In [1]:

```
import pandas as pd
```

In [2]:

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

In [3]:

```
df.head()
```

Out[3]:

	date	mmm yy	week no	day_type
0	01-May-22	May 22	W 19	weekend
1	02-May-22	May 22	W 19	weekeday
2	03-May-22	May 22	W 19	weekeday
3	04-May-22	May 22	W 19	weekeday
4	05-May-22	May 22	W 19	weekeday

In [4]:

```
import pandas as pd

df_agg = pd.read_csv("fact_aggregated_bookings.csv")
df_bookings = pd.read_csv("fact_bookings.csv")
df_hotels = pd.read_csv("dim_hotels.csv")
df_rooms = pd.read_csv("dim_rooms.csv")
df_date = pd.read_csv("dim_date.csv")
```

In [5]:

```
unique_properties = df_agg['property_id'].unique()
print(unique_properties)
```

```
[16559 19562 19563 17558 16558 17560 19558 19560 17561 16560 16561 16562
 16563 17559 17562 17563 18558 18559 18561 18562 18563 19559 19561 17564
 18560]
```

In [9]:

```
highest_capacity = df_agg.groupby('property_id')['capacity'].max().reset_index()
highest_capacity = highest_capacity.sort_values(by='capacity', ascending=False)
print(highest_capacity.head())
```

```
    property_id  capacity
6         17558      50.0
8         17560      45.0
22        19561      45.0
24        19563      45.0
11        17563      44.0
```

In [16]:

```
print(df_agg.columns.tolist())
```

```
['property_id', 'check_in_date', 'room_category', 'successful_bookings', 'capacity']
```

In [17]:

```
total_bookings = df_agg.groupby('property_id')['successful_bookings'].sum().reset_index(
print(total_bookings)
```

```
    property_id  successful_bookings
0         16558                 3153
1         16559                 7338
2         16560                 4693
3         16561                 4418
4         16562                 4820
5         16563                 7211
6         17558                 5053
7         17559                 6142
8         17560                 6013
9         17561                 5183
10        17562                 3424
11        17563                 6337
12        17564                 3982
13        18558                 4475
14        18559                 5256
15        18560                 6638
16        18561                 6458
17        18562                 7333
18        18563                 4737
19        19558                 4400
20        19559                 4729
21        19560                 6079
22        19561                 5736
23        19562                 5812
24        19563                 5413
```

In [18]:

```
overbooked_days = df_agg[df_agg['successful_bookings'] > df_agg['capacity']]
print(overbooked_days[['check_in_date', 'property_id', 'successful_bookings', 'capacity'
```

```
     check_in_date  property_id  successful_bookings  capacity
3          1-May-22        17558                   30      19.0
12         1-May-22        16563                  100      41.0
4136     11-Jun-22        19558                   50      39.0
6209      2-Jul-22        19560                  123      26.0
8522     25-Jul-22        19559                   35      24.0
9194     31-Jul-22        18563                   20      18.0
```

In [19]:

```
df_bookings.describe()
```

Out[19]:

	property_id	no_guests	ratings_given	revenue_generated	revenue_realized
count	134590.000000	134587.000000	56683.000000	1.345900e+05	134590.000000
mean	18061.113493	2.036170	3.619004	1.537805e+04	12696.123256
std	1093.055847	1.034885	1.235009	9.303604e+04	6928.108124
min	16558.000000	-17.000000	1.000000	6.500000e+03	2600.000000
25%	17558.000000	1.000000	3.000000	9.900000e+03	7600.000000
50%	17564.000000	2.000000	4.000000	1.350000e+04	11700.000000
75%	18563.000000	2.000000	5.000000	1.800000e+04	15300.000000
max	19563.000000	6.000000	5.000000	2.856000e+07	45220.000000

In [20]:

```
df_bookings[df_bookings.no_guests<=0]
```

Out[20]:

	booking_id	property_id	booking_date	check_in_date	checkout_date	no_guests	roo
0	May012216558RT11	16558	27-04-22	1/5/2022	2/5/2022	-3.0	
3	May012216558RT14	16558	28-04-22	1/5/2022	2/5/2022	-2.0	
17924	May122218559RT44	18559	12/5/2022	12/5/2022	14-05-22	-10.0	
18020	May122218561RT22	18561	8/5/2022	12/5/2022	14-05-22	-12.0	
18119	May122218562RT311	18562	5/5/2022	12/5/2022	17-05-22	-6.0	
18121	May122218562RT313	18562	10/5/2022	12/5/2022	17-05-22	-4.0	
56715	Jun082218562RT12	18562	5/6/2022	8/6/2022	13-06-22	-17.0	
119765	Jul202219560RT220	19560	19-07-22	20-07-22	22-07-22	-1.0	
134586	Jul312217564RT47	17564	30-07-22	31-07-22	1/8/2022	-4.0	

In [21]:

```
df_bookings.shape
```

Out[21]:

```
(134590, 12)
```

In [22]:

```
df_bookings[df_bookings.no_guests>0]
```

Out[22]:

	booking_id	property_id	booking_date	check_in_date	checkout_date	no_guests	room
1	May012216558RT12	16558	30-04-22	1/5/2022	2/5/2022	2.0	
2	May012216558RT13	16558	28-04-22	1/5/2022	4/5/2022	2.0	
4	May012216558RT15	16558	27-04-22	1/5/2022	2/5/2022	4.0	
5	May012216558RT16	16558	1/5/2022	1/5/2022	3/5/2022	2.0	
6	May012216558RT17	16558	28-04-22	1/5/2022	6/5/2022	2.0	
...	...	...	...	...	...	...	
134584	Jul312217564RT45	17564	30-07-22	31-07-22	1/8/2022	2.0	
134585	Jul312217564RT46	17564	29-07-22	31-07-22	3/8/2022	1.0	
134587	Jul312217564RT48	17564	30-07-22	31-07-22	2/8/2022	1.0	
134588	Jul312217564RT49	17564	29-07-22	31-07-22	1/8/2022	2.0	
134589	Jul312217564RT410	17564	31-07-22	31-07-22	1/8/2022	2.0	

134578 rows × 12 columns

In [24]:

```
df_bookings = df_bookings[df_bookings.no_guests>0]
df_bookings.shape
```

Out[24]:

(134578, 12)

In [25]:

```
df_bookings.revenue_generated.min(), df_bookings.revenue_generated.max()
```

Out[25]:
(6500, 28560000)

In [26]:

```
avg, std = df_bookings.revenue_generated.mean(), df_bookings.revenue_generated.std()
avg, std
```

Out[26]:
(np.float64(15378.036937686695), 93040.15493143328)

In [27]:

```
higher_limit = avg + 3*std
higher_limit
```

Out[27]:
np.float64(294498.50173198653)

In [29]:

```
lower_limit = avg - 3*std
lower_limit
```

Out[29]:
np.float64(-263742.4278566132)

In [30]:

```
df_bookings[df_bookings.revenue_generated>higher_limit]
```

Out[30]:

	booking_id	property_id	booking_date	check_in_date	checkout_date	no_guests	roo
2	May012216558RT13	16558	28-04-22	1/5/2022	4/5/2022	2.0	
111	May012216559RT32	16559	29-04-22	1/5/2022	2/5/2022	6.0	
315	May012216562RT22	16562	28-04-22	1/5/2022	4/5/2022	2.0	
562	May012217559RT118	17559	26-04-22	1/5/2022	2/5/2022	2.0	
129176	Jul282216562RT26	16562	21-07-22	28-07-22	29-07-22	2.0	

In [31]:

```
df_bookings[df_bookings.revenue_generated<higher_limit]
df_bookings.shape
```

Out[31]:
(134578, 12)

In [32]:

```
df_bookings.revenue_realized.describe()
```

Out[32]:

```
count    134578.000000
mean      12696.011822
std        6927.841641
min        2600.000000
25%        7600.000000
50%       11700.000000
75%       15300.000000
```

```
max       45220.000000
Name: revenue_realized, dtype: float64
```

In [33]:

```
higher_limit = df_bookings.revenue_realized.mean() + 3*df_bookings.revenue_realized.std(
higher_limit
```

Out[33]:

```
np.float64(33479.53674501789)
```

In [34]:

```
df_bookings[df_bookings.revenue_realized>higher_limit]
```

Out[34]:

	booking_id	property_id	booking_date	check_in_date	checkout_date	no_guests	roo
137	May012216559RT41	16559	27-04-22	1/5/2022	7/5/2022	4.0	
139	May012216559RT43	16559	1/5/2022	1/5/2022	2/5/2022	6.0	
143	May012216559RT47	16559	28-04-22	1/5/2022	3/5/2022	3.0	
149	May012216559RT413	16559	24-04-22	1/5/2022	7/5/2022	5.0	
222	May012216560RT45	16560	30-04-22	1/5/2022	3/5/2022	5.0	
...	...	...	...	...	...	...	
134328	Jul312219560RT49	19560	31-07-22	31-07-22	2/8/2022	6.0	
134331	Jul312219560RT412	19560	31-07-22	31-07-22	1/8/2022	6.0	
134467	Jul312219562RT45	19562	28-07-22	31-07-22	1/8/2022	6.0	
134474	Jul312219562RT412	19562	25-07-22	31-07-22	6/8/2022	5.0	
134581	Jul312217564RT42	17564	31-07-22	31-07-22	1/8/2022	4.0	

1299 rows × 12 columns

In [35]:

```
df_bookings[df_bookings.room_category=="RT4"].revenue_realized.describe()
```

Out[35]:

```
count    16071.000000
mean     23439.308444
std       9048.599076
min       7600.000000
25%      19000.000000
50%      26600.000000
75%      32300.000000
max      45220.000000
Name: revenue_realized, dtype: float64
```

In [36]:

```
df_bookings.isnull().sum()
```

Out[36]:

```
booking_id              0
property_id             0
booking_date            0
check_in_date           0
checkout_date           0
```

```
no_guests                0
room_category            0
booking_platform         0
ratings_given        77899
booking_status           0
revenue_generated        0
revenue_realized         0
dtype: int64
```

In [41]:

```
df_agg.head(5)
```

Out[41]:

	property_id	check_in_date	room_category	successful_bookings	capacity
0	16559	1-May-22	RT1	25	30.0
1	19562	1-May-22	RT1	28	30.0
2	19563	1-May-22	RT1	23	30.0
3	17558	1-May-22	RT1	30	19.0
4	16558	1-May-22	RT1	18	19.0

In [48]:

```
df_agg["occ_pct"] = df_agg["successful_bookings"] / df_agg["capacity"]
df_agg["occ_pct"] = df_agg["occ_pct"].apply(lambda x: round(x*100, 2))
df_agg.head(4)
```

Out[48]:

	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

In [50]:

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

Out[50]:

```
room_category
RT1    58.22
RT2    58.04
RT3    58.03
RT4    59.30
Name: occ_pct, dtype: float64
```

In [51]:

```
df = pd.merge(df_agg, df_rooms, left_on="room_category", right_on="room_id")
df.head(4)
```

Out[51]:

	property_id	check_in_date	room_category	successful_bookings	capacity	occ_pct	room_id	room_
0	16559	1-May-22	RT1	25	30.0	83.33	RT1	Sta

	property_id	check_in_date	room_category	successful_bookings	capacity	occ_pct	room_id	room_
1	19562	1-May-22	RT1	28	30.0	93.33	RT1	Sta
2	19563	1-May-22	RT1	23	30.0	76.67	RT1	Sta
3	17558	1-May-22	RT1	30	19.0	157.89	RT1	Sta

In [52]:

```
df.groupby("room_class")["occ_pct"].mean()
```

Out[52]:

```
room_class
Elite           58.040278
Premium         58.028213
Presidential    59.300461
Standard        58.224247
Name: occ_pct, dtype: float64
```

In [53]:

```
df.drop("room_id",axis=1, inplace=True)
df.head(4)
```

Out[53]:

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

In [54]:

```
df_hotels.head(3)
```

Out[54]:

	property_id	property_name	category	city
0	16558	Atliq Grands	Luxury	Delhi
1	16559	Atliq Exotica	Luxury	Mumbai
2	16560	Atliq City	Business	Delhi

In [55]:

```
df = pd.merge(df, df_hotels, on="property_id")
df.head(3)
```

Out[55]:

	property_id	check_in_date	room_category	successful_bookings	capacity	occ_pct	room_class	pr
0	16559	1-May-22	RT1	25	30.0	83.33	Standard	
1	19562	1-May-22	RT1	28	30.0	93.33	Standard	
2	19563	1-May-22	RT1	23	30.0	76.67	Standard	

In [56]:

```
df.groupby("city")["occ_pct"].mean()
```

Out[56]:

```
city
Bangalore    56.594207
Delhi        61.606467
Hyderabad    58.144651
Mumbai       57.936305
Name: occ_pct, dtype: float64
```

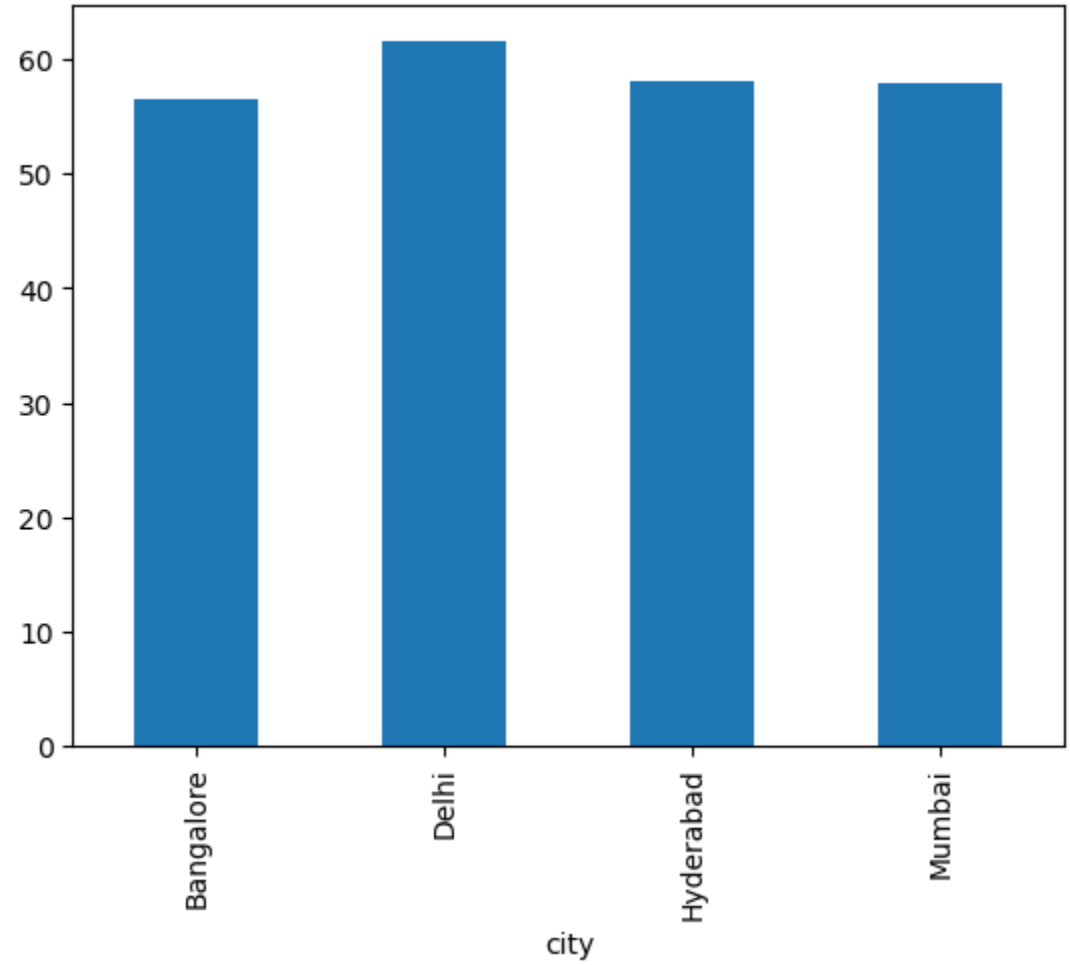
In [57]:

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

```
Matplotlib is building the font cache; this may take a moment.
```

Out[57]:

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



In [58]:

```
df_date.head(3)
```

Out[58]:

	date	mmm yy	week no	day_type
0	01-May-22	May 22	W 19	weekend
1	02-May-22	May 22	W 19	weekeday
2	03-May-22	May 22	W 19	weekeday

In [59]:

```
df = pd.merge(df, df_date, left_on="check_in_date", right_on="date")
df.head(3)
```

Out[59]:

	property_id	check_in_date	room_category	successful_bookings	capacity	occ_pct	room_class	pr
0	19563	10-May-22	RT3	15	29.0	51.72	Premium	
1	18560	10-May-22	RT1	19	30.0	63.33	Standard	
2	19562	10-May-22	RT1	18	30.0	60.00	Standard	

In [60]:

```
df.groupby("day_type")["occ_pct"].mean().round(2)
```

Out[60]:

```
day_type
weekeday    50.90
weekend     72.39
Name: occ_pct, dtype: float64
```

In [61]:

```
df_june_22 = df[df["mmm yy"]=="Jun 22"]
df_june_22.head(4)
```

Out[61]:

	property_id	check_in_date	room_category	successful_bookings	capacity	occ_pct	room_class
2200	16559	10-Jun-22	RT1	20	30.0	66.67	Standard
2201	19562	10-Jun-22	RT1	19	30.0	63.33	Standard
2202	19563	10-Jun-22	RT1	17	30.0	56.67	Standard
2203	17558	10-Jun-22	RT1	9	19.0	47.37	Standard

In [62]:

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

Out[62]:

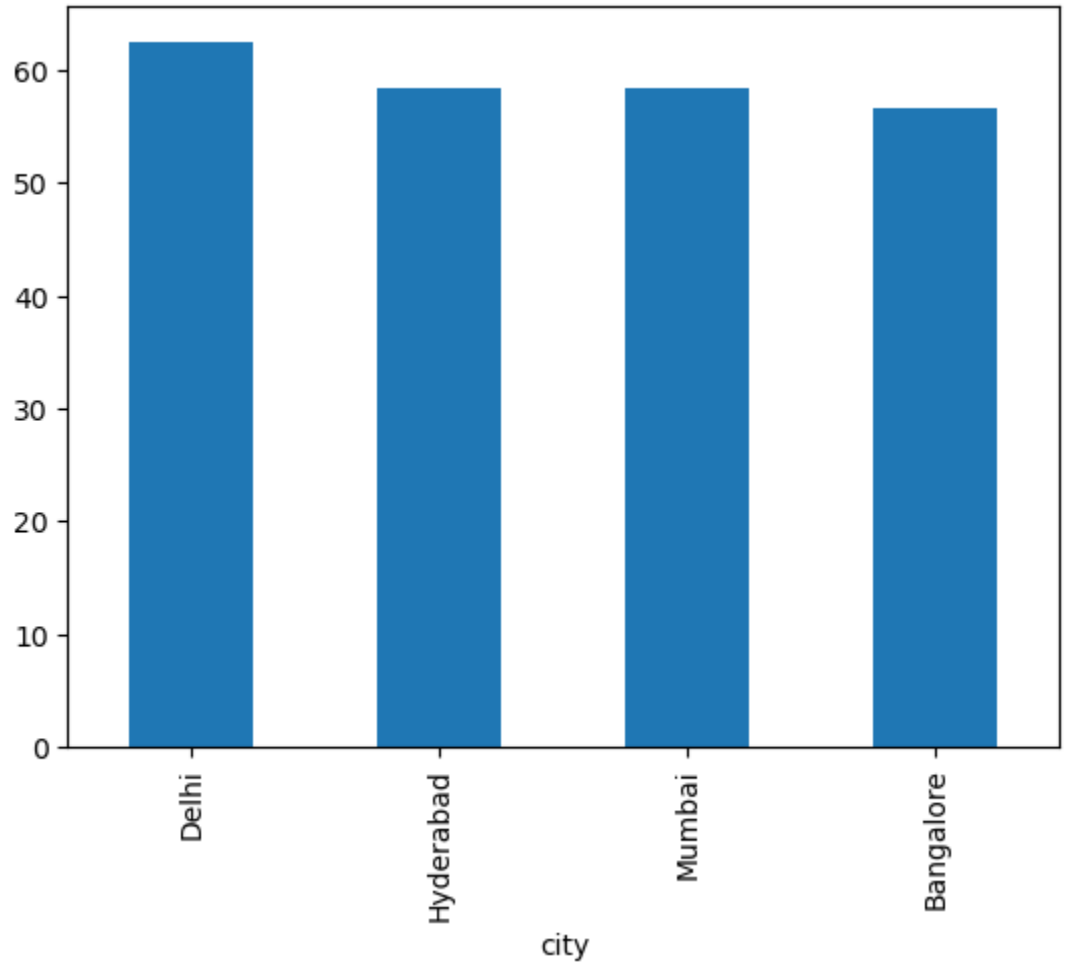
```
city
Delhi        62.47
Hyderabad    58.46
Mumbai       58.38
Bangalore    56.58
Name: occ_pct, dtype: float64
```

In [63]:

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

Out[63]:

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



In [ ]: