In [24]:
```
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns
pd.set_option('display.max_rows', 500)
pd.set_option('display.max_columns', 500)
pd.set_option('display.width', 1000)
import numpy as np
```

Assumsion:

- This year is 2020
- We are travel booking company and we provide the flight booking and hotel booking services for all flight and hotel over the world
- We are charging 10% commision for each booking (flight and hotel).
- So in this analysis we are trying to figure out is there any way to increase our revenue and trade off if any
- Flight:
  - We assume each flight from A --> B then go back B --> A is 1 round trip flight

Data Preparation

In [2]:
```
flight_df = pd.read_csv('./data/flights.csv')
hotel_df = pd.read_csv('./data/hotels.csv')
user_df = pd.read_csv('./data/users.csv')
```

User data

In [3]:
```
user_df.head()
```

Out[3]:

	code	company	name	gender	age
0	0	4You	Roy Braun	male	21
1	1	4You	Joseph Holsten	male	37
2	2	4You	Wilma Mcinnis	female	48
3	3	4You	Paula Daniel	female	23
4	4	4You	Patricia Carson	female	44

Flight data

In [4]:
```
flight_df['date'] = pd.to_datetime(flight_df['date'])
flight_df['month']= pd.DatetimeIndex(flight_df['date']).month
flight_df['year']= pd.DatetimeIndex(flight_df['date']).year
flight_df['flight_trip']= flight_df.groupby('travelCode')['travelCode'].rank(method=
#flight_df['flight_trip_by_user']= flight_df.groupby('userCode')['travelCode'].rank(
departure_flight_df = flight_df[(flight_df['flight_trip']==1)]
arrival_flight_df = flight_df[(flight_df['flight_trip']==2)]
new_flight_df = departure_flight_df.merge(arrival_flight_df, left_on =['travelCode',
print(len(new_flight_df))
new_flight_df.head()
```

```
135944
```

Out[4]:

	travelCode	userCode	from_departure	to_departure	flightType_departure	price_departure	time_

	travelCode	userCode	from_departure	to_departure	flightType_departure	price_departure	time_
0	0	0	Recife (PE)	Florianopolis (SC)	firstClass	1434.38	
1	1	0	Brasilia (DF)	Florianopolis (SC)	firstClass	1487.52	
2	2	0	Aracaju (SE)	Salvador (BH)	firstClass	1684.05	
3	3	0	Aracaju (SE)	Campo Grande (MS)	economic	743.54	
4	4	0	Recife (PE)	Florianopolis (SC)	economic	803.39	

Hotel Data

In [5]:
```
hotel_df['month']= pd.DatetimeIndex(hotel_df['date']).month
hotel_df['year']= pd.DatetimeIndex(hotel_df['date']).year
hotel_df['hotel_revenue'] = hotel_df['total']*0.1
hotel_df.head()
```

Out[5]:

	travelCode	userCode	name	place	days	price	total	date	month	year	hote
0	0	0	Hotel A	Florianopolis (SC)	4	313.02	1252.08	09/26/2019	9	2019	
1	2	0	Hotel K	Salvador (BH)	2	263.41	526.82	10/10/2019	10	2019	
2	7	0	Hotel K	Salvador (BH)	3	263.41	790.23	11/14/2019	11	2019	
3	11	0	Hotel K	Salvador (BH)	4	263.41	1053.64	12/12/2019	12	2019	
4	13	0	Hotel A	Florianopolis (SC)	1	313.02	313.02	12/26/2019	12	2019	

Combine User, flight and hotel data

In [6]:
```
new_df = user_df.merge(new_flight_df, left_on ='code', right_on ='userCode', how ='l
new_df = new_df.merge(hotel_df, left_on =['code', 'travelCode'], right_on =['userCod
new_df = new_df.drop(columns=['userCode','userCode_hotel'])
new_df = new_df.rename(columns={'travelCode_user':'travelCode'})
new_df['date_diff'] = new_df['date_arrival'] - new_df['date_departure']
new_df['date_diff'] = new_df['date_diff'] / np.timedelta64(1, 'D')
new_df.loc[new_df['travelCode'].isnull(),'flight'] = 'No'
new_df.loc[new_df['travelCode'].notnull(), 'flight'] = 'Yes'
new_df.loc[new_df['name_hotel'].isnull(),'flight_with_hotel'] = 'No'
new_df.loc[new_df['name_hotel'].notnull(), 'flight_with_hotel'] = 'Yes'
new_df['flight_revenue'] = (new_df['price_departure']*0.1) + (new_df['price_arrival'
new_df.loc[new_df['hotel_revenue'].isnull(),'hotel_revenue'] = 0
new_df['total_revenue'] = new_df['flight_revenue'] +new_df['hotel_revenue']
new_df = new_df[(new_df['flight']=='Yes')]
print(len(new_df))
new_df.head(10)
```

```
135944
```

Out[6]:

	code	company	name	gender	age	travelCode	from_departure	to_departure	flightType_depar
0	0	4You	Roy Braun	male	21	0.0	Recife (PE)	Florianopolis (SC)	first(
1	0	4You	Roy Braun	male	21	1.0	Brasilia (DF)	Florianopolis (SC)	first(
2	0	4You	Roy Braun	male	21	2.0	Aracaju (SE)	Salvador (BH)	first(
3	0	4You	Roy Braun	male	21	3.0	Aracaju (SE)	Campo Grande (MS)	econ(
4	0	4You	Roy Braun	male	21	4.0	Recife (PE)	Florianopolis (SC)	econ(
5	0	4You	Roy Braun	male	21	5.0	Brasilia (DF)	Aracaju (SE)	first(
6	0	4You	Roy Braun	male	21	6.0	Recife (PE)	Florianopolis (SC)	pren
7	0	4You	Roy Braun	male	21	7.0	Aracaju (SE)	Salvador (BH)	econ(
8	0	4You	Roy Braun	male	21	8.0	Recife (PE)	Sao Paulo (SP)	econ(
9	0	4You	Roy Braun	male	21	9.0	Brasilia (DF)	Campo Grande (MS)	econ(

In [17]:

```
revenue_info = new_df[['total_revenue','year_departure']].groupby('year_departure').
revenu_info1 = new_df[['flight_revenue','year_departure']].groupby('year_departure')
revenu_info2 = new_df[['hotel_revenue','year_departure']].groupby('year_departure').
flight_info = new_df[['travelCode','year_departure']].groupby('year_departure').coun
user_info0 = new_df[['code','year_departure']].groupby(['code','year_departure']).co
user_info0 = user_info0.groupby('year_departure').count().reset_index().rename(colum
revenue_info = revenue_info.merge(flight_info, on ='year_departure')
revenue_info = revenue_info.merge(revenu_info1, on ='year_departure')
revenue_info = revenue_info.merge(revenu_info2, on ='year_departure')
revenue_info = revenue_info.merge(user_info0, on ='year_departure')
revenue_info['avg_flight_per_user'] = revenue_info['total_flight']/revenue_info['tot
revenue_info['avg_flight_revenue_per_user'] = revenue_info['flight_revenue']/revenue
revenue_info
```

Out[17]:

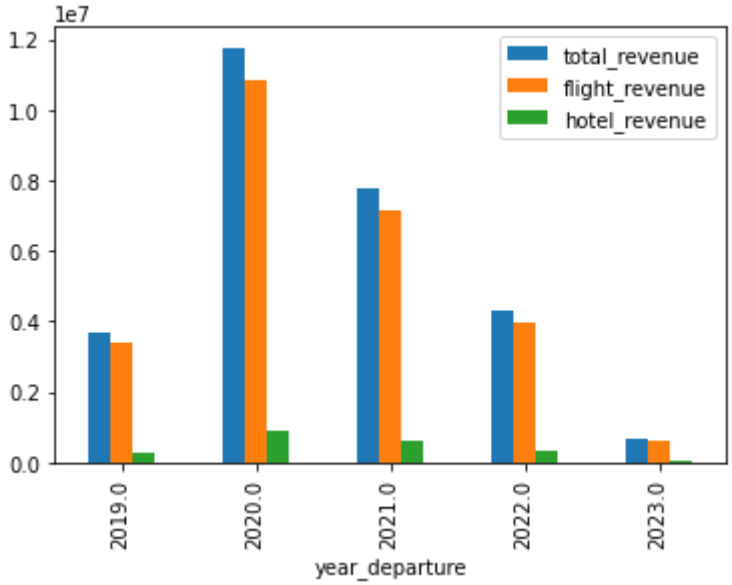
	year_departure	total_revenue	total_flight	flight_revenue	hotel_revenue	total_user_flight	avg_fli
0	2019.0	3.698317e+06	17913	3.412428e+06	285889.214	1335	
1	2020.0	1.176454e+07	56735	1.085945e+07	905089.748	1233	
2	2021.0	7.767330e+06	37449	7.167351e+06	599979.705	896	
3	2022.0	4.321939e+06	20716	3.989000e+06	332939.678	553	
4	2023.0	6.522739e+05	3131	6.016543e+05	50619.576	211	

In [18]:

```
revenue_info[['total_revenue','flight_revenue','hotel_revenue','year_departure']].pl
```

Out[18]:

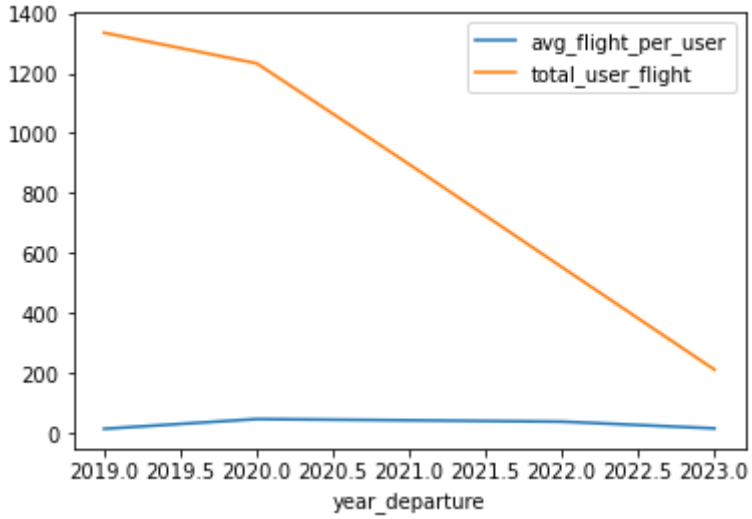
```
<AxesSubplot:xlabel='year_departure'>
```



From the chart we saw that most of the revenue come from flight revenue but the revenue will decrease in 2021, 2022 and 2023

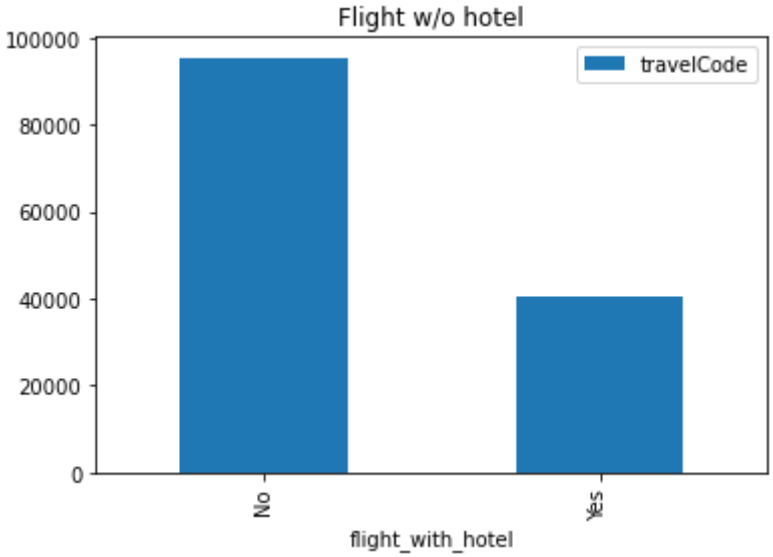
In [19]:
```
revenue_info[['avg_flight_per_user','total_user_flight','year_departure']].plot(x= '
```

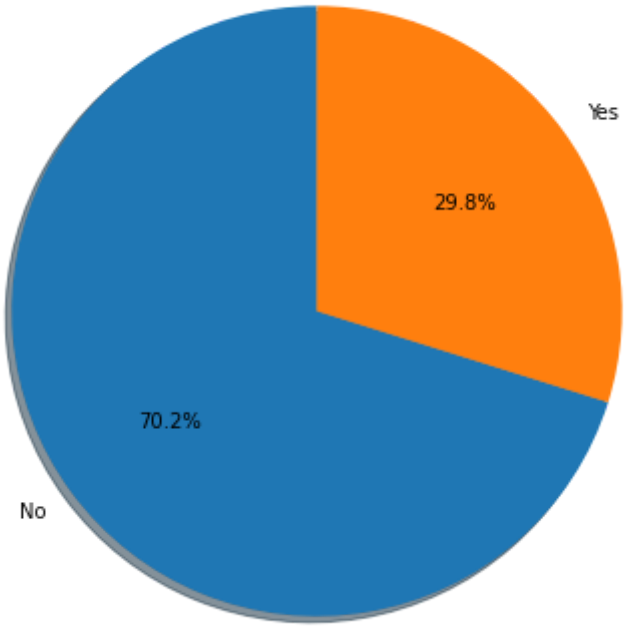
Out[19]: <AxesSubplot:xlabel='year_departure'>



Total unique user flight will be decreased in 2021 to 2023 but the Avg flight per user slightly remain in 2021 and 2023. it mean there are a group of user who flight more than other

In [10]:
```
flight_w_hotel_df = new_df[['travelCode','flight_with_hotel']].groupby(['travelCode'
flight_w_hotel_df.groupby('flight_with_hotel').count().reset_index().plot(kind='bar'
labels = flight_w_hotel_df['flight_with_hotel'].value_counts(sort = True).index
sizes = flight_w_hotel_df['flight_with_hotel'].value_counts(sort = True)
plt.figure(figsize=(7,7))
plt.pie(sizes, labels=labels, autopct='%1.1f%%', shadow=True, startangle=90,)
plt.title('Flight w/o Hotel distribution')
plt.show()
```





Interm of hotel. 70% number of flight without booking hotel so if we can increase the number of flight with hotel booking we can increase the hotel revenue and total revenue

We will split the flight with and without hotel to see what difference between them

In [11]:
```
flight_w_hotel = new_df[(new_df['flight_with_hotel']=='Yes')]
flight_wo_hotel = new_df[(new_df['flight_with_hotel']=='No')]
flight_wo_hotel = flight_wo_hotel[(flight_wo_hotel['year_departure']>2020)]
flight_wo_hotel.head()
```

Out[11]:

	code	company	name	gender	age	travelCode	from_departure	to_departure	flightType_depa
67	0	4You	Roy Braun	male	21	67.0	Recife (PE)	Aracaju (SE)	eco
68	0	4You	Roy Braun	male	21	68.0	Recife (PE)	Salvador (BH)	firs

	code	company	name	gender	age	travelCode	from_departure	to_departure	flightType_depa
69	0	4You	Roy Braun	male	21	69.0	Recife (PE)	Sao Paulo (SP)	pre
70	0	4You	Roy Braun	male	21	70.0	Aracaju (SE)	Sao Paulo (SP)	firs
71	0	4You	Roy Braun	male	21	71.0	Brasilia (DF)	Salvador (BH)	firs

To boost the hotel revenue, We will create a simple program to:

- Offer the customer in same company, departure, arrival, same lenght of stay date but without hotel booking
- To stay in the same hotel with same price to see how much revenue do we generated in 2021 onward

In [12]:
```
flight_w_hotel_offer = flight_wo_hotel[['company','from_departure','to_departure','f
flight_w_hotel_offer= flight_w_hotel_offer.drop_duplicates()
additional_revenue = flight_w_hotel_offer[['hotel_revenue','year_departure']].groupb
print('Additional revenue will be generated if we offer hotel booking is',round(addi
additional_revenue
```

```
Additional revenue will be generated if we offer hotel booking is 509305.66
```

Out[12]:

	year_departure	additional_revenue
0	2021.0	307793.344
1	2022.0	182919.654
2	2023.0	18592.660

In [22]:
```
revenue_info = revenue_info.merge(additional_revenue, on ='year_departure', how ='le
revenue_info.loc[revenue_info['additional_revenue'].isnull(),'additional_revenue'] =
revenue_info['total_revenue_new'] = revenue_info['flight_revenue']+revenue_info['hot
revenue_info
```

Out[22]:

	year_departure	total_revenue	total_flight	flight_revenue	hotel_revenue	total_user_flight	avg_fli
0	2019.0	3.698317e+06	17913	3.412428e+06	285889.214	1335	
1	2020.0	1.176454e+07	56735	1.085945e+07	905089.748	1233	
2	2021.0	7.767330e+06	37449	7.167351e+06	599979.705	896	
3	2022.0	4.321939e+06	20716	3.989000e+06	332939.678	553	
4	2023.0	6.522739e+05	3131	6.016543e+05	50619.576	211	

- We will be generated more than 500k revenue if we offer hotel booking to customer. But life is not dream, we have to give some promo to customer to make the offer more interesting.
- Example if we give 10$ discount on total price for every booking how much does it cost and how much revenue do we generated

In [23]:
```
flight_w_hotel_offer1 = flight_wo_hotel[['company','from_departure','to_departure','
flight_w_hotel_offer1= flight_w_hotel_offer1.drop_duplicates()
flight_w_hotel_offer1['cost'] = 10
flight_w_hotel_offer1['hotel_revenue_new'] = flight_w_hotel_offer1['hotel_revenue']
additional_revenue1 = flight_w_hotel_offer1[['hotel_revenue_new','year_departure']].
additional_cost1 = flight_w_hotel_offer1[['cost','year_departure']].groupby('year_de
print('Additional revenue will be generated if we offer hotel booking with 10$ vouch
additional_revenue1
```

```
Additional revenue will be generated if we offer hotel booking with 10$ voucher off
in total price for each booking is 414285.66
And the cost of this promo campain is 95020
```

Out[23]:

	year_departure	additional_revenue_senario_2
0	2021.0	250153.344
1	2022.0	149159.654
2	2023.0	14972.660

In this case our revenue generated has been decreased because we already spent 95k for promo voucher

Besides this senario we can create other senario such as:

- Offer promo based on number of day stay at hotel
- Offer based on the frequency of user travel
- Offer based on difference group of user
- Offer user to buy subscription voucher to have discount on flight and hotel

There are lot of senario we can think about and estimate but for each senario we need to consider on trade off on cost and user engagement then we can do AB test to see hoow does it work

In [ ]:
```

```