In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

In [4]:
```
data=pd.read_csv('https://raw.githubusercontent.com/dsrscientist/dataset3/main/Salaries.csv')
```

data.head()

In [5]:
```
data.describe()
```

Out[5]:

	yrs.since.phd	yrs.service	salary
count	397.000000	397.000000	397.000000
mean	22.314861	17.614610	113706.458438
std	12.887003	13.006024	30289.038695
min	1.000000	0.000000	57800.000000
25%	12.000000	7.000000	91000.000000
50%	21.000000	16.000000	107300.000000
75%	32.000000	27.000000	134185.000000
max	56.000000	60.000000	231545.000000

In [6]:
```
df_features=data.drop('salary',axis=1)
```

In [17]:
```
q1=data.quantile(0.25)
print(q1)
q3=data.quantile(0.75)
print(q3)
iqr=q3-q1
```

```
yrs.since.phd       12.0
yrs.service          7.0
salary           91000.0
Name: 0.25, dtype: float64
yrs.since.phd        32.0
yrs.service          27.0
salary           134185.0
Name: 0.75, dtype: float64
yrs.since.phd       20.0
yrs.service         20.0
salary           43185.0
dtype: float64
```

In [32]:
```
print(iqr)
```

```
yrs.since.phd       20.0
yrs.service         20.0
salary           43185.0
dtype: float64
```

In [20]:
```
yrs_sin=50
```

In [23]:
```
index=np.where(data['yrs.since.phd']>50)
print(index)
```

```
(array([125, 131, 276, 282, 350], dtype=int64),)
```

In [25]:
```
data=data.drop(data.index[index])
```

In [26]:
```
data.reset_index()
```

Out[26]:

	index	rank	discipline	yrs.since.phd	yrs.service	sex	salary
0	0	Prof	B	19	18	Male	139750
1	1	Prof	B	20	16	Male	173200
2	2	AsstProf	B	4	3	Male	79750
3	3	Prof	B	45	39	Male	115000
4	4	Prof	B	40	41	Male	141500
...	...	...	...	...	...	...	...
382	392	Prof	A	33	30	Male	103106
383	393	Prof	A	31	19	Male	150564
384	394	Prof	A	42	25	Male	101738
385	395	Prof	A	25	15	Male	95329
386	396	AsstProf	A	8	4	Male	81035

387 rows × 7 columns

In [27]:
```
yrs_ser=50
```

In [28]:
```
index=np.where(data['yrs.service']>50)
print(index)
```

```
(array([190, 322], dtype=int64),)
```

In [29]:
```
data=data.drop(data.index[index])
```

In [31]:
```
data.reset_index()
```

Out[31]:

	index	rank	discipline	yrs.since.phd	yrs.service	sex	salary
0	0	Prof	B	19	18	Male	139750
1	1	Prof	B	20	16	Male	173200
2	2	AsstProf	B	4	3	Male	79750
3	3	Prof	B	45	39	Male	115000
4	4	Prof	B	40	41	Male	141500
...	...	...	...	...	...	...	...
380	392	Prof	A	33	30	Male	103106
381	393	Prof	A	31	19	Male	150564
382	394	Prof	A	42	25	Male	101738
383	395	Prof	A	25	15	Male	95329
384	396	AsstProf	A	8	4	Male	81035

385 rows × 7 columns

In [33]:
```
sal=107962.5
```

In [34]:
```
index=np.where(data['salary']>107962.5)
print(index)
```

```
(array([  0,   1,   3,   4,   6,   7,   8,   9,  10,  15,  18,  19,  23,
        26,  29,  30,  32,  36,  38,  40,  43,  45,  47,  48,  50,  51,
        56,  62,  68,  70,  71,  74,  76,  77,  80,  81,  82,  84,  85,
        86,  88,  93,  94,  97, 100, 101, 102, 103, 105, 109, 110, 115,
       116, 117, 120, 121, 126, 130, 131, 132, 135, 140, 141, 143, 144,
       146, 148, 151, 152, 155, 157, 162, 163, 165, 167, 169, 171, 174,
       176, 177, 179, 181, 185, 186, 187, 188, 190, 193, 194, 196, 197,
       198, 199, 201, 202, 204, 206, 207, 208, 209, 210, 211, 213, 214,
       215, 216, 218, 220, 224, 227, 228, 231, 236, 238, 243, 244, 245,
```

```
       247, 249, 251, 257, 258, 260, 261, 265, 266, 271, 273, 275, 279,
       281, 282, 283, 287, 288, 292, 293, 295, 296, 298, 300, 302, 305,
       309, 310, 313, 314, 315, 317, 318, 320, 321, 323, 325, 327, 328,
       329, 331, 332, 333, 334, 335, 336, 337, 339, 341, 342, 346, 348,
       349, 350, 352, 354, 355, 356, 357, 359, 360, 361, 366, 369, 372,
       373, 374, 375, 376, 377, 378, 379, 381], dtype=int64),)
```

In [35]:
```
data=data.drop(data.index[index])
```

In [36]:
```
data.reset_index()
```

Out[36]:

	index	rank	discipline	yrs.since.phd	yrs.service	sex	salary
0	2	AsstProf	B	4	3	Male	79750
1	5	AssocProf	B	6	6	Male	97000
2	11	AsstProf	B	7	2	Male	79800
3	12	AsstProf	B	1	1	Male	77700
4	13	AsstProf	B	2	0	Male	78000
...	...	...	...	...	...	...	...
190	383	Prof	A	44	44	Male	105000
191	392	Prof	A	33	30	Male	103106
192	394	Prof	A	42	25	Male	101738
193	395	Prof	A	25	15	Male	95329
194	396	AsstProf	A	8	4	Male	81035

195 rows × 7 columns

In [38]:
```
yrs_sin=-18
index=np.where(data['yrs.since.phd']<-18)
print(index)
```

```
(array([], dtype=int64),)
```

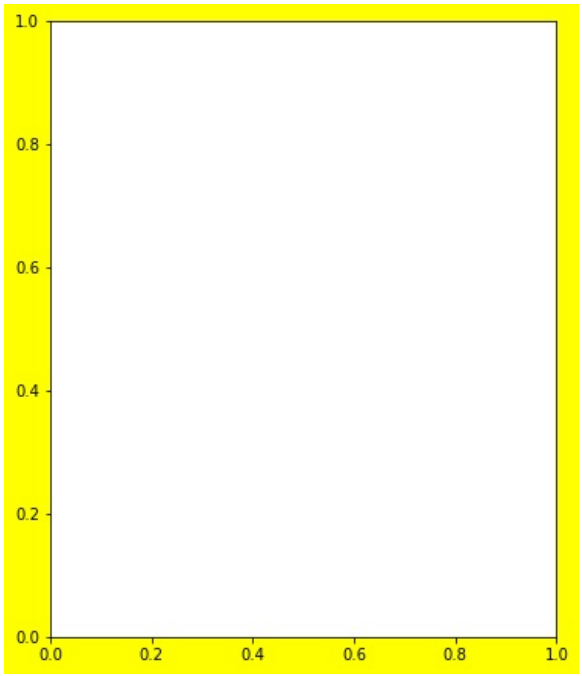
In [39]:
```
data.reset_index()
```

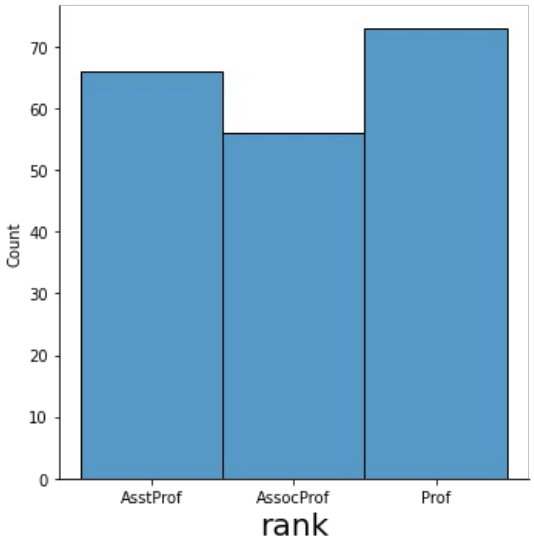
Out[39]:

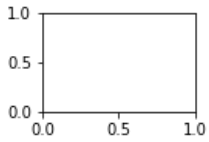
	index	rank	discipline	yrs.since.phd	yrs.service	sex	salary
0	2	AsstProf	B	4	3	Male	79750
1	5	AssocProf	B	6	6	Male	97000
2	11	AsstProf	B	7	2	Male	79800
3	12	AsstProf	B	1	1	Male	77700
4	13	AsstProf	B	2	0	Male	78000
...	...	...	...	...	...	...	...
190	383	Prof	A	44	44	Male	105000
191	392	Prof	A	33	30	Male	103106
192	394	Prof	A	42	25	Male	101738
193	395	Prof	A	25	15	Male	95329
194	396	AsstProf	A	8	4	Male	81035

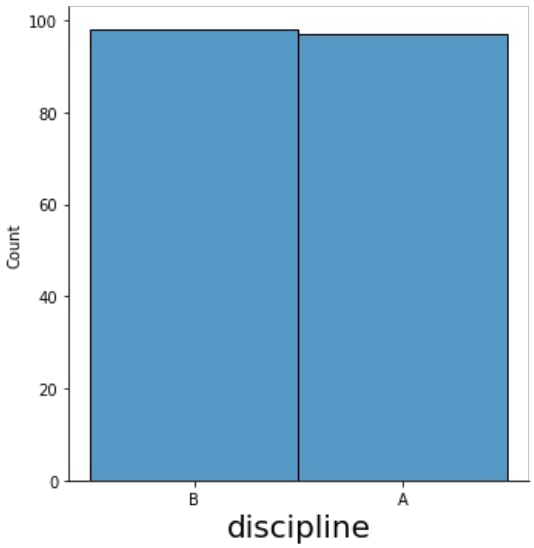
195 rows × 7 columns

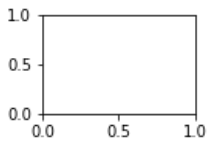
In [40]:
```
plt.figure(figsize=(20,25),facecolor='yellow')
plotnumber=1
for column in data:
    if plotnumber<=5:
        ax=plt.subplot(3,3,plotnumber)
        sns.displot(data[column])
        plt.xlabel(column,fontsize=20)
        plotnumber+=1
        plt.show()
```

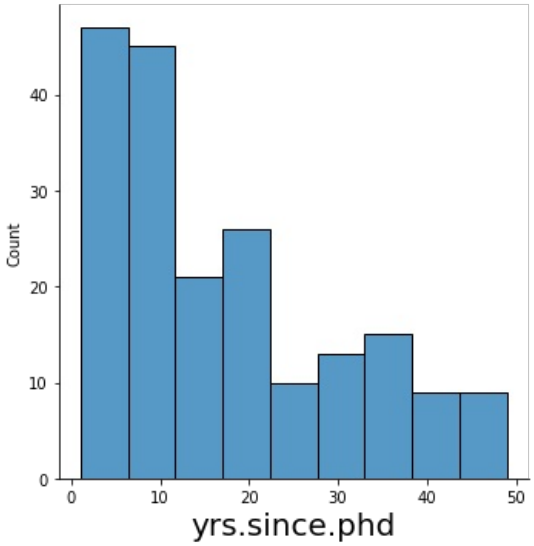

1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0

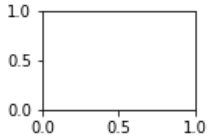

70
60
50
40
30
20
10
0
Count
AsstProf
AssocProf
Prof
rank

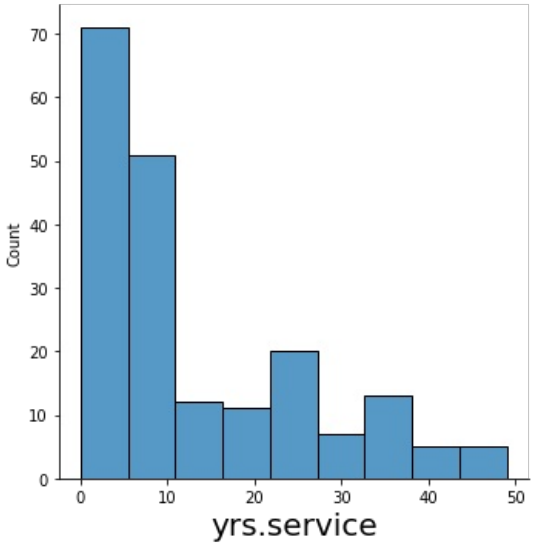

1.0
0.5
0.0
0.0
0.5
1.0

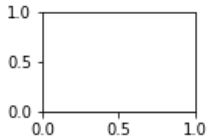

100
80
60
40
20
0
Count
B
A
discipline

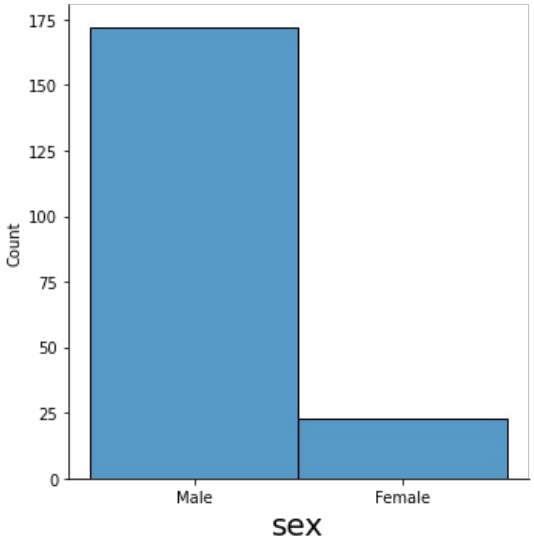

1.0
0.5
0.0
0.0
0.5
1.0


Count
0
10
20
30
40
0
10
20
30
40
50
yrs.since.phd


1.0
0.5
0.0
0.0
0.5
1.0


Count
0
10
20
30
40
50
60
70
0
10
20
30
40
50
yrs.service


1.0
0.5
0.0
0.0
0.5
1.0

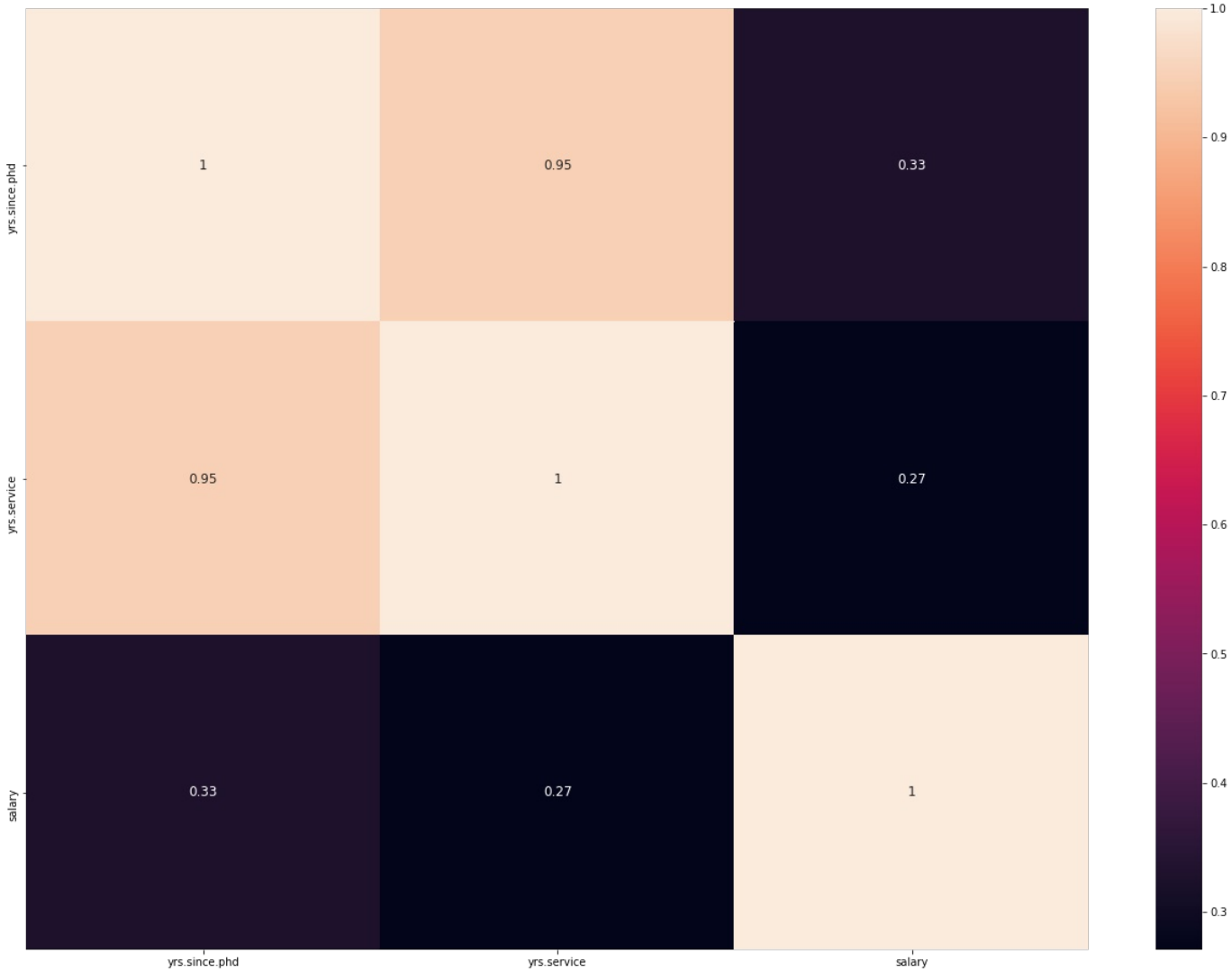

Count
0
25
50
75
100
125
150
175
Male
Female
sex

In [41]:
```
data.describe()
```

Out[41]:

	yrs.since.phd	yrs.service	salary
count	195.000000	195.000000	195.000000
mean	17.010256	12.825641	90248.276923
std	13.019410	12.445773	11928.469764
min	1.000000	0.000000	62884.000000
25%	7.000000	3.000000	80225.000000
50%	12.000000	8.000000	91300.000000
75%	25.000000	22.000000	101018.000000
max	49.000000	49.000000	107550.000000

In [42]:
```
df_corr=data.corr().abs()
plt.figure(figsize=(22,16))
sns.heatmap(df_corr,annot=True,annot_kws={'size':12})
plt.show()
```



In [43]:
```
##x=data.drop(columns=['salary','yrs.service'])
```

In [57]:
```
##print(x)
```

In [53]:
```
from sklearn.preprocessing import LabelEncoder
lab_enc=LabelEncoder()
df2=lab_enc.fit_transform(data['rank'])
data['rank']=df2
print(data)
```

```
     rank discipline  yrs.since.phd  yrs.service   sex  salary
2       1          B              4            3  Male   79750
```

```
5       0          B              6             6  Male   97000
11      1          B              7             2  Male   79800
12      1          B              1             1  Male   77700
13      1          B              2             0  Male   78000
..    ...        ...            ...           ...   ...     ...
383     2          A             44            44  Male  105000
392     2          A             33            30  Male  103106
394     2          A             42            25  Male  101738
395     2          A             25            15  Male   95329
396     1          A              8             4  Male   81035

[195 rows x 6 columns]
```

In [58]:
```
#print(x)
```

In [59]:
```
data.drop(columns=['sex'])
```

Out[59]:

	rank	discipline	yrs.since.phd	yrs.service	salary
2	1	B	4	3	79750
5	0	B	6	6	97000
11	1	B	7	2	79800
12	1	B	1	1	77700
13	1	B	2	0	78000
...	...	...	...	...	...
383	2	A	44	44	105000
392	2	A	33	30	103106
394	2	A	42	25	101738
395	2	A	25	15	95329
396	1	A	8	4	81035

195 rows × 5 columns

In [104…
```
x=data.drop(columns=['salary','sex','yrs.since.phd','descipline','rank','discipline'],axis=1)
print(x)
```

```
     yrs.service
2              3
5              6
11             2
12             1
13             0
..           ...
383           44
392           30
394           25
395           15
396            4

[195 rows x 1 columns]
```

In [105…
```
print(x)
```

```
     yrs.service
2              3
5              6
11             2
12             1
13             0
..           ...
383           44
392           30
394           25
395           15
396            4

[195 rows x 1 columns]
```

In [106…
```
y=data.salary
print(y)
```

```
2       79750
5       97000
11      79800
12      77700
13      78000
        ...
383    105000
392    103106
394    101738
395     95329
396     81035
Name: salary, Length: 195, dtype: int64
```

In [107…
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.25,random_state=41)
```

In [108…
```
from sklearn.linear_model import LinearRegression
```

In [109…
```
lr=LinearRegression()
lr.fit(x_train,y_train)
```

Out[109… LinearRegression()

In [110…
```
lr.intercept_
```

Out[110… 86679.72266424124

In [111…
```
lr.coef_
```

Out[111… array([278.80811048])

In [112…
```
y_pred=lr.intercept_ + lr.coef_* x
```

In [113…
```
print(y_pred)
```

```
      yrs.service
2    87516.146996
5    88352.571327
11   87237.338885
12   86958.530775
13   86679.722664
..            ...
383  98947.279525
392  95043.965979
394  93649.925426
395  90861.844321
396  87794.955106

[195 rows x 1 columns]
```

In [114…
```
y_pred=lr.predict(x_test)
```

In [115…
```
print(y_pred)
```

```
[89188.99565858 86679.72266424 99226.08763594 95043.9659787
 92813.50109485 95322.77408919 87794.95510617 88352.57132713
 88631.37943762 88631.37943762 88910.1875481  93371.11731581
 88073.76321665 86679.72266424 94207.54164726 88352.57132713
```

```
88910.1875481  88073.76321665 90025.41999003 95043.9659787
87794.95510617 88910.1875481  88073.76321665 88910.1875481
89188.99565858 89188.99565858 87516.14699569 87237.33888521
86958.53077472 87794.95510617 92813.50109485 93092.30920533
94207.54164726 90583.03621099 86679.72266424 87237.33888521
88352.57132713 96159.19842063 91977.0767634  87794.95510617
93928.73353678 91419.46054244 87794.95510617 86679.72266424
89467.80376906 91977.0767634  87237.33888521 92813.50109485
86958.53077472]
```

In [127…
```
from sklearn.metrics import mean_squared_error,mean_absolute_error
```

In [128…
```
mean_absolute_error(y_test,y_pred)
```

Out[128… 9823.370789077942

In [ ]:
```
```