```
import pandas as pd


# xl = pd.ExcelFile("DATA-Set.csv")
df = pd.read_csv("DATA-Set.csv")


df
```

	Date	Open	High	Low	Close
0	15-12-2011	154.740005	154.949997	151.710007	152.330002
1	16-12-2011	154.309998	155.369995	153.899994	155.229996
2	19-12-2011	155.479996	155.860001	154.360001	154.869995
3	20-12-2011	156.820007	157.429993	156.580002	156.979996
4	21-12-2011	156.979996	157.529999	156.130005	157.160004
...	...	...	...	...	...
1713	24-12-2018	119.570000	120.139999	119.570000	120.019997
1714	26-12-2018	120.620003	121.000000	119.570000	119.660004
1715	27-12-2018	120.570000	120.900002	120.139999	120.570000
1716	28-12-2018	120.800003	121.080002	120.720001	121.059998
1717	31-12-2018	120.980003	121.260002	120.830002	121.250000

1718 rows × 5 columns

```
# checking for NULL values
df.isnull().values.any()
```

```
True
```

```
# number of NuLL values
df.isnull().sum()
```

```
Date     0
Open     0
High     3
Low      3
Close    1
dtype: int64
```

Add mean value to missing values or Null values

```
mean_value=df['Open'].mean()

df['Open'].fillna(value=mean_value, inplace=True)


mean_value=df['High'].mean()

df['High'].fillna(value=mean_value, inplace=True)


mean_value=df['Low'].mean()

df['Low'].fillna(value=mean_value, inplace=True)


mean_value=df['Close'].mean()

df['Close'].fillna(value=mean_value, inplace=True)
```

Data Redction using Random sampeling

```
d = df.sample(150)

d
```

	Date	Open	High	Low	Close
507	13-02-2014	124.699997	125.599998	124.660004	125.489998
263	11-01-2013	161.860001	161.899994	160.110001	161.059998
657	24-09-2014	117.120003	117.720001	116.900002	117.050003
1261	02-03-2017	117.760002	118.339996	117.230003	117.580002
1043	14-04-2016	118.089996	118.190002	116.949997	117.110001
...	...	...	...	...	...
267	17-01-2013	161.940002	164.399994	161.830002	163.350006
897	11-09-2015	105.650002	106.199997	105.269997	106.160004
1190	15-11-2016	116.459999	117.239998	116.290001	117.120003
1551	01-05-2018	123.900002	123.980003	123.389999	123.709999
266	16-01-2013	162.419998	163.029999	162.149994	162.649994

150 rows × 5 columns

DATA Transformation using Z-score

```
open_1=d['Open'].mean()
op=d['Open'].std()

d['Open']=d['Open'].apply(lambda x:(x-open_1)/op)
```

```
high_1=d['High'].mean()
hi=d['High'].std()

d['High']=d['High'].apply(lambda x:(x-high_1)/hi)
```

```
low_1=d['Low'].mean()
lo=d['Low'].std()

d['Low']=d['Low'].apply(lambda x:(x-low_1)/lo)
```

```
close_1=d['Close'].mean()
co=d['Close'].std()

d['Close']=d['Close'].apply(lambda x:(x-close_1)/co)
```

```
d
```

	Date	Open	High	Low	Close
507	13-02-2014	-0.077975	-0.057232	-0.044789	-0.030469
263	11-01-2013	2.280810	2.239054	2.225832	2.229278
657	24-09-2014	-0.559126	-0.555710	-0.541827	-0.566658
1261	02-03-2017	-0.518501	-0.516490	-0.520690	-0.532988
1043	14-04-2016	-0.497554	-0.525978	-0.538625	-0.562846
...	...	...	...	...	...
267	17-01-2013	2.285888	2.397201	2.336001	2.374761
897	11-09-2015	-1.287201	-1.284449	-1.286745	-1.258495
1190	15-11-2016	-0.601020	-0.586074	-0.580899	-0.562211
1551	01-05-2018	-0.128755	-0.159711	-0.126134	-0.143551
266	16-01-2013	2.316357	2.310537	2.356496	2.330289

150 rows × 5 columns

✓ 0s completed at 10:00 PM

AP19110010138
Sai Sen Ram Kumar Mocha

__/__/____

TID	Item lists
T100	I_1, I_2, I_3, I_4
T200	I_2, I_4, I_2, I_5
T300	I_3, I_4, I_5
T400	I_1, I_3, I_4
T500	$I_3 \quad I_4, \quad I_6$
T600	I_1, I_3, I_4, I_6
T700	I_1, I_6
T800	I_1, I_4, I_4
T900	I_3, I_4, I_2, I_6
T1000	I_1, I_4, I_3

Items	
I_1	6
I_3	8
I_4	9
I_2	3
I_5	2
I_6	4

min support % = 40 %

i.e. $= \frac{40}{100} \times 10$

$= 4$

min sup = 4

I_1		I_2		
I_1	=>	I_1, I_3	— 5	
I_3		I_1, I_4	— 5	
I_4		I_1, I_6	— 2	×
I_6		I_3, I_4	— 7	
		I_4, I_6	— 3	×
		I_3, I_6	— 3	×

I_1, I_3, I_4 - 4

$\text{support}(I_1, I_3, I_4) = \frac{4^2}{10 \cdot 5} = \frac{2}{5} = 40\%$

$\text{confidence}\{I_1 \rightarrow (I_3, I_4)\} = \frac{4}{6} = \frac{2}{3} = 66.\bar{6}\%$

$\text{confidence}\{I_3 \rightarrow (I_1, I_4)\} = \frac{4}{6} = \frac{2}{3} = 66.\bar{6}\%$

$\text{confidence}\{I_4 \rightarrow (I_1, I_3)\} = \frac{4}{9} = 44.\bar{4}\%$

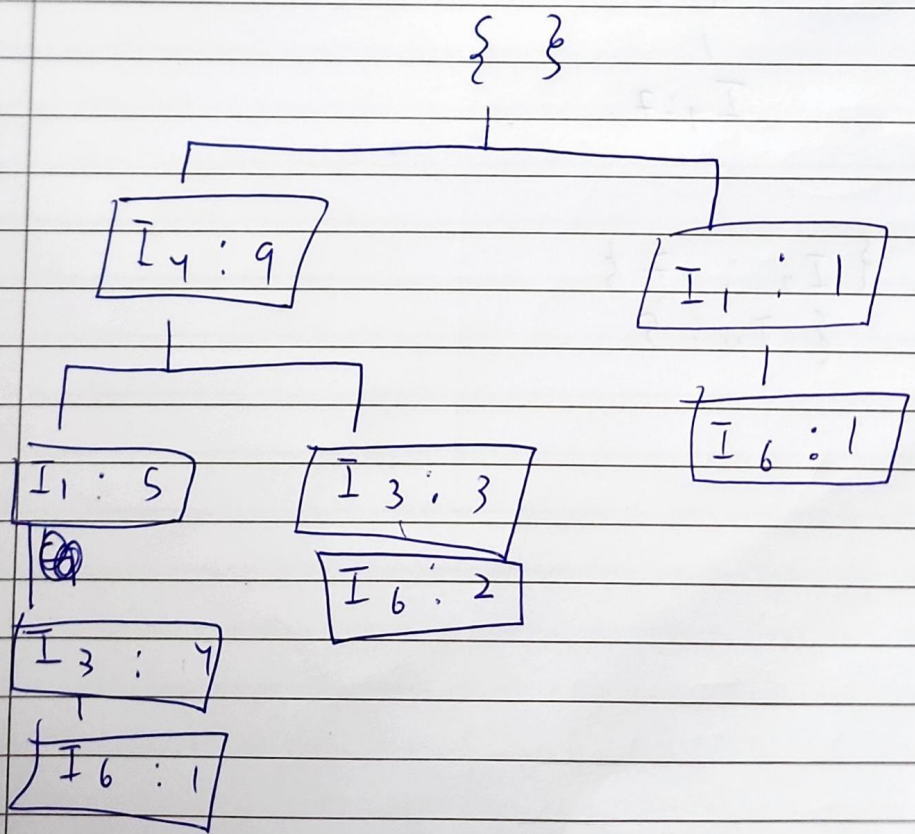
FP Growth

I_1 - 6	sorted Freq
I_3 - 6	I_4 - 10
I_4 - 10	I_1 - 6
I_6 - 4	I_3 - 6
	I_6 - 4

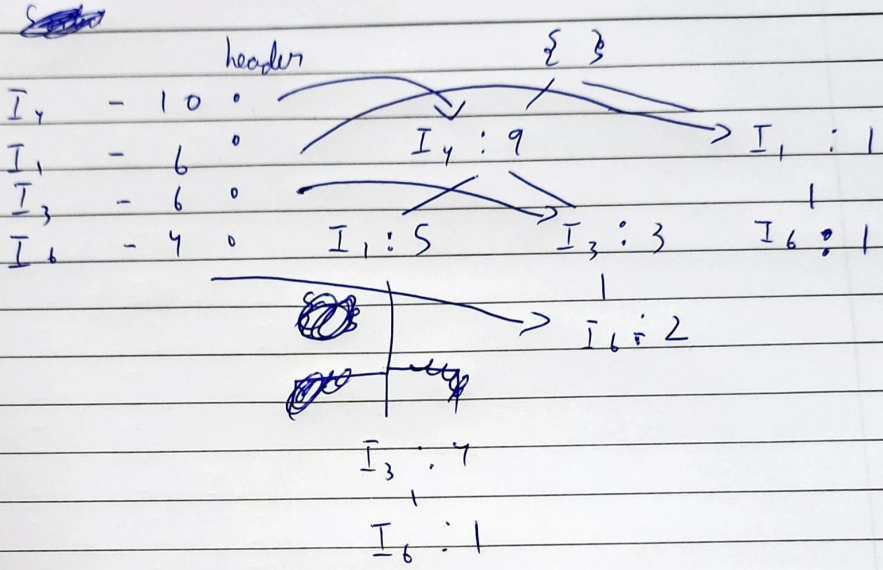
__/__/____

Set

1) I_4 I_1 I_3
2) I_4
3) I_4 , I_3
4) I_4, I_1, I_3
5) I_4, I_3, I_6
6) I_4, I_1, I_3, I_6
7) I_1, I_6
8) I_4, I_4, I_1
9) I_4, I_3, I_6
10) I_4, I_1, I_3



__/__/____


header
{ }
I_4 - 10 :
I_1 - 6 :
I_3 - 6 :
I_6 - 4 :
$I_4 : 9$
$I_1 : 1$
$I_1 : 5$
$I_3 : 3$
$I_6 : 1$
$I_6 : 2$
$I_3 : 7$
$I_6 : 1$

__/__/____

I_1 : { I_4 : 5 }

I_3 : { I_4 : 3, $I_4 I_1$: 4 }

I_6 : $I_4 I_3$: 2, $I_4 I_1 I_3$: 1, I_1 : 1 }

I_1 : { }
/
I_4 : 5

I_3 : { }
/
I_4 : 7

I_3 { I_4 : 7 }
I_1 { I_4 : 5