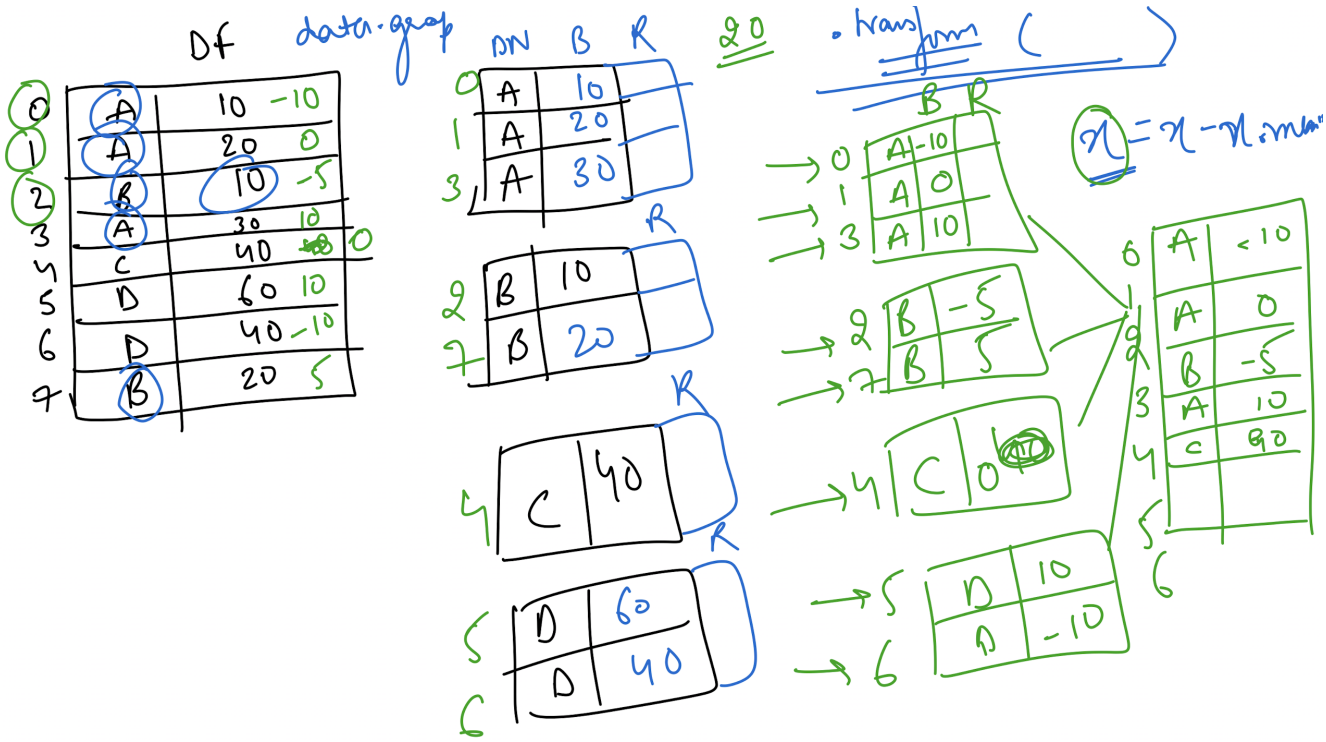
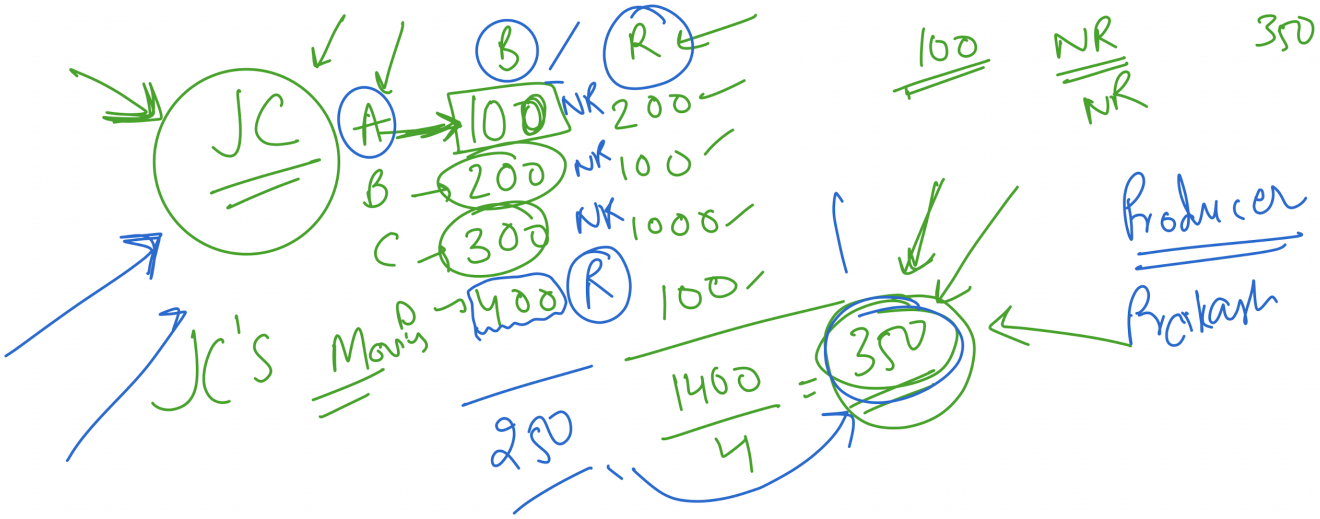
```
import pandas as pd
import numpy as np
!gdown 1s2TkjSpzNc4SyxqRrQleZyDIHlc7bxnd
!gdown 1Ws-_s1fHZ9nHfGLVUQurbHDvStePlEJm
movies = pd.read_csv('movies.csv', index_col=0)
directors = pd.read_csv('directors.csv',index_col=0)
data = movies.merge(directors, how='left', left_on='director_id',right_on='id')
data.drop(['director_id','id_y'],axis=1,inplace=True)
```

```
Downloading...
From: https://drive.google.com/uc?id=1s2TkjSpzNc4SyxqRrQleZyDIHlc7bxnd
To: /content/movies.csv
100% 112k/112k [00:00<00:00, 76.1MB/s]
Downloading...
From: https://drive.google.com/uc?id=1Ws-_s1fHZ9nHfGLVUQurbHDvStePlEJm
To: /content/directors.csv
100% 65.4k/65.4k [00:00<00:00, 95.7MB/s]
```

```
data["budget"]=(data["budget"]/1000000).round(2)
data["revenue"]=(data["revenue"]/1000000).round(2)
data
```

	id_x	budget	popularity	revenue	title	vote_average	vote_count	year
0	43597	237.00	150	2787.97	Avatar	7.2	11800	2009
1	43598	300.00	139	961.00	Pirates of the Caribbean: At World's End	6.9	4500	2007
2	43599	245.00	107	880.67	Spectre	6.3	4466	2015
3	43600	250.00	112	1084.94	The Dark Knight Rises	7.6	9106	2012
4	43602	258.00	115	890.87	Spider-Man 3	5.9	3576	2007
...	...	...	...	...	...	...	...	...
1460	48363	0.00	3	0.32	The Last Waltz	7.9	64	1978






```
def calc_exp(x):
    x["budget"]= x["budget"] - x["budget"].mean()
    x["budget"]

data.groupby("director_name").transform(calc_exp)
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
/usr/local/lib/python3.7/dist-packages/pandas/core/indexes/base.py in get_loc(self, k
   3360             try:
-> 3361                 return self._engine.get_loc(casted_key)
   3362             except KeyError as err:

                                15 frames
pandas/_libs/index_class_helper.pxi in pandas._libs.index.Int64Engine._check_type()

pandas/_libs/index_class_helper.pxi in pandas._libs.index.Int64Engine._check_type()

KeyError: 'budget'

The above exception was the direct cause of the following exception:

KeyError                                  Traceback (most recent call last)
/usr/local/lib/python3.7/dist-packages/pandas/core/indexes/base.py in get_loc(self, k
   3361                 return self._engine.get_loc(casted_key)
   3362             except KeyError as err:
-> 3363                 raise KeyError(key) from err
   3364
   3365         if is_scalar(key) and isna(key) and not self.hasnans:
```

```
def calc_exp(x):
    x=x-x.mean()
    return x

data["exp_or_inexp"]=data.groupby("director_name")["budget"].transform(calc_exp)
data
```

	id_x	budget	popularity	revenue	title	vote_average	vote_count
0	43597	237000000	150	2787965087	Avatar	7.2	11800

```
data.shape
```

```
(1465, 12)
```

At World's

```
data
```

	id_x	budget	popularity	revenue	title	vote_average	vote_count	year
0	43597	237.00	150	2787.97	Avatar	7.2	11800	2009
1	43598	300.00	139	961.00	Pirates of the Caribbean: At World's End	6.9	4500	2007
2	43599	245.00	107	880.67	Spectre	6.3	4466	2015
3	43600	250.00	112	1084.94	The Dark Knight Rises	7.6	9106	2012
4	43602	258.00	115	890.87	Spider-Man 3	5.9	3576	2007
...	...	...	...	...	...	...	...	...
1460	48363	0.00	3	0.32	The Last Waltz	7.9	64	1978

```
data.loc[data['director_name']=="James Cameron",["budget"]].mean()
```

```
budget    106.7
dtype: float64
```

```
def calc_risky(x):
    x["risky"]=(x["budget"]-x["revenue"].mean())>=0
    return x

data_risky=data.groupby("director_name").apply(calc_risky)
data_risky
```

	id_x	budget	popularity	revenue	title	vote_average	vote_count	year
0	43597	237.00	150	2787.97	Avatar	7.2	11800	2009
1	43598	300.00	139	961.00	Pirates of the Caribbean: At World's End	6.9	4500	2007
2	43599	245.00	107	880.67	Spectre	6.3	4466	2015
3	43600	250.00	112	1084.94	The Dark Knight Rises	7.6	9106	2012
4	43602	258.00	115	890.87	Spider-Man 3	5.9	3576	2007

```
data_risky.loc[data_risky["risky"]==True].head(10)
```

	id_x	budget	popularity	revenue	title	vote_average	vote_count	year	m
7	43608	200.0	107	586.09	Quantum of Solace	6.1	2965	2008	
12	43614	380.0	135	1045.71	Pirates of the Caribbean: On Stranger Tides	6.4	4948	2011	
15	43618	200.0	37	310.67	Robin Hood	6.2	1398	2010	
20	43624	209.0	64	303.03	Battleship	5.5	2114	2012	
24	43630	210.0	3	459.36	X-Men: The Last Stand	6.3	3525	2006	
29	43640	200.0	71	371.35	Terminator Salvation	5.9	2463	2009	
31	43642	200.0	81	531.86	World War Z	6.7	5560	2013	

```
data_risky.loc[data_risky["director_name"]=="Marc Forster",["budget","revenue"]].mean()
```

	budget	revenue
7	200.0	586.09
31	200.0	531.86
811	30.0	53.65
828	30.0	2.53
914	25.0	116.77
1023	30.0	72.28

```
data_risky.loc[data_risky["director_name"]=="Marc Forster",["budget","revenue"]].mean()
```

```
budget      63.68750
revenue    176.13625
dtype: float64
```

```
data
```

	id_x	budget	popularity	revenue	title	vote_average	vote_count	year
0	43597	237.00	150	2787.97	Avatar	7.2	11800	2009
1	43598	300.00	139	961.00	Pirates of the Caribbean: At World's End	6.9	4500	2007
2	43599	245.00	107	880.67	Spectre	6.3	4466	2015
3	43600	250.00	112	1084.94	The Dark Knight Rises	7.6	9106	2012
4	43602	258.00	115	890.87	Spider-Man 3	5.9	3576	2007
...	...	...	...	...	...	...	...	...
1460	48363	0.00	3	0.32	The Last Waltz	7.9	64	1978
1461	48370	0.03	19	3.15	Clerks	7.4	755	1994
1462	48375	0.00	7	0.00	Rampage	6.0	131	2009

```
data.loc[:,["revenue","budget","vote_average"]]
```

	revenue	budget	vote_average
0	2787.97	237.00	7.2
1	961.00	300.00	6.9
2	880.67	245.00	6.3
3	1084.94	250.00	7.6
4	890.87	258.00	5.9
...	...	...	...
1460	0.32	0.00	7.9
1461	3.15	0.03	7.4
1462	0.00	0.00	6.0
1463	0.00	0.00	6.4
1464	2.04	0.22	6.6

1465 rows × 3 columns

```
data.loc[:,["revenue","budget","vote_average"]].apply(np.mean,axis=0)
```

```
revenue         143.253952
budget           48.022949
vote_average      6.368191
dtype: float64
```

```
data.apply?
```

```
data.loc[:,["revenue","budget","vote_average"]].apply(np.mean,axis=1)
```

```
0       1010.723333
1        422.633333
2        377.323333
3        447.513333
4        384.923333
            ...
1460       2.740000
1461       3.526667
1462       2.000000
1463       2.133333
1464       2.953333
Length: 1465, dtype: float64
```

```
!gdown 173A59xh2mnpmljCCB9bhC4C5eP2IS6qZ
```

```
Downloading...
```

```
From: https://drive.google.com/uc?id=173A59xh2mnpmljCCB9bhC4C5eP2IS6qZ
To: /Users/nikhilsanghi/Downloads/dsml-course-main-live/batches/May-Beg-Aug-Adv/Pfize
100%|████████████████████████████████████████| 1.51k/1.51k [00:00<00:00, 1.43MB/s]
```

```
df=pd.read_csv("Pfizer_1.csv")
df
```

	Date	Drug_Name	Parameter	1:30:00	2:30:00	3:30:00	4:30:00	5:30:00	6:30:

```
pd.Series([1,np.nan,2,np.nan,3])
```

```
0    1.0
1    NaN
2    2.0
3    NaN
4    3.0
dtype: float64
```

```
pd.Series([1,np.nan,2,np.nan,3,None])
```

```
0    1.0
1    NaN
2    2.0
3    NaN
4    3.0
5    NaN
dtype: float64
```

```
type(None)
```

```
NoneType
```

```
type(np.nan)
```

```
float
```

```
pd.Series(["1","np.nan","2","np.nan","3",None])
```

```
0         1
1    np.nan
2         2
3    np.nan
4         3
5      None
dtype: object
```

```
df.isna()
```

	Date	Drug_Name	Parameter	1:30:00	2:30:00	3:30:00	4:30:00	5:30:00	6:30:00
0	False	False	False	False	False	True	False	False	False
1	False	False	False	False	False	True	False	False	False
2	False	False	False	True	False	False	True	False	False
3	False	False	False	True	False	False	True	False	False
4	False	False	False	False	True	True	False	True	False
5	False	False	False	False	True	True	False	True	False
6	False	False	False	False	False	False	False	False	False
7	False	False	False	False	False	False	False	False	False
8	False	False	False	False	False	True	False	False	False
9	False	False	False	False	False	True	False	False	False
10	False	False	False	False	False	False	True	False	False
11	False	False	False	False	False	False	True	False	False

```
df.isnull()
```

```
pd.isna
```

```
<function pandas.core.dtypes.missing.isna(obj)>
```

```
pd.isnull
```

```
<function pandas.core.dtypes.missing.isna(obj)>
```

4	False	False	False	False	True	True	False	True	False
6	False	False	False	False	False	False	False	False	False

```
df
```

	Date	Drug_Name	Parameter	1:30:00	2:30:00	3:30:00	4:30:00	5:30:00	6:30:
0	15-10-2020	diltiazem hydrochloride	Temperature	23.0	22.0	NaN	21.0	21.0	
1	15-10-2020	diltiazem hydrochloride	Pressure	12.0	13.0	NaN	11.0	13.0	

```
df.isna().sum?
```

```
df.isna().sum()
```

```
Date         0
Drug_Name    0
Parameter    0
1:30:00      2
2:30:00      2
3:30:00      6
4:30:00      4
5:30:00      2
6:30:00      0
7:30:00      2
8:30:00      4
9:30:00      2
10:30:00     0
11:30:00     2
12:30:00     0
dtype: int64
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18 entries, 0 to 17
Data columns (total 15 columns):
 #   Column     Non-Null Count  Dtype  
---  ------     --------------  -----  
 0   Date       18 non-null     object 
 1   Drug_Name  18 non-null     object 
 2   Parameter  18 non-null     object 
 3   1:30:00    16 non-null     float64
 4   2:30:00    16 non-null     float64
 5   3:30:00    12 non-null     float64
 6   4:30:00    14 non-null     float64
 7   5:30:00    16 non-null     float64
 8   6:30:00    18 non-null     int64  
 9   7:30:00    16 non-null     float64
 10  8:30:00    14 non-null     float64
 11  9:30:00    16 non-null     float64
 12  10:30:00   18 non-null     int64  
 13  11:30:00   16 non-null     float64
 14  12:30:00   18 non-null     int64  
dtypes: float64(9), int64(3), object(3)
memory usage: 2.2+ KB
```

```
df
```

	Date	Drug_Name	Parameter	1:30:00	2:30:00	3:30:00	4:30:00	5:30:00	6:30:
0	15-10-2020	diltiazem hydrochloride	Temperature	23.0	22.0	NaN	21.0	21.0	
1	15-10-2020	diltiazem hydrochloride	Pressure	12.0	13.0	NaN	11.0	13.0	
2	15-10-2020	docetaxel injection	Temperature	NaN	17.0	18.0	NaN	17.0	
3	15-10-2020	docetaxel injection	Pressure	NaN	22.0	22.0	NaN	22.0	
4	15-10-2020	ketamine hydrochloride	Temperature	24.0	NaN	NaN	27.0	NaN	
5	15-10-2020	ketamine hydrochloride	Pressure	8.0	NaN	NaN	7.0	NaN	
6	16-10-2020	diltiazem hydrochloride	Temperature	34.0	35.0	36.0	36.0	37.0	
7	16-10-2020	diltiazem hydrochloride	Pressure	18.0	19.0	20.0	21.0	22.0	
8	16-10-2020	docetaxel injection	Temperature	46.0	47.0	NaN	48.0	48.0	
9	16-10-2020	docetaxel injection	Pressure	23.0	24.0	NaN	25.0	26.0	
10	16-10-2020	ketamine hydrochloride	Temperature	8.0	9.0	10.0	NaN	11.0	
11	16-10-2020	ketamine hydrochloride	Pressure	12.0	12.0	13.0	NaN	15.0	
12	17-10-2020	diltiazem hydrochloride	Temperature	20.0	19.0	19.0	18.0	17.0	
13	17-10-	diltiazem							

```
df.dropna()
```

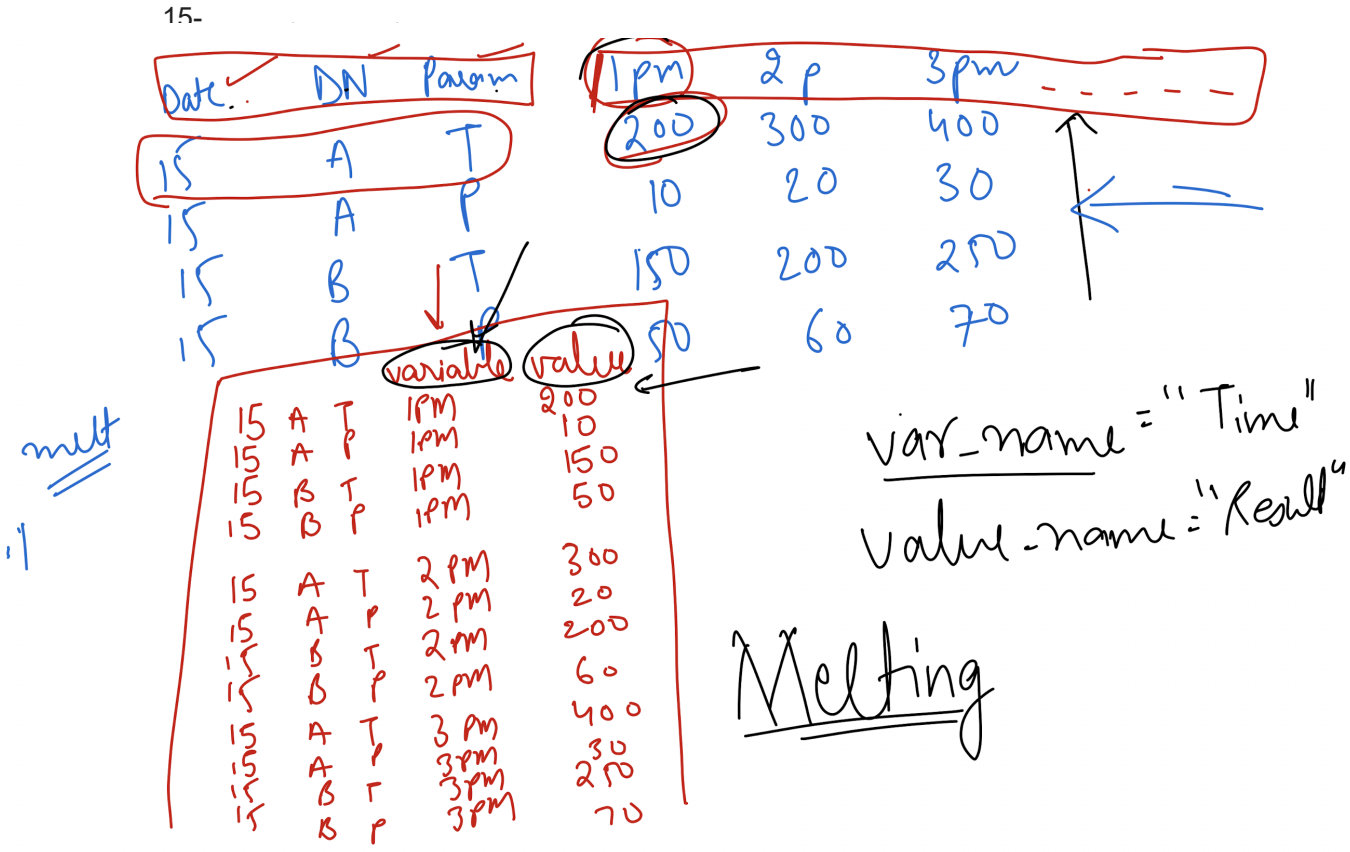
	Date	Drug_Name	Parameter	1:30:00	2:30:00	3:30:00	4:30:00	5:30:00	6:30:
14	17-10-2020	docetaxel injection	Temperature	12.0	13.0	14.0	15.0	16.0	
15	17-10-2020	docetaxel injection	Pressure	20.0	22.0	22.0	22.0	22.0	
16	17-10-2020	ketamine hydrochloride	Temperature	13.0	14.0	15.0	16.0	17.0	

```
df
```

	Date	Drug_Name	Parameter	1:30:00	2:30:00	3:30:00	4:30:00	5:30:00	6:30:
0	15-10-2020	diltiazem hydrochloride	Temperature	23.0	22.0	NaN	21.0	21.0	
1	15-10-2020	diltiazem hydrochloride	Pressure	12.0	13.0	NaN	11.0	13.0	
2	10-	docetaxel	Temperature	NaN	17.0	18.0	NaN	17.0	

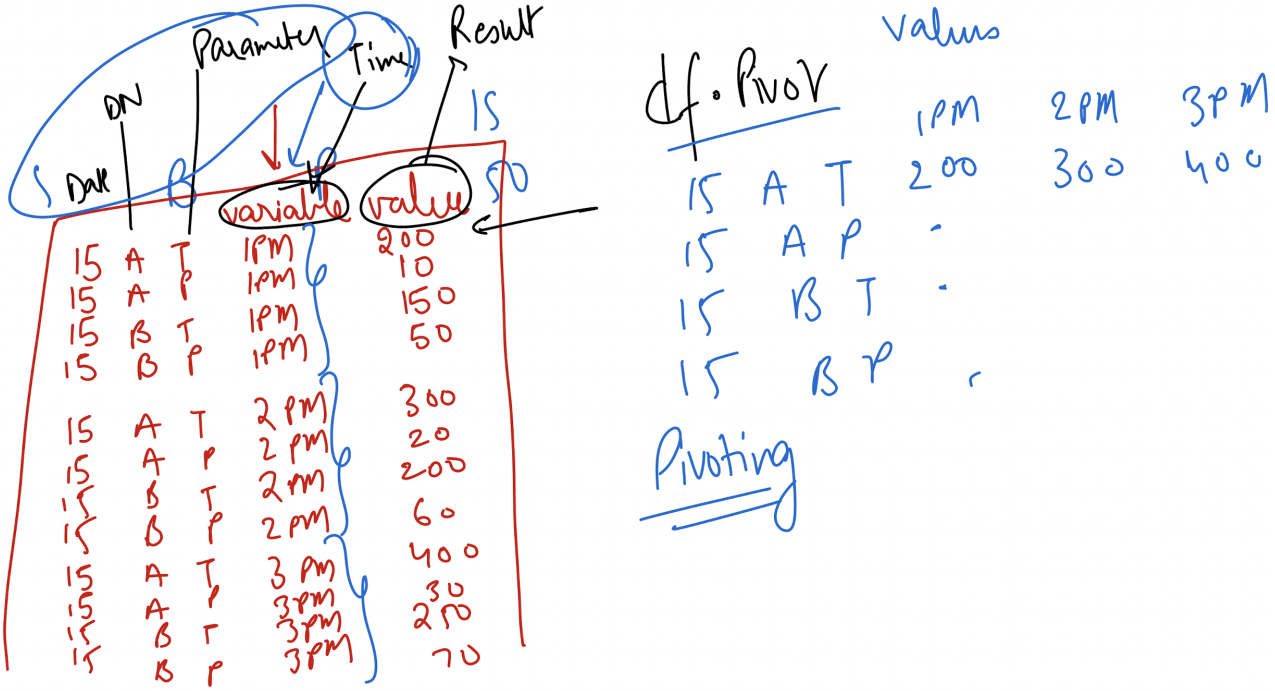
```
df.fillna(0)
```

	Date	Drug_Name	Parameter	1:30:00	2:30:00	3:30:00	4:30:00	5:30:00	6:30:
0	15-10-2020	diltiazem hydrochloride	Temperature	23.0	22.0	0.0	21.0	21.0	
1	15-10	diltiazem	Pressure	12.0	13.0	0.0	11.0	13.0	




```
df_melt=pd.melt(df,
       id_vars=["Date","Drug_Name","Parameter"],
       var_name="Time",
       value_name="Result")
df_melt
```

	Date	Drug_Name	Parameter	Time	Result
0	15-10-2020	diltiazem hydrochloride	Temperature	1:30:00	23.0
1	15-10-2020	diltiazem hydrochloride	Pressure	1:30:00	12.0
2	15-10-2020	docetaxel injection	Temperature	1:30:00	NaN




```
ab=df_melt.pivot(index=["Date","Drug_Name","Parameter"],
                 columns="Time",
                 values="Result")
ab
```

		Time	10:30:00	11:30:00	12:30:00	1:30:00	2:30:00	3:3
Date	Drug_Name	Parameter						
15-10-2020	diltiazem hydrochloride	Pressure	18.0	19.0	20.0	12.0	13.0	
		Temperature	20.0	20.0	21.0	23.0	22.0	
	docetaxel injection	Pressure	26.0	29.0	28.0	NaN	22.0	
		Temperature	23.0	25.0	25.0	NaN	17.0	
	ketamine hydrochloride	Pressure	9.0	9.0	11.0	8.0	NaN	
		Temperature	22.0	21.0	20.0	24.0	NaN	
16-	diltiazem	Pressure	24.0	NaN	27.0	18.0	19.0	

```
movies_directors["budget"] = (movies_directors["budget"] / 100000).round(2)
movies_directors.rename(columns={"budget":"budget_in_mill"})
```

	ketamine	Pressure	16.0	17.0	18.0	12.0	12.0	
2020		Temperature	14.0	11.0	10.0	20.0	19.0	
	docetaxel injection	Pressure	28.0	29.0	28.0	20.0	22.0	
		Temperature	21.0	22.0	23.0	12.0	13.0	
	ketamine hydrochloride	Pressure	13.0	14.0	15.0	8.0	9.0	
		Temperature	22.0	23.0	24.0	13.0	14.0	