Libraries

In [97]:
```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
%matplotlib inline
import plotly.graph_objects as go
from plotly.subplots import make_subplots
```

Load the data

In [22]:
```python
df=pd.read_csv('C:/Users/SLEFPL/Desktop/DSA/data_analyst/udemy_dataset/udemy_courses.csv')
```

Out[22]:

	course_id	course_title	url	is_paid	price	num_subscribers	num_reviews	num_lectures	level	content_duration	published_timestamp	subject
0	1070968	Ultimate Investment Banking Course	https://www.udemy.com/ultimate-investment-bank...	True	200	2147	23	51	All Levels	1.5	2017-01-18T20:58:58Z	Business Finance
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	2017-03-09T16:34:20Z	Business Finance
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	2016-12-19T19:26:30Z	Business Finance
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	2017-05-30T20:07:24Z	Business Finance
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	2016-12-13T14:57:18Z	Business Finance
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	2014-05-02T15:13:30Z	Business Finance
6	739964	Investing And Trading For Beginners: Mastering...	https://www.udemy.com/investing-and-trading-fo...	True	65	1540	178	26	Beginner Level	1.0	2016-02-21T18:23:12Z	Business Finance
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	2015-01-30T22:13:03Z	Business Finance
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	2015-05-28T00:14:03Z	Business Finance
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	2017-04-18T18:13:32Z	Business Finance

In [42]:
```python
df['published_date']=df['published_timestamp'].str.split('T').str.get(0)
df['published_date']=pd.to_datetime(df['published_date'], format='%Y-%m-%d')
df['year']=df['published_date'].dt.year
```

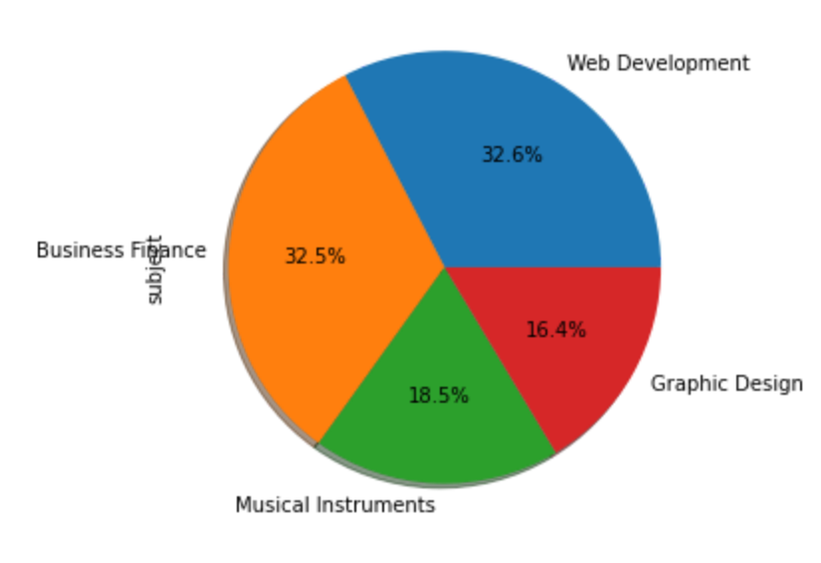
What is the distribution of subjects

In [116…
```python
# Distribution of subjects
df['subject'].unique()
```

Out[116…
```
array(['Business Finance', 'Graphic Design', 'Musical Instruments',
       'Web Development'], dtype=object)
```

In [115…
```python
# Distribution of subjects(course-wise)
print(df['subject'].value_counts())
plt.figure(figsize=(5,5))
df['subject'].value_counts().plot(kind='pie',autopct='%1.1f%%', shadow=True)
plt.show()
```

```
Web Development        1200
Business Finance       1195
Musical Instruments     680
Graphic Design          603
Name: subject, dtype: int64
```

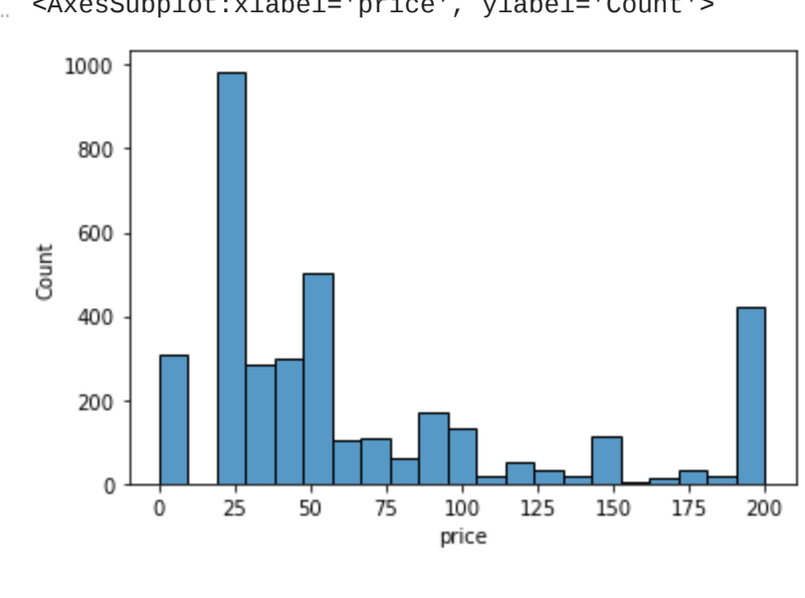


- It can be seen here, that 4 subjects in total are present and most number of courses are in Web Development

Price for different courses

In [119…
```python
# Price range of courses
sns.histplot(data=df, x='price')
```

Out[119… <AxesSubplot:xlabel='price', ylabel='Count'>

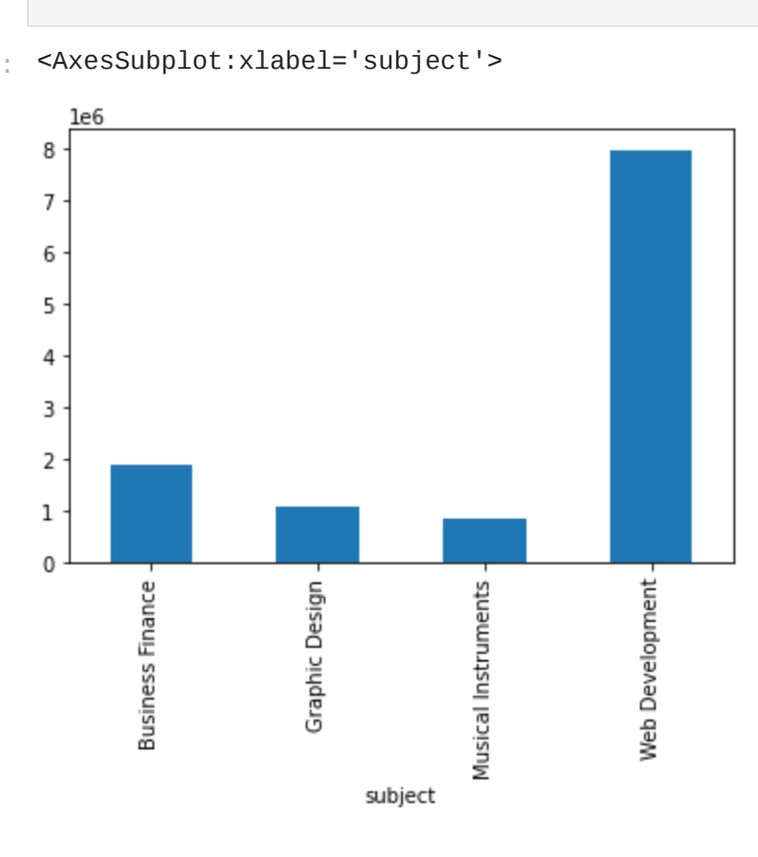


- The maximum number of courses cost around $25

Number of subscribers per subject

In [31]:
```python
# subscribers per subject
df.groupby('subject')['num_subscribers'].sum().plot(kind='bar')
```

Out[31]: <AxesSubplot:xlabel='subject'>



- Although the maximum number of courses are in Business Finance, the number of subscribers are maximum in Web Development.

Course with maximum number of subscribers

In [101…
```python
# Course with max subscribers
id=df['num_subscribers'].idxmax()
df.iloc[id]
```

Out[101…
```
course_id                                              41295
course_title                   Learn HTML5 Programming From Scratch
url                    https://www.udemy.com/learn-html5-programming-...
is_paid                                                False
price                                                      0
num_subscribers                                       268923
num_reviews                                             8629
num_lectures                                              45
level                                             All Levels
content_duration                                        10.5
published_timestamp                     2013-02-14T07:03:41Z
subject                                      Web Development
published_date                           2013-02-14 00:00:00
year                                                    2013
Engagement                                            277552
Name: 2827, dtype: object
```

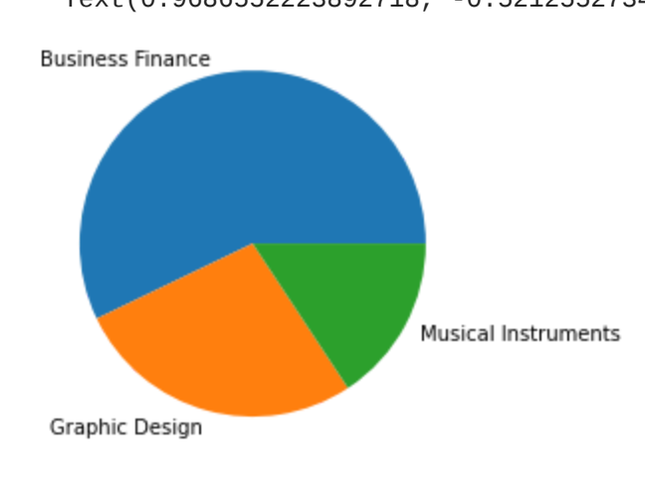
- course with maximum number of subscribers comes out to be Web Development course titled 'Learn HTML5 Programming From Scratch' with 268923 subscribers

Course with minimum number of subscribers

In [102…
```python
# Courses with 0 subscribers in each subject
# no. of Courses with 0 subscribers
couses_no_subscribers = df.loc[df.num_subscribers == 0]
print('No. of courses with 0 subscribers: '+str(couses_no_subscribers.num_subscribers.value_counts()[0]))
data_nouse_courses = couses_no_subscribers.subject.value_counts().reset_index()
plt.pie(data_nouse_courses['subject'], labels = data_nouse_courses['index'])
```

```
No. of courses with 0 subscribers: 70
```

Out[102…
```
([<matplotlib.patches.Wedge at 0x10cd89b8>,
  <matplotlib.patches.Wedge at 0x10cd8c28>,
  <matplotlib.patches.Wedge at 0x10cd8e68>],
 [Text(-0.24477311341532162, 1.0724206837565984, 'Business Finance'),
  Text(-0.2926403145482764, -1.060359206260353, 'Graphic Design'),
  Text(0.9686552223892718, -0.5212552734869842, 'Musical Instruments')])
```



- The minimum number of subscribers comes out to be 0.
- There are 70 such courses most of them from Business Finance. It can be noted that each Web development course has atleast one subscriber

Levels classification under subjects

In [45]:
```python
# How many levels do we have
df['level'].unique()
```

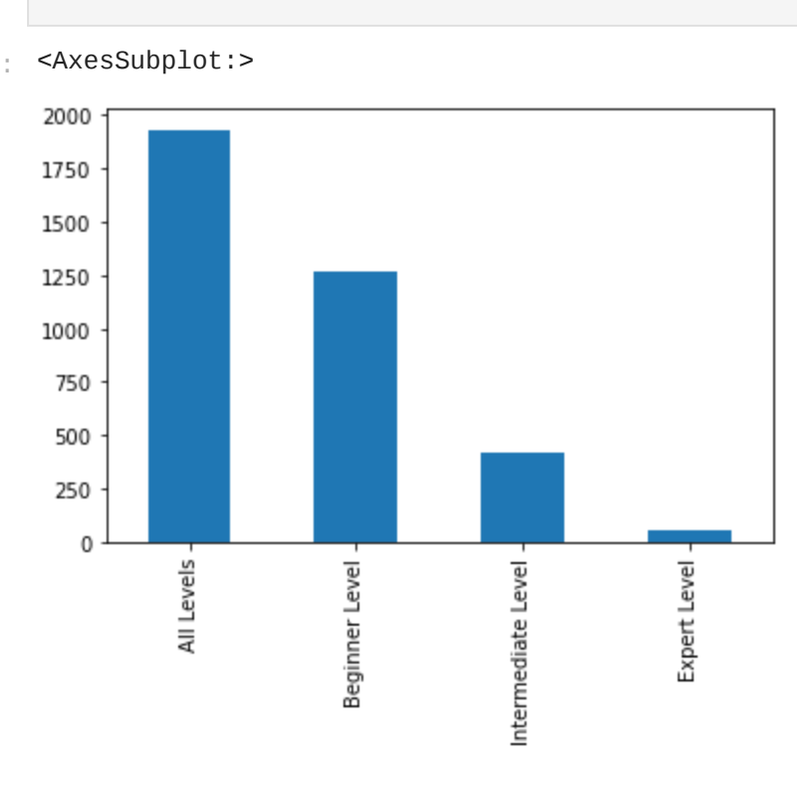
Out[45]:
```
array(['All Levels', 'Intermediate Level', 'Beginner Level',
       'Expert Level'], dtype=object)
```

- We have 4 different levels:
  - All Levels
  - Intermediate Levels
  - Beginner Levels
  - Expert Level

Number of courses per level

In [48]:
```python
#distribution of courses per level
df['level'].value_counts().plot(kind='bar')
```

Out[48]: <AxesSubplot:>



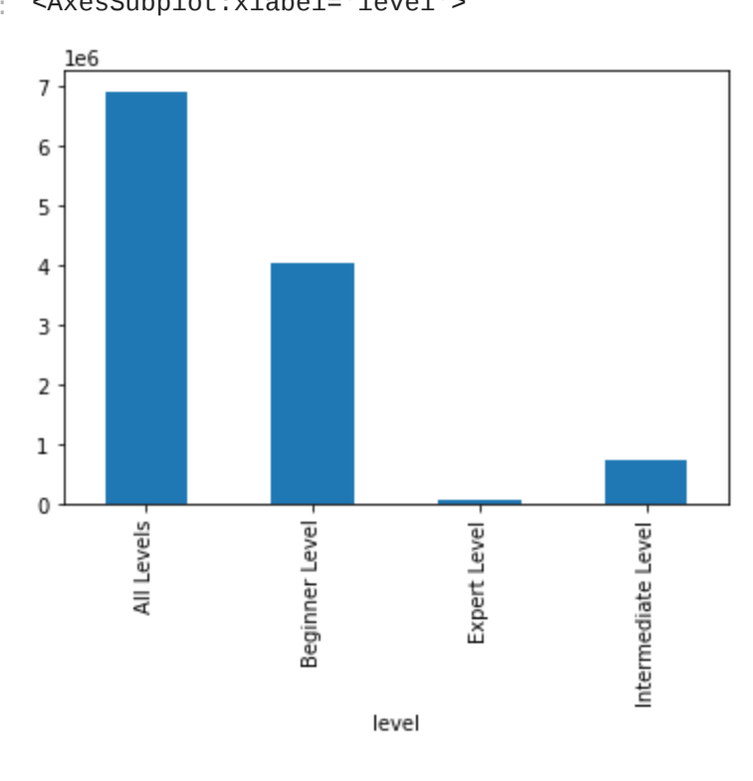
- Maximum number of courses lie under All Levels and there are minimum courses for Expert Level

Subscribers for each level

In [55]:
```python
# how many subscribers per level
print(df.groupby('level')['num_subscribers'].sum())
df.groupby('level')['num_subscribers'].sum().plot(kind = 'bar')
```

```
level
All Levels            6915076
Beginner Level        4051843
Expert Level            50196
Intermediate Level     742005
Name: num_subscribers, dtype: int64
```

Out[55]: <AxesSubplot:xlabel='level'>

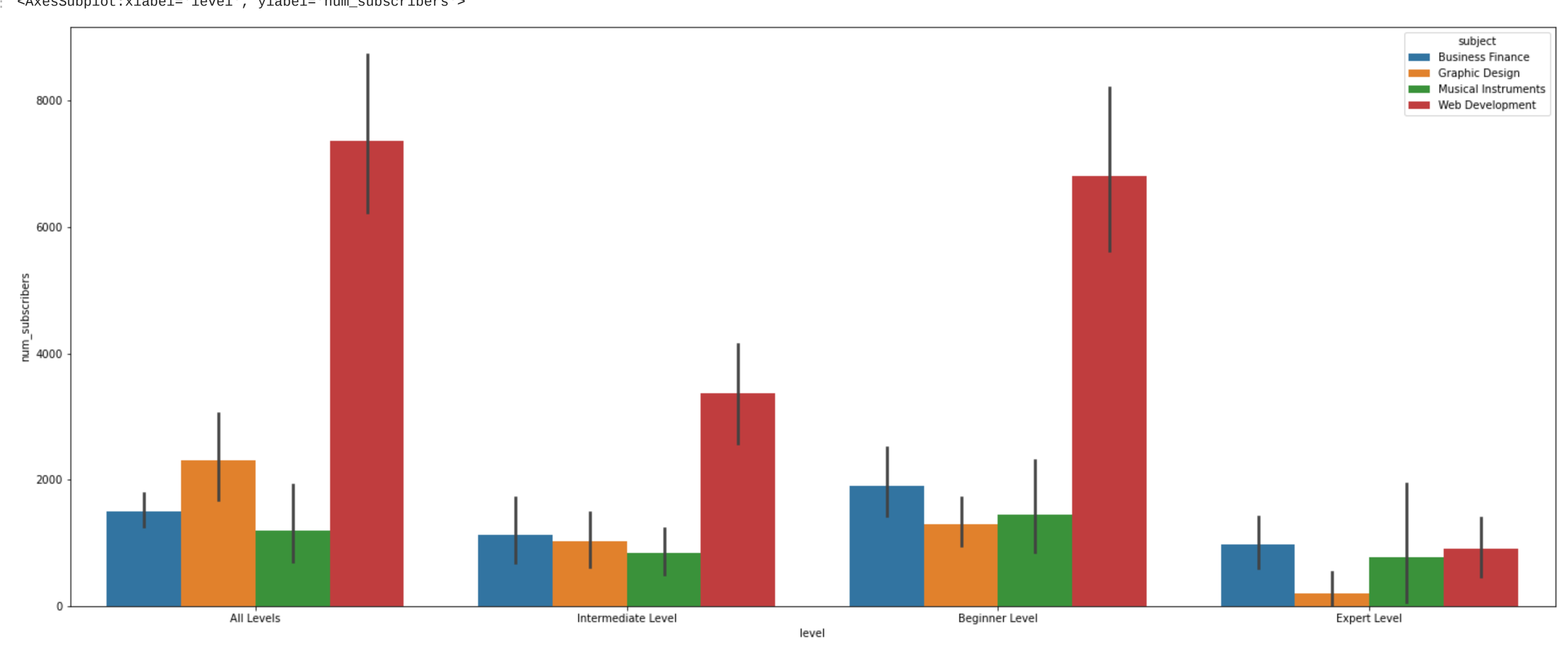


- Maximum number of courses are subscribed under All Level and the Expert Level courses are least subscribed

Subscribers for each subject under different levels

In [62]:
```python
# subscribers comparision per subject for each level
df.groupby('subject')['level'].value_counts()
plt.figure(figsize=(25,10))
sns.barplot(x='level', y='num_subscribers',hue='subject', data=df)
```

Out[62]: <AxesSubplot:xlabel='level', ylabel='num_subscribers'>



- Under all levels the Web Development courses are subscribed the most

Median Price

In [121…
```python
df['price'].median()
```

Out[121… 45.0

- The median price comes out to be $45

Total revenue collected

In [109…
```python
# total earning
df['profit']=df['price']*df['num_subscribers']
sum=df['profit'].sum()
sum
```

Out[109… 884921315

- The total profit comes out to be $884921315 till date for all the udemy courses purchased.

Highest profitable course

In [115…
```python
#highest profitable course
id=df['profit'].idxmax()
df.iloc[id]
```

Out[115…
```
course_id                                             625204
course_title                          The Web Developer Bootcamp
url                    https://www.udemy.com/the-web-developer-bootcamp/
is_paid                                                 True
price                                                    200
num_subscribers                                       121584
num_reviews                                            27445
num_lectures                                             342
level                                             All Levels
content_duration                                        43.0
published_timestamp                     2015-11-02T21:13:27Z
subject                                      Web Development
published_date                           2015-11-02 00:00:00
year                                                    2015
Engagement                                            149029
profit                                              24316800
Name: 3230, dtype: object
```

In [116…
```python
total_profit=df.loc[df['course_id']==625204]
print(total_profit['profit'])
```

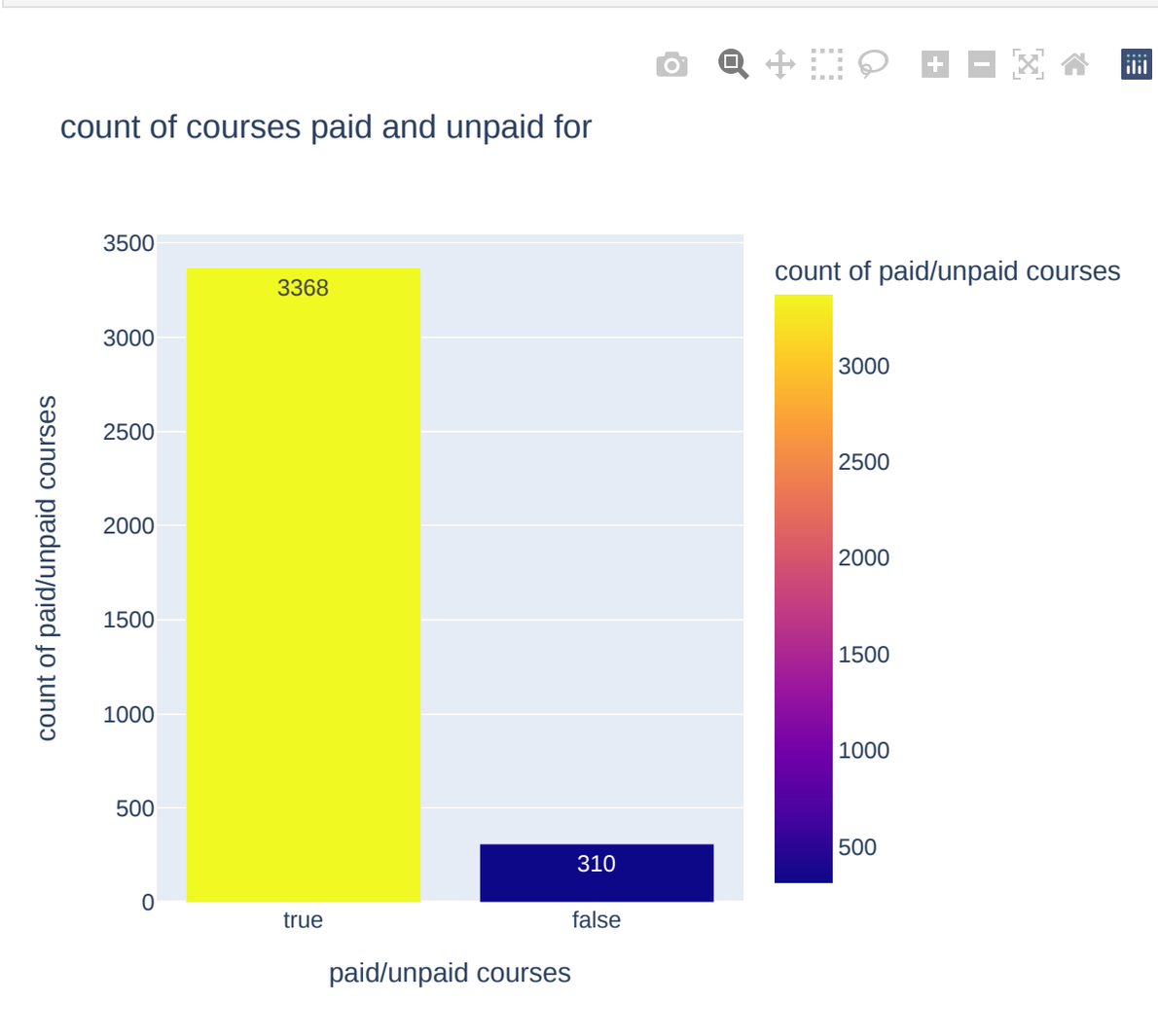
```
3230    24316800
Name: profit, dtype: int64
```

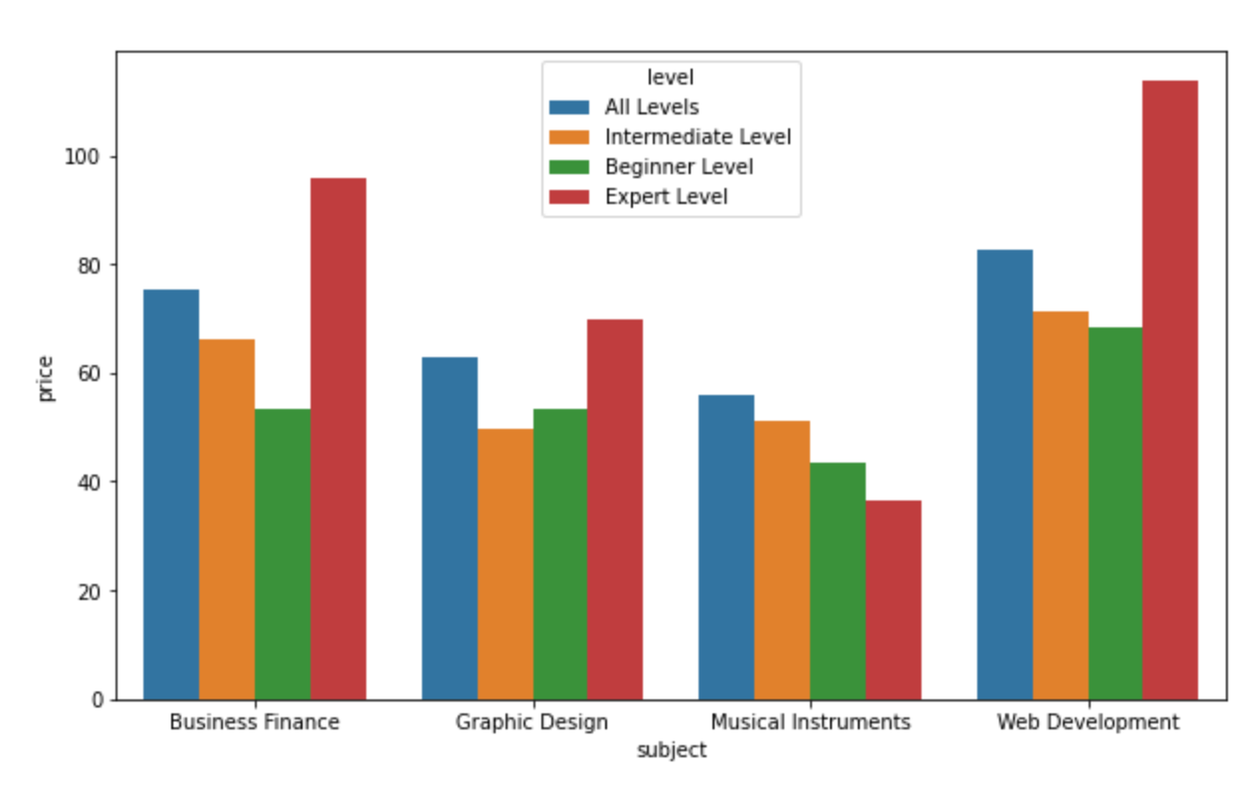
- The course which has contributed the most in total revenue is 'The Web Developer Bootcamp' and has brought a revenue of $24316800

Paid or free courses count

In [92]:
```python
# paid/free courses in each subject based on level
plt.figure(figsize=(10,6))
v6 = sns.barplot(x='subject',y='price',data = df ,hue = 'level',ci=False)

# count of paid/unpaid courses
price_list=df["is_paid"].unique()
price_count=df['is_paid'].value_counts().reset_index()
fig=px.bar(price_count, x='index', y='is_paid', text='is_paid', color='is_paid',
           title='count of courses paid and unpaid for',
           labels={'index':'paid/unpaid courses','is_paid':'count of paid/unpaid courses'})
fig.update_layout(showlegend=False, width=600)
fig.show()
```



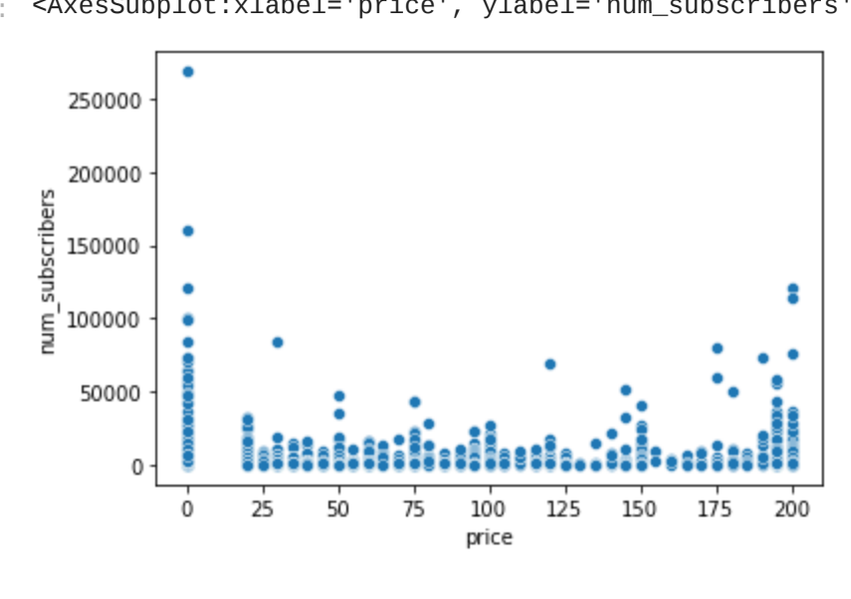


- Number of paid courses are 3368 and free courses are 310
- The Expert Level course in all the subjects leaving Musical Instruments subject are the most expensive ones.

Subscription dependency on price

In [96]:
```python
# price vs users
sns.scatterplot(data=df,x='price',y='num_subscribers')
```

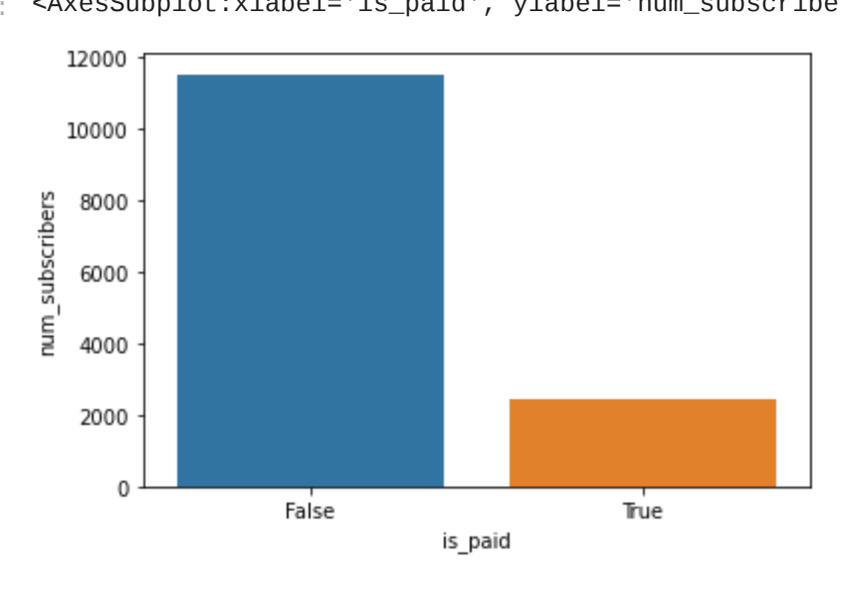
Out[96]: <AxesSubplot:xlabel='price', ylabel='num_subscribers'>



Subscribers dependencies on course is paid/free

In [9]:
```python
# subscription Vs paid
sns.barplot(x='is_paid', y='num_subscribers', data=df, ci=False)
```

Out[9]: <AxesSubplot:xlabel='is_paid', ylabel='num_subscribers'>

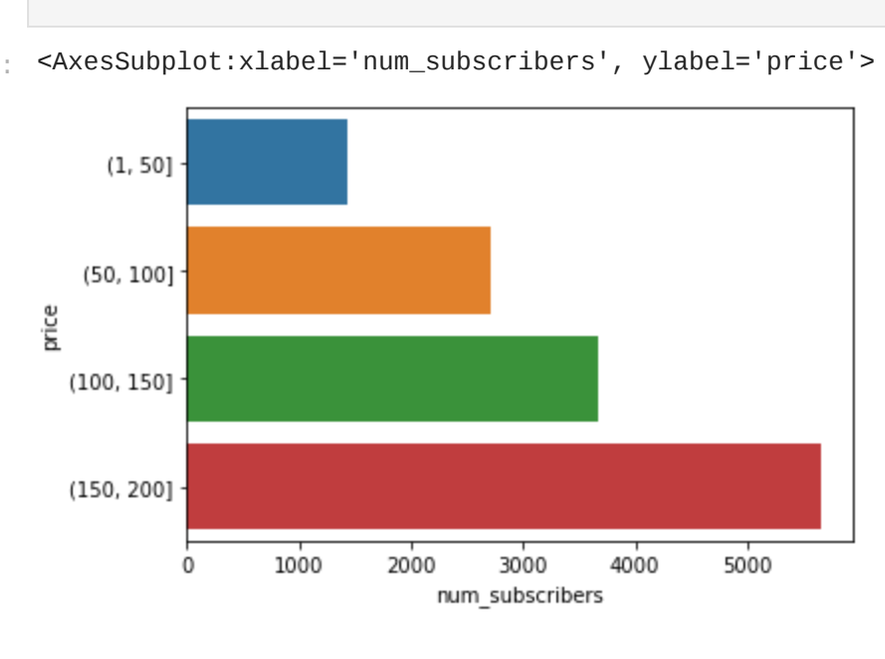


- Subscription is more for free courses rather than the paid ones

Subscriptions in different price ranges

In [36]:
```python
# Subscriptions Vs price
price=pd.cut(x=df['price'], bins=[1, 50, 100, 150, 200])
sns.barplot(y=price, x='num_subscribers', data=df, ci=False)
```

Out[36]: <AxesSubplot:xlabel='num_subscribers', ylabel='price'>

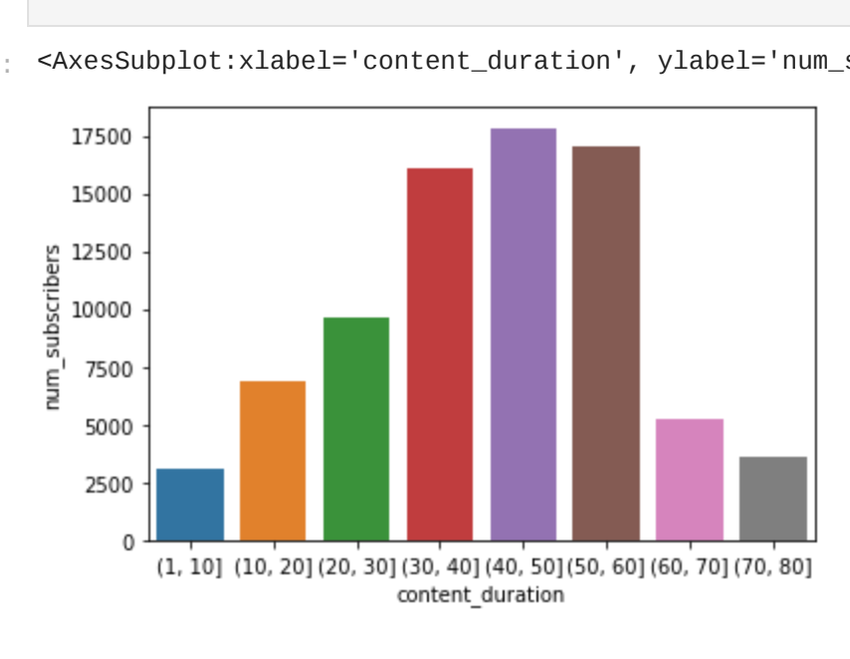


- Even though the number of subscribers opting for free courses was much higher than the paid ones, those who opted for the paid ones chose the course in range $150-200 the most expensive range.

Subscriptions depending on the duration of the course

In [27]:
```python
# Subscriptions Vs Duration
duration=pd.cut(x=df['content_duration'], bins=[1, 10, 20, 30, 40,50,60,70,80])
sns.barplot(x=duration, y='num_subscribers', data=df, ci=False)
```

Out[27]: <AxesSubplot:xlabel='content_duration', ylabel='num_subscribers'>

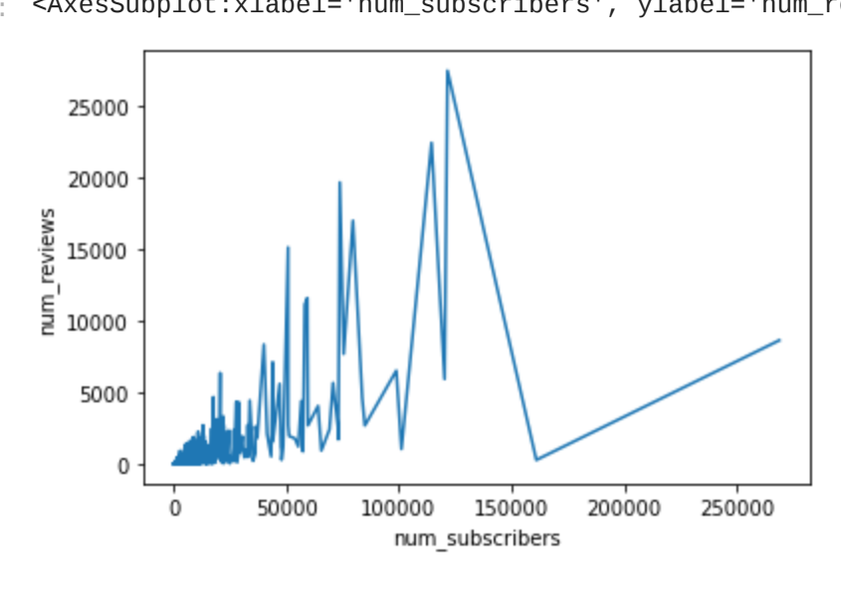


- Maximum number of subscribers lie in the range 30-60 hours range

Subscriptions depending on Reviews

In [38]:
```python
#Subscriptions Vs Reviews
sns.lineplot(y='num_reviews', x='num_subscribers', data=df, ci=False)
```

Out[38]: <AxesSubplot:xlabel='num_subscribers', ylabel='num_reviews'>

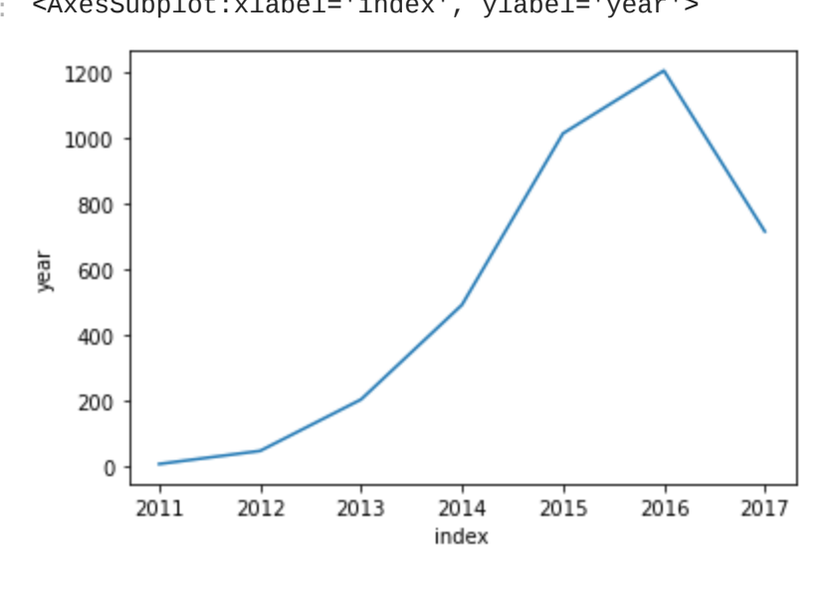


- No particular pattern can be seen here

Courses published per year

In [69]:
```python
#Publications per year
data_year_count = df['year'].value_counts().reset_index()
data_year_count
sns.lineplot(x = "index",y = "year", data = data_year_count)
```

Out[69]: <AxesSubplot:xlabel='index', ylabel='year'>



- Year 2011 to 2016 saw an increase in number of courses being published every year. But there was a sudden fall in 2017.

Most popular courses(depends upon number of subscribers)

In [71]:
```python
# Most Popular courses
df[['course_title', 'num_subscribers','subject']].sort_values('num_subscribers', ascending = False).head(5)
```

Out[71]:

	course_title	num_subscribers	subject
2827	Learn HTML5 Programming From Scratch	268923	Web Development
3032	Coding for Entrepreneurs Basic	161029	Web Development
3230	The Web Developer Bootcamp	121584	Web Development
2783	Build Your First Website in 1 Week with HTML5 ...	120291	Web Development
3232	The Complete Web Developer Course 2.0	114512	Web Development

- All of the courses belong to web development

Most engaging courses(depends on number of reviews and subscribers)

In [89]:
```python
# Most engaging Courses
df['Engagement'] = df['num_subscribers'] + df['num_reviews']
df[['course_title', 'Engagement','subject']].sort_values('Engagement', ascending = False).head(5)
```

Out[89]:

	course_title	Engagement	subject
2827	Learn HTML5 Programming From Scratch	277552	Web Development
3032	Coding for Entrepreneurs Basic	161308	Web Development
3230	The Web Developer Bootcamp	149029	Web Development
3232	The Complete Web Developer Course 2.0	136924	Web Development
2783	Build Your First Website in 1 Week with HTML5 ...	126215	Web Development

- The most engaging course belongs to Web Development.

Best cost beneficial course

In [91]:
```python
# What courses offer the best cost benefit
# we can determine this by finding the course with the least amount paid, but has the highest number of subscribers.
cost_benefit = df[(df['price']<=df['price'].mean()) &
(df['num_subscribers']>=df['num_subscribers'].mean())].sort_values(('num_subscribers'),ascending=False)['course_title'].head(1).unique()
print("The course which offers the best cost benefit is : \n", cost_benefit)
```

```
The course which offers the best cost benefit is :
 ['Learn HTML5 Programming From Scratch']
```

- 'Learn HTML5 Programming From Scratch' comes out to be the most beneficial course.

In [130…
```python
jupyter nbconvert --execute --to pdf Udemy_analysis.ipynb
```

```
  File "<ipython-input-130-6e4012f6b3ce>", line 1
    jupyter nbconvert --execute --to pdf Udemy_analysis.ipynb
            ^
SyntaxError: invalid syntax
```