In [2]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error,r2_score
from sklearn.linear_model import LinearRegression
```

In [3]:
```
df=pd.read_csv("C://Users//user//Downloads//Ecommerce Customers")
df
```

Out[3]:

	Email	Address	Avatar	Avg. Session Length	Time on App	
0	mstephenson@fernandez.com	835 Frank Tunnel\nWrightmouth, MI 82180-9605	Violet	34.497268	12.655651	39
1	hduke@hotmail.com	4547 Archer Common\nDiazchester, CA 06566-8576	DarkGreen	31.926272	11.109461	37
2	pallen@yahoo.com	24645 Valerie Unions Suite 582\nCobbborough, D...	Bisque	33.000915	11.330278	37
3	riverarebecca@gmail.com	1414 David Throughway\nPort Jason, OH 22070-1220	SaddleBrown	34.305557	13.717514	36
4	mstephens@davidson-herman.com	14023 Rodriguez Passage\nPort Jacobville, PR 3...	MediumAquaMarine	33.330673	12.795189	37
...	...	...	...	...	...	
495	lewisjessica@craig-evans.com	4483 Jones Motorway Suite 872\nLake Jamiefurt,...	Tan	33.237660	13.566160	36
496	katrina56@gmail.com	172 Owen Divide Suite 497\nWest Richard, CA 19320	PaleVioletRed	34.702529	11.695736	37
497	dale88@hotmail.com	0787 Andrews Ranch Apt. 633\nSouth Chadburgh, ...	Cornsilk	32.646777	11.499409	38
498	cwilson@hotmail.com	680 Jennifer Lodge Apt. 808\nBrendachester, TX...	Teal	33.322501	12.391423	36
499	hannahwilson@davidson.com	49791 Rachel Heights Apt. 898\nEast Drewboroug...	DarkMagenta	33.715981	12.418808	35

500 rows × 8 columns

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

Out[4]:

	Email	Address	Avatar	Avg. Session Length	Time on App	Tir We
0	mstephenson@fernandez.com	835 Frank Tunnel\nWrightmouth, MI 82180-9605	Violet	34.497268	12.655651	39.5
1	hduke@hotmail.com	4547 Archer Common\nDiazchester, CA 06566-8576	DarkGreen	31.926272	11.109461	37.2
2	pallen@yahoo.com	24645 Valerie Unions Suite 582\nCobbborough, D...	Bisque	33.000915	11.330278	37.1
3	riverarebecca@gmail.com	1414 David Throughway\nPort Jason, OH 22070-1220	SaddleBrown	34.305557	13.717514	36.7
4	mstephens@davidson-herman.com	14023 Rodriguez Passage\nPort Jacobville, PR 3...	MediumAquaMarine	33.330673	12.795189	37.5

In [5]:
```
df.shape
```

Out[5]:
```
(500, 8)
```

In [6]:
```
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 [7]:
```
df.describe()
```

Out[7]:

	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 [8]:
```
df.isnull().sum()
```

Out[8]:
```
Email                 0
Address               0
Avatar                0
Avg. Session Length   0
Time on App           0
Time on Website       0
Length of Membership  0
Yearly Amount Spent   0
dtype: int64
```

In [10]:
```
df.columns=df.columns.str.strip().str.lower()
```

In [9]:
```
df.duplicated().sum()
```

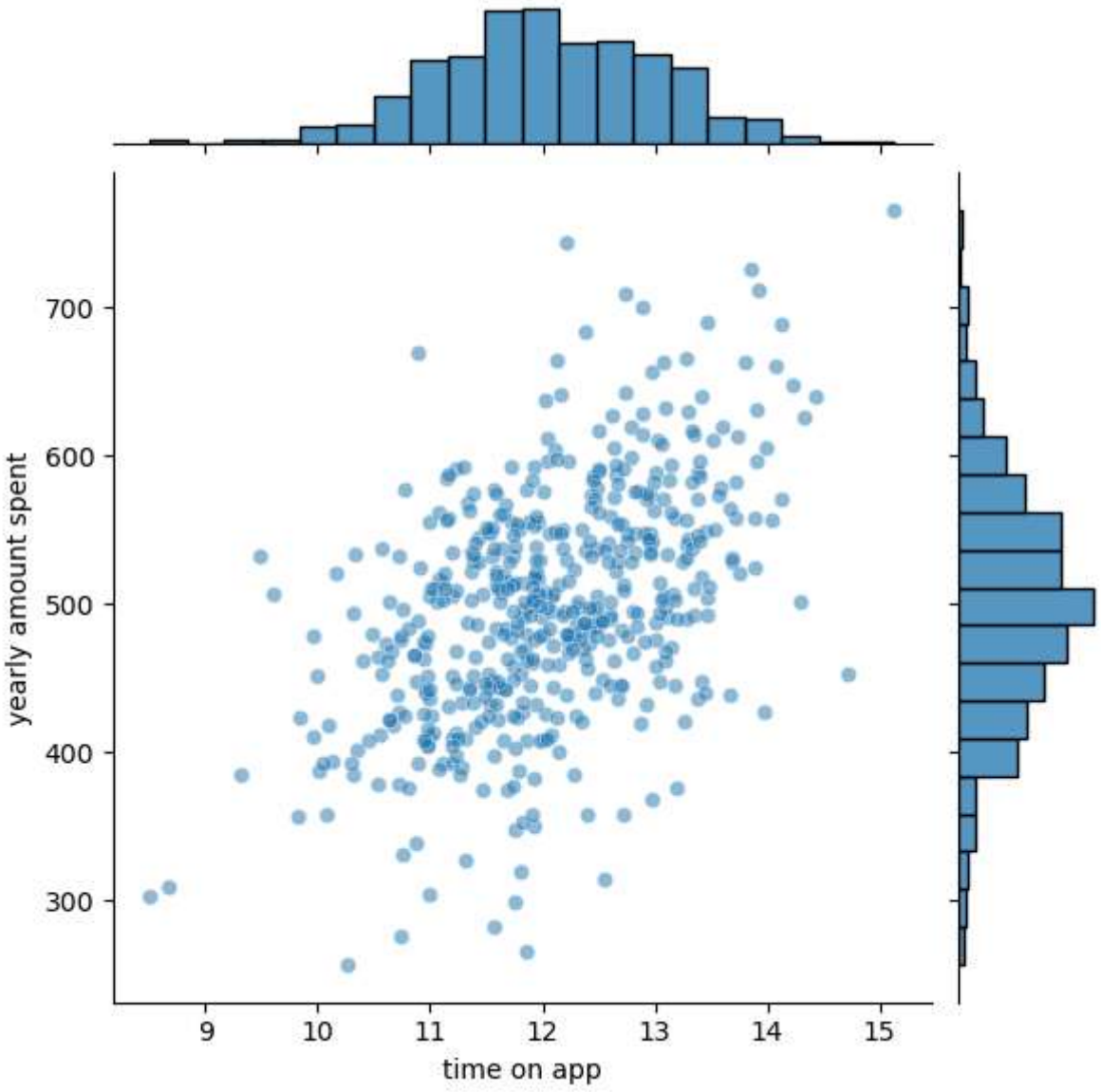
Out[9]:
```
0
```

In [12]:
```
df.columns
```

Out[12]:
```
Index(['email', 'address', 'avatar', 'avg. session length', 'time on app',
       'time on website', 'length of membership', 'yearly amount spent'],
      dtype='object')
```

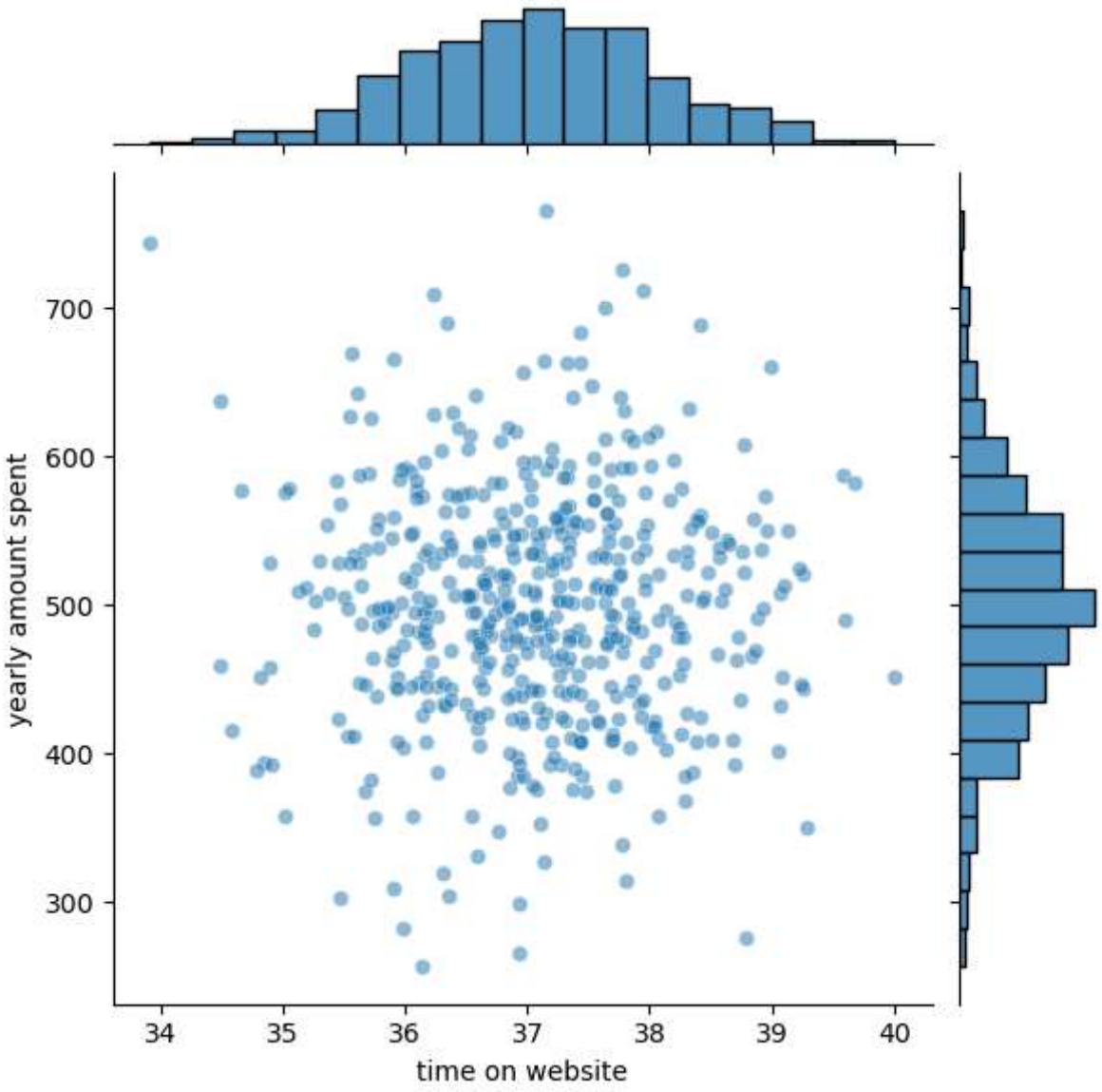
In [14]:
```
sns.jointplot(x="time on app",y="yearly amount spent",data=df,alpha=0.5)
```

Out[14]:
```
<seaborn.axisgrid.JointGrid at 0x1adff3a25d0>
```



In [15]:
```
sns.jointplot(x="time on website",y="yearly amount spent",data=df,alpha=0.5)
```

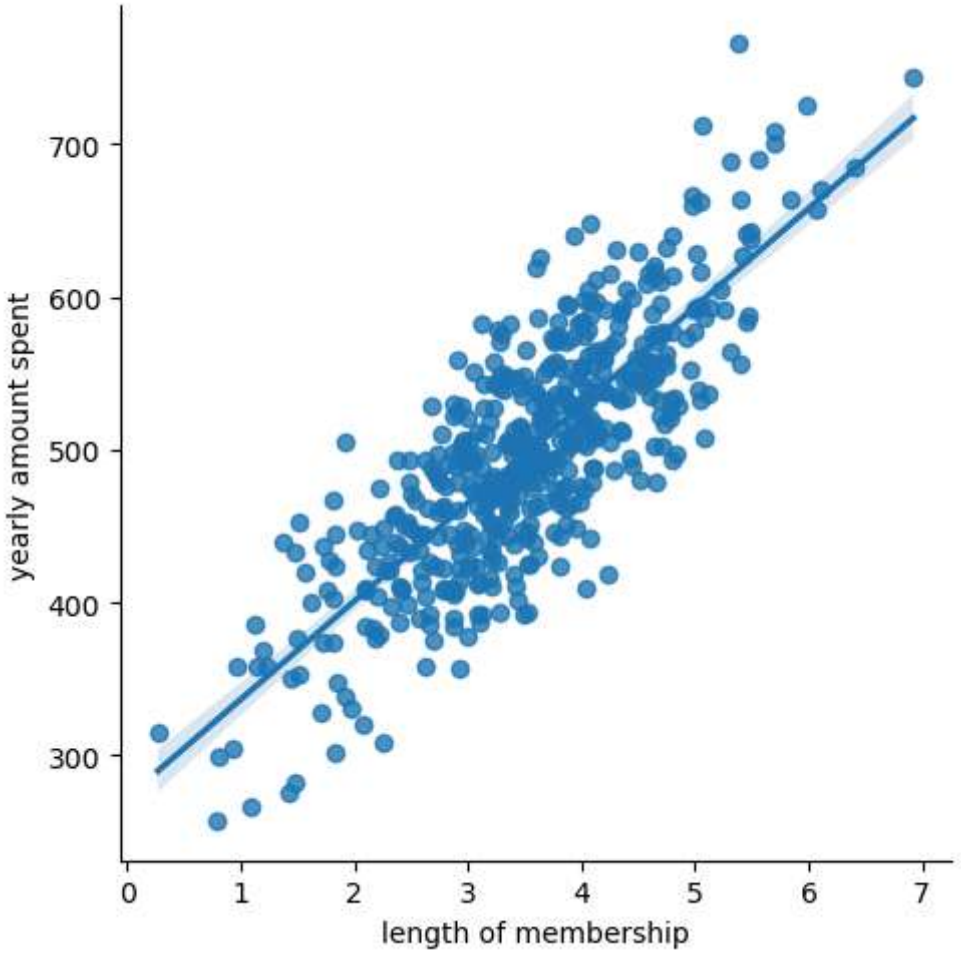
Out[15]:
```
<seaborn.axisgrid.JointGrid at 0x1ad8178fe50>
```



In [18]:
```
sns.lmplot(x='length of membership',y="yearly amount spent",data=df)
```

```
C:\Users\user\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning:
The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```

Out[18]:
```
<seaborn.axisgrid.FacetGrid at 0x1adff5bd510>
```



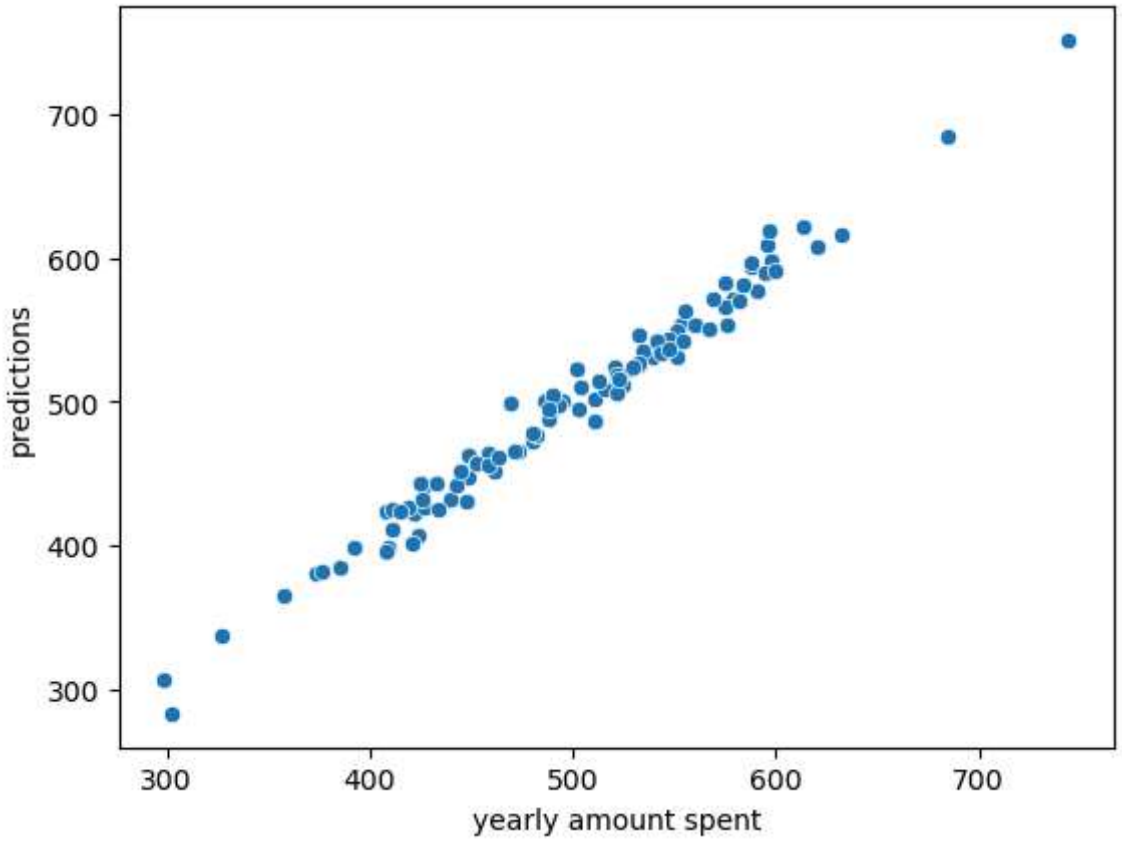
In [25]:
```
y=df['yearly amount spent']
x=df[["avg. session length", 'time on app',
       'time on website', 'length of membership']]
```

In [26]:
```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=40)
```

In [27]:
```
lr=LinearRegression()
lr.fit(x_train,y_train)
y_pred=lr.predict(x_test)
```

In [30]:
```
sns.scatterplot(x=y_test,y=y_pred)
plt.ylabel("predictions")
```

Out[30]:
```
Text(0, 0.5, 'predictions')
```



In [33]:
```
print("Mean Square Error",mean_squared_error(y_test,y_pred))
print("r2_score",r2_score(y_test,y_pred))
```

```
Mean Square Error 111.06329597249865
r2_score 0.9819676748095943
```

In [37]:
```
my_test_data=[[38.7896,56.5432,23.4567,43.5674]]
prediction=lr.predict(my_test_data)
print("Predicted Health Score:", prediction[0])
```

```
Predicted Health Score: 4833.883496251193

C:\Users\user\anaconda3\Lib\site-packages\sklearn\utils\validation.py:2739: Use
rWarning: X does not have valid feature names, but LinearRegression was fitted
with feature names
  warnings.warn(
```

In [ ]:
```

```