1.what is an average occupancy rate in each of the room categories?

```
df_agg_bookings["occ_pct"] = df_agg_bookings["successful_bookings"]/df_agg_bookings["capacity"]
```

```
df_agg_bookings.head(4)
```

	property_id	check_in_date	room_category	successful_bookings	capacity	occ_pct
0	16559	1-May-22	RT1	25	30.0	83.33
1	19562	1-May-22	RT1	28	30.0	93.33
2	19563	1-May-22	RT1	23	30.0	76.67
3	17558	1-May-22	RT1	30	19.0	157.89

```
df_agg_bookings["occ_pct"] = df_agg_bookings["occ_pct"].apply(lambda x: round(x*100,2))
```

```
df_agg_bookings.head(4)
```

	property_id	check_in_date	room_category	successful_bookings	capacity	occ_pct
0	16559	1-May-22	RT1	25	30.0	83.33
1	19562	1-May-22	RT1	28	30.0	93.33
2	19563	1-May-22	RT1	23	30.0	76.67
3	17558	1-May-22	RT1	30	19.0	157.89

```
df_agg_bookings.groupby("room_category")["occ_pct"].mean().round(2)
```

```
room_category
RT1    58.22
RT2    58.04
RT3    58.03
RT4    59.30
```

2.print average occupancy rate per city ?

```
2..print average  occupancy rate per city based on property id we have to join hotels and agg_booking
```

```
df_hotels = pd.read_csv("/users/AAA-09/dim_hotels.csv")
```

```
df = pd.merge(df,df_hotels,on = "property_id")
df.head(4)
```

	property_id	check_in_date	room_category	successful_bookings	capacity	occ_pct	room_id	room_class	property_name	category	city
0	16559	1-May-22	RT1	25	30.0	83.33	RT1	Standard	Atliq Exotica	Luxury	Mumbai
1	16559	2-May-22	RT1	20	30.0	66.67	RT1	Standard	Atliq Exotica	Luxury	Mumbai
2	16559	3-May-22	RT1	17	30.0	56.67	RT1	Standard	Atliq Exotica	Luxury	Mumbai
3	16559	4-May-22	RT1	21	30.0	70.00	RT1	Standard	Atliq Exotica	Luxury	Mumbai

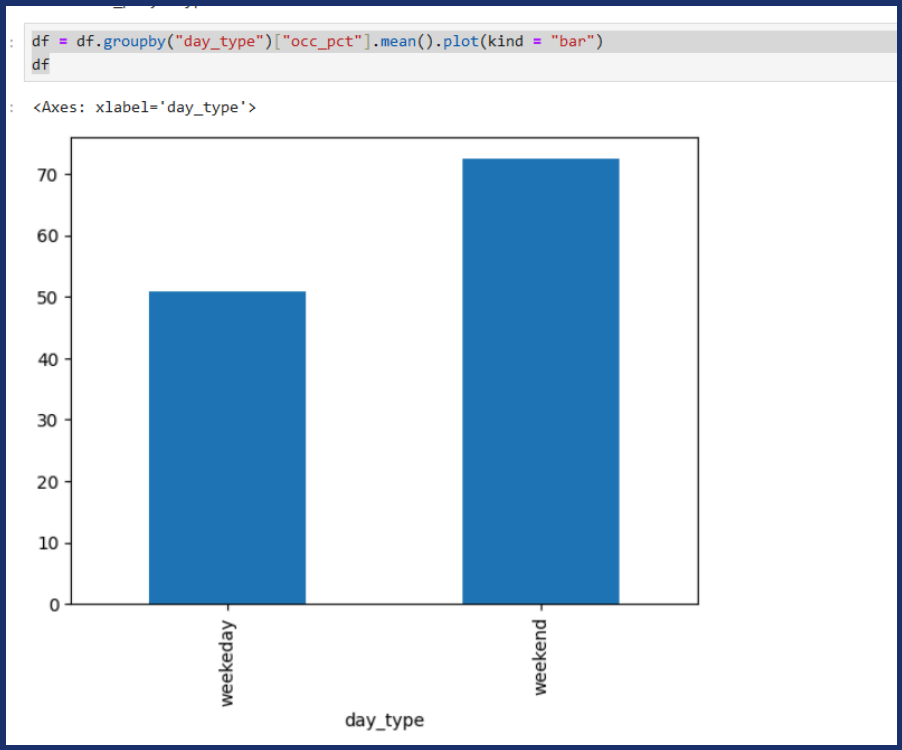
```
import pandas as pd
df = df.groupby("city")["occ_pct"].mean().round(2)
df
```

```
city
Bangalore    56.59
Delhi        61.61
Hyderabad    58.14
Mumbai       57.94
Name: occ_pct, dtype: float64
```

3.when was the occupancy better weedays or weekends?

```
df = df.groupby("day_type")["occ_pct"].mean().plot(kind = "bar")
df
```

```
<Axes: xlabel='day_type'>
```



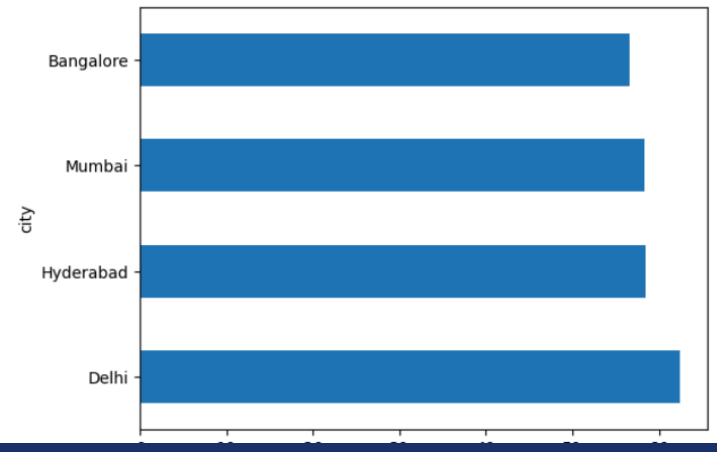
4.In the month of june what is the occupancy for different cities?

```
df = df_june_22.groupby("city")["occ_pct"].mean().round(2).sort_values(ascending=False)
df
```

```
city
Delhi        62.47
Hyderabad    58.46
Mumbai       58.38
Bangalore    56.58
Name: occ_pct, dtype: float64
```

```
df = df_june_22.groupby("city")["occ_pct"].mean().round(2).sort_values(ascending=False).plot(kind = "barh")
df
```

```
<Axes: ylabel='city'>
```



5.How to add august data in the file?

```
[ ]: 5.How to add august data in the file?
```

```
[39]: df_august = pd.read_csv("/users/AAA-09/new_data_august.csv")
      df_august
```

[39]:

	property_id	property_name	category	city	room_category	room_class	check_in_date	mmm yy	week no	day_type	successful_bookings	capacity	occ%
0	16559	Atliq Exotica	Luxury	Mumbai	RT1	Standard	01-Aug-22	Aug-22	W 32	weekeday	30	30	100.00
1	19562	Atliq Bay	Luxury	Bangalore	RT1	Standard	01-Aug-22	Aug-22	W 32	weekeday	21	30	70.00
2	19563	Atliq Palace	Business	Bangalore	RT1	Standard	01-Aug-22	Aug-22	W 32	weekeday	23	30	76.67
3	19558	Atliq Grands	Luxury	Bangalore	RT1	Standard	01-Aug-22	Aug-22	W 32	weekeday	30	40	75.00
4	19560	Atliq City	Business	Bangalore	RT1	Standard	01-Aug-22	Aug-22	W 32	weekeday	20	26	76.92
5	17561	Atliq Blu	Luxury	Mumbai	RT1	Standard	01-Aug-22	Aug-22	W 32	weekeday	18	26	69.23
6	17564	Atliq Seasons	Business	Mumbai	RT1	Standard	01-Aug-22	Aug-22	W 32	weekeday	10	16	62.50

```
[2]: import pandas as pd
     latest_df = pd.concat([df,df_august], ignore_index=True, axis=0)
     latest.df.tail(5)
```

6.print revenue realized per city?

```
6.print revenue realized per city?
```

```
df_bookings_all = pd.merge(df_hotels,df_bookings,on ="property_id")
df_bookings_all.head(4)
```

	property_id	property_name	category	city	booking_id	booking_date	check_in_date	checkout_date	no_guests	room_category	booking_platform	ratings
0	16558	Atliq Grands	Luxury	Delhi	May012216558RT11	27-04-22	1/5/2022	2/5/2022	-3.0	RT1	direct online	
1	16558	Atliq Grands	Luxury	Delhi	May012216558RT12	30-04-22	1/5/2022	2/5/2022	2.0	RT1	others	
2	16558	Atliq Grands	Luxury	Delhi	May012216558RT13	28-04-22	1/5/2022	4/5/2022	2.0	RT1	logtrip	
3	16558	Atliq Grands	Luxury	Delhi	May012216558RT14	28-04-22	1/5/2022	2/5/2022	-2.0	RT1	others	

```
df_bookings_all.groupby("city")["revenue_realized"].sum()
```

```
city
Bangalore    420397050
Delhi        294500318
Hyderabad    325232870
Mumbai       668640991
Name: revenue_realized, dtype: int64
```

7.print month by revenue?

```
: df_bookings_all.groupby("mmm yy")["revenue_realized"].sum()

: mmm yy
  Jul 22    389940912
  Jun 22    377191229
  May 22    408375641
  Name: revenue_realized, dtype: int64
```

8.print revenue realized per hotel type?

```
df_bookings_all2 = pd.merge(df_bookings_all,df_hotels,on = "property_id")
df_bookings_all2.head(4)
```

	property_id	property_name_x	category_x	city_x	booking_id	booking_date	check_in_date	checkout_date	no_guests	room_category	booking_platform	rat
0	16558	Atliq Grands	Luxury	Delhi	May012216558RT11	27-04-22	1/5/2022	2/5/2022	-3.0	RT1	direct online	
1	16558	Atliq Grands	Luxury	Delhi	May012216558RT12	30-04-22	1/5/2022	2/5/2022	2.0	RT1	others	
2	16558	Atliq Grands	Luxury	Delhi	May012216558RT13	28-04-22	1/5/2022	4/5/2022	2.0	RT1	logtrip	
3	16558	Atliq Grands	Luxury	Delhi	May012216558RT14	28-04-22	1/5/2022	2/5/2022	-2.0	RT1	others	

```
df_bookings_all2.groupby("category_y")["revenue_realized"].sum()
```

```
category_y
Business     656019297
Luxury      1052751932
Name: revenue_realized, dtype: int64
```

9.print average rating per city?

```
[ ]: 9.print average rating per city

36]: df_bookings_all.groupby("city")["ratings_given"].mean().round(2)

36]: city
     Bangalore    3.41
     Delhi        3.78
     Hyderabad    3.66
     Mumbai       3.65
     Name: ratings_given, dtype: float64
```

10.print a pie chart of revenue realized per booking platform?

```
10.print a pie chart of revenue realized per booking platform
```

```
df_bookings.groupby("booking_platform")["revenue_realized"].sum().plot(kind = "pie")
```

```
<Axes: ylabel='revenue_realized'>
```