

```
[1]: # Incentive Table
import pandas as pd
data = pd.read_excel("Goals.xlsx")
data
```

```
print[1]:
```

	Manager ID	Manager Name	City	Q3 - 2020 Goals (SU)
0	10015	Rahul S	Gangtok	235000
1	10019	Kirti S	Coimbatore	160000
2	10017	Vishwa P	Mumbai	190000
3	10012	Kalpana K	Amritsar	200000
4	10013	Ram S	Chandigarh	225000
5	10016	Kirti S	Goa	145000
6	10014	Taran S	Durgapur	170000
7	10020	Abhishek S	Hyderabad	250000
8	10011	Rakesh S	Agra	170000
9	10018	Ankit J	Chennai	180000

```
[2]: # Attainment Table
data2 = pd.read_excel("att.xlsx")
data2.head()
```

```
print[2]:
```

	Wholesaler ID	Date of Purchase	Month	Product Type	Capacity (ml)	Quantity Purchased	ZIP	City	Manager ID	Manager Name
0	R00014	2020-01-01	2020-01-31	Pulp Orange 250ml	250ml	28	36000198	Durgapur	10014	Taran S
1	R00015	2020-01-01	2020-01-31	Pulp Orange 1000ml	1000ml	111	36450915	Gangtok	10015	Rahul S
2	R00016	2020-01-01	2020-01-31	Pulp Orange 500ml	500ml	28	38041001	Goa	10016	Kirti S
3	R00018	2020-01-01	2020-01-31	Pulp Orange 1000ml	1000ml	34	39058313	Chennai	10018	Ankit J
4	R00022	2020-01-01	2020-01-31	Pulp Orange 1000ml	1000ml	36	41145792	Amritsar	10012	Kalpana K

```
[4]: # Removing ml from end of Capacity, converting it into a number and adding a new column new_capa for it

def func(x):
    return int(x[:-2])
data2["new_capa"] = data2["Capacity (ml)"].apply(func)
data2.head()
```

[4]: *# Removing ml from end of Capacity, converting it into a number and adding a new column new_capa for it*

```
def func(x):  
    return int(x[:-2])  
data2["new_capa"] = data2["Capacity (ml)"].apply(func)  
data2.head()
```

print[4]:

	Wholesaler ID	Date of Purchase	Month	Product Type	Capacity (ml)	Quantity Purchased	ZIP	City	Manager ID	Manager Name	new_capa
0	R00014	2020-01-01	2020-01-31	Pulp Orange 250ml	250ml	28	36000198	Durgapur	10014	Taran S	250
1	R00015	2020-01-01	2020-01-31	Pulp Orange 1000ml	1000ml	111	36450915	Gangtok	10015	Rahul S	1000
2	R00016	2020-01-01	2020-01-31	Pulp Orange 500ml	500ml	28	38041001	Goa	10016	Kirti S	500
3	R00018	2020-01-01	2020-01-31	Pulp Orange 1000ml	1000ml	34	39058313	Chennai	10018	Ankit J	1000
4	R00022	2020-01-01	2020-01-31	Pulp Orange 1000ml	1000ml	36	41145792	Amritsar	10012	Kalpana K	1000

[5]: *# Calculating the SU for each row and adding a new column Actual Sales for it*

```
data2["Actual Sales (SU)"] = data2["Quantity Purchased"]*data2["new_capa"]/500  
data2.head()
```

print[5]:

	Wholesaler ID	Date of Purchase	Month	Product Type	Capacity (ml)	Quantity Purchased	ZIP	City	Manager ID	Manager Name	new_capa	Actual Sales (SU)
0	R00014	2020-01-01	2020-01-31	Pulp Orange 250ml	250ml	28	36000198	Durgapur	10014	Taran S	250	14.0
1	R00015	2020-01-01	2020-01-31	Pulp Orange 1000ml	1000ml	111	36450915	Gangtok	10015	Rahul S	1000	222.0
2	R00016	2020-01-01	2020-01-31	Pulp Orange 500ml	500ml	28	38041001	Goa	10016	Kirti S	500	28.0
3	R00018	2020-01-01	2020-01-31	Pulp Orange 1000ml	1000ml	34	39058313	Chennai	10018	Ankit J	1000	68.0
4	R00022	2020-01-01	2020-01-31	Pulp Orange 1000ml	1000ml	36	41145792	Amritsar	10012	Kalpana K	1000	72.0

[6]: *# Grouping data by Manager ID to get total sales for each employee*

```
new_data = data2[["Manager ID", "Actual Sales (SU)"]]
```

```
[6]: # Grouping data by Manager ID to get total sales for each employee
new_data = data2[["Manager ID", "Actual Sales (SU)"]]
grouped_data = new_data.groupby("Manager ID").sum()
grouped_data.reset_index(level=0, inplace=True)
grouped_data
```

Out[6]:

	Manager ID	Actual Sales (SU)
0	10011	124695.5
1	10012	216948.0
2	10013	241285.5
3	10014	173087.5
4	10015	1002223.5
5	10016	162732.0
6	10017	231673.5
7	10018	126104.0
8	10019	226131.0
9	10020	211652.0

```
[7]: # Merging Goals and Actual Sales
data = data[["Manager ID", "Q3 - 2020 Goals (SU)"]]
result = pd.merge(grouped_data, data, on = 'Manager ID')
result
```

Out[7]:

	Manager ID	Actual Sales (SU)	Q3 - 2020 Goals (SU)
0	10011	124695.5	170000
1	10012	216948.0	200000
2	10013	241285.5	225000
3	10014	173087.5	170000
4	10015	1002223.5	235000
5	10016	162732.0	145000
6	10017	231673.5	190000
7	10018	126104.0	180000
8	10019	226131.0	160000
9	10020	211652.0	250000


```
[8]: # Calculating Attainment for each Manager
```

```
import numpy as np
r = result.values
new_result = []
for each in r:
    each = each.tolist()
    new_result.append(each)

for each in new_result:
    each.append(each[1]/each[2]*100)
new_result
```

```
Out[8]: [[10011.0, 124695.5, 170000.0, 73.35029411764707],
 [10012.0, 216948.0, 200000.0, 108.474],
 [10013.0, 241285.5, 225000.0, 107.23800000000001],
 [10014.0, 173087.5, 170000.0, 101.81617647058823],
 [10015.0, 1002223.5, 235000.0, 426.478085106383],
 [10016.0, 162732.0, 145000.0, 112.22896551724138],
 [10017.0, 231673.5, 190000.0, 121.93342105263159],
 [10018.0, 126104.0, 180000.0, 70.05777777777777],
 [10019.0, 226131.0, 160000.0, 141.331875],
 [10020.0, 211652.0, 250000.0, 84.66080000000001]]
```

```
[10]: df = pd.DataFrame(new_result, columns=['Manager ID', 'Actual Sales (SU)', 'Q3 - 2020 Goals (SU)', 'Attainment'])
df
```

```
[10]:
```

	Manager ID	Actual Sales (SU)	Q3 - 2020 Goals (SU)	Attainment
0	10011.0	124695.5	170000.0	73.350294
1	10012.0	216948.0	200000.0	108.474000
2	10013.0	241285.5	225000.0	107.238000
3	10014.0	173087.5	170000.0	101.816176
4	10015.0	1002223.5	235000.0	426.478085
5	10016.0	162732.0	145000.0	112.228966
6	10017.0	231673.5	190000.0	121.933421
7	10018.0	126104.0	180000.0	70.057778
8	10019.0	226131.0	160000.0	141.331875
9	10020.0	211652.0	250000.0	84.660800

```
[ ]:
```