

UDEMY_COURSE_EDA_PROJECT_VIVEK_CHAUHAN

```
In [1]: # import necessary libraries to work with dataset
```

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

```
In [2]: # Load the dataset
```

```
data = pd.read_csv("C:/Users/VIVEK CHAUHAN/Desktop/FUTURE COMPUTER VISION INSTITUTE/EDA_PROJECT_HOME/udemy_courses.csv")
data
```

Out[2]:

	id	title	url	paid	price	suscribers	reviews	lectures	level	duration	day	month
0	1070968	Ultimate Investment Banking Course	https://www.udemy.com/ultimate-investment-bank...	True	200	2147	23	51	All Levels	1.5	18	
1	1113822	Complete GST Course & Certification - Grow You...	https://www.udemy.com/goods-and-services-tax/	True	75	2792	923	274	All Levels	39.0	9	
2	1006314	Financial Modeling for Business Analysts and C...	https://www.udemy.com/financial-modeling-for-b...	True	45	2174	74	51	Intermediate Level	2.5	19	1
3	1210588	Beginner to Pro - Financial Analysis in Excel ...	https://www.udemy.com/complete-excel-finance-c...	True	95	2451	11	36	All Levels	3.0	30	
4	1011058	How To Maximize Your Profits Trading Options	https://www.udemy.com/how-to-maximize-your-pro...	True	200	1276	45	26	Intermediate Level	2.0	13	1
...
3673	775618	Learn jQuery from Scratch - Master of JavaScri...	https://www.udemy.com/easy-jquery-for-beginner...	True	100	1040	14	21	All Levels	2.0	14	
3674	1088178	How To Design A	https://www.udemy.com/how-to-make-a-wordpress-...	True	25	306	3	42	Beginner Level	3.5	10	

	id	title		url	paid	price	subscribers	reviews	lectures	level	duration	day	month
		WordPress Website With No Codi...											
3675	635248	Learn and Build using Polymer	https://www.udemy.com/learn-and-build-using-polymer/	True	40		513	169	48	All Levels	3.5	30	1
3676	905096	CSS Animations: Create Amazing Effects on Your...	https://www.udemy.com/css-animations-create-amazing-effects-on-your-website/	True	50		300	31	38	All Levels	3.0	11	
3677	297602	Using MODX CMS to Build Websites: A Beginner's...	https://www.udemy.com/using-modx-cms-to-build-websites-a-beginners-guide/	True	45		901	36	20	Beginner Level	2.0	28	

3678 rows × 17 columns

In [3]: *# top 10 data of the dataset*`data.head(10)`

Out[3]:

	id	title	url	paid	price	suscribers	reviews	lectures	level	duration	day	month
0	1070968	Ultimate Investment Banking Course	https://www.udemy.com/ultimate-investment-bank...	True	200	2147	23	51	All Levels	1.5	18	1
1	1113822	Complete GST Course & Certification - Grow You...	https://www.udemy.com/goods-and-services-tax/	True	75	2792	923	274	All Levels	39.0	9	3
2	1006314	Financial Modeling for Business Analysts and C...	https://www.udemy.com/financial-modeling-for-b...	True	45	2174	74	51	Intermediate Level	2.5	19	12
3	1210588	Beginner to Pro - Financial Analysis in Excel ...	https://www.udemy.com/complete-excel-finance-c...	True	95	2451	11	36	All Levels	3.0	30	5
4	1011058	How To Maximize Your Profits Trading Options	https://www.udemy.com/how-to-maximize-your-pro...	True	200	1276	45	26	Intermediate Level	2.0	13	12
5	192870	Trading Penny Stocks: A Guide for All Levels I...	https://www.udemy.com/trading-penny-stocks-a-g...	True	150	9221	138	25	All Levels	3.0	2	5
6	739964	Investing And Trading For	https://www.udemy.com/investing-and-trading-fo...	True	65	1540	178	26	Beginner Level	1.0	21	2

	id	title		url	paid	price	suscribers	reviews	lectures	level	duration	day	month
		Beginners: Mastering...											
7	403100	Trading Stock Chart Patterns For Immediate, Ex...	https://www.udemy.com/trading-chart-patterns-f...	True	95	2917	148	23	All Levels	2.5	30	1	
8	476268	Options Trading 3 : Advanced Stock Profit and ...	https://www.udemy.com/day-trading-stock-option...	True	195	5172	34	38	Expert Level	2.5	28	5	
9	1167710	The Only Investment Strategy You Need For Your...	https://www.udemy.com/the-only-investment-stra...	True	200	827	14	15	All Levels	1.0	18	4	

In [4]: # Last 10 data of the dataset

```
data.tail(10)
```

Out[4]:

	id	title	url	paid	price	subscribers	reviews	lectures	level	d
3668	270976	A how to guide in HTML	https://www.udemy.com/a-how-to-guide-in-html/	False	0	7318	205	8	Beginner Level	C
3669	679992	Building Better APIs with GraphQL	https://www.udemy.com/building-better-apis-with-graphql/	True	50	555	89	16	All Levels	2
3670	330900	Learn Grunt with Examples: Automate Your Front...	https://www.udemy.com/learn-grunt-automate-your-front.../	True	20	496	113	17	All Levels	1
3671	667122	Build A Stock Downloader With Visual Studio 20...	https://www.udemy.com/csharp-yahoo-stock-downloader/	True	20	436	36	22	Intermediate Level	1
3672	865438	jQuery UI in Action: Build 5 jQuery UI Projects	https://www.udemy.com/jquery-ui-practical-build.../	True	150	382	28	140	All Levels	15
3673	775618	Learn jQuery from Scratch - Master of JavaScript	https://www.udemy.com/easy-jquery-for-beginner.../	True	100	1040	14	21	All Levels	2
3674	1088178	How To Design A WordPress Website With No Codi...	https://www.udemy.com/how-to-make-a-wordpress-.../	True	25	306	3	42	Beginner Level	3

	id	title		url	paid	price	subscribers	reviews	lectures	level	duration
3675	635248	Learn and Build using Polymer		https://www.udemy.com/learn-and-build-using-po...	True	40	513	169	48	All Levels	3
3676	905096	CSS Animations: Create Amazing Effects on Your...		https://www.udemy.com/css-animations-create-am...	True	50	300	31	38	All Levels	3
3677	297602	Using MODX CMS to Build Websites: A Beginner's...		https://www.udemy.com/using-modx-cms-to-build-...	True	45	901	36	20	Beginner Level	2

In [5]: `# checking the column datatypes`

```
data.dtypes
```

```
Out[5]: id          int64
         title       object
         url         object
         paid        bool
         price       int64
         subscribers int64
         reviews     int64
         lectures    int64
         level        object
         duration    float64
         day          int64
         month        int64
         year          int64
         hours        int64
         minutes      int64
         seconds      int64
         subject      object
         dtype: object
```

```
In [6]: # information of our dataset
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3678 entries, 0 to 3677
Data columns (total 17 columns):
 #   Column      Non-Null Count  Dtype  
--- 
 0   id          3678 non-null    int64  
 1   title        3678 non-null    object  
 2   url          3678 non-null    object  
 3   paid          3678 non-null    bool   
 4   price         3678 non-null    int64  
 5   suscribers   3678 non-null    int64  
 6   reviews       3678 non-null    int64  
 7   lectures      3678 non-null    int64  
 8   level         3678 non-null    object  
 9   duration      3678 non-null    float64
 10  day           3678 non-null    int64  
 11  month          3678 non-null    int64  
 12  year           3678 non-null    int64  
 13  hours          3678 non-null    int64  
 14  minutes         3678 non-null    int64  
 15  seconds         3678 non-null    int64  
 16  subject        3678 non-null    object  
dtypes: bool(1), float64(1), int64(11), object(4)
memory usage: 463.5+ KB
```

In [7]: `# print the no of rows & columns`

```
print("Number of Rows",data.shape[0])
print("Number of Rows",data.shape[1])
```

Number of Rows 3678

Number of Rows 17

In [8]: `# check the null is present in the dataset`

```
data.isnull()
```

Out[8]:

	id	title	url	paid	price	suscribers	reviews	lectures	level	duration	day	month	year	hours	minutes	seconds	subject
0	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
...
3673	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
3674	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
3675	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
3676	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False
3677	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False	False

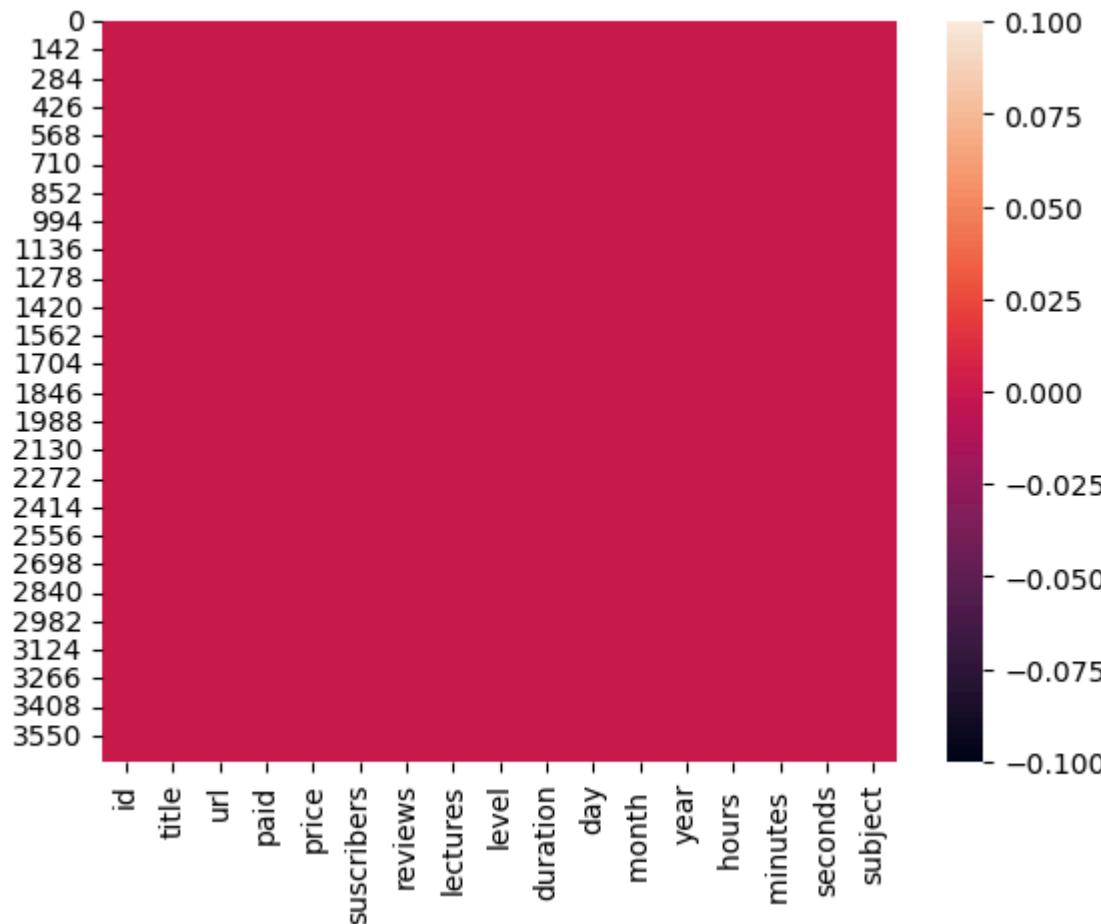
3678 rows × 17 columns



In [9]: # check the null is present in the dataset

sns.heatmap(data.isnull())

Out[9]: <Axes: >



```
In [10]: # statistics of the dataset  
data.describe()
```

Out[10]:

	id	price	suscribers	reviews	lectures	duration	day	month	year	helpfulness_votes
count	3.678000e+03	3678.000000	3678.000000	3678.000000	3678.000000	3678.000000	3678.000000	3678.000000	3678.000000	3678.000000
mean	6.759720e+05	66.049483	3197.150625	156.259108	40.108755	4.094517	15.831158	6.164220	2015.431213	15.418
std	3.432732e+05	61.005755	9504.117010	935.452044	50.383346	6.053840	8.781086	3.380535	1.185317	6.820
min	8.324000e+03	0.000000	0.000000	0.000000	0.000000	0.000000	1.000000	1.000000	2011.000000	0.000
25%	4.076925e+05	20.000000	111.000000	4.000000	15.000000	1.000000	8.000000	3.000000	2015.000000	14.000
50%	6.879170e+05	45.000000	911.500000	18.000000	25.000000	2.000000	16.000000	6.000000	2016.000000	17.000
75%	9.613555e+05	95.000000	2546.000000	67.000000	45.750000	4.500000	23.000000	9.000000	2016.000000	20.000
max	1.282064e+06	200.000000	268923.000000	27445.000000	779.000000	78.500000	31.000000	12.000000	2017.000000	23.000



In [11]: # overall statistics of the dataset

data.describe(include="all")

Out[11]:

	id	title	url	paid	price	suscribers	reviews	lectures	level	durat
count	3.678000e+03	3678		3678	3678	3678.000000	3678.000000	3678.000000	3678	3678.000000
unique	Nan	3663		3672	2	Nan	Nan	Nan	4	Nan
top	Nan	Acoustic Blues Guitar Lessons	https://www.udemy.com/cfa-level-2-quantitative.../	True	Nan	Nan	Nan	Nan	All Levels	Nan
freq	Nan	3		2	3368	Nan	Nan	Nan	1929	Nan
mean	6.759720e+05	Nan		Nan	NaN	66.049483	3197.150625	156.259108	40.108755	NaN
std	3.432732e+05	Nan		Nan	NaN	61.005755	9504.117010	935.452044	50.383346	NaN
min	8.324000e+03	Nan		Nan	NaN	0.000000	0.000000	0.000000	0.000000	NaN
25%	4.076925e+05	Nan		Nan	NaN	20.000000	111.000000	4.000000	15.000000	NaN
50%	6.879170e+05	Nan		Nan	NaN	45.000000	911.500000	18.000000	25.000000	NaN
75%	9.613555e+05	Nan		Nan	NaN	95.000000	2546.000000	67.000000	45.750000	NaN
max	1.282064e+06	Nan		Nan	NaN	200.000000	268923.000000	27445.000000	779.000000	NaN



In [12]: # print all the columns-names of the database

data.columns

Out[12]: Index(['id', 'title', 'url', 'paid', 'price', 'suscribers', 'reviews', 'lectures', 'level', 'duration', 'day', 'month ', 'year', 'hours ', 'minutes ', 'seconds', 'subject'], dtype='object')

In [13]: # check the duplicate data is present in the data

data = data.drop_duplicates()

```
In [14]: # check the duplicate data is present in the data  
  
dup = data.duplicated().any()  
print(dup)
```

False

```
In [15]: # highest price of the course subject  
  
a = data.price.max()  
print("highest price of the course is:",a)  
data["subject"][data["price"]==a]
```

highest price of the course is: 200

```
Out[15]: 0    Business Finance  
4    Business Finance  
9    Business Finance  
10   Business Finance  
11   Business Finance  
     ...  
3592  Web Development  
3620  Web Development  
3642  Web Development  
3652  Web Development  
3654  Web Development  
Name: subject, Length: 295, dtype: object
```

```
In [16]: # Lowest price of the course subject  
  
a = data.price.min()  
print("lowest price of the course is:",a)  
data["subject"][data["price"]==a]
```

lowest price of the course is: 0

```
Out[16]: 95      Business Finance
          103     Business Finance
          106     Business Finance
          108     Business Finance
          112     Business Finance
          ...
          3638    Web Development
          3643    Web Development
          3651    Web Development
          3665    Web Development
          3668    Web Development
Name: subject, Length: 310, dtype: object
```

```
In [17]: # highest price of the course levels
```

```
a = data.price.max()
print("highest price of the course level is:",a)
data["level"][data["price"]==a]
```

```
highest price of the course level is: 200
```

```
Out[17]: 0      All Levels
          4      Intermediate Level
          9      All Levels
          10     All Levels
          11     All Levels
          ...
          3592    Intermediate Level
          3620    Beginner Level
          3642    Beginner Level
          3652    All Levels
          3654    All Levels
Name: level, Length: 295, dtype: object
```

```
In [18]: # Lowest price of the course levels
```

```
a = data.price.min()
print("lowest price of the course level is:",a)
data["level"][data["price"]==a]
```

```
lowest price of the course level is: 0
```

```
Out[18]: 95      Beginner Level  
103      Beginner Level  
106      All Levels  
108      All Levels  
112      Beginner Level  
         ...  
3638      All Levels  
3643      All Levels  
3651      All Levels  
3665      All Levels  
3668      Beginner Level  
Name: level, Length: 310, dtype: object
```

```
In [19]: # highest suscribers of the course subject  
  
a = data.suscribers.max()  
print("highest suscribers of the course is:",a)  
data["subject"][data["suscribers"]==a]
```

highest suscribers of the course is: 268923

```
Out[19]: 2827    Web Development  
Name: subject, dtype: object
```

```
In [20]: # Lowest suscribers of the course subject  
  
a = data.suscribers.min()  
print("lowest suscribers of the course is:",a)  
data["subject"][data["suscribers"]==a]
```

lowest suscribers of the course is: 0

```
Out[20]: 453      Business Finance
454      Business Finance
455      Business Finance
456      Business Finance
457      Business Finance
...
2402     Musical Instruments
2403     Musical Instruments
2404     Musical Instruments
2405     Musical Instruments
2418     Musical Instruments
Name: subject, Length: 65, dtype: object
```

```
In [21]: # Lowest reviews of the course subject
```

```
a = data.reviews.min()
print("highest reviews of the course is:",a)
data["subject"][data["reviews"]==a]
```

highest reviews of the course is: 0

```
Out[21]: 213      Business Finance
256      Business Finance
379      Business Finance
405      Business Finance
408      Business Finance
...
2588     Web Development
2645     Web Development
3580     Web Development
3626     Web Development
3648     Web Development
Name: subject, Length: 284, dtype: object
```

```
In [22]: # highest reviews of the course subject
```

```
a = data.reviews.max()
print("highest reviews of the course subject is:",a)
data["subject"][data["reviews"]==a]
```

highest reviews of the course subject is: 27445

```
Out[22]: 3230    Web Development  
Name: subject, dtype: object
```

```
In [23]: # Lowest lectures of the course subject
```

```
a = data.lectures.min()  
print("lowest lectures of the course subject is:",a)  
data["subject"][data["lectures"]==a]
```

```
lowest lectures of the course subject is: 0
```

```
Out[23]: 892    Business Finance  
Name: subject, dtype: object
```

```
In [24]: # highest lectures of the course subject
```

```
a = data.lectures.max()  
print("highest lectures of the course subject is:",a)  
data["subject"][data["lectures"]==a]
```

```
highest lectures of the course subject is: 779
```

```
Out[24]: 2707    Web Development  
Name: subject, dtype: object
```

```
In [25]: # highest price of the course level
```

```
a = data.price.max()  
print("highest price of the course level is:",a)  
data["level"][data["price"]==a]
```

```
highest price of the course level is: 200
```

```
Out[25]: 0           All Levels
        4       Intermediate Level
        9           All Levels
       10          All Levels
       11          All Levels
       ...
      3592   Intermediate Level
      3620     Beginner Level
      3642     Beginner Level
      3652          All Levels
      3654          All Levels
Name: level, Length: 295, dtype: object
```

```
In [26]: # Lowest price of the course level
```

```
a = data.price.min()
print("lowest price of the course level is:",a)
data["level"][data["price"]==a]
```

```
lowest price of the course level is: 0
```

```
Out[26]: 95     Beginner Level
      103    Beginner Level
      106      All Levels
      108      All Levels
      112    Beginner Level
      ...
      3638      All Levels
      3643      All Levels
      3651      All Levels
      3665      All Levels
      3668    Beginner Level
Name: level, Length: 310, dtype: object
```

```
In [27]: # highest suscribers of the course level
```

```
a = data.suscribers.max()
print("highest suscribers of the course level is:",a)
data["level"][data["suscribers"]==a]
```

```
highest suscribers of the course level is: 268923
```

```
Out[27]: 2827    All Levels  
          Name: level, dtype: object
```

```
In [28]: # Lowest suscribers of the course level
```

```
a = data.suscribers.min()  
print("lowest suscribers of the course level is:",a)  
data["level"][data["suscribers"]==a]
```

```
lowest suscribers of the course level is: 0
```

```
Out[28]: 453      All Levels  
        454      Beginner Level  
        455      Beginner Level  
        456      Beginner Level  
        457      Intermediate Level  
        ...  
        2402     All Levels  
        2403     All Levels  
        2404     Beginner Level  
        2405     Beginner Level  
        2418     Intermediate Level  
          Name: level, Length: 65, dtype: object
```

```
In [29]: # highest reviews of the course level
```

```
a = data.reviews.max()  
print("highest reviews of the course level is:",a)  
data["level"][data["reviews"]==a]
```

```
highest reviews of the course level is: 27445
```

```
Out[29]: 3230    All Levels  
          Name: level, dtype: object
```

```
In [30]: # Lowest reviews of the course level
```

```
a = data.reviews.min()  
print("lowest reviews of the course level is:",a)  
data["level"][data["reviews"]==a]
```

```
lowest reviews of the course level is: 0
```

```
Out[30]: 213      Intermediate Level  
256      Beginner Level  
379      Beginner Level  
405      Intermediate Level  
408      Beginner Level  
...  
2588     All Levels  
2645     All Levels  
3580     All Levels  
3626     All Levels  
3648     All Levels  
Name: level, Length: 284, dtype: object
```

```
In [31]: # highest lectures of the course level  
  
a = data.lectures.max()  
print("highest lectures of the course level is:",a)  
data["level"][data["lectures"]==a]
```

highest lectures of the course level is: 779

```
Out[31]: 2707      Beginner Level  
Name: level, dtype: object
```

```
In [32]: # Lowest lectures of the course level  
  
a = data.lectures.max()  
print("lowest lectures of the course level is:",a)  
data["level"][data["lectures"]==a]
```

lowest lectures of the course level is: 779

```
Out[32]: 2707      Beginner Level  
Name: level, dtype: object
```

```
In [33]: # highest duration of the course level  
  
a = data.duration.max()  
print("highest duration of the course level is:",a)  
data["level"][data["duration"]==a]
```

highest duration of the course level is: 78.5

```
Out[33]: 1658    Beginner Level  
Name: level, dtype: object
```

```
In [34]: # Lowest duration of the course level
```

```
a = data.duration.min()  
print("lowest duration of the course level is:",a)  
data["level"][data["duration"]==a]
```

```
lowest duration of the course level is: 0.0
```

```
Out[34]: 892    All Levels  
Name: level, dtype: object
```

```
In [35]: # Let's count the total paid & free courses
```

```
a = data.paid  
a.value_counts()
```

```
Out[35]: paid  
True      3362  
False     310  
Name: count, dtype: int64
```

```
In [36]: data.columns
```

```
Out[36]: Index(['id', 'title', 'url', 'paid', 'price', 'suscribers', 'reviews',  
       'lectures', 'level', 'duration', 'day', 'month ', 'year', 'hours ',  
       'minutes ', 'seconds', 'subject'],  
       dtype='object')
```

```
In [37]: # Let's count year wise publications of the course
```

```
a = data.year  
b = a.value_counts()  
print("total course publications by",b)
```

```
total course publications by year
2016    1204
2015    1014
2017     713
2014     490
2013     201
2012      45
2011       5
Name: count, dtype: int64
```

```
In [38]: # Let's count the publish day

a = data.day
b = a.value_counts()
print("total course publications by",b)
```

```
total course publications by day
29    150
13    147
12    141
19    135
1     132
17    131
23    131
24    128
3     128
27    128
8     127
28    126
22    125
18    122
9     121
2     121
26    118
6     117
21    115
16    114
14    112
20    111
7     111
10    110
11    107
15    106
25    106
5     102
4     101
30    94
31    55
Name: count, dtype: int64
```

```
In [39]: # change the month column data type
          data[ "month " ].astype( "int64" )
```

```
Out[39]: 0      1
         1      3
         2     12
         3      5
         4     12
         ..
        3673     6
        3674     3
        3675    12
        3676     8
        3677     9
Name: month , Length: 3672, dtype: int64
```

```
In [40]: # Let's count the publish month
```

```
b = data["month "]
b.value_counts()
```

```
Out[40]: month
5      375
3      364
4      355
6      349
2      339
11     298
1      292
10     286
7      277
8      254
12     244
9      239
Name: count, dtype: int64
```

```
In [41]: # most common purchases course level
```

```
a = data.level.mode()
print("most common purchases course level is:",a)
```

```
most common purchases course level is: 0      All Levels
Name: level, dtype: object
```

```
In [42]: # most common purchases course subject
```

```
a = data.subject.mode()  
print("most common purchases course subject is:",a)
```

```
most common purchases course subject is: 0      Web Development  
Name: subject, dtype: object
```

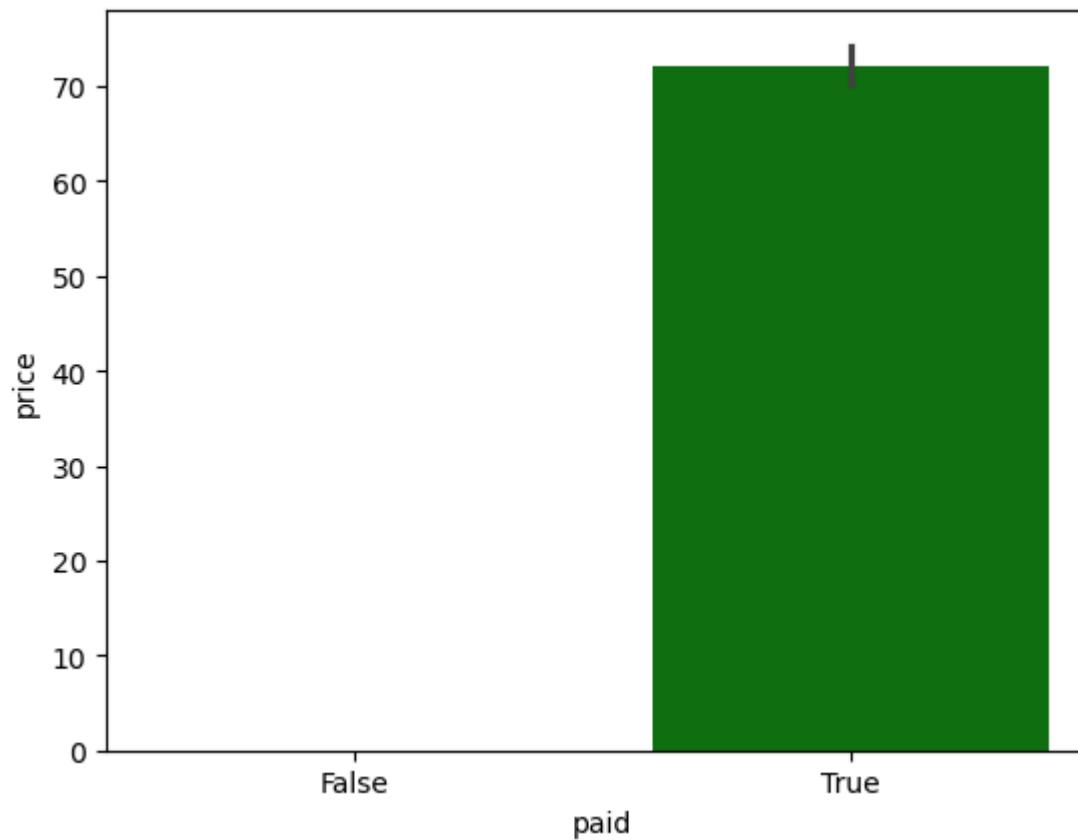
```
In [43]: # most common purchases course title
```

```
a = data.title.mode()  
print("most common purchases course title is:",a)
```

```
most common purchases course title is: 0          Acoustic Blues Guitar Lessons  
1    Creating an animated greeting card via Google ...  
Name: title, dtype: object
```

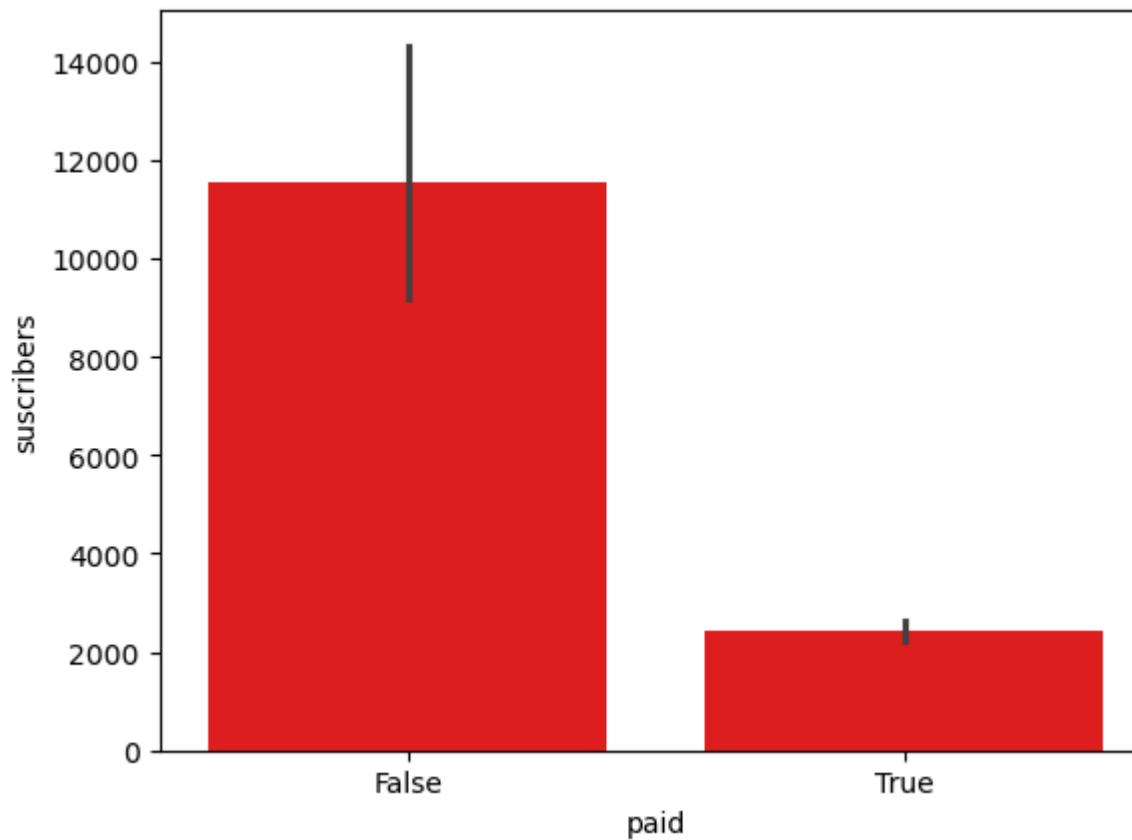
```
In [44]: # paid wise course prices
```

```
sns.barplot(x="paid",y="price",data=data,color="green")  
plt.show()
```

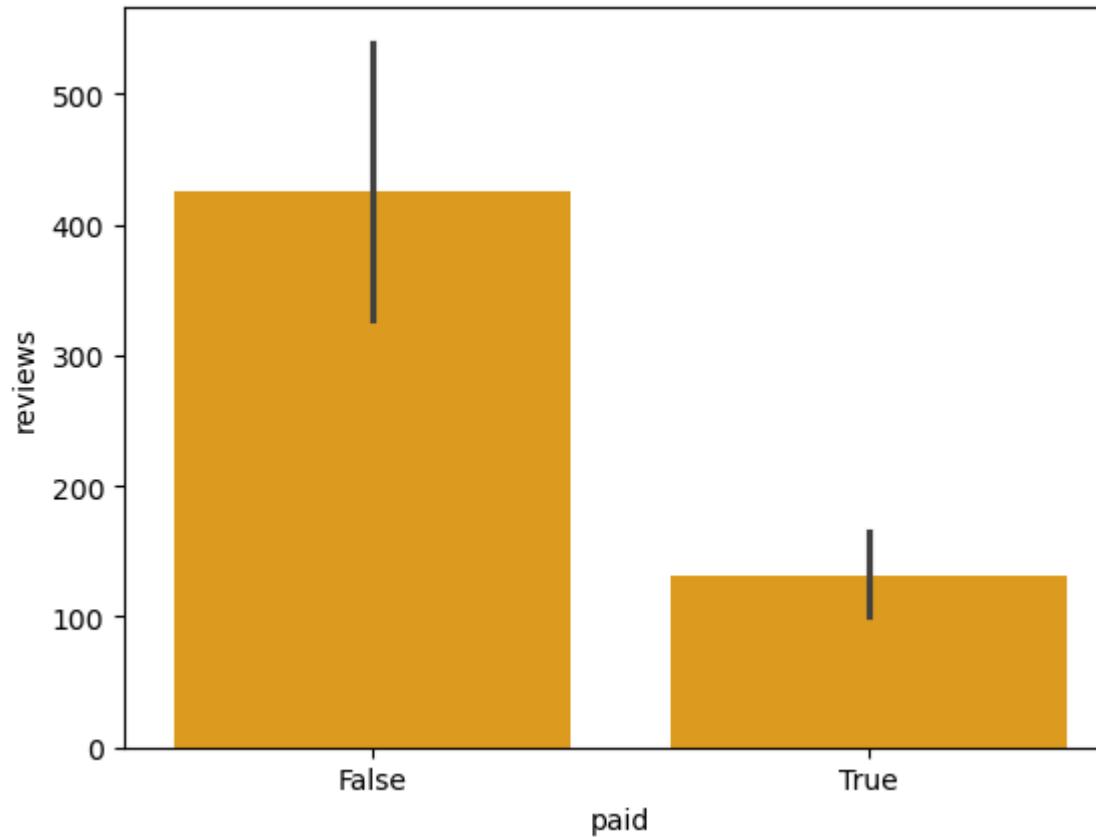


```
In [45]: # paid wise course suscribers
```

```
sns.barplot(x="paid",y="suscribers",data=data,color="red")
plt.show()
```

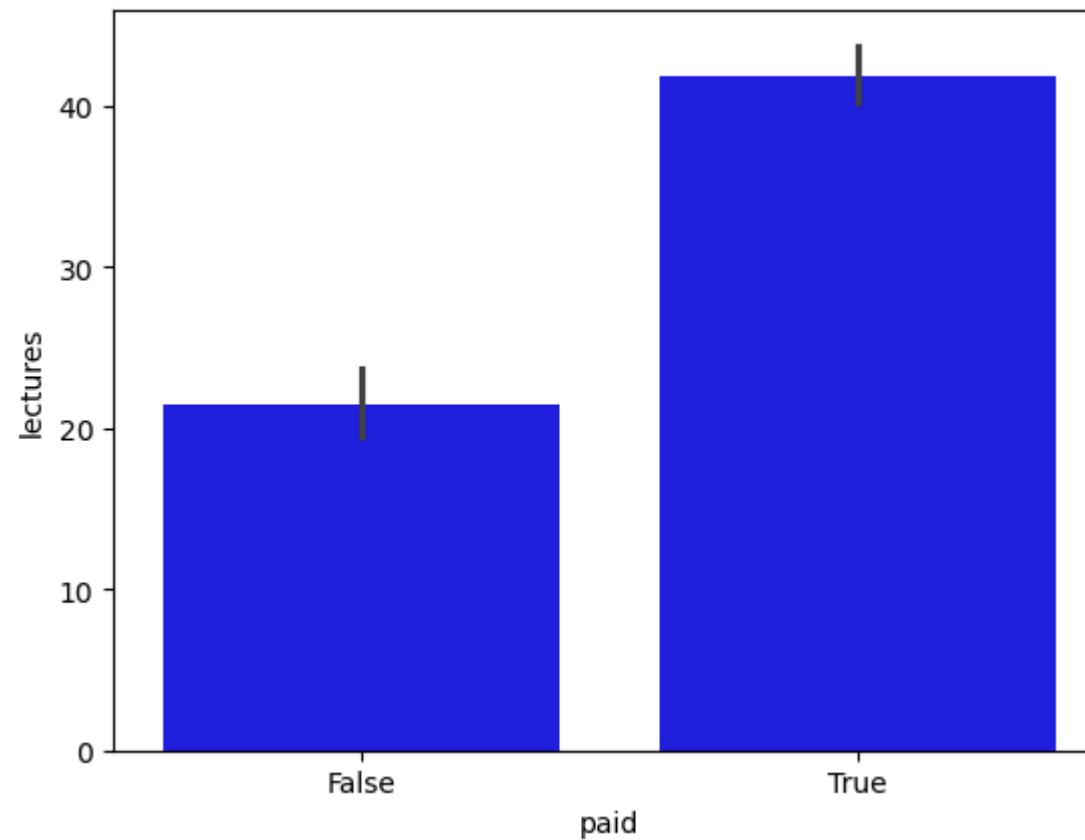


```
In [46]: # paid wise course reviews  
sns.barplot(x="paid",y="reviews",data=data,color="orange")  
plt.show()
```

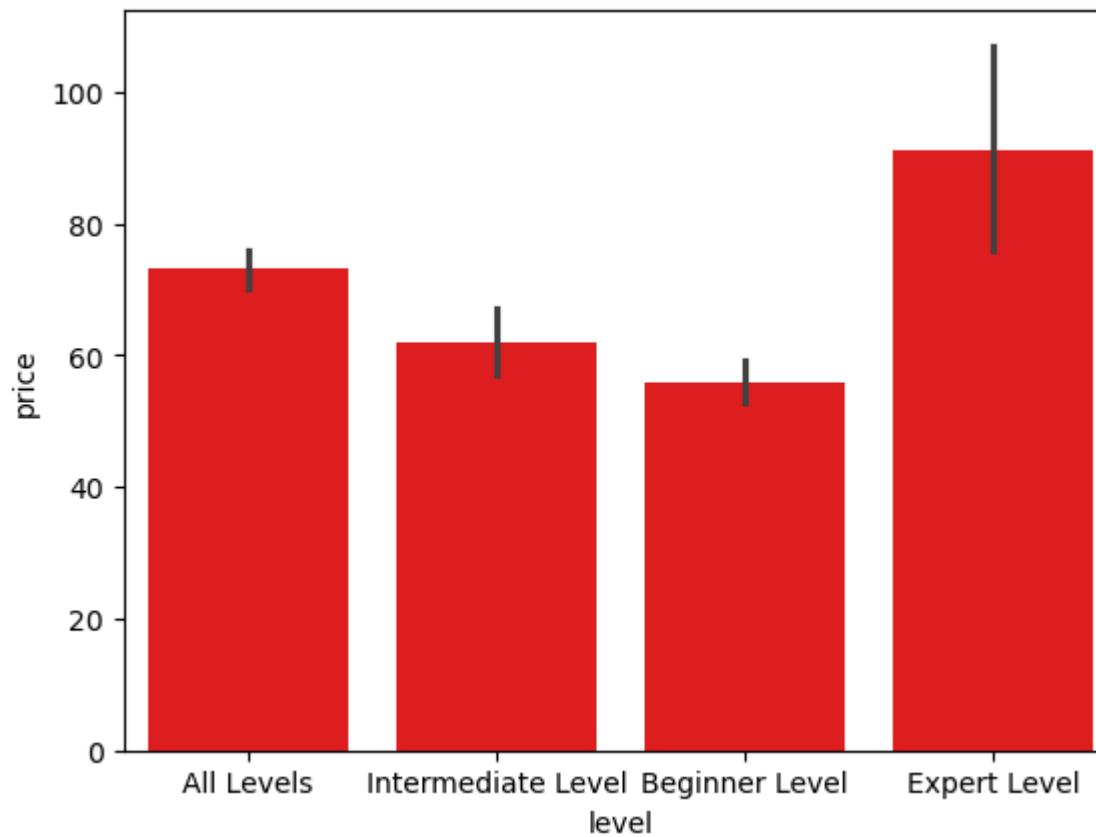


```
In [47]: # paid wise course Lectures
```

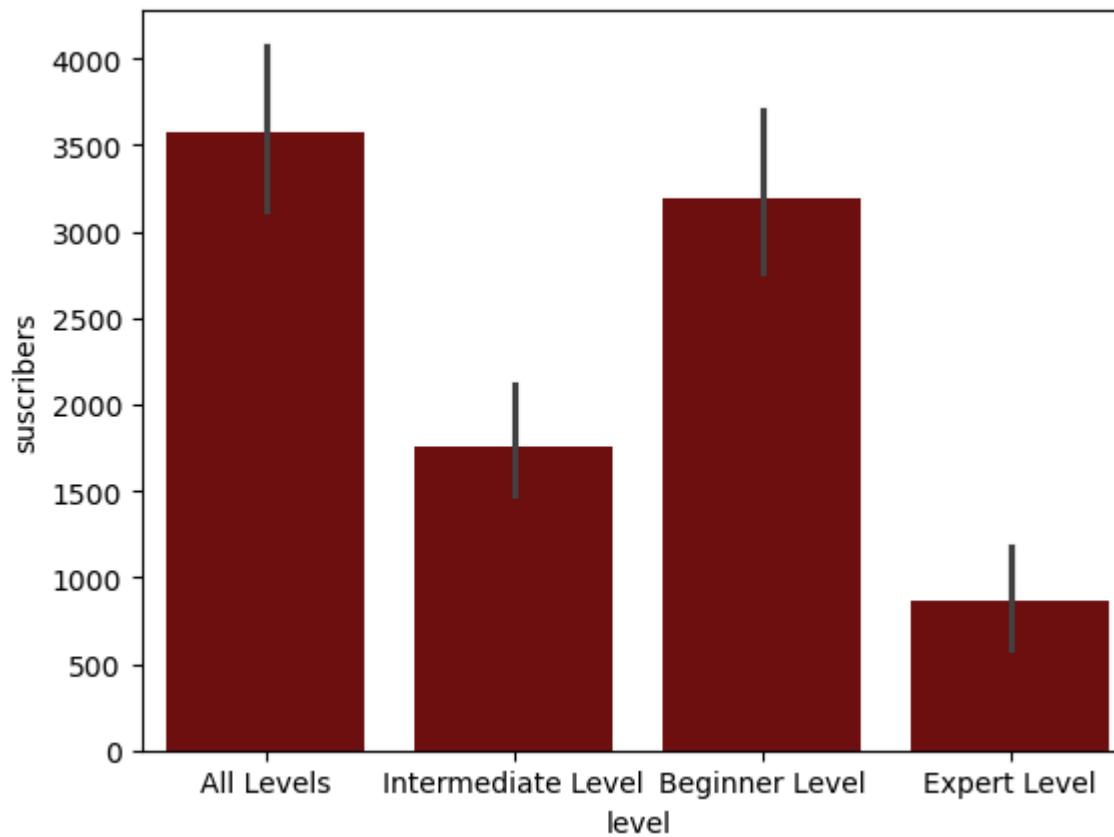
```
sns.barplot(x="paid",y="lectures",data=data,color="blue")
plt.show()
```



```
In [48]: # Level wise course prices  
sns.barplot(x="level",y="price",data=data,color="red")  
plt.show()
```

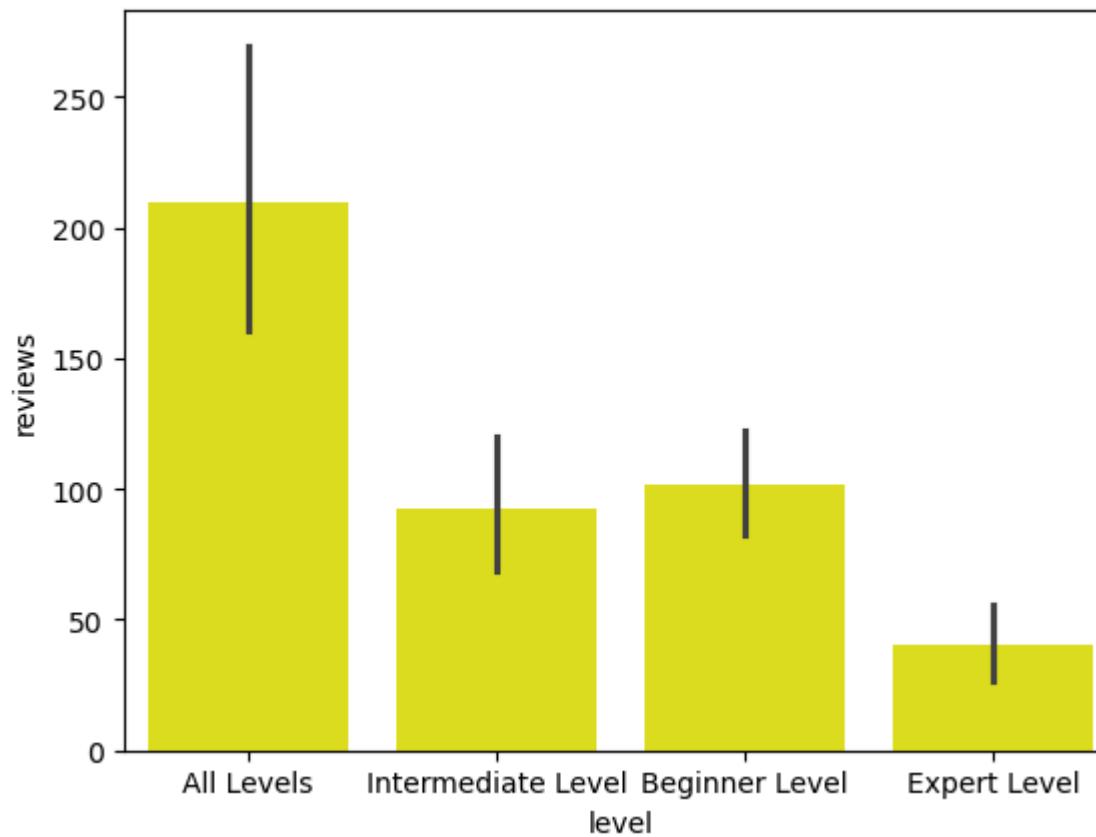


```
In [49]: # Level wise course suscribers  
sns.barplot(x="level",y="suscribers",data=data,color="maroon")  
plt.show()
```

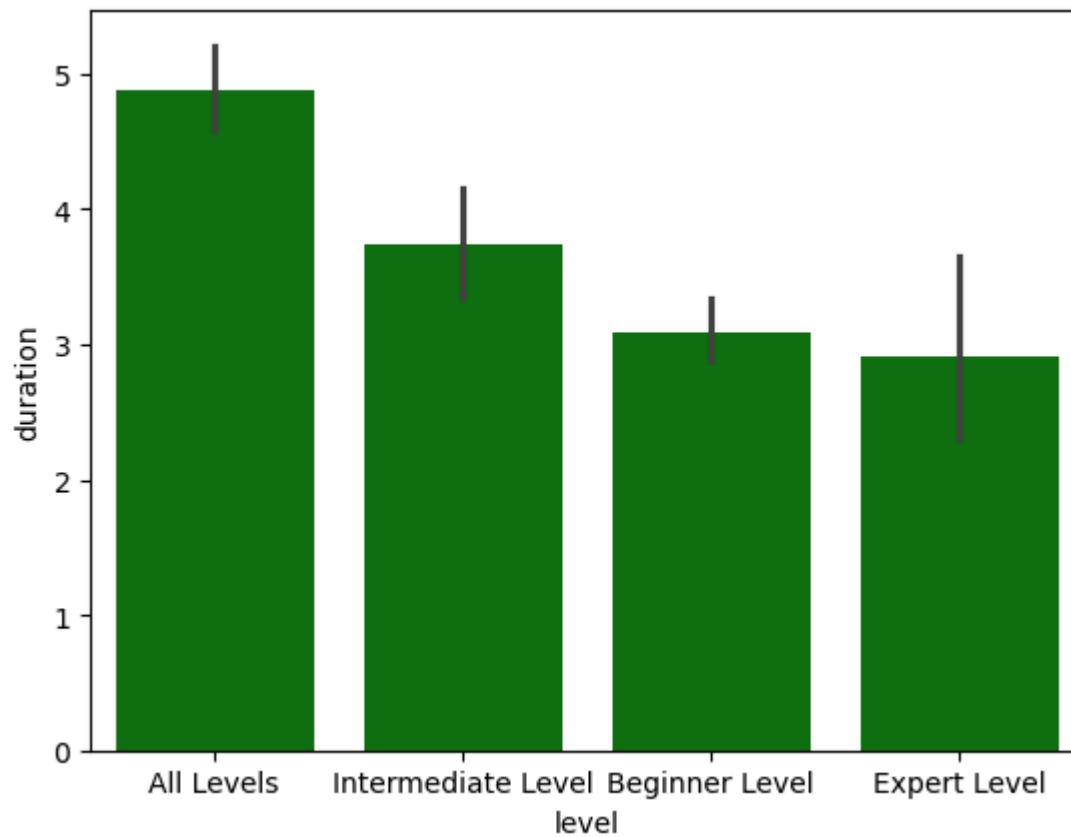


```
In [50]: # Level wise course reviews
```

```
sns.barplot(x="level",y="reviews",data=data,color="yellow")
plt.show()
```

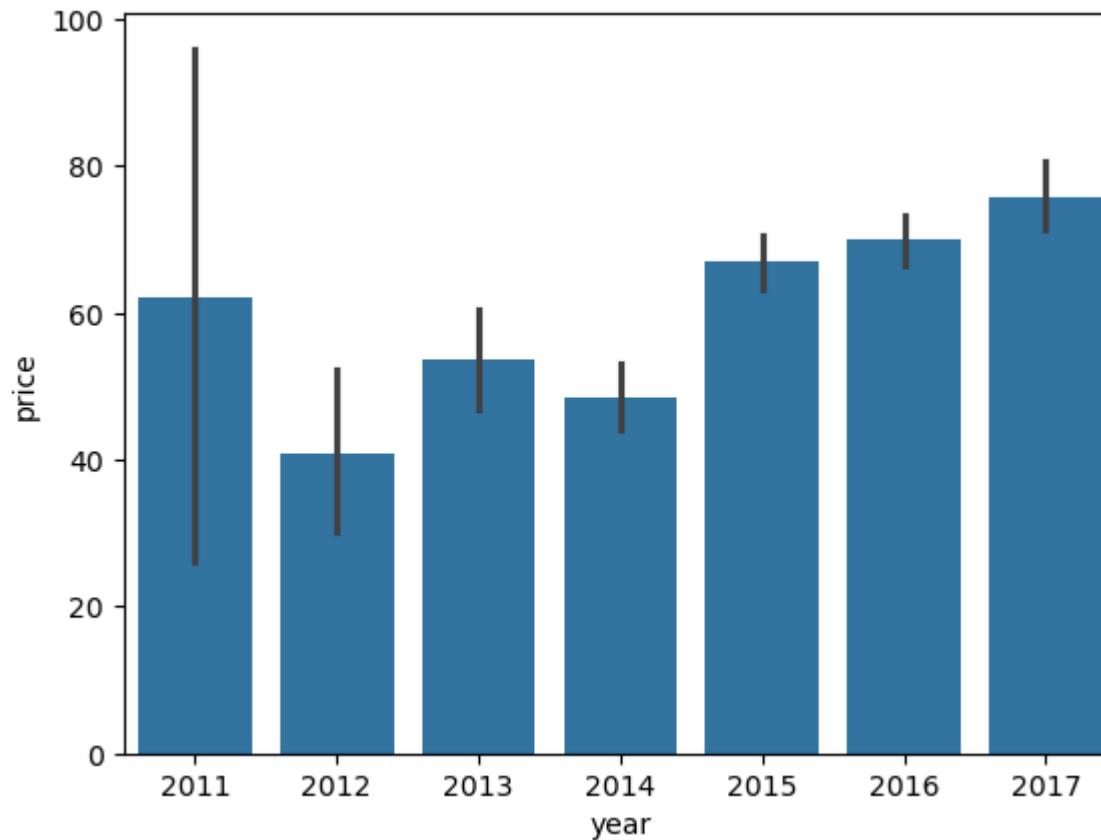


```
In [51]: # Level wise course duration  
sns.barplot(x="level",y="duration",data=data,color="green")  
plt.show()
```



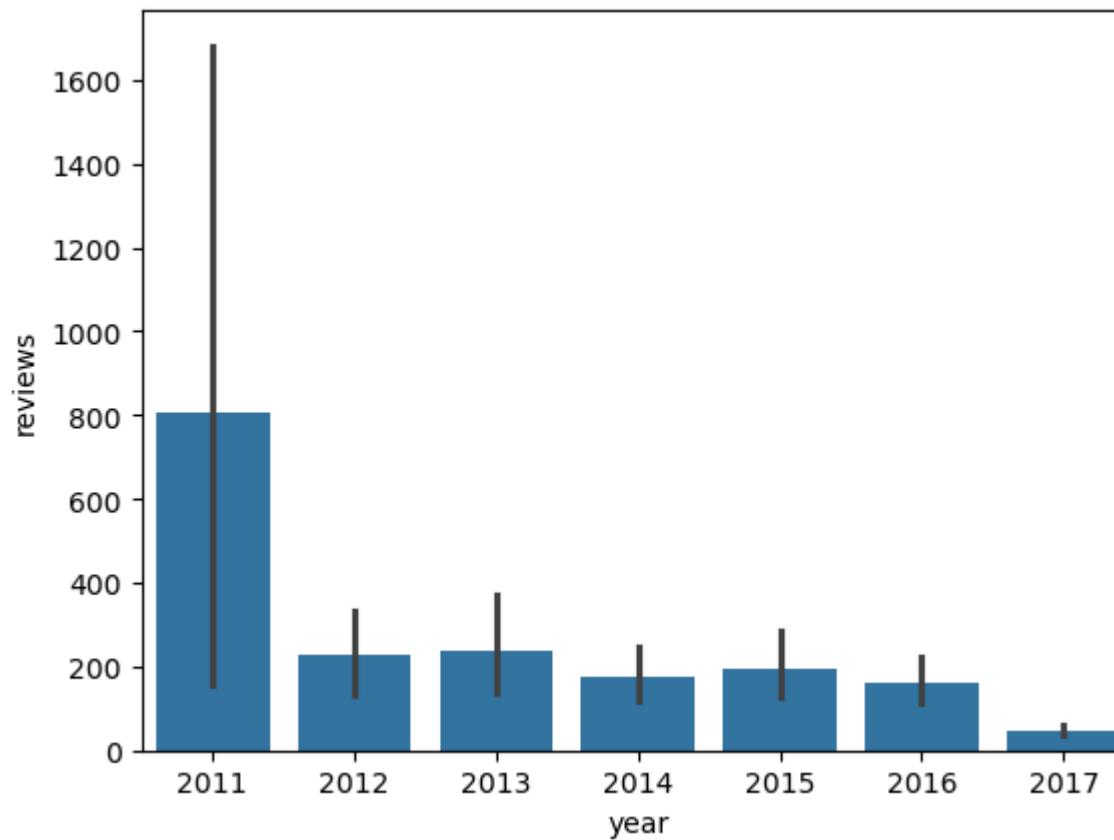
```
In [52]: # year wise course prices is increases or decreases Let's check it
```

```
sns.barplot(x="year",y="price",data=data)
plt.show()
```



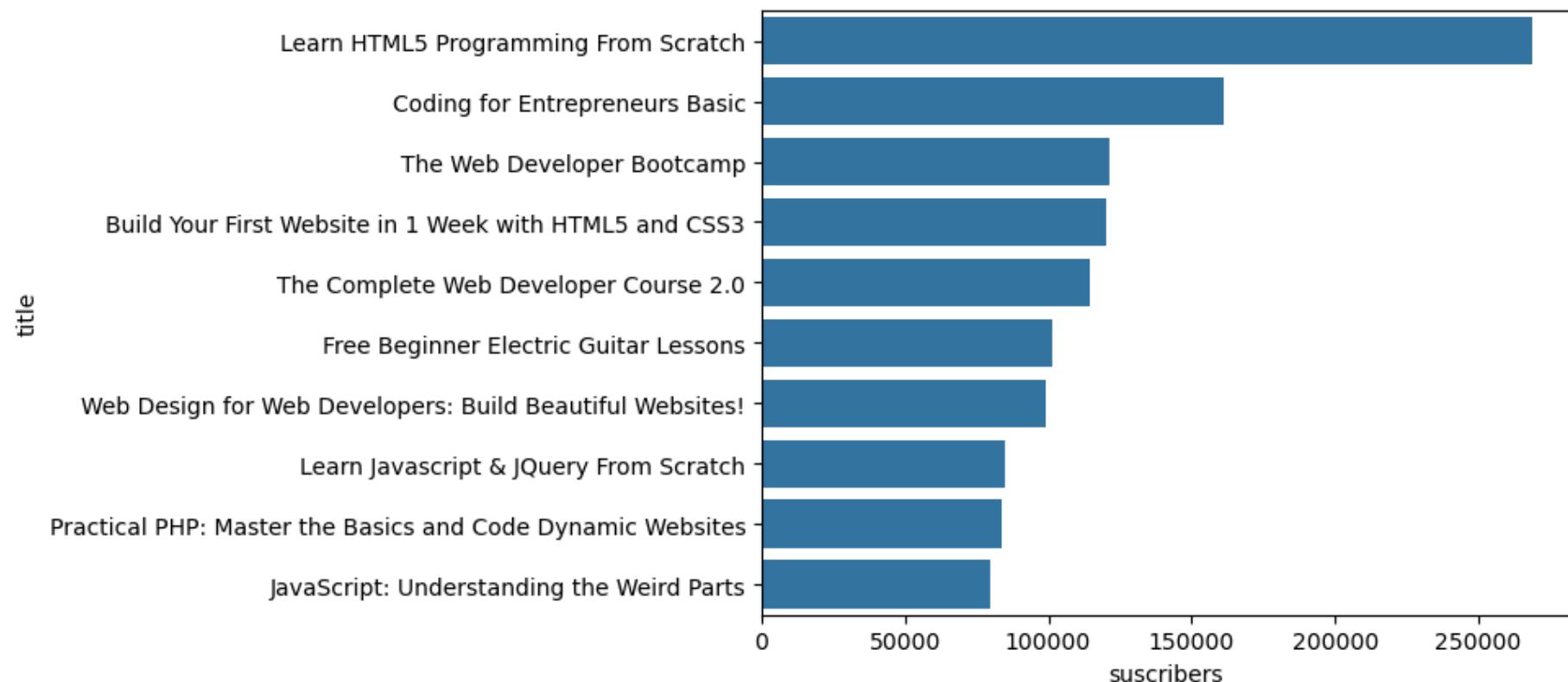
```
In [53]: # year wise course reviews is increases or decreases Let's check it
```

```
sns.barplot(x="year",y="reviews",data=data)
plt.show()
```



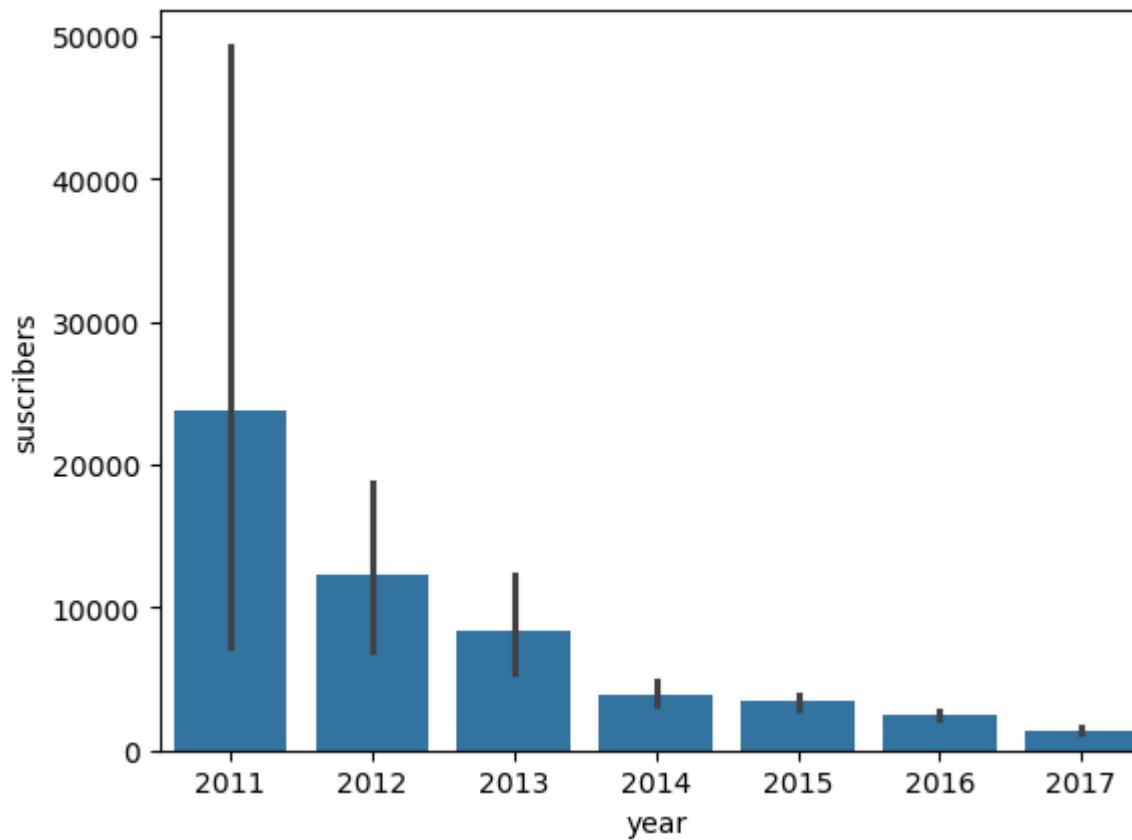
```
In [54]: # course title wise suscribers
# we consider the highest top10 suscribers from the dataset

top10 = data.sort_values(by = "suscribers", ascending=False).head(10)
sns.barplot(x = "suscribers",y = "title",data = top10)
plt.show()
```



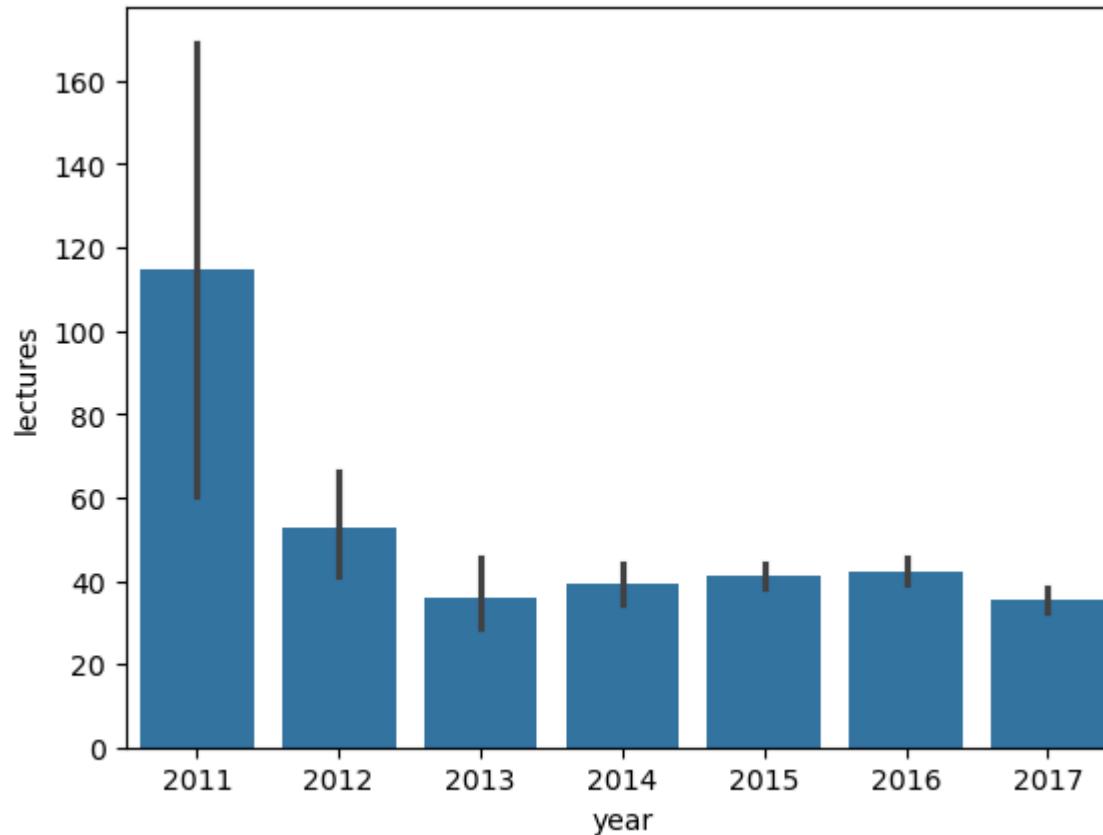
```
In [55]: # year wise course suscribers is increases or decreases let's check it
```

```
sns.barplot(x="year",y="suscribers",data=data)
plt.show()
```



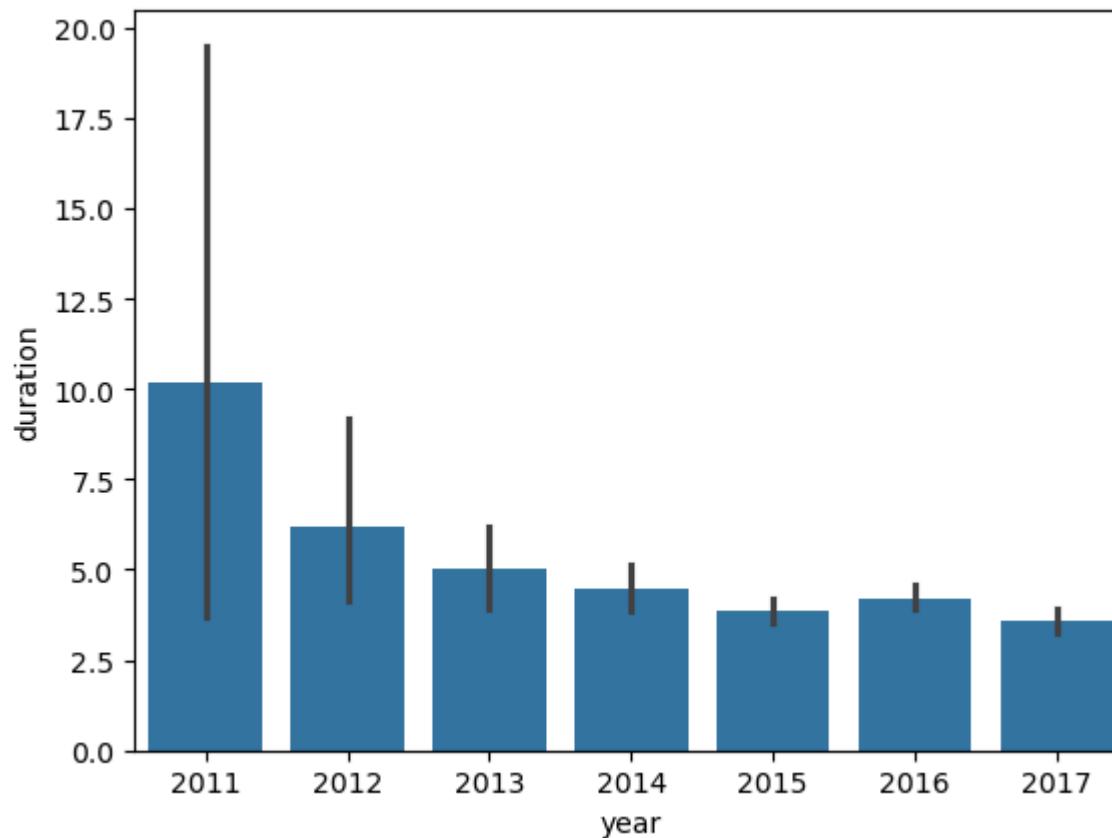
```
In [56]: # year wise course Lectures is increases or decreases Let's check it
```

```
sns.barplot(x="year",y="lectures",data=data)
plt.show()
```



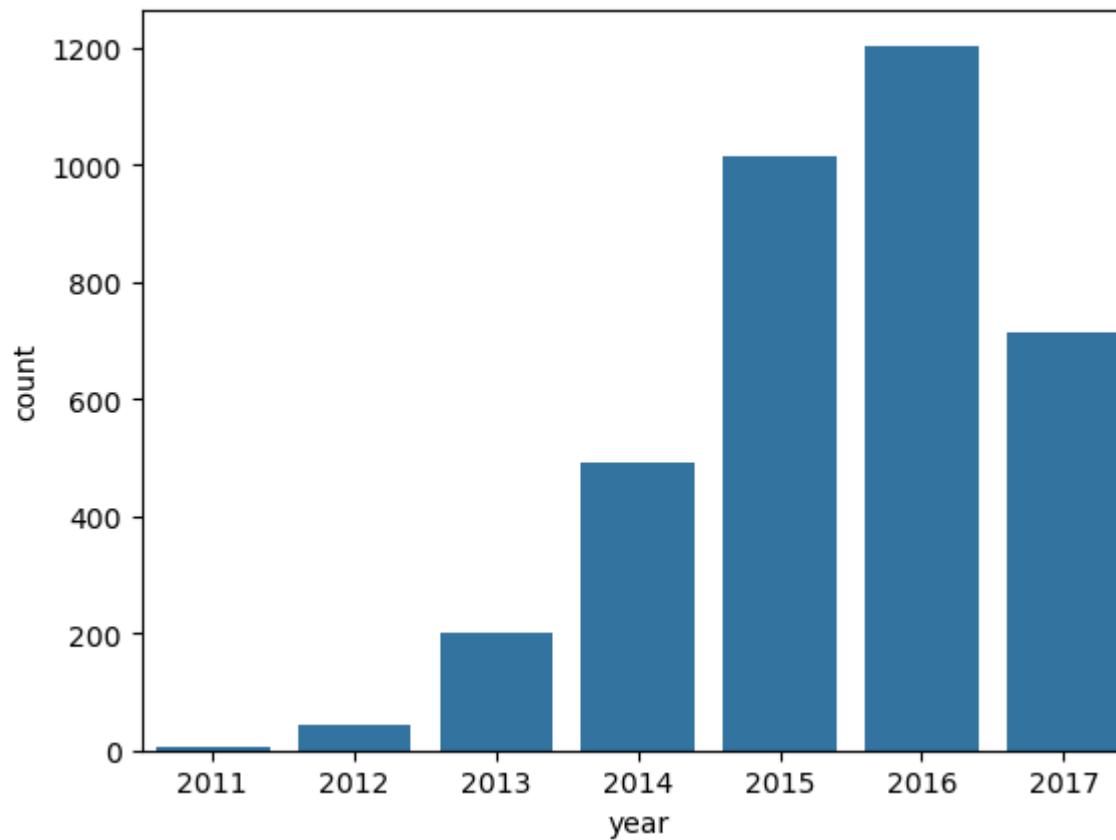
```
In [57]: # year wise course duration is increases or decreases Let's check it
```

```
sns.barplot(x="year",y="duration",data=data)
plt.show()
```



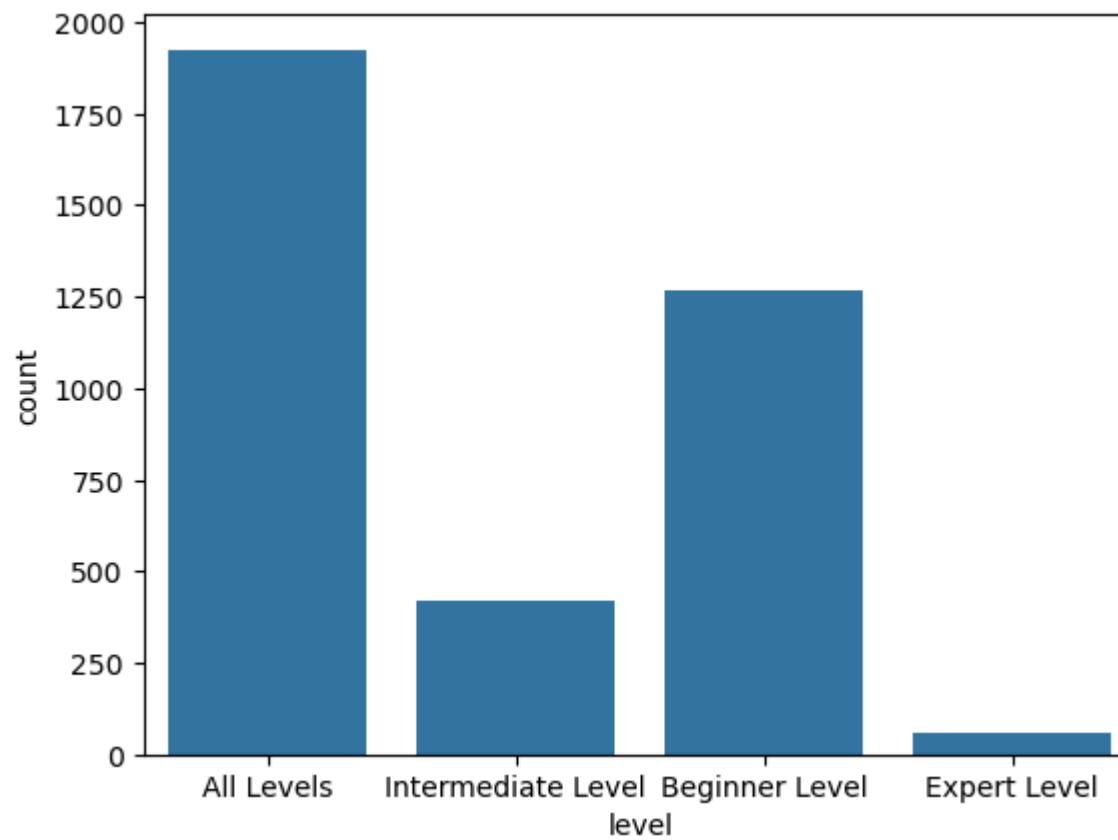
```
In [58]: # Let's check the highest course publish year
```

```
sns.countplot(x="year", data=data)
plt.show()
```



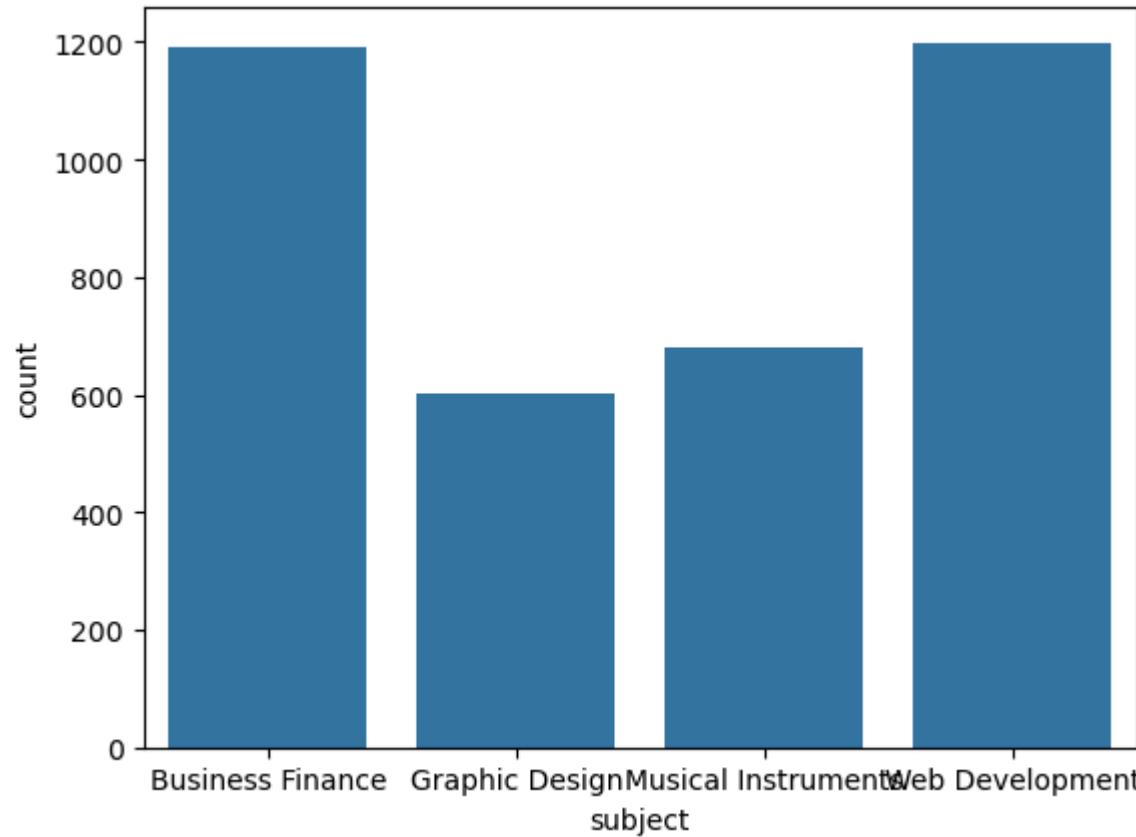
```
In [59]: # Let's count the highest publish course level
```

```
sns.countplot(x="level", data=data)
plt.show()
```



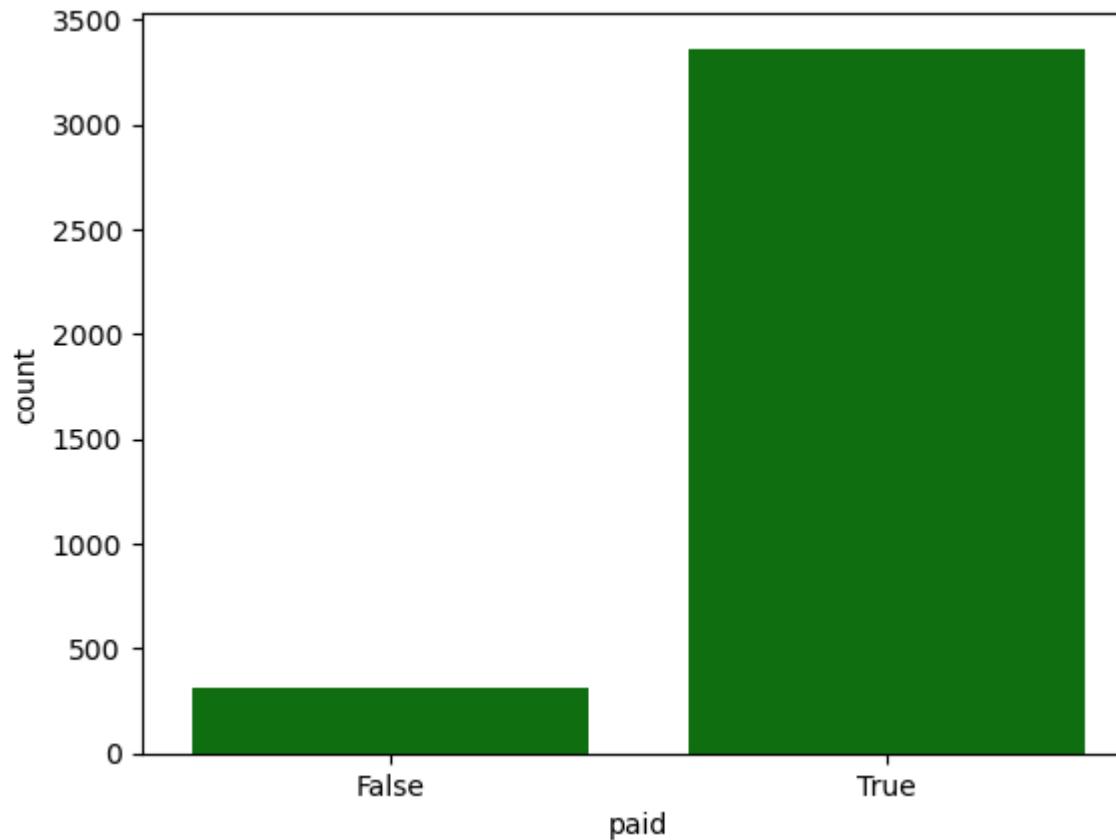
```
In [60]: # Let's count the highest publish course subject
```

```
    sns.countplot(x="subject", data=data)
    plt.show()
```

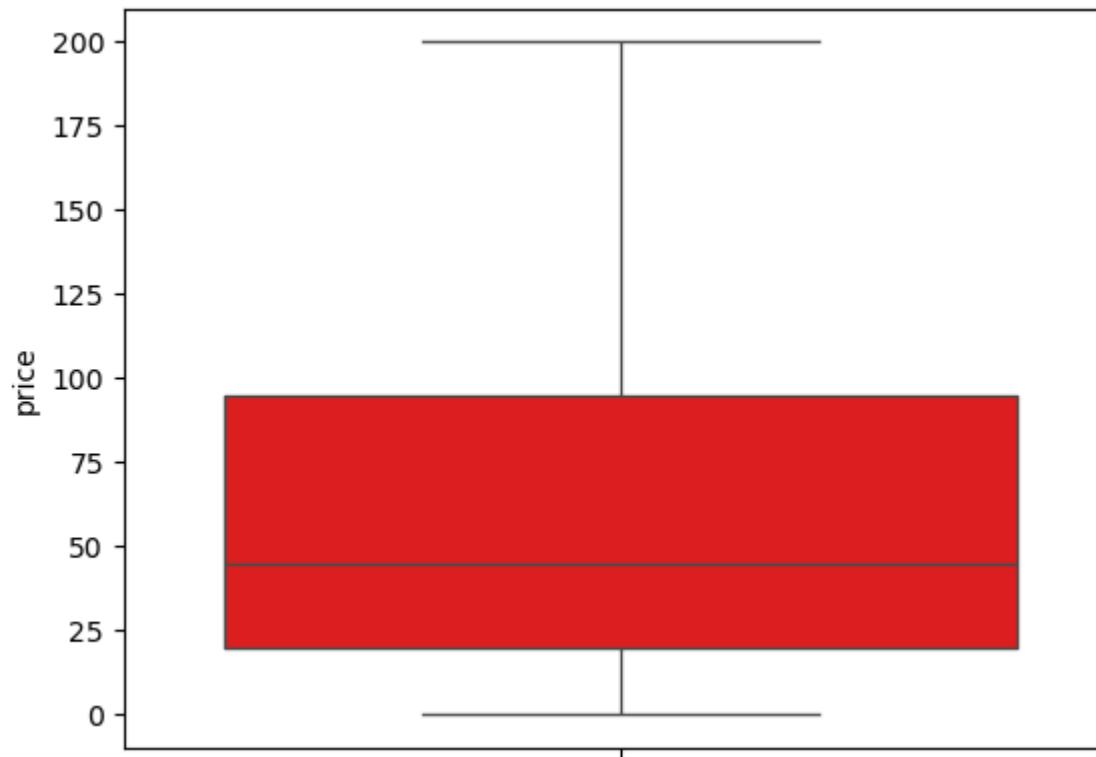


```
In [61]: # Let's count the publish course type means free or paid
```

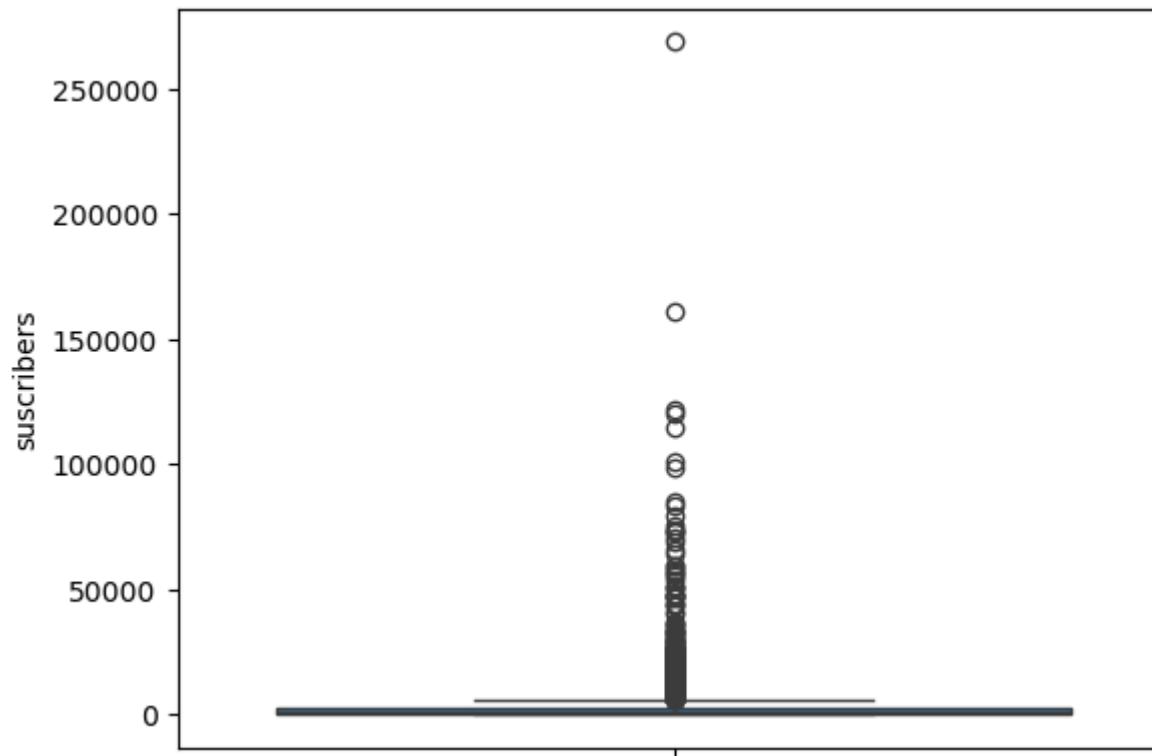
```
sns.countplot(x="paid", data=data, color="green")
plt.show()
```



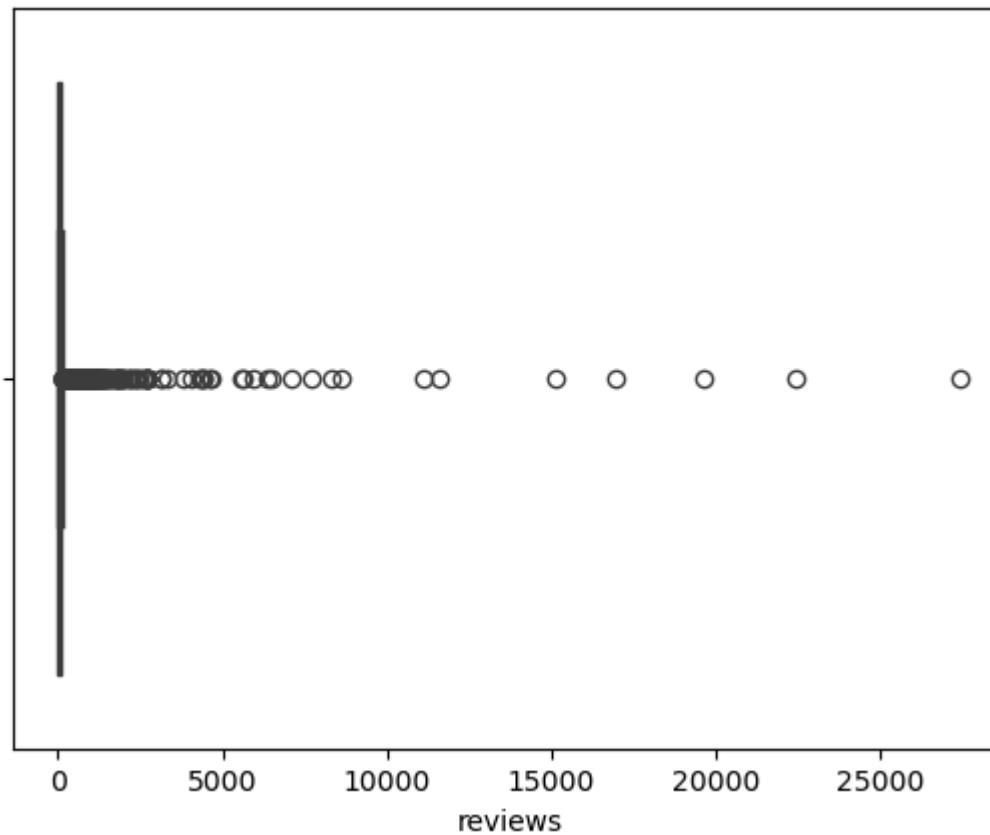
```
In [62]: # Let's check is there any outliers in our dataset  
sns.boxplot(y="price", data=data, color="red")  
plt.show()
```



```
In [63]: # Let's check if there are any outliers in our dataset  
sns.boxplot(y="suscribers", data=data)  
plt.show()
```

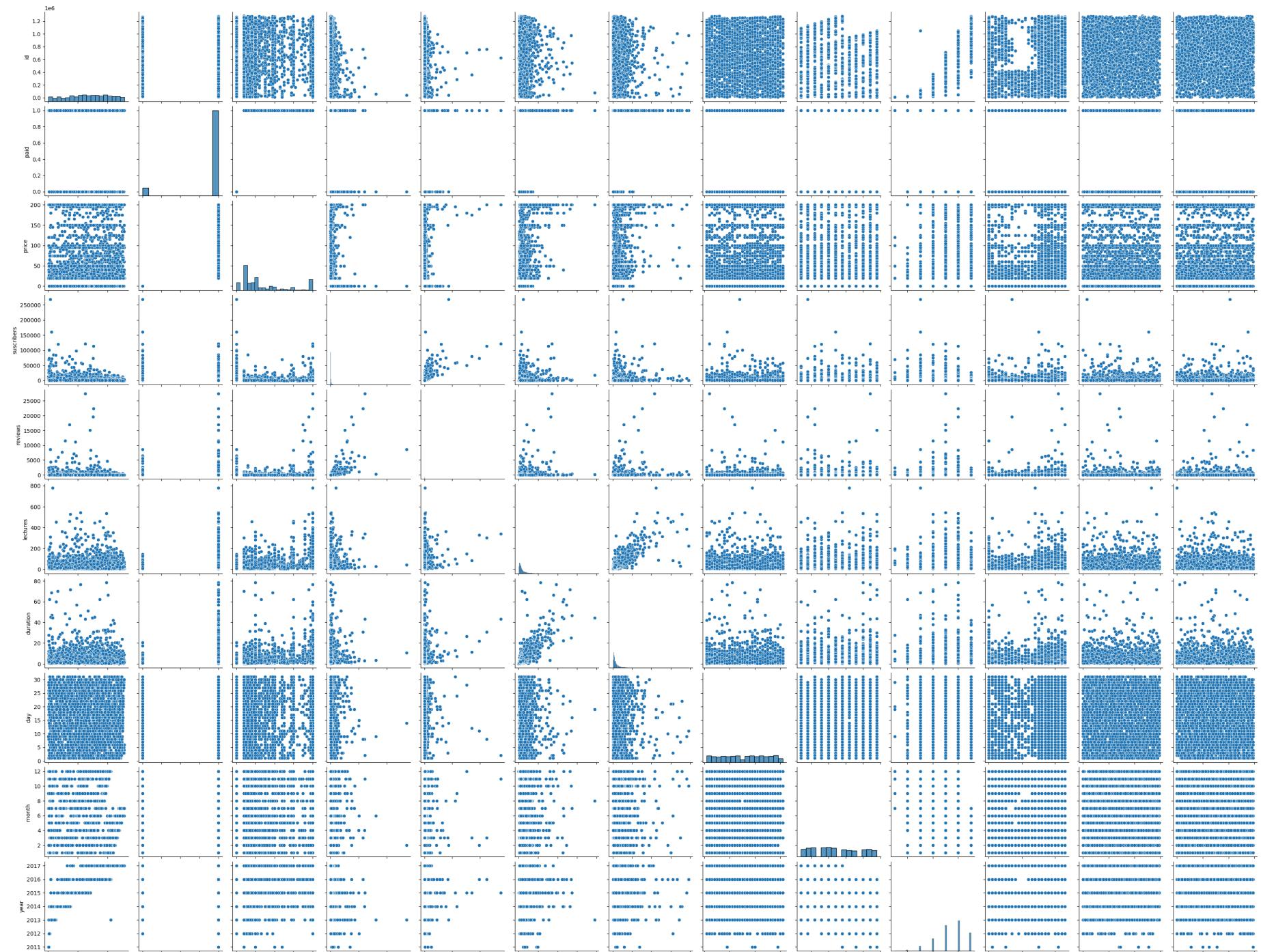


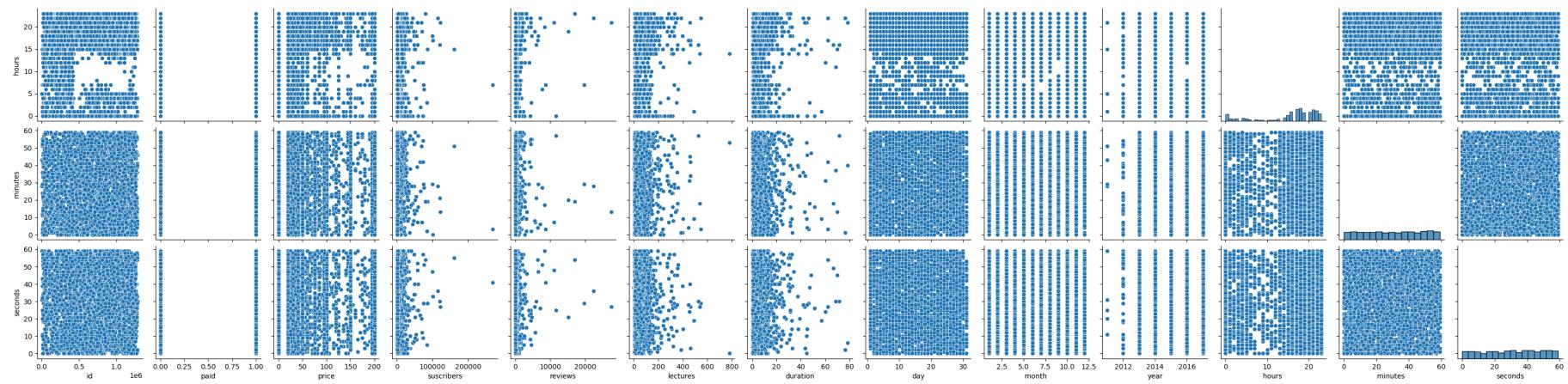
```
In [64]: # Let's check if there are any outliers in our dataset  
sns.boxplot(x="reviews", data=data)  
plt.show()
```



In [65]: *# Let's check the distribution in our dataset*

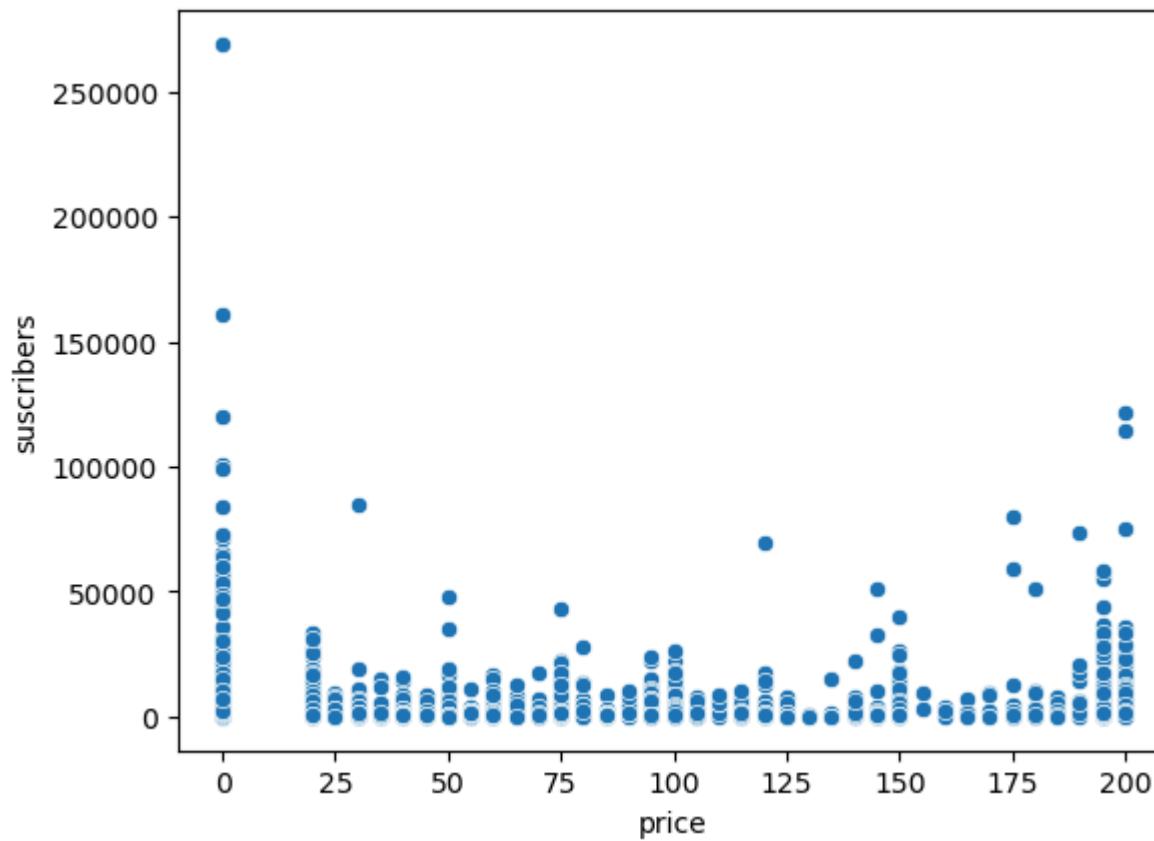
```
sns.pairplot(data=data)
plt.show()
```





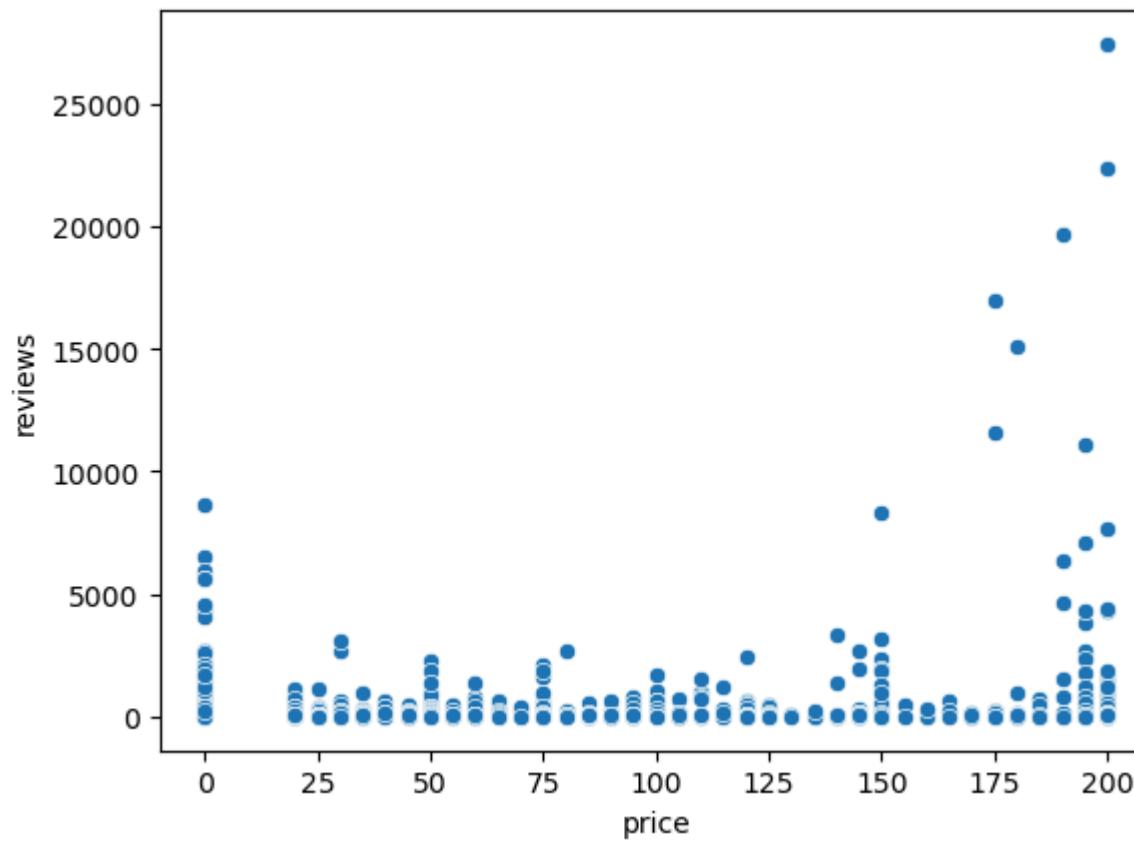
```
In [66]: # Let's check the relation ship between two variables
```

```
sns.scatterplot(x="price",y="suscribers",data=data)
plt.show()
```



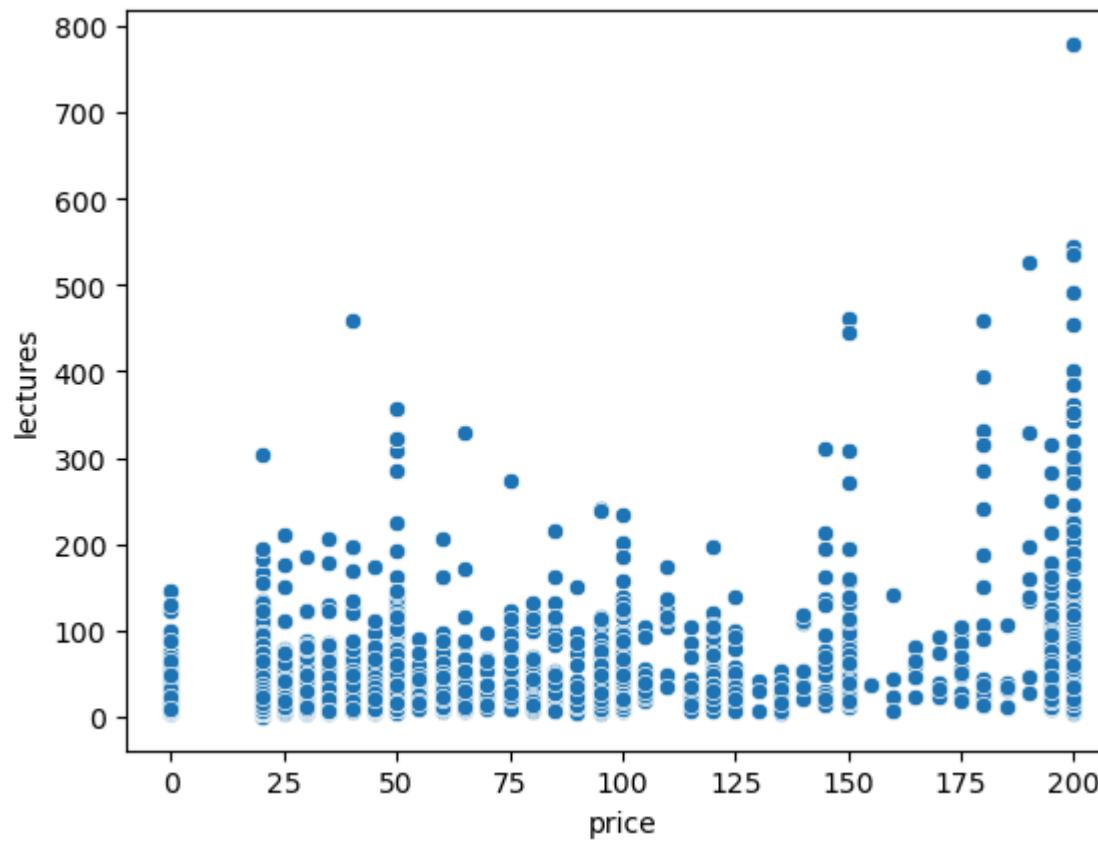
```
In [67]: # Let's check the relation ship between two variables
```

```
sns.scatterplot(x="price",y="reviews",data=data)
plt.show()
```



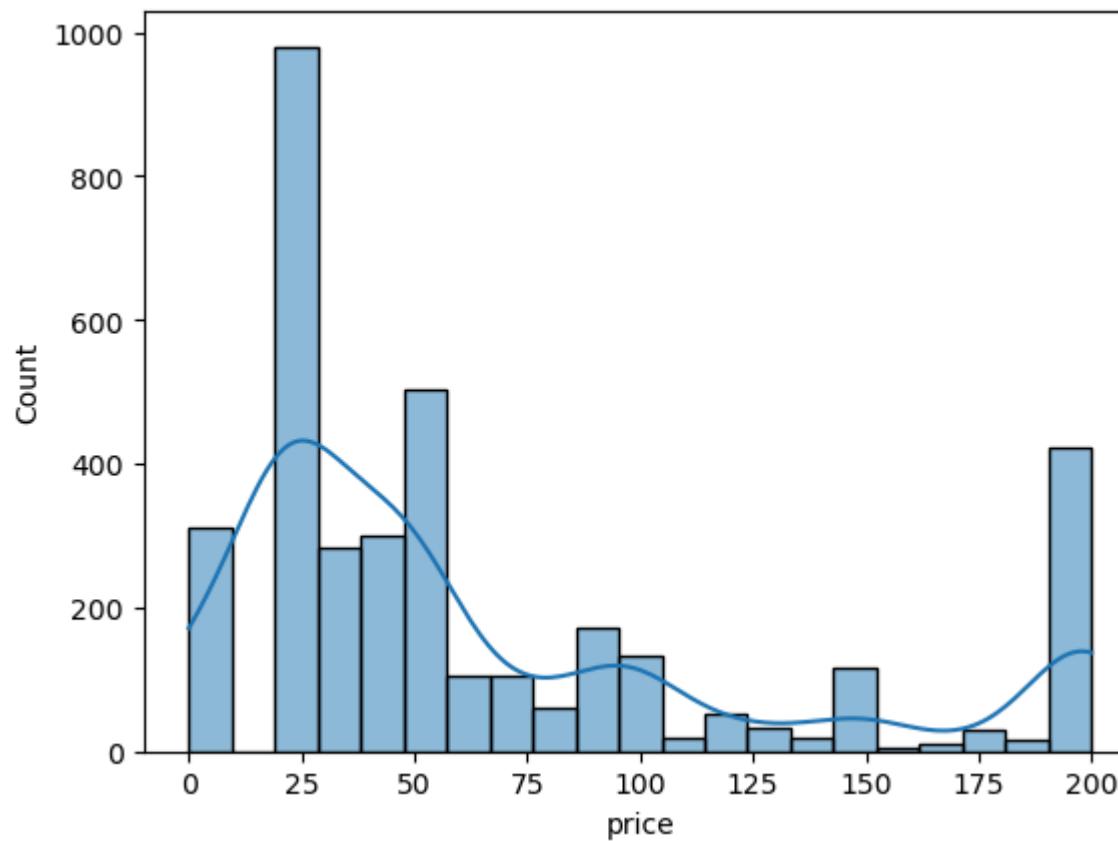
```
In [68]: # Let's check the relation ship betweeen two variables
```

```
sns.scatterplot(x="price",y="lectures",data=data)
plt.show()
```



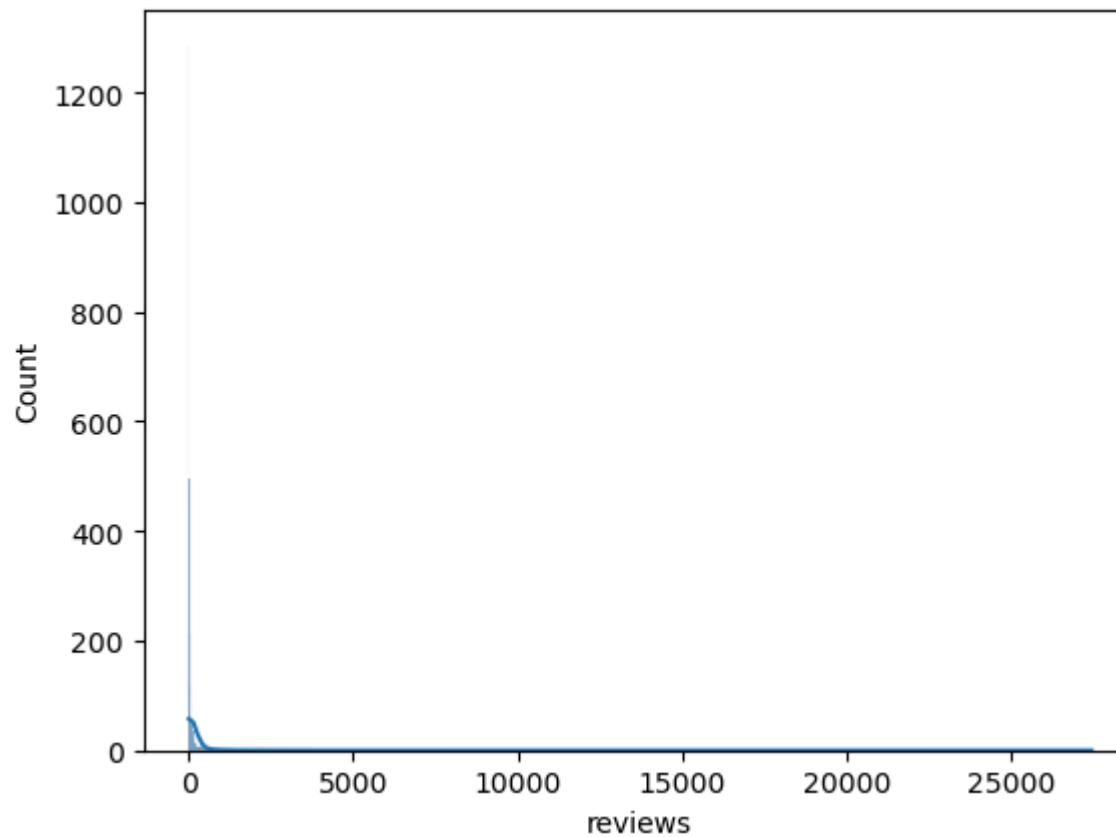
```
In [69]: # to check the distribution
```

```
sns.histplot(x="price", data=data, kde=True)  
plt.show()
```

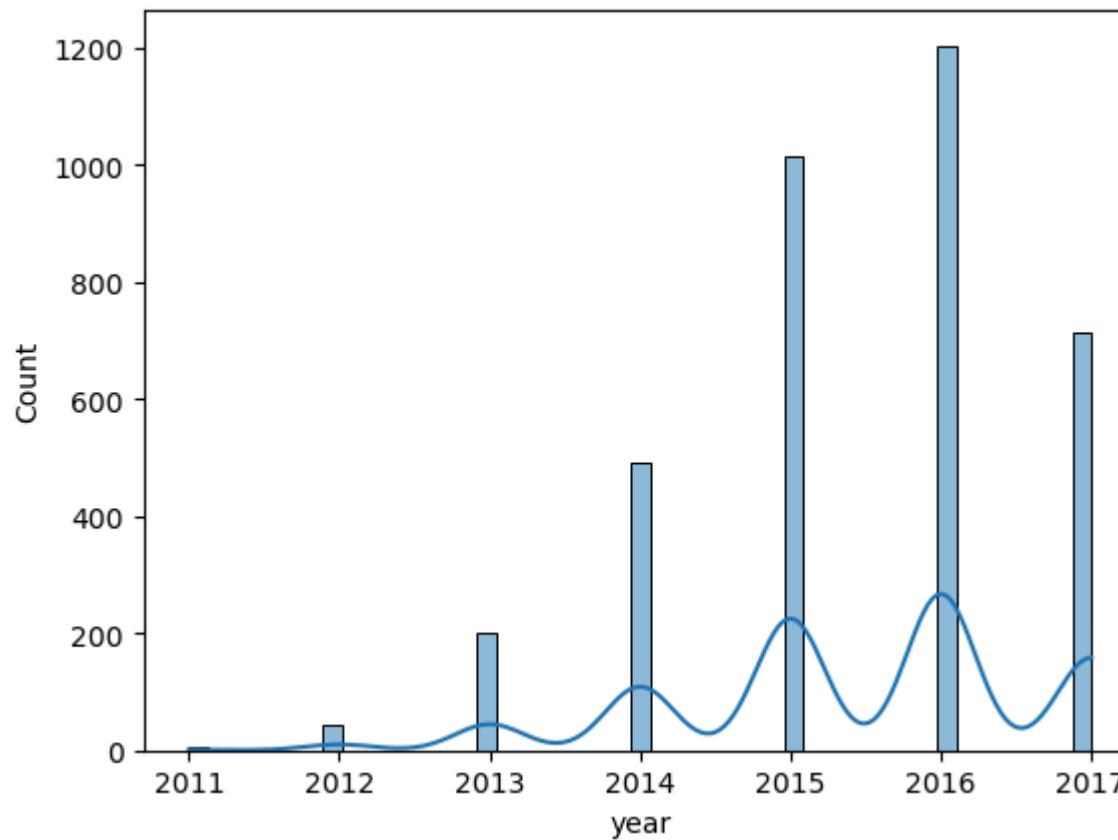


```
In [70]: # to check the distribution
```

```
sns.histplot(x="reviews", data=data, kde=True)
plt.show()
```



```
In [71]: # to check the distribution  
  
sns.histplot(x="year", data=data, kde=True)  
plt.show()
```



In [72]: *# column wise standard deviations*

```
data.std(axis=1,skipna=True,numeric_only=True)
```

```
Out[72]: 0      296927.844221
         1      308776.568608
         2      278999.051349
         3      335649.095582
         4      280332.666768
         ...
        3673    215041.979322
        3674    301749.309982
        3675    176119.226256
        3676    250971.333672
        3677    82469.48903
Length: 3672, dtype: object
```

```
In [73]: # row wise standard deviations
```

```
data.std(axis=0,skipna=True,numeric_only=True)
```

```
Out[73]: id          343071.951301
         paid        0.278058
         price       61.035920
         suscribers  9488.105448
         reviews     936.178649
         lectures    50.417102
         duration    6.057830
         day          8.781920
         month        3.380390
         year          1.184731
         hours        6.820321
         minutes      17.523770
         seconds      17.099919
dtype: float64
```

```
In [74]: # column wise variance
```

```
data.var(axis=1,skipna=True,numeric_only=True)
```

```
Out[74]: 0      88166144673.557678
         1      95342969321.52565
         2      77840470653.57692
         3      112660315364.897415
         4      78586404057.358994
         ...
        3673    46243052870.858978
        3674    91052646074.358963
        3675    31017981857.141022
        3676    62986610324.910248
        3677    6801216620.858975
Length: 3672, dtype: object
```

```
In [75]: # row wise variance
```

```
data.var(axis=0,skipna=True,numeric_only=True)
```

```
Out[75]: id      1.176984e+11
          paid     7.731653e-02
          price    3.725384e+03
          suscribers 9.002415e+07
          reviews   8.764305e+05
          lectures  2.541884e+03
          duration  3.669730e+01
          day       7.712212e+01
          month     1.142704e+01
          year      1.403587e+00
          hours     4.651678e+01
          minutes   3.070825e+02
          seconds   2.924072e+02
          dtype: float64
```

```
In [76]: # Let's check the bell shape curve means proper distribution curve is made by our dataset or not? row wise
```

```
a = data.quantile(q=0.25,axis=1,numeric_only=True)
sns.kdeplot(a)
```

```
Out[76]: <Axes: xlabel='0.25', ylabel='Density'>
```

