Libraries used:- pandas, numpy , json and datetime

Load Data :-

Data preview :-

preview of dataframe

```
In [3]: data
```

Out[3]:

	_id	EmailIds	Experience	Domain	Education	Patents	Publications	CreatedOn	UpdateO
0	5be55d2e895bbd0ecc542e17	[murali.vasudevan@wal-mart.com]	[{'title': 'Assistant Manager - Human Resource...	[]	[{'fieldOfStudy': 'Human Resources', 'projects...	[]	[]	2018-11-09T10:10:54.185Z	2021-0 23T10:10:54.185
1	5bfb9cb547b87d5b22be6f6a	[neeraj@formulateip.com]	[{'companyName': 'Chapter.AI', 'startDate': '2...	[]	[{'startDate': '2002', 'endDate': '2004', 'deg...	[]	[]	2018-11-26T07:11:49.636Z	2018-1 26T07:11:49.636
2	5dd565ca99209e3e656a2ac7	[suhas.mayekar.ee@gmail.com]	[{'title': 'Regional Sales Manager', 'startDat...	NaN	[{'fieldOfStudy': 'aggregate of', 'degreeName'...	[]	[]	2019-11-20 16:11:54.136107	2019-11-2 16:11:54.13610
3	5dd77779ea1fb9f90ce5d21e	[rahulbhatikp@gmail.com]	[{'title': 'Channel Sales Manager, IT Value Bu...	NaN	[{'fieldOfStudy': '', 'degreeName': '', 'start...	[]	[]	2019-11-22 05:51:53.485023	2019-11-2 05:51:53.48502
4	5dc00363ca2cb5dda0a4b120	[Anand.amit@live.com]	[{'title': 'Corporate Sales Manager', 'locatio...	[]	[{'schoolName': 'Delhi University', 'fieldOfSt...	[]	[]	2019-11-04 10:54:26.958080	2019-11-0 10:54:26.95808
5	5d3302210807d15ee9c4856e	[himanshu.v.tanwar@gmail.com]	[{'title': 'Manager, Product Marketing', 'loca...	[]	[{'schoolName': 'SCMHRD Pune', 'fieldOfStudy':...	[]	[]	2019-07-20T11:59:29.201Z	2019-07 20T11:59:29.201
6	5ddbd5a08d5aefa5e38ca1fd	[gauravkenue@gmail.com]	[{'title': 'Senior Subject Matter Expert (Auto...	NaN	[{'fieldOfStudy': 'Business Analytics',	[]	[]	2019-11-25 13:22:40.069845	2019-11-2 13:22:40.06989

As depicted in the Data, there is **Experience** column in which our data is present which we want to process for further analysis

```
In [5]: data['Experience'][0]
```

Out[5]:

```
[{'title': 'Assistant Manager - Human Resources',
  'location': '',
  'projects': '',
  'startDate': 'October 2003',
  'endDate': 'January 2005',
  'promotion': '',
  'honors': '',
  'description': '',
  'recommendations': '',
  'companyName': 'HCL Technologies',
  'courses': '',
  'organizations': ''},
 {'title': 'Senior Manager Human Resources',
  'location': '',
  'projects': '',
  'startDate': 'August 2010',
  'endDate': '',
  'promotion': '',
  'honors': '',
  'description': 'Managed Walmart Global Technologies expansion plans in India from inception – Set up of the legal entity\nPar
tnering with business on the talent development strategy - devising development plans and delivery of HR programs\nBuild compen
sation strategy which is aligned to organization's comp philosophy – Pay for Performance\nStreamlining of HR process to achieve
standardization across locations\nFormulated and implemented new policies understanding market practices and assists alignment
at an organization level\nDrive engagement initiatives through employee opinion survey\nIntroduce benefit plans to competitivel
y position as best place to work and preferred employer\n\n\n',
  'recommendations': '',
  'companyName': '@WalmartLabs India',
  'courses': '',
  'organizations': ''},
 {'title': 'Staffing Lead - Human Resources',
  'location': '',
  'projects': '',
  'startDate': 'August 2006',
```

From above we can see that Experience column of each row is in the form of dictionary So for each row we convert Experience column into dataframe

Data Pre-processing :-

1. First we make a empty data frame to append the extracted value of each column
2. Now we expand experience sequence into a dataframe for each column

for i in range(len(data)):

#Making a dataframe of data['Experience'][i] column

t = pd.DataFrame(data['Experience'][i])

3. Futher process occur in t dataframe
4. Remove the row whose company name is not mentioned

```
#If company name is not there than remove that row
for j in range(len(t)):
        if t['companyName'][j] == None:
            t=t.drop(j)
t=t.reset_index(drop=True)
```

5. Now if **t** Dataframe is empty than we mark the value we want to extract as Na or 0

```
# dataframe is null
if len(t) ==0:
    ids = data['_id'][i]
    Most_Worked_Company_Name = "Na"
    Least_Worked_Company_Name = "Na"
    Total_experience=0
    Carrer_gap=0
    data_info1 = {'Ids':ids,'Most_Worked_Company_Name':Most_Worked_Company_Name,
                  'Least_Worked_Company_Name':Least_Worked_Company_Name,'Total_experience':Total_experience,
                  'Carrer_gap':Carrer_gap}
    experience = experience.append(data_info1,ignore_index=True).reset_index(drop=True)
    continue
```

6. Replace current or Present with todays date
   t = t.replace(to_replace =['current','Present'], value =date.today())

7. Conerting start_date and end_date string column to date fromate
   t['startDate'] = pd.to_datetime(t['startDate'], utc=True)
   t['endDate'] = pd.to_datetime(t['endDate'], utc=True)

8. Sort the dateframe with respect to startdate and reset index
   t=t.sort_values(by=['startDate'], ascending=True)
   t=t.reset_index(drop=True)

9. For dealing the case in which a person is working in the same company and its position change
Thus, its experience is total time spend in that company

```
for j in range(1,len(t)):
    if j >= len(t):
        break
    if t['companyName'][j] == t['companyName'][j-1] and t['endDate'][j-1] == t['startDate'][j]:
        t['endDate'][j-1] = t['endDate'][j]
        t=t.drop(j)
        t=t.reset_index(drop=True)
```

10. Condition for removing row don't having both StartDate and EndDate not present

```
# Condition for removing row don't having both StartDate and EndDate not present
for j in range(len(t)):
    if j < len(t):
        if type(t['endDate'][j]) ==pd._libs.tslibs.nattype.NaTType and
           type(t['startDate'][j]) ==pd._libs.tslibs.nattype.NaTType:
            t=t.drop(j)
            t=t.reset_index(drop=True)
```

11. If 'employmentDuration' column is present and endate is not than we add employee duration in startDate :-

```
# if 'employmentDuration' column  is present and endate is not than we add employee duration in startDate
if 'employmentDuration' in t:
    for j in range(len(t)):
        if type(t['endDate'][j]) ==pd._libs.tslibs.nattype.NaTType:
            t['endDate'][j] = t['startDate'][j]+pd.offsets.DateOffset(years=t['employmentDuration'][j]['years'],
                                                                      months=t['employmentDuration'][j]['months'])
            pd.to_datetime(t['endDate'][j], utc=True)
```

12. Last row of dataframe dont have enddate than make it todays date

```
if type(t['endDate'][len(t)-1])==pd._libs.tslibs.nattype.NaTType:
        t['endDate'][len(t)-1] = pd.to_datetime(date.today(), utc=True)
```

13. **If other than last row, enddate is not present than make endate equal start date of next row:-**

```
for j in range(len(t)-1):
    if type(t['endDate'][j]) ==pd._libs.tslibs.nattype.NaTType:
        t['endDate'][j] = t['startDate'][j+1]
```

14. For handling duplicate value with consideration that, a person can work only at 1 company in a duration
#if start date of any 2 column is same then lowest enddate value row get deleted

```
# for handling duplicate value with consideration that, a person can work only at 1 company in an duration
# if start date of any 2 column is same than lowest enddate value row get delted
for j in range(1,len(t)):
    if j >= len(t):
        break

    if t['startDate'][j] == t['startDate'][j-1]:
          if t['endDate'][j] > t['endDate'][j-1]:
              t=t.drop(j-1)
              t=t.reset_index(drop=True)


          else:
              t=t.drop(j)
              j=j-1;
              t=t.reset_index(drop=True)
```

15. Extracting id of 't' dataframe from original data
ids = data['_id'][i]

16. **For get total experience of a particular person :-**
➔ first we make a sperate column in which equals to t['endDate']-t['startDate'] and then convert it to years
t['exp'] =(t['endDate']-t['startDate'])/np.timedelta64(1, 'Y')

➔ Then take the total sum of that column as "Total_experience"
Total_experience= float("{:.2f}".format(t['exp'].sum()))

17. **For finding most worked company :-** we get the index value of row which have highest exp than make than index company name as Most_Worked_Company_Name

Most_Worked_Company_Name=t['companyName'][t['exp'].idxmax(axis = 0)]

18. **For finding Least worked company** :- we get the index value of row which have minimum exp than make than index company name as Least_Worked_Company_Name

Least_Worked_Company_Name=t['companyName'][t['exp'].idxmin(axis = 0)]

19. For getting extracting career gap :-

```
Career_gap= (
                    float("{:.2f}".format((pd.to_datetime(date.today(), utc=True)
                    -pd.to_datetime(t['startDate'][0]))/np.timedelta64(1, 'Y')))
                     )
                     -  Total_experience
```

20. **Now adding the information we exteract of particular row in experience dataframe** :-

```
# now adding the information we exteract of particular row in experience datframe
data_info1 = {'Ids':ids,'Most_Worked_Company_Name':Most_Worked_Company_Name,
              'Least_Worked_Company_Name':Least_Worked_Company_Name,'Total_experience':Total_experience,
              'Career_gap':Career_gap}
experience = experience.append(data_info1,ignore_index=True).reset_index(drop=True)
```

20. Extract output : - experience dataframe:-

Out[57]:

	Ids	Most_Worked_Company_Name	Least_Worked_Company_Name	Total_experience	Carrer_gap
0	5be55d2e895bbd0ecc542e17	@WalmartLabs India	HCL Technologies	21.80	0.00
1	5bfb9cb547b87d5b22be6f6a	FormulateIP (www.formulateip.com)	Chapter.AI	15.38	0.00
2	5dd565ca99209e3e656a2ac7	SkyBridge Solutions Private Limited	PathInfotech Limited	11.71	2.17
3	5dd77779ea1fb9f90ce5d21e	Redington Gulf	HCL Infosystem Limited	11.46	0.09
4	5dc00363ca2cb5dda0a4b120	HDFC LIFE LTD	YES BANK	8.80	8.25
5	5d3302210807d15ee9c4856e	biggest conglomerate Reliance Industries Ltd	Reliance headquarters	5.63	3.00
6	5ddbd5a08d5aefa5e38ca1fd	Amdocs Development Centre (Delhi) Pvt Ltd.	Vodafone	3.25	9.72
7	61dd83c1ac3cdd88dabf62a3	Hindustan Unilever Ltd at Nasik	Abbott Healthcare Pvt. Ltd at Jhagadia	18.05	1.75
8	61d6b8dcac3cdd88dabf5def	Tally Solutions Private Limited	OTO Capital	4.88	0.00
9	61d52758ac3cdd88dabf5c5c	ZS Associates	ZS Associates	1.13	0.00
10	61dd07c208d9ebf21498a5e9	TRANSEARCH	Purple Quarter	9.88	0.09
11	5c5ea676f4f04f3b2e5cdf11	Amazon	Apeejay Shipping Ltd.	12.38	2.00
12	5db7f3b07bb56edeb8d34e1e	Syndicate Bank	MetricFox	10.08	5.22
13	5db93ae79198b31eefec95e2	Redbus India Pvt. Ltd	Bluemeric Technologies Pvt. Ltd	8.05	0.66
14	5ddbd3078d5aefa5e38ca017	VIP Industries Ltd.	internal EdTech startup database	6.38	1.34
15	5dde9a8bcd014ac56e7e23ff	SKILLS	Cisco Systems	8.05	0.42
16	5f1abd28d6c68a3921475bbc	Na	Na	0.00	0.00
17	5f1eaa48d6c68a3921476051	Adobe Systems	Tata Consultancy Services	9.58	2.97
18	5f460945d6c68a3921477c1f	innoValus Technologies	innoValus Technologies	6.83	2.55
19	5fc8baf407cd0c7a4e867602	Na	Na	0.00	0.00
20	5bfc066d47b87d5b22be6f81	GruBox	Max India	14.38	1.00

For desire outcome above dataframe is used