda3/lib/python3.7/site-packages/scipy/stats/stats.py:1713: FutureWarning: Using a non-tuple sequence for multidimensional indexing is deprecated; use `arr[tuple(seq)]` instead of `arr[seq]`. In the future this will be interpreted as an array index, `arr[np.array(seq)]`, which will result either in an error or a different result.
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```



Time on App

In [9]:

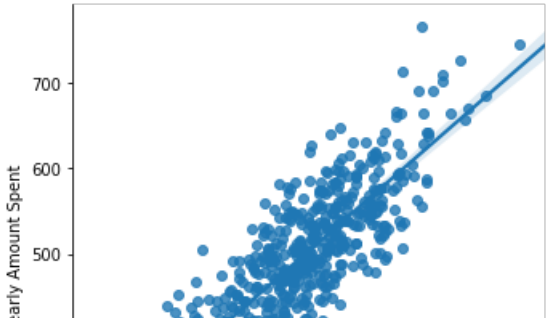
```
sns.pairplot(customers);
```

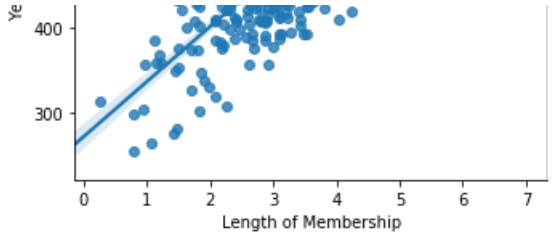


In [11]:

```
sns.lmplot(data=customers, x='Length of Membership', y='Yearly Amount Spent');
```

```
/Users/sudeng/anaconda3/lib/python3.7/site-packages/scipy/stats/stats.py:1713: FutureWarning: Using a non-tuple sequence for multidimensional indexing is deprecated; use `arr[tuple(seq)]` instead of `arr[seq]`. In the future this will be interpreted as an array index, `arr[np.array(seq)]`, which will result either in an error or a different result.
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```





In [12]:

```
customers.columns
```

Out[12]:

```
Index(['Email', 'Address', 'Avatar', 'Avg. Session Length', 'Time on App',
       'Time on Website', 'Length of Membership', 'Yearly Amount Spent'],
      dtype='object')
```

In [13]:

```
x = customers[['Avg. Session Length', 'Time on App',
       'Time on Website', 'Length of Membership']]
y = customers['Yearly Amount Spent']
```

In [14]:

```
from sklearn.cross_validation import train_test_split
```

In [18]:

```
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.3, 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(copy_X=True, fit_intercept=True, n_jobs=1, normalize=False)
```

In [22]:

```
lm.coef_
```

Out[22]:

```
array([25.98154972, 38.59015875,  0.19040528, 61.27909654])
```

In [23]:

```
lm.intercept_
```

Out[23]:

```
-1047.9327822502382
```

In [24]:

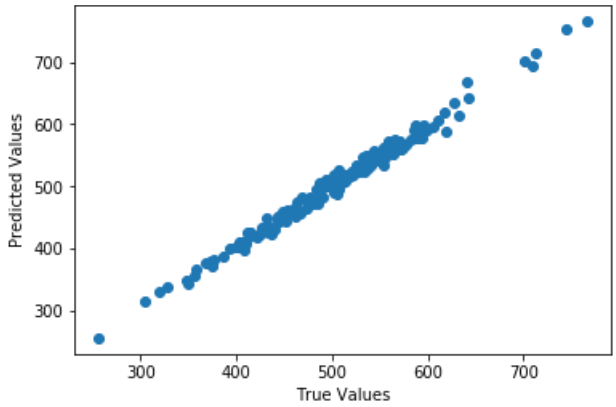
```
predictions = lm.predict(x_test)
```

In [25]:

```
plt.scatter(y_test, predictions);
plt.xlabel('True Values')
plt.ylabel('Predicted Values')
```

Out[25]:

```
Text(0,0.5,'Predicted Values')
```



In [26]:

```
from sklearn import metrics
```

In [27]:

```
metrics.mean_absolute_error(y_test, predictions) #mae
```

Out[27]:

```
7.228148653430835
```

In [28]:

```
metrics.mean_squared_error(y_test, predictions) #mse
```

Out[28]:

```
79.81305165097467
```

In [29]:

```
np.sqrt(metrics.mean_squared_error(y_test, predictions)) #rmse
```

Out[29]:

```
8.933815066978646
```

In [30]:

```
(y_test - predictions).describe()
```

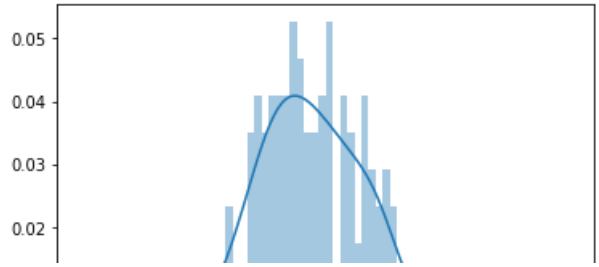
Out[30]:

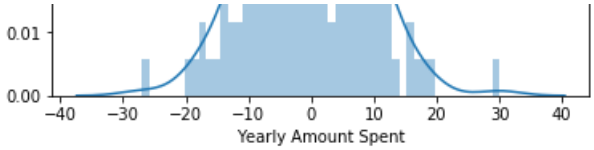
```
count    150.000000
mean      -0.725432
std        8.934144
min      -26.955731
25%       -6.971724
50%       -1.424168
75%        5.398126
max       29.998572
Name: Yearly Amount Spent, dtype: float64
```

In [31]:

```
sns.distplot((y_test - predictions), bins=50);
```

```
/Users/sudeng/anaconda3/lib/python3.7/site-packages/scipy/stats/stats.py:1713: FutureWarning: Using a non-tuple sequence for multidimensional indexing is deprecated; use `arr[tuple(seq)]` instead of `arr[seq]`. In the future this will be interpreted as an array index, `arr[np.array(seq)]`, which will result either in an error or a different result.
  return np.add.reduce(sorted[indexer] * weights, axis=axis) / sumval
```





In [32]:

```
cdf = pd.DataFrame(lm.coef_, x.columns, columns=['Coeff'])
cdf
```

Out[32]:

	Coeff
Avg. Session Length	25.981550
Time on App	38.590159
Time on Website	0.190405
Length of Membership	61.279097

In [ ]:

```

```