Naukri.com job sample data analysis and visualization

Problem: Anticipate job market demand, (forecast)

solution : lets tackle this promblem , i.e forecasting the job market demand by analysis of job sample of naukri.com

project :

Major analysis we are going to do with the naukri.comjobsample data are :

1.Company's analysis

2.Industry's analysis

3.Jobtitle's analysis

4.Skill's analysis

5.Pay Rate's analysis i.e MIN and MAX Pay Rates and we will add this columns to our updated_data

6.Experience analysis i.e MIN and MAX Experience and we will add this columns to our updated_data

5.1 now the updated data contains 3 extra columns i.e minimum,maximum,average pay

6.1 now the updated data contains 3 extra columns i.e minimum,maximum,average experience

7.Experience and Pay_Rates

7.1) co-relation between min_experience and min_pay

a) seaborn stripplots

b) seaborn pointplots

7.2) co-relation between max_experience and max_pay

a) seaborn stripplots

b) seaborn pointplots

8. MINIMUM,MAXIMUM EXPERIENCE AND MINIMUM ,MAXIMUM PAY

8.1) co-relation between MIN_experience,MAX_experience and MIN_pay

8.2) co-relation between MIN_experience,MAX_experience and MAX_pay

9. co-relation between avg_experience and avg_pay

9.1) seaborn stripplots

9.2) seaborn pointplots

10. comparison between

10.1) MINIMUM PAY & INDUSTRIES

10.2) MAXIMUM PAY & INDUSTRIES

10.3) AVERAGE PAY AND SKILLS

10.4) AVERAGE PAY AND JOBTITLES

11. SUMMARY:

let's begin with importing the necessary libraries

In [2]:

```
import numpy as np #for algebric calculations
import pandas as pd #essential for data reading,writing etc
import seaborn as sns #visualization library
import matplotlib.pyplot as plt  #visualization library
%matplotlib inline
plt.rcParams['figure.figsize'] = (10, 7) #plotting parameters size's

import warnings
warnings.filterwarnings('ignore')
```

After successfully importing libraries , now let's import the data

In [3]:

```
data = pd.read_csv("naukri_com-job_sample.csv")
```

let's read the first 10 columns of our imported data by using a pandas function head(10)

In [4]:

```
data.head(10)
```

Out[4]:

	company	education	experience	industry	jobdescription	
0	MM Media Pvt Ltd	UG: B.Tech/B.E. - Any Specialization PG:Any Po...	0 - 1 yrs	Media / Entertainment / Internet	Job Description Send me Jobs like this Quali...	210516
1	find live infotech	UG: B.Tech/B.E. - Any Specialization PG:MBA/PG...	0 - 0 yrs	Advertising / PR / MR / Event Management	Job Description Send me Jobs like this Quali...	210516
2	Softtech Career Infosystem Pvt. Ltd	UG: Any Graduate - Any Specialization PG:Any P...	4 - 8 yrs	IT-Software / Software Services	Job Description Send me Jobs like this - as ...	101016
3	Onboard HRServices LLP	UG: Any Graduate - Any Specialization PG:CA Do...	11 - 15 yrs	Banking / Financial Services / Broking	Job Description Send me Jobs like this - Inv...	810169
4	Spire Technologies and Solutions Pvt. Ltd.	UG: B.Tech/B.E. - Any Specialization PG:Any Po...	6 - 8 yrs	IT-Software / Software Services	Job Description Send me Jobs like this Pleas...	120916
5	PFS Web Global Services Pvt Ltd	UG: B.Tech/B.E. - Any Specialization PG:MCA - ...	2 - 5 yrs	IT-Software / Software Services	Job Description Send me Jobs like this We ar...	131016
6	Kinesis Management Consultant Pvt. Ltd	NaN	1 - 3 yrs	IT-Software / Software Services	Job Description Send me Jobs like this exper...	131016
7	Agile HR consultancy Pvt. Ltd. hiring for Ross...	UG: Diploma - Any Specialization, Electrical, ...	2 - 7 yrs	Aviation / Aerospace Firms	Job Description Send me Jobs like this Job D...	121016
8	HANSUM INDIA ELECTRONICS PVT.LTD.	UG: Diploma - Any Specialization, Electronics/...	1 - 3 yrs	Industrial Products / Heavy Machinery	Job Description Send me Jobs like this Indep...	131016

	company	education	experience	industry	jobdescription	
9	Accenture	UG: Any Graduate - Any Specialization	1 - 5 yrs	IT-Software / Software Services	Job Description Send me Jobs like this	121016

now we have an idea regarding the features we have in the data

let's begin the analysis, as we will get more in-depth useful insights of the data

let's see what are the top 10 companies?

In [5]:

```
data['company'].value_counts().head(10)
```

Out[5]:

```
Indian Institute of Technology Bombay          403
Confidential                                   393
National Institute of Industrial Engineering   185
Oracle India Pvt. Ltd.                         151
JPMorgan Chase                                 135
Godrej Industries Ltd                          125
Unitforce technologies Pvt. Ltd.               100
Capgemini                                       98
HCL Technologies                                95
Axis Jobs                                       92
Name: company, dtype: int64
```

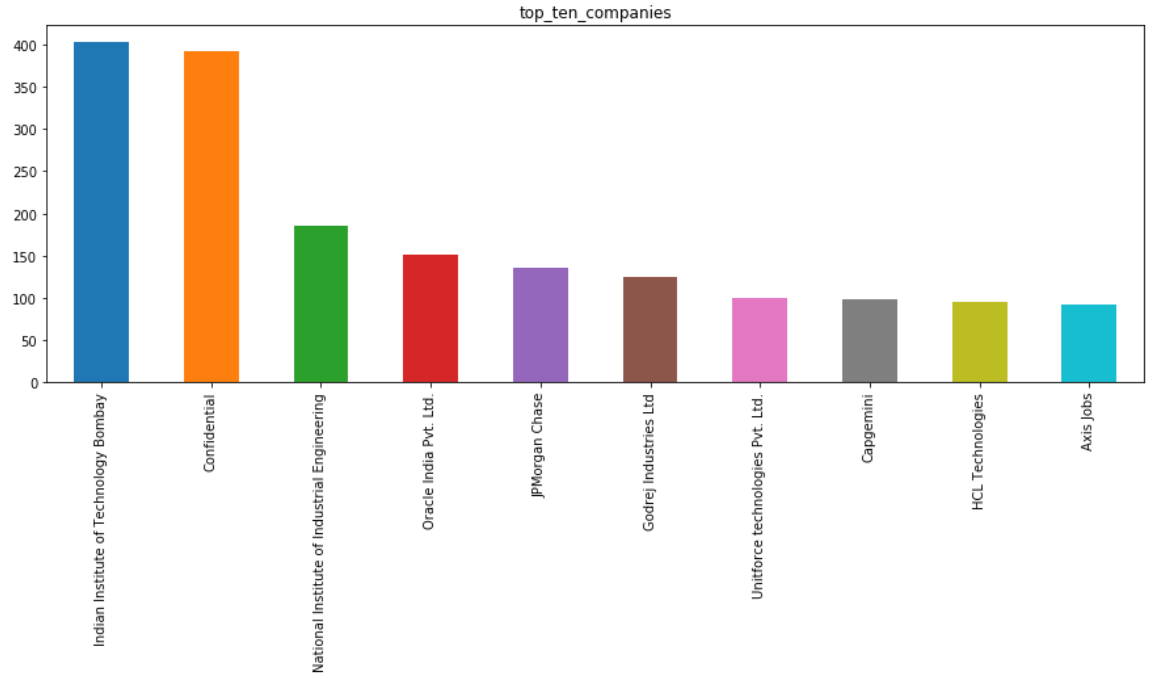
let's plot plot 10 companies for indepth analysis.

In [6]:

```
top_ten_companies=data['company'].value_counts().head(10)
f,ax=plt.subplots(figsize=(15,5))
top_ten_companies.plot(kind = 'bar')
plt.title('top_ten_companies')
```

Out[6]:

```
Text(0.5,1,'top_ten_companies')
```



it is clear that IITbombay is NO.1 among the top 1o company of our data

let's see what are the top 10 industries?

let's plot plot 10 industries for indepth insights.

In [7]:

```
data['industry'].value_counts().head(10)
```

Out[7]:

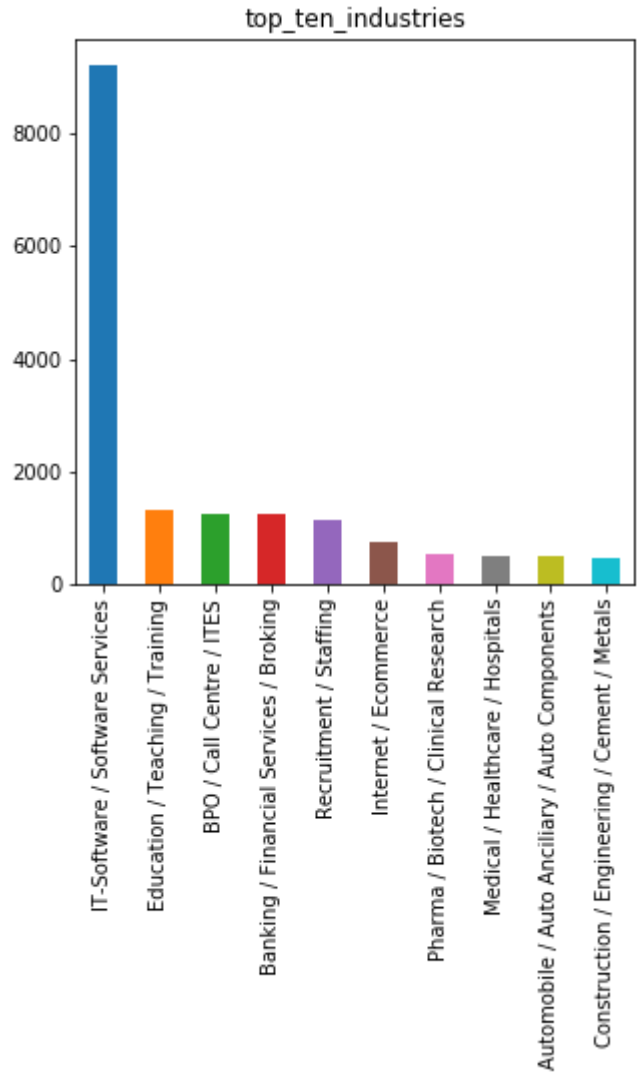
```
IT-Software / Software Services                  9216
Education / Teaching / Training                  1322
BPO / Call Centre / ITES                         1254
Banking / Financial Services / Broking           1238
Recruitment / Staffing                           1129
Internet / Ecommerce                              738
Pharma / Biotech / Clinical Research              525
Medical / Healthcare / Hospitals                  495
Automobile / Auto Anciliary / Auto Components     478
Construction / Engineering / Cement / Metals      449
Name: industry, dtype: int64
```

In [8]:

```
top_ten_industries = data['industry'].value_counts().head(10)
f,ax=plt.subplots(figsize=(5,5))
top_ten_industries.plot(kind='bar')
plt.title('top_ten_industries')
```

Out[8]:

```
Text(0.5,1,'top_ten_industries')
```



let's see what are the top 10 jobtitle's?

let's plot plot 10 jobtitle's for indepth insights.

In [9]:

```
data['jobtitle'].value_counts().head(10)
```

Out[9]:

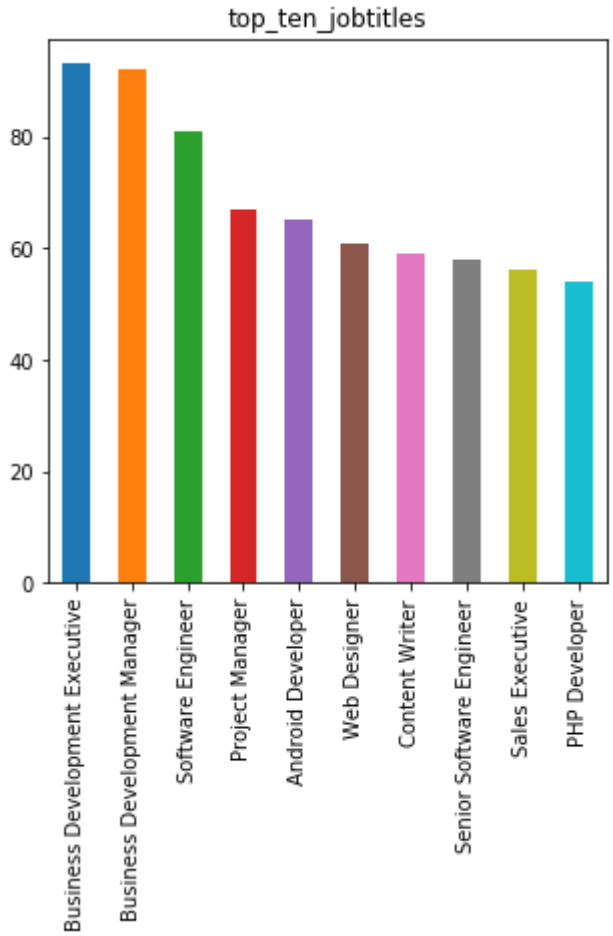
```
Business Development Executive    93
Business Development Manager      92
Software Engineer                 81
Project Manager                   67
Android Developer                 65
Web Designer                      61
Content Writer                    59
Senior Software Engineer          58
Sales Executive                   56
PHP Developer                     54
Name: jobtitle, dtype: int64
```

In [10]:

```
top_ten_jobtitles = data['jobtitle'].value_counts().head(10)
f,ax=plt.subplots(figsize=(5,5))
top_ten_jobtitles.plot(kind='bar')
plt.title('top_ten_jobtitles')
```

Out[10]:

```
Text(0.5,1,'top_ten_jobtitles')
```



let's see what are the top 10 skill's?

let's plot plot 10 skill's for indepth insights.

In [11]:

```
data['skills'].value_counts().head(10)
```

Out[11]:

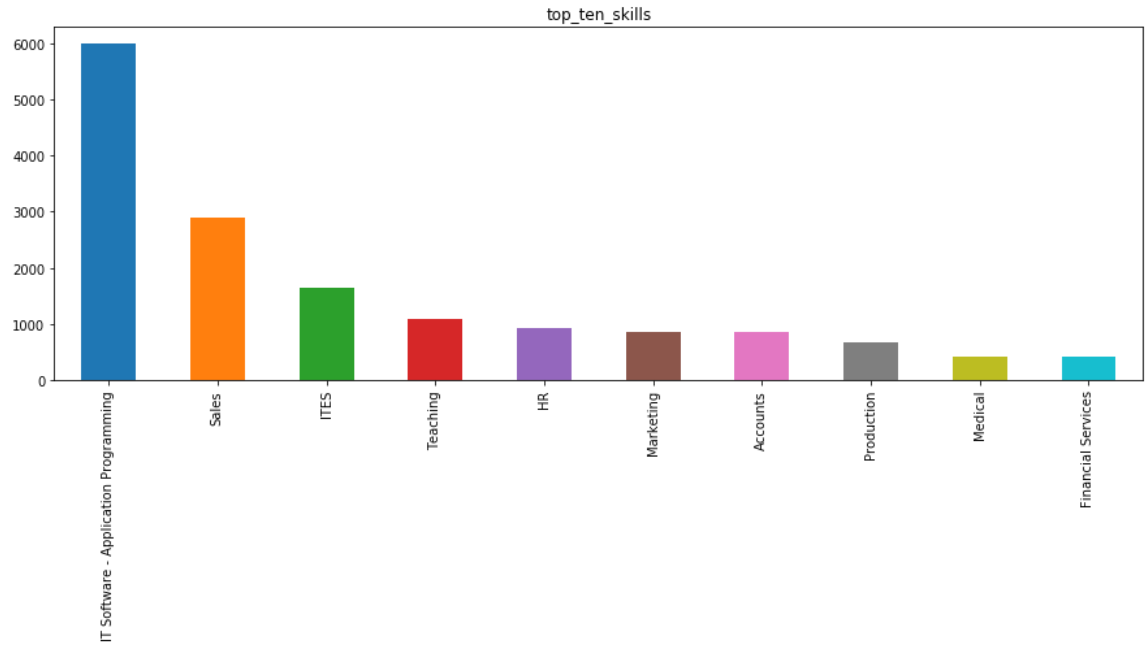
```
IT Software - Application Programming    5989
Sales                                    2893
ITES                                     1640
Teaching                                 1091
HR                                        928
Marketing                                 868
Accounts                                  860
Production                                667
Medical                                   418
Financial Services                        413
Name: skills, dtype: int64
```

In [12]:

```
top_ten_skills = data['skills'].value_counts().head(10)
f,ax=plt.subplots(figsize=(15,5))
top_ten_skills.plot(kind='bar')
plt.title('top_ten_skills')
```

Out[12]:

```
Text(0.5,1,'top_ten_skills')
```



on looking carefully the data given here contains the name of the old names of the cities along with some new names

this will cause error's in the analysis resulting in major mistakes

to avoid that , we will replace the old names with the new names and we will modify the data witgh the new names

In [13]:

```
replacements = {
   'joblocation_address': {
      r'(Bengaluru/Bangalore)': 'Bangalore',
      r'Bengaluru': 'Bangalore',
      r'Hyderabad / Secunderabad': 'Hyderabad',
      r'Mumbai , Mumbai': 'Mumbai',
      r'Noida': 'NCR',
      r'Delhi': 'NCR',
      r'Gurgaon': 'NCR',
      r'Delhi/NCR(National Capital Region)': 'NCR',
      r'Delhi , Delhi': 'NCR',
      r'Noida , Noida/Greater Noida': 'NCR',
      r'Ghaziabad': 'NCR',
      r'Delhi/NCR(National Capital Region) , Gurgaon': 'NCR',
      r'NCR , NCR': 'NCR',
      r'NCR/NCR(National Capital Region)': 'NCR',
      r'NCR , NCR/Greater NCR': 'NCR',
      r'NCR/NCR(National Capital Region) , NCR': 'NCR',
      r'NCR , NCR/NCR(National Capital Region)': 'NCR',
      r'Bangalore , Bangalore / Bangalore': 'Bangalore',
      r'Bangalore , karnataka': 'Bangalore',
      r'NCR/NCR(National Capital Region)': 'NCR',
      r'NCR/Greater NCR': 'NCR',
      r'NCR/NCR(National Capital Region) , NCR': 'NCR'

   }
}

data.replace(replacements, regex=True, inplace=True)
y = data['joblocation_address'].value_counts()
```

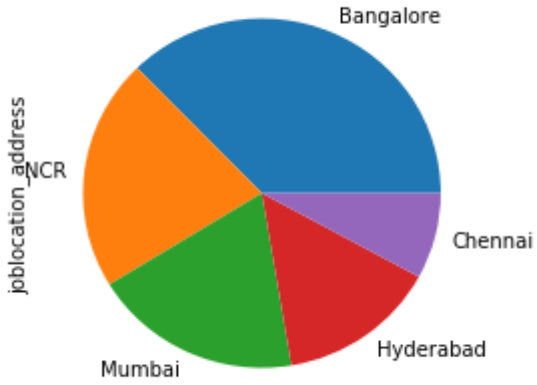
let's examine from where the more job's are coming ?

In [14]:

```
most_job_posting_city=data['joblocation_address'].value_counts().head()
f ,ax=plt.subplots(figsize=(4,4))
most_job_posting_city.plot(kind = 'pie')
```

Out[14]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x131c69f9f60>
```



it's clear now , that BENGALURU holds the first place in more joblocation.

thats a pretty good insights regading the

1.top10companies

2.top10industries

3.top10skills

4.jobtitles

5.top city for joblocations

lets examine the payrates:)

In [15]:

```
pay_split = data['payrate'].str[1:-1].str.split('-', expand=True)
pay_split.head()
```

Out[15]:

	0	1	2	3	4	5	6
0	,50,000	2,25,000 P.	None	None	None	None	None
1	,50,000	2,50,000 P.A. 2000	None	None	None	None	None
2	ot Disclosed by Recruite	None	None	None	None	None	None
3	ot Disclosed by Recruite	None	None	None	None	None	None
4	ot Disclosed by Recruite	None	None	None	None	None	None

In [16]:

```
#let's clean the payrates more

#remove space in left and right
pay_split[0] =  pay_split[0].str.strip()
#remove comma
pay_split[0] = pay_split[0].str.replace(',', '')
#remove all character in two condition
# 1 remove if only character
# 2 if start in number remove after all character
pay_split[0] = pay_split[0].str.replace(r'\D.*', '')
#display
pay_split[0].head()
```

Out[16]:

```
0    50000
1    50000
2
3
4
Name: 0, dtype: object
```

In [17]:

```
#remove space in left and right
pay_split[1] = pay_split[1].str.strip()
#remove comma
pay_split[1] = pay_split[1].str.replace(',', '')
#remove all character in two condition
# 1 remove if only character
# 2 if start in number remove after all character
pay_split[1] = pay_split[1].str.replace(r'\D.*', '')
#display
pay_split[1].head()
```

Out[17]:

```
0    225000
1    250000
2      None
3      None
4      None
Name: 1, dtype: object
```

In [18]:

```
pay_split[0] = pd.to_numeric(pay_split[0], errors='coerce')
pay_split[1] = pd.to_numeric(pay_split[1], errors='coerce')
```

that's the end of massive cleaning part , now its time to concatenate the new results

In [19]:

```
pay=pd.concat([pay_split[0], pay_split[1]], axis=1, sort=False)
```

hurrey! we done it , lets name the new borns:)

In [20]:

```
pay.rename(columns={0:'min_pay', 1:'max_pay'}, inplace=True)
pay.head()
```

Out[20]:

	min_pay	max_pay
0	50000.0	225000.0
1	50000.0	250000.0
2	NaN	NaN
3	NaN	NaN
4	NaN	NaN

the big work, now lets add this new bornone's to the data

In [21]:

```
#sometimes while testing , we may get multiple min_pay,max_pay columns as we are using
 conaction function for previous data to new data
#in such cases , we can use following code for removal of extra min-pay,max_pay columns
#data = data.drop("min_pay",axis = 1)
#data = data.drop("max_pay",axis = 1)
#data.head()

data=pd.concat([data, pay], axis=1, sort=False)

data.head()
```

Out[21]:

	company	education	experience	industry	jobdescription	job
0	MM Media Pvt Ltd	UG: B.Tech/B.E. - Any Specialization PG:Any Po...	0 - 1 yrs	Media / Entertainment / Internet	Job Description Send me Jobs like this Quali...	21051600226
1	find live infotech	UG: B.Tech/B.E. - Any Specialization PG:MBA/PG...	0 - 0 yrs	Advertising / PR / MR / Event Management	Job Description Send me Jobs like this Quali...	21051600239
2	Softtech Career Infosystem Pvt. Ltd	UG: Any Graduate - Any Specialization PG:Any P...	4 - 8 yrs	IT-Software / Software Services	Job Description Send me Jobs like this - as ...	10101690053
3	Onboard HRServices LLP	UG: Any Graduate - Any Specialization PG:CA Do...	11 - 15 yrs	Banking / Financial Services / Broking	Job Description Send me Jobs like this - Inv...	81016900536
4	Spire Technologies and Solutions Pvt. Ltd.	UG: B.Tech/B.E. - Any Specialization PG:Any Po...	6 - 8 yrs	IT-Software / Software Services	Job Description Send me Jobs like this Pleas...	12091600212

In [22]:

```
experience_split = data['experience'].str[0:-1].str.split('-', expand=True)
experience_split.head()
```

Out[22]:

	0	1	2
0	0	1 yr	None
1	0	0 yr	None
2	4	8 yr	None
3	11	15 yr	None
4	6	8 yr	None

In [23]:

```
#let's clean the experience more

#remove space in left and right
experience_split[1] =  experience_split[1].str.strip()
#remove comma
experience_split[1] = experience_split[1].str.replace('yr', '')
#remove all character in two condition
# 1 remove if only character
# 2 if start in number remove after all character
experience_split[1] = experience_split[1].str.replace(r'yr', '')
#display
experience_split[1].head()
```

Out[23]:

```
0     1
1     0
2     8
3    15
4     8
Name: 1, dtype: object
```

In [24]:

```
experience_split[0] = pd.to_numeric(experience_split[0], errors='coerce')
experience_split[1] = pd.to_numeric(experience_split[1], errors='coerce')
```

In [25]:

```
experience=pd.concat([experience_split[0], experience_split[1]], axis=1, sort=False)
```

In [26]:

```
experience.rename(columns={0:'min_experience', 1:'max_experience'}, inplace=True)
experience.head()
```

Out[26]:

	min_experience	max_experience
0	0.0	1.0
1	0.0	0.0
2	4.0	8.0
3	11.0	15.0
4	6.0	8.0

In [27]:

```
data=pd.concat([data, experience], axis=1, sort=False)
data.head()
```

Out[27]:

	company	education	experience	industry	jobdescription	job
0	MM Media Pvt Ltd	UG: B.Tech/B.E. - Any Specialization PG:Any Po...	0 - 1 yrs	Media / Entertainment / Internet	Job Description Send me Jobs like this Quali...	21051600226
1	find live infotech	UG: B.Tech/B.E. - Any Specialization PG:MBA/PG...	0 - 0 yrs	Advertising / PR / MR / Event Management	Job Description Send me Jobs like this Quali...	21051600239
2	Softtech Career Infosystem Pvt. Ltd	UG: Any Graduate - Any Specialization PG:Any P...	4 - 8 yrs	IT-Software / Software Services	Job Description Send me Jobs like this - as ...	10101690053
3	Onboard HRServices LLP	UG: Any Graduate - Any Specialization PG:CA Do...	11 - 15 yrs	Banking / Financial Services / Broking	Job Description Send me Jobs like this - Inv...	81016900536
4	Spire Technologies and Solutions Pvt. Ltd.	UG: B.Tech/B.E. - Any Specialization PG:Any Po...	6 - 8 yrs	IT-Software / Software Services	Job Description Send me Jobs like this Pleas...	12091600212

5.1 & 6.1

Now it's time to add some average values of payrate's and experience , giving us more deeper insights

In [28]:

```
data['avg_pay']=(data['min_pay'].values + data['max_pay'].values)/2
data['avg_experience']=(data['min_experience'].values + data['max_experience'].values)/
2
data.head()
```

Out[28]:

	company	education	experience	industry	jobdescription	job
0	MM Media Pvt Ltd	UG: B.Tech/B.E. - Any Specialization PG:Any Po...	0 - 1 yrs	Media / Entertainment / Internet	Job Description Send me Jobs like this Quali...	21051600226
1	find live infotech	UG: B.Tech/B.E. - Any Specialization PG:MBA/PG...	0 - 0 yrs	Advertising / PR / MR / Event Management	Job Description Send me Jobs like this Quali...	21051600239
2	Softtech Career Infosystem Pvt. Ltd	UG: Any Graduate - Any Specialization PG:Any P...	4 - 8 yrs	IT-Software / Software Services	Job Description Send me Jobs like this - as ...	10101690053
3	Onboard HRServices LLP	UG: Any Graduate - Any Specialization PG:CA Do...	11 - 15 yrs	Banking / Financial Services / Broking	Job Description Send me Jobs like this - Inv...	81016900536
4	Spire Technologies and Solutions Pvt. Ltd.	UG: B.Tech/B.E. - Any Specialization PG:Any Po...	6 - 8 yrs	IT-Software / Software Services	Job Description Send me Jobs like this Pleas...	12091600212

7. Experience and Pay_Rates

Relation between experience and pay_rates

this will give us an idea how your experience impacts your pay_rates

7.1) co-relation between min_experience and min_pay

let's get the insights by using

A)seaborn stripplots

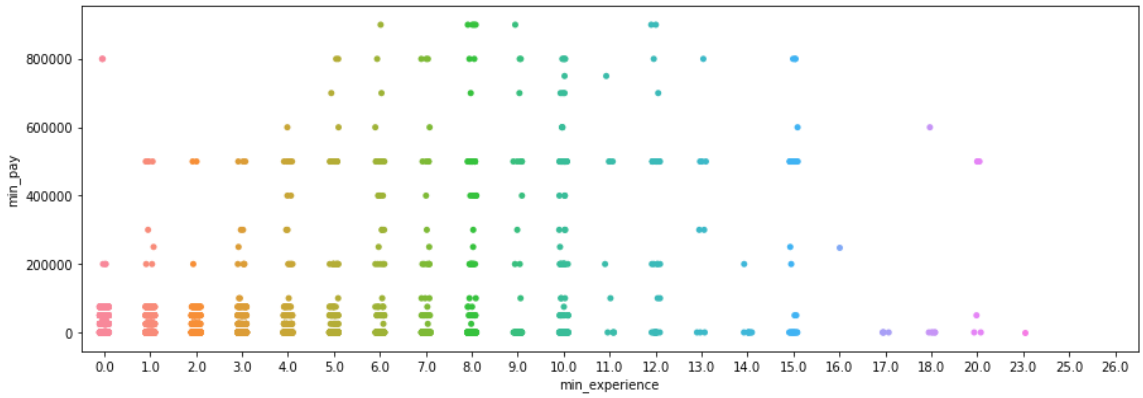
In [29]:

```
f,ax=plt.subplots(figsize=(15,5))

sns.stripplot(x='min_experience', y='min_pay', data=data, jitter=True)
```

Out[29]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x131c74443c8>
```



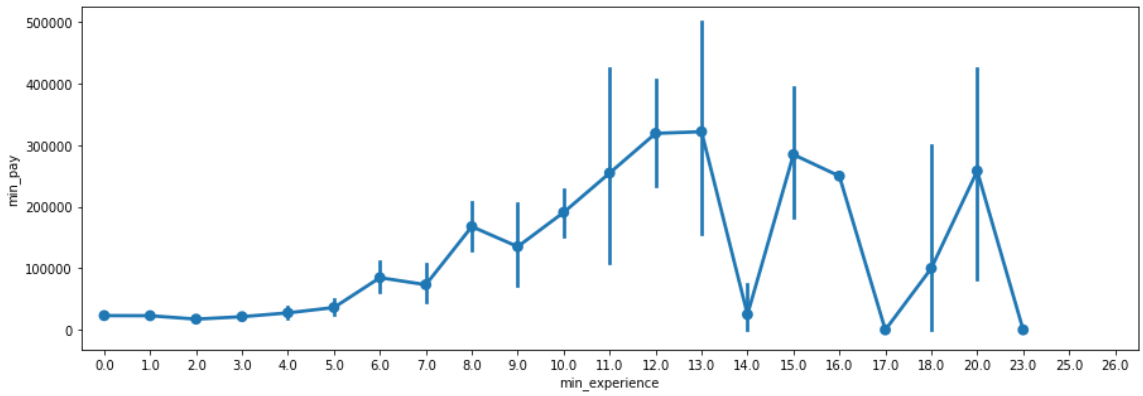
B)seaborn pointplots

In [30]:

```
f,ax=plt.subplots(figsize=(15,5))
sns.pointplot(x='min_experience', y='min_pay', data=data)
```

Out[30]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x131c723ae48>
```



7.2) co-relation between max_experience and max_pay

let's get the insights by using

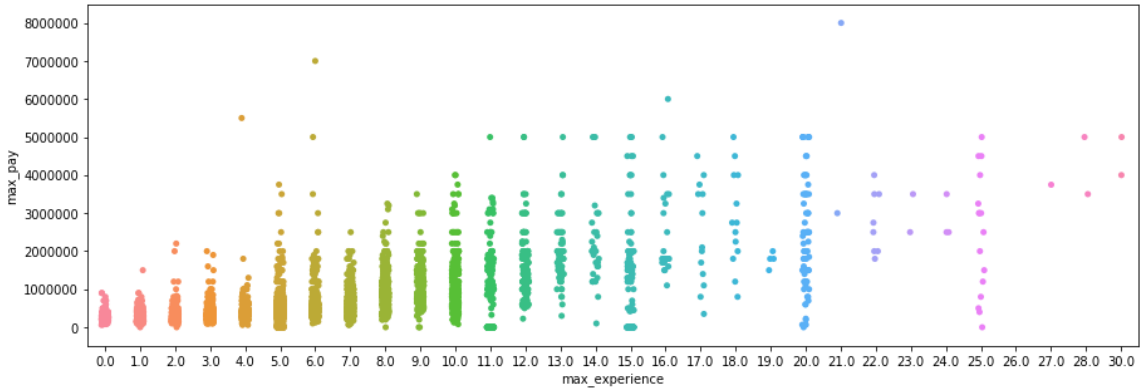
A)seaborn stripplots

In [31]:

```
f,ax=plt.subplots(figsize=(15,5))
sns.stripplot(x='max_experience', y='max_pay', data=data, jitter=True)
```

Out[31]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x131c7899400>
```



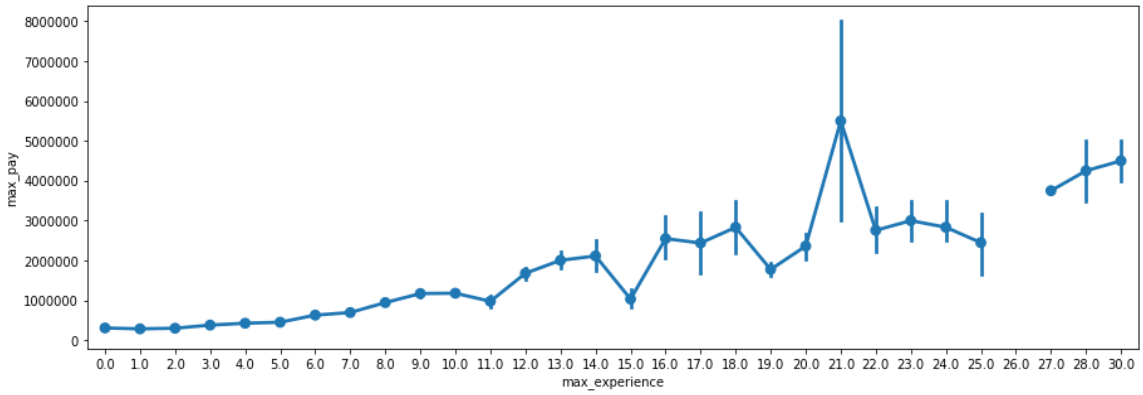
B) seaborn pointplots

In [32]:

```
f,ax=plt.subplots(figsize=(15,5))
sns.pointplot(x='max_experience', y='max_pay', data=data)
```

Out[32]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x131ca980908>
```



8) MINIMUM,MAXIMUM experience and MINIMUM,MAXIMUM pay

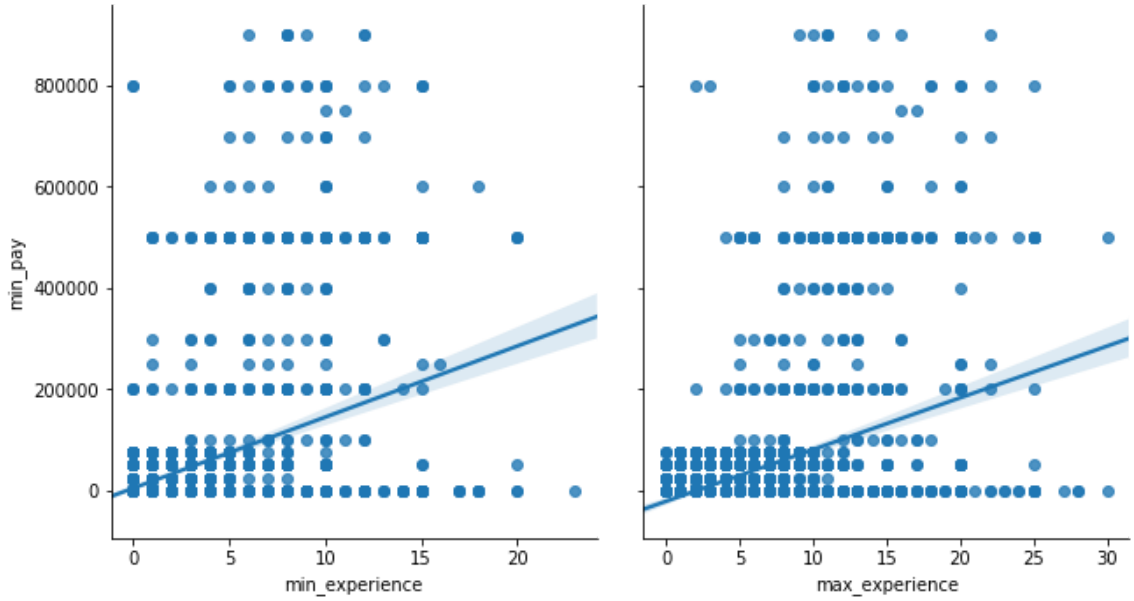
8.1) Relation between MIN,MAX experience and MIN pay

In [33]:

```
sns.pairplot(data,
             size=5, aspect=0.9,
             x_vars=["min_experience","max_experience"],
             y_vars=["min_pay"],
             kind="reg")
```

Out[33]:

```
<seaborn.axisgrid.PairGrid at 0x131c78f4cc0>
```



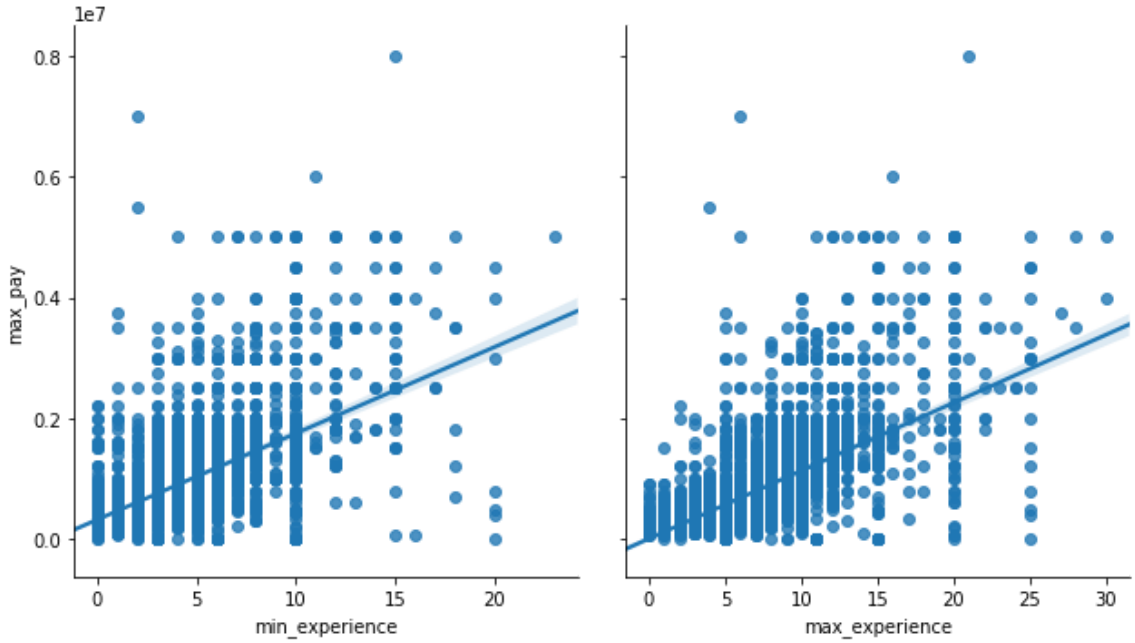
8.2) Relation between MIN,MAX experience and MAX pay

In [34]:

```
sns.pairplot(data,
             size=5, aspect=0.9,
             x_vars=["min_experience","max_experience"],
             y_vars=["max_pay"],
             kind="reg")
```

Out[34]:

```
<seaborn.axisgrid.PairGrid at 0x131c757c7b8>
```



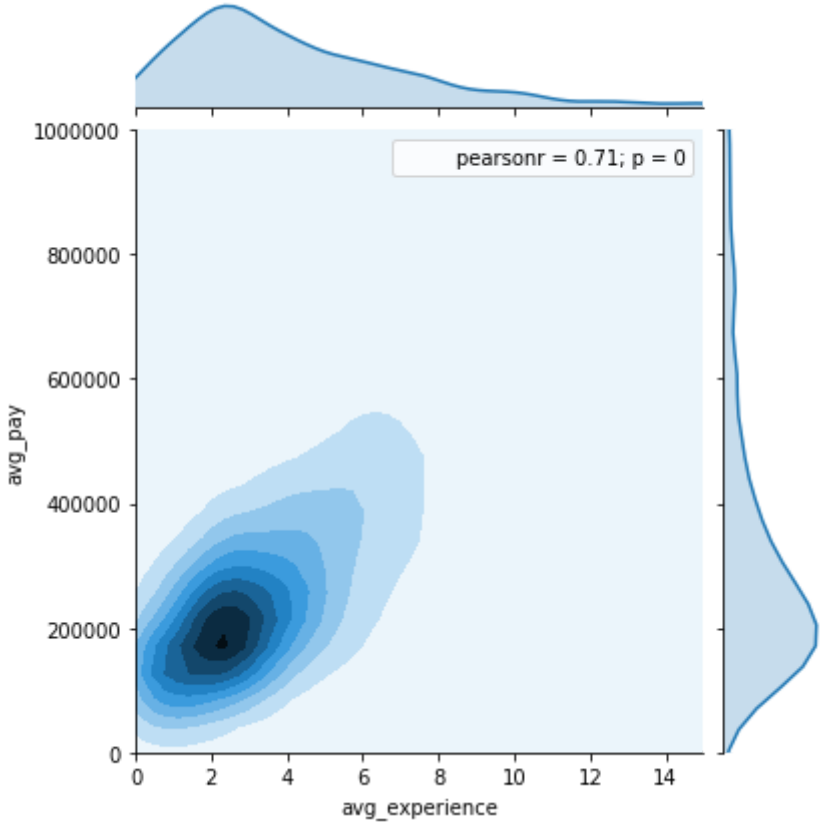
9) Relation between AVERAGE experience and AVERAGE pay_rates

In [35]:

```
sns.jointplot(x='avg_experience', y='avg_pay', data=data,
              kind="kde",xlim={0,15}, ylim={0,1000000})
```

Out[35]:

```
<seaborn.axisgrid.JointGrid at 0x131c731bcc0>
```



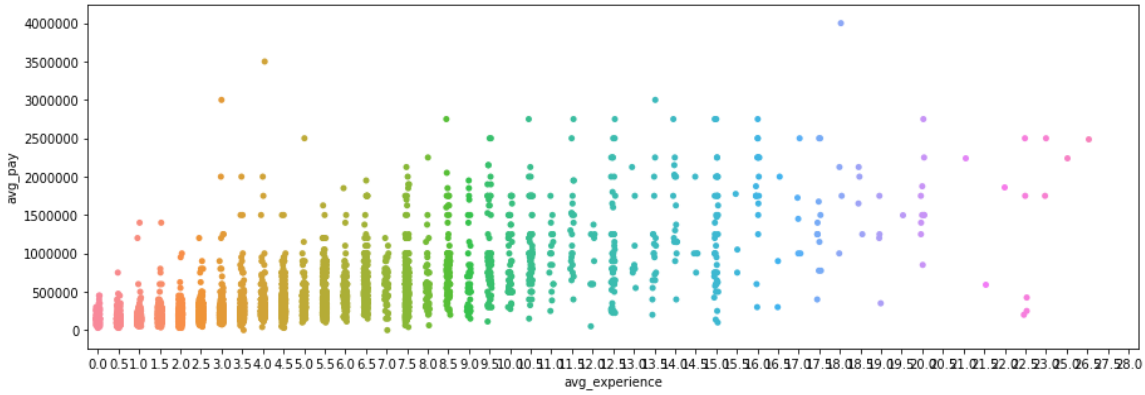
9.1) co-relation between avg_experience and avg_pay by using seaborn.stripplots

In [36]:

```
f,ax=plt.subplots(figsize=(15,5))
sns.stripplot(x='avg_experience', y='avg_pay', data=data, jitter=True)
```

Out[36]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x131c6e0d3c8>
```



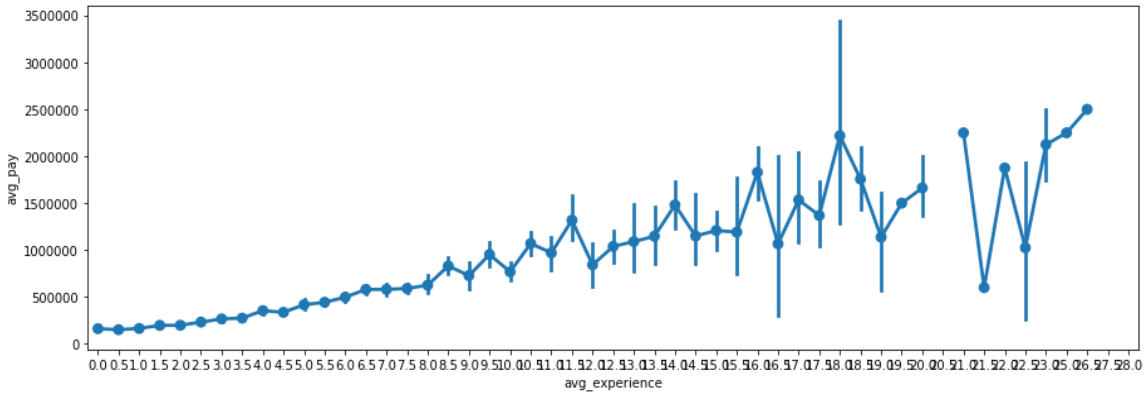
9.2) co-relation between avg_experience and avg_pay by using seaborn.pointplots

In [37]:

```
f,ax=plt.subplots(figsize=(15,5))
sns.pointplot(x='avg_experience', y='avg_pay', data=data)
```

Out[37]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x131cace8588>
```



10. comparison between

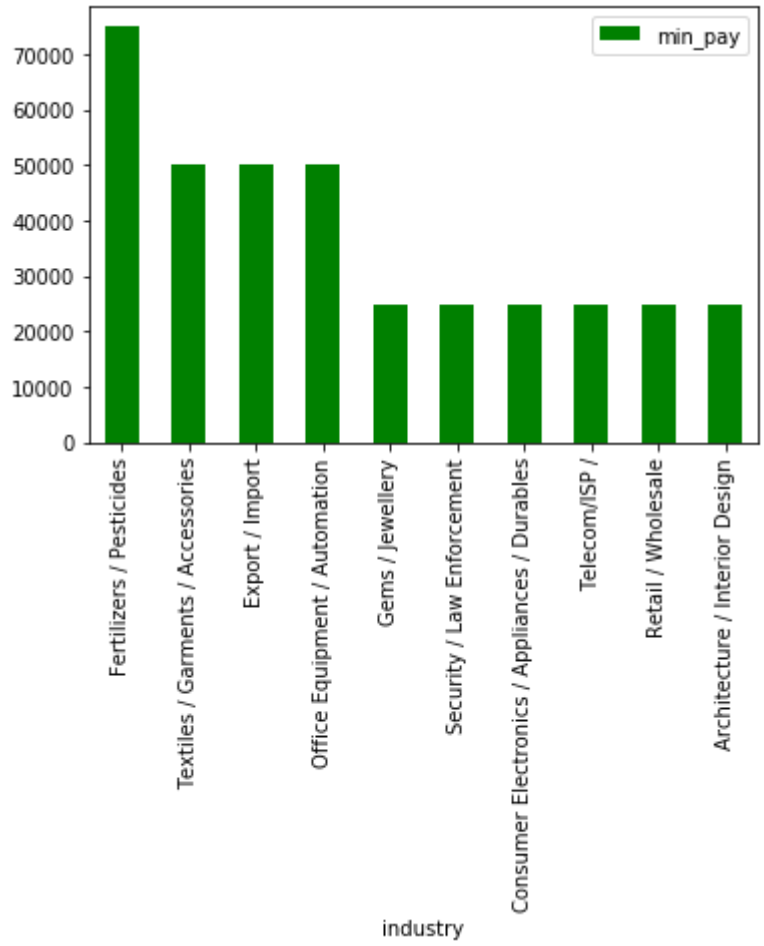
10.1) MINIMUM PAY & INDUSTRIES

In [38]:

```
data[['min_pay','industry']].groupby(["industry"]).median().sort_values(by='min_pay',as
cending=False).head(10).plot.bar(color='green')
```

Out[38]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x131cadfc860>
```



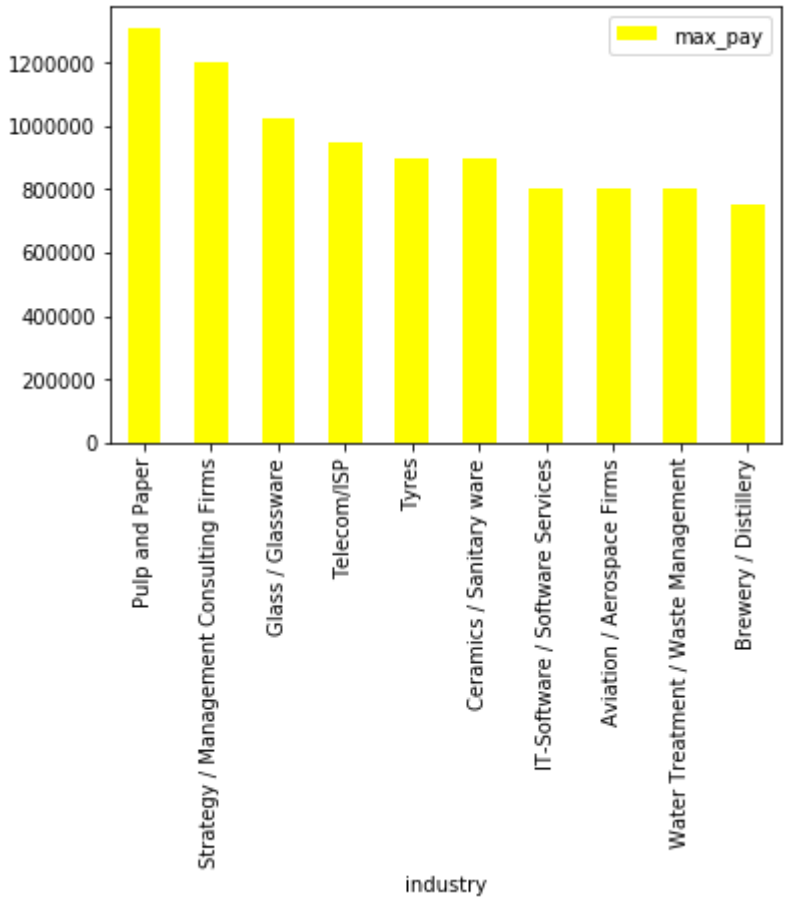
10.2) MAXIMUM PAY & INDUSTRIES

In [39]:

```
data[['max_pay','industry']].groupby(["industry"]).median().sort_values(by='max_pay',ascending=False).head(10).plot.bar(color='yellow')
```

Out[39]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x131cc1a6a58>
```



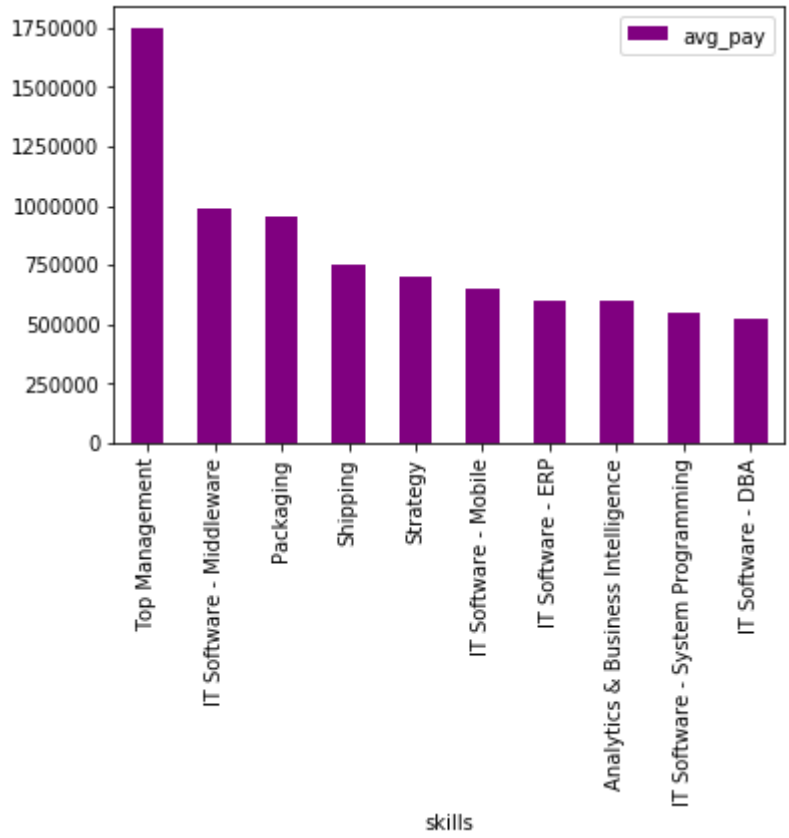
10.3) AVERAGE PAY AND SKILLS

In [40]:

```
data[['avg_pay','skills']].groupby(["skills"]).median().sort_values(by='avg_pay',ascending=False).head(10).plot.bar(color='purple')
```

Out[40]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x131cc22c7b8>
```



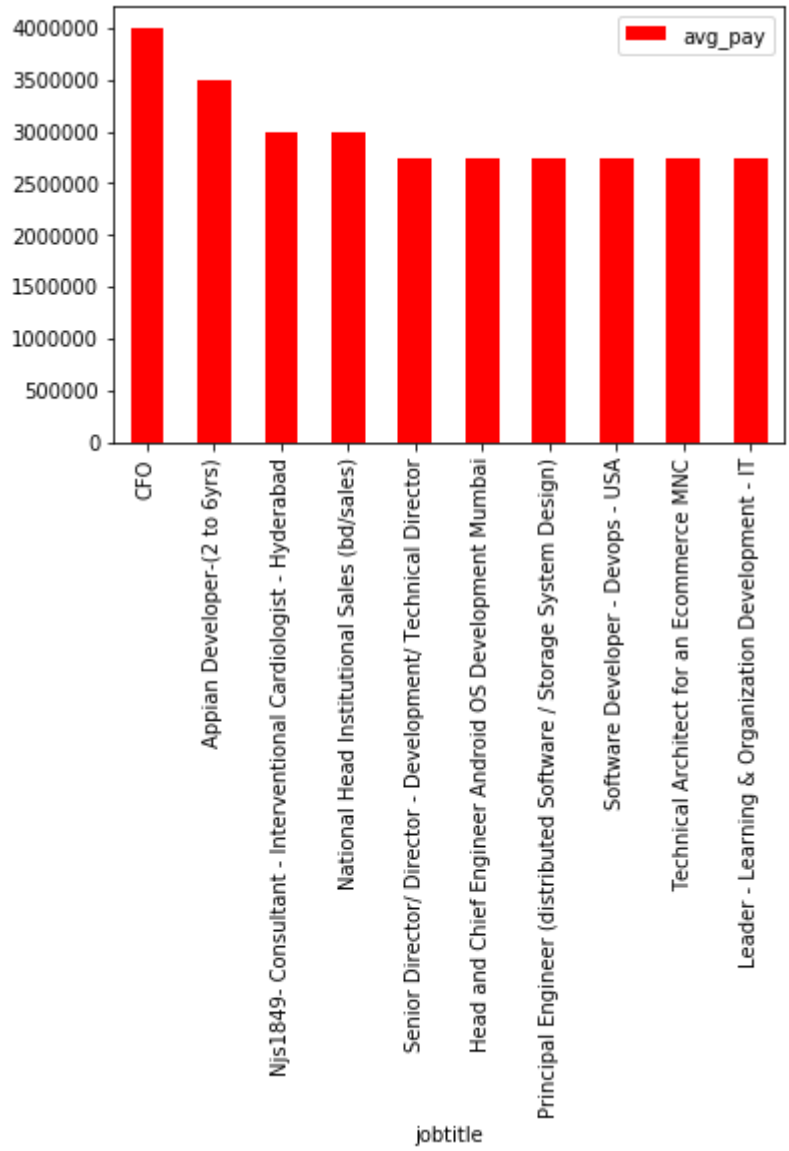
10.4) AVERAGE PAY AND JOBTITLES

In [41]:

```
data[['avg_pay','jobtitle']].groupby(["jobtitle"]).median().sort_values(by='avg_pay',ascending=False).head(10).plot.bar(color='r')
```

Out[41]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x131cc2ae240>
```



Result

1. we majorly speak here about the expereince's and pay_rates.

2. secondly , we will speak about the comaparison between the

a) industries and pay_rates

b) avg. pay and skills jobtitles

Thanking you , satyamsharma(Highly Passionate Machine Learning Engineer)