

## Name : Arul kumar ARK

Roll No. : 225229103

### Lab 6 :

In [ ]: Pandas Data Cleaning

```
In [1]: import pandas as pd
df = pd.read_csv("train_hr.csv")
df.head(10)
```

```
Out[1]:
```

	employee_id	department	region	education	gender	recruitment_channel	no_of_trainings	age	previous_year_rating	length_of_service	KPIs_met >80%	awards
0	65438	Sales & Marketing	region_7	Master's & above	f	sourcing	1	35	5.0	8	1	
1	65141	Operations	region_22	Bachelor's	m	other	1	30	5.0	4	0	
2	7513	Sales & Marketing	region_19	Bachelor's	m	sourcing	1	34	3.0	7	0	
3	2542	Sales & Marketing	region_23	Bachelor's	m	other	2	39	1.0	10	0	
4	48945	Technology	region_26	Bachelor's	m	other	1	45	3.0	2	0	
5	58896	Analytics	region_2	Bachelor's	m	sourcing	2	31	3.0	7	0	
6	20379	Operations	region_20	Bachelor's	f	other	1	31	3.0	5	0	
7	16290	Operations	region_34	Master's & above	m	sourcing	1	33	3.0	6	0	
8	73202	Analytics	region_20	Bachelor's	m	other	1	28	4.0	5	0	
9	28911	Sales & Marketing	region_1	Master's & above	m	sourcing	1	32	5.0	5	1	

```
In [21]: column_names = df.columns
print(column_names)
df.dtypes
for i in column_names:
    print("{} is unique : {}".format(i,df[i].is_unique))

Index(['department', 'region', 'education', 'gender', 'recruitment_channel',
       'no_of_trainings', 'age', 'awards_won?', 'avg_training_score',
       'is_promoted'],
      dtype='object')
department is unique : False
region is unique : False
education is unique : False
gender is unique : False
recruitment_channel is unique : False
no_of_trainings is unique : False
age is unique : False
awards_won? is unique : False
avg_training_score is unique : False
is_promoted is unique : False
```

```
In [3]: df.index.values
```

```
Out[3]: array([ 0, 1, 2, ..., 54805, 54806, 54807], dtype=int64)
```

```
In [4]: 0 in df.index.values
```

```
Out[4]: True
```

```
In [5]: df.set_index("employee_id",inplace=True)
```

```
In [6]: df
```

Out[6]:

	department	region	education	gender	recruitment_channel	no_of_trainings	age	previous_year_rating	length_of_service	KPIs_met >80%	awards_w
employee_id											
65438	Sales & Marketing	region_7	Master's & above	f	sourcing	1	35	5.0	8	1	
65141	Operations	region_22	Bachelor's	m	other	1	30	5.0	4	0	
7513	Sales & Marketing	region_19	Bachelor's	m	sourcing	1	34	3.0	7	0	
2542	Sales & Marketing	region_23	Bachelor's	m	other	2	39	1.0	10	0	
48945	Technology	region_26	Bachelor's	m	other	1	45	3.0	2	0	
...	...	...	...	...	...	...	...	...	...	...	
3030	Technology	region_14	Bachelor's	m	sourcing	1	48	3.0	17	0	
74592	Operations	region_27	Master's & above	f	other	1	37	2.0	6	0	
13918	Analytics	region_1	Bachelor's	m	other	1	27	5.0	3	1	
13614	Sales & Marketing	region_9	NaN	m	sourcing	1	29	1.0	2	0	
51526	HR	region_22	Bachelor's	m	other	1	27	1.0	5	0	

54808 rows × 13 columns



```
In [7]: columns_to_drop = [column_names[i] for i in [8,9,10]]
```

```
In [8]: df.drop(columns_to_drop, inplace=True, axis=1)
```

```
In [9]: df
```

Out[9]:

	department	region	education	gender	recruitment_channel	no_of_trainings	age	awards_won?	avg_training_score	is_promoted
employee_id										
65438	Sales & Marketing	region_7	Master's & above	f	sourcing	1	35	0	49	0
65141	Operations	region_22	Bachelor's	m	other	1	30	0	60	0
7513	Sales & Marketing	region_19	Bachelor's	m	sourcing	1	34	0	50	0
2542	Sales & Marketing	region_23	Bachelor's	m	other	2	39	0	50	0
48945	Technology	region_26	Bachelor's	m	other	1	45	0	73	0
...	...	...	...	...	...	...	...	...	...	...
3030	Technology	region_14	Bachelor's	m	sourcing	1	48	0	78	0
74592	Operations	region_27	Master's & above	f	other	1	37	0	56	0
13918	Analytics	region_1	Bachelor's	m	other	1	27	0	79	0
13614	Sales & Marketing	region_9	NaN	m	sourcing	1	29	0	45	0
51526	HR	region_22	Bachelor's	m	other	1	27	0	49	0

54808 rows × 10 columns

```
In [10]: df['department'] = df['department'].fillna(' ')
df
```

```
Out[10]:
```

	department	region	education	gender	recruitment_channel	no_of_trainings	age	awards_won?	avg_training_score	is_promoted
employee_id										
65438	Sales & Marketing	region_7	Master's & above	f	sourcing	1	35	0	49	0
65141	Operations	region_22	Bachelor's	m	other	1	30	0	60	0
7513	Sales & Marketing	region_19	Bachelor's	m	sourcing	1	34	0	50	0
2542	Sales & Marketing	region_23	Bachelor's	m	other	2	39	0	50	0
48945	Technology	region_26	Bachelor's	m	other	1	45	0	73	0
...	...	...	...	...	...	...	...	...	...	...
3030	Technology	region_14	Bachelor's	m	sourcing	1	48	0	78	0
74592	Operations	region_27	Master's & above	f	other	1	37	0	56	0
13918	Analytics	region_1	Bachelor's	m	other	1	27	0	79	0
13614	Sales & Marketing	region_9	NaN	m	sourcing	1	29	0	45	0
51526	HR	region_22	Bachelor's	m	other	1	27	0	49	0

54808 rows × 10 columns

```
In [11]: df['education'] = df['education'].fillna(99)
df
```

```
Out[11]:
```

	department	region	education	gender	recruitment_channel	no_of_trainings	age	awards_won?	avg_training_score	is_promoted
employee_id										
65438	Sales & Marketing	region_7	Master's & above	f	sourcing	1	35	0	49	0
65141	Operations	region_22	Bachelor's	m	other	1	30	0	60	0
7513	Sales & Marketing	region_19	Bachelor's	m	sourcing	1	34	0	50	0
2542	Sales & Marketing	region_23	Bachelor's	m	other	2	39	0	50	0
48945	Technology	region_26	Bachelor's	m	other	1	45	0	73	0
...	...	...	...	...	...	...	...	...	...	...
3030	Technology	region_14	Bachelor's	m	sourcing	1	48	0	78	0
74592	Operations	region_27	Master's & above	f	other	1	37	0	56	0
13918	Analytics	region_1	Bachelor's	m	other	1	27	0	79	0
13614	Sales & Marketing	region_9	99	m	sourcing	1	29	0	45	0
51526	HR	region_22	Bachelor's	m	other	1	27	0	49	0

54808 rows × 10 columns

```
In [12]: df['age'] = df['age'].fillna(df['age'].mean())
df
```

```
Out[12]:
```

	department	region	education	gender	recruitment_channel	no_of_trainings	age	awards_won?	avg_training_score	is_promoted
employee_id										
65438	Sales & Marketing	region_7	Master's & above	f	sourcing	1	35	0	49	0
65141	Operations	region_22	Bachelor's	m	other	1	30	0	60	0
7513	Sales & Marketing	region_19	Bachelor's	m	sourcing	1	34	0	50	0
2542	Sales & Marketing	region_23	Bachelor's	m	other	2	39	0	50	0
48945	Technology	region_26	Bachelor's	m	other	1	45	0	73	0
...	...	...	...	...	...	...	...	...	...	...
3030	Technology	region_14	Bachelor's	m	sourcing	1	48	0	78	0
74592	Operations	region_27	Master's & above	f	other	1	37	0	56	0
13918	Analytics	region_1	Bachelor's	m	other	1	27	0	79	0
13614	Sales & Marketing	region_9	99	m	sourcing	1	29	0	45	0
51526	HR	region_22	Bachelor's	m	other	1	27	0	49	0

54808 rows × 10 columns

```
In [13]: import numpy as np
```

```
In [14]: df1 = pd.DataFrame(data={'col1':[np.nan,np.nan,2,3,4,np.nan,np.nan]})
```

```
In [15]: df1.fillna(method='pad', limit=1)
```

```
Out[15]:
```

	col1
0	NaN
1	NaN
2	2.0
3	3.0
4	4.0
5	4.0
6	NaN

```
In [16]: df1.fillna(method='pad', limit=1)
```

```
Out[16]:
```

	col1
0	NaN
1	NaN
2	2.0
3	3.0
4	4.0
5	4.0
6	NaN

```
In [17]: df1.fillna(method = 'bfill')
```

```
Out[17]:
```

	col1
0	2.0
1	2.0
2	2.0
3	3.0
4	4.0
5	NaN
6	NaN

```
In [18]: df1.dropna()
```

```
Out[18]:
```

	col1
2	2.0
3	3.0
4	4.0

```
In [19]: df1.dropna(axis=1)
```

```
Out[19]:
```

0	
1	
2	
3	
4	
5	
6	

```
In [20]: df1.dropna(thresh=int(df1.shape[0] * .9), axis=1)
```

```
Out[20]:  
0  
1  
2  
3  
4  
5  
6
```

```
In [ ]:
```