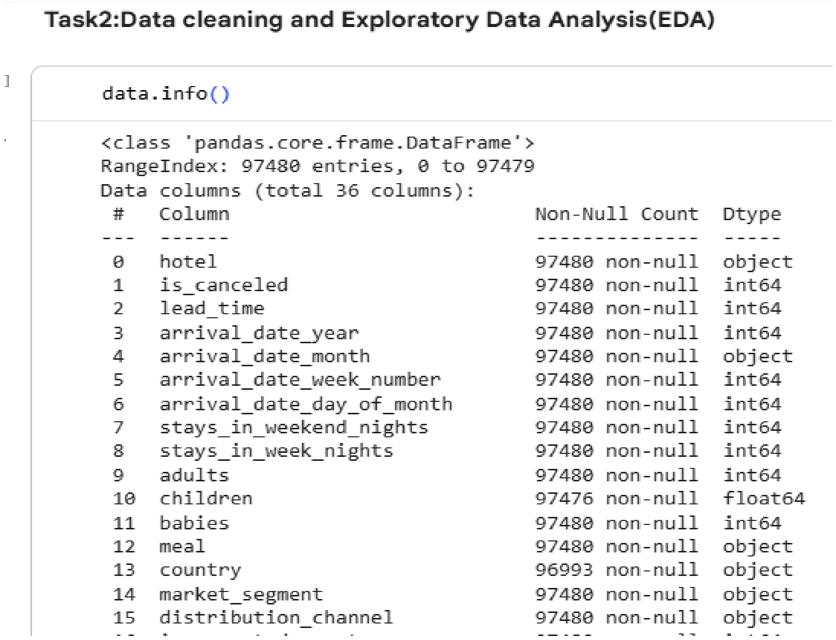
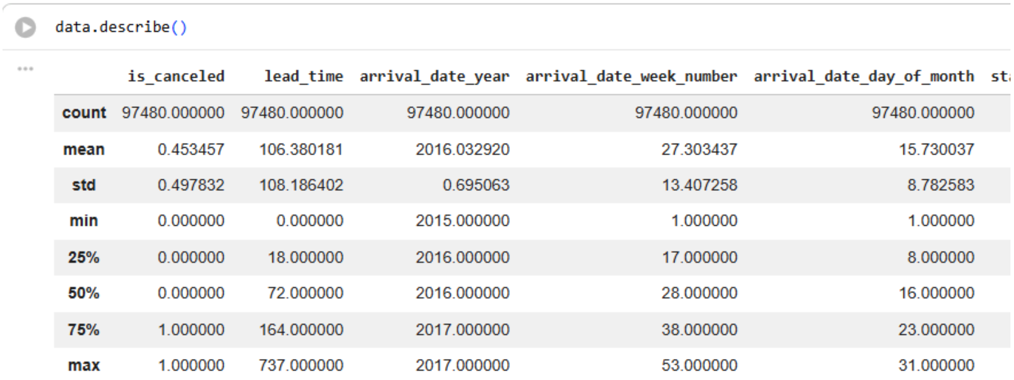
Task no 2



Task2:Data cleaning and Exploratory Data Analysis(EDA)

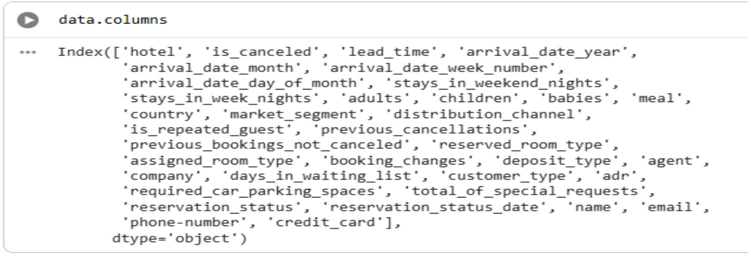
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 97480 entries, 0 to 97479
Data columns (total 36 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   hotel                           97480 non-null  object
 1   is_canceled                     97480 non-null  int64
 2   lead_time                       97480 non-null  int64
 3   arrival_date_year               97480 non-null  int64
 4   arrival_date_month              97480 non-null  object
 5   arrival_date_week_number        97480 non-null  int64
 6   arrival_date_day_of_month       97480 non-null  int64
 7   stays_in_weekend_nights         97480 non-null  int64
 8   stays_in_week_nights            97480 non-null  int64
 9   adults                          97480 non-null  int64
 10  children                        97476 non-null  float64
 11  babies                          97480 non-null  int64
 12  meal                            97480 non-null  object
 13  country                         96993 non-null  object
 14  market_segment                  97480 non-null  object
 15  distribution_channel            97480 non-null  object
```



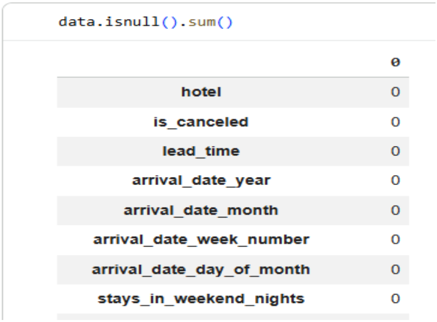
```
data.describe()
```

	is_canceled	lead_time	arrival_date_year	arrival_date_week_number	arrival_date_day_of_month	st
count	97480.000000	97480.000000	97480.000000	97480.000000	97480.000000	
mean	0.453457	106.380181	2016.032920	27.303437	15.730037	
std	0.497832	108.186402	0.695063	13.407258	8.782583	
min	0.000000	0.000000	2015.000000	1.000000	1.000000	
25%	0.000000	18.000000	2016.000000	17.000000	8.000000	
50%	0.000000	72.000000	2016.000000	28.000000	16.000000	
75%	1.000000	164.000000	2017.000000	38.000000	23.000000	
max	1.000000	737.000000	2017.000000	53.000000	31.000000	



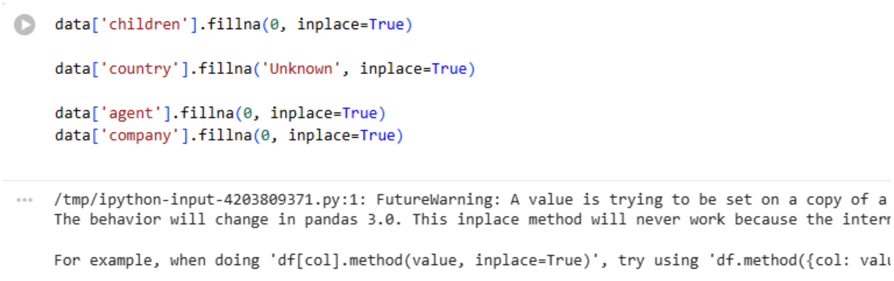
```
data.columns
```

```
Index(['hotel', 'is_canceled', 'lead_time', 'arrival_date_year',
       'arrival_date_month', 'arrival_date_week_number',
       'arrival_date_day_of_month', 'stays_in_weekend_nights',
       'stays_in_week_nights', 'adults', 'children', 'babies', 'meal',
       'country', 'market_segment', 'distribution_channel',
       'is_repeated_guest', 'previous_cancellations',
       'previous_bookings_not_canceled', 'reserved_room_type',
       'assigned_room_type', 'booking_changes', 'deposit_type', 'agent',
       'company', 'days_in_waiting_list', 'customer_type', 'adr',
       'required_car_parking_spaces', 'total_of_special_requests',
       'reservation_status', 'reservation_status_date', 'name', 'email',
       'phone-number', 'credit_card'],
      dtype='object')
```



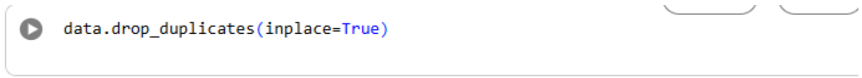
```
data.isnull().sum()
```

	0
hotel	0
is_canceled	0
lead_time	0
arrival_date_year	0
arrival_date_month	0
arrival_date_week_number	0
arrival_date_day_of_month	0
stays_in_weekend_nights	0

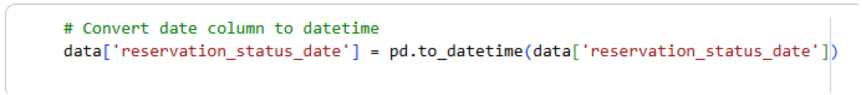


```
data['children'].fillna(0, inplace=True)

data['country'].fillna('Unknown', inplace=True)

data['agent'].fillna(0, inplace=True)
data['company'].fillna(0, inplace=True)
```

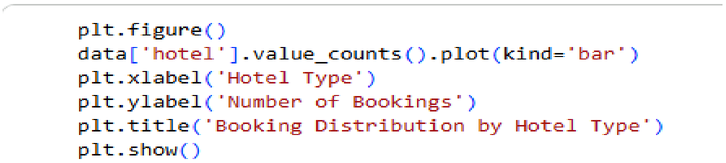
```
/tmp/ipython-input-4203809371.py:1: FutureWarning: A value is trying to be set on a copy of a
The behavior will change in pandas 3.0. This inplace method will never work because the intern

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: valu
```



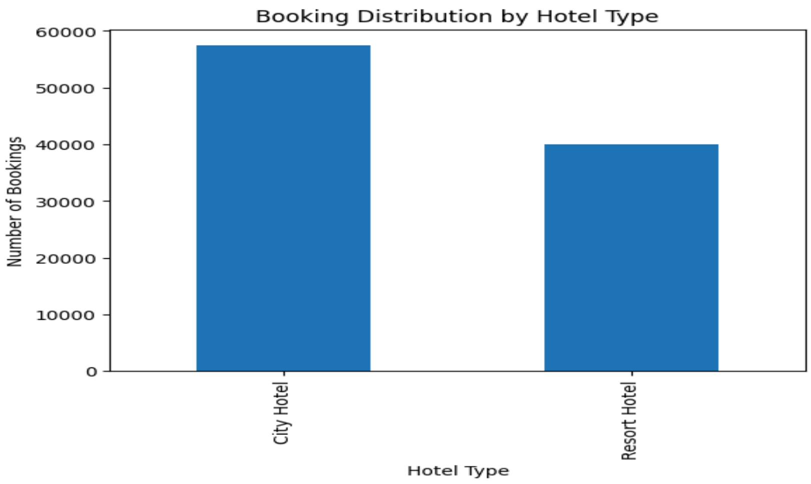
```
data.drop_duplicates(inplace=True)
```

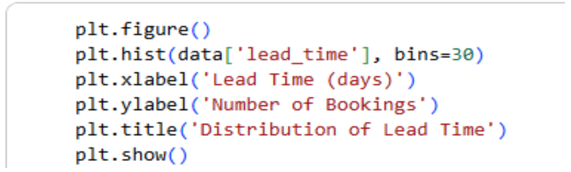


```
# Convert date column to datetime
data['reservation_status_date'] = pd.to_datetime(data['reservation_status_date'])
```

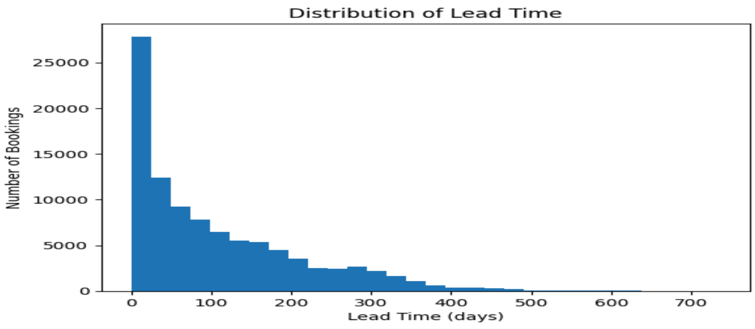


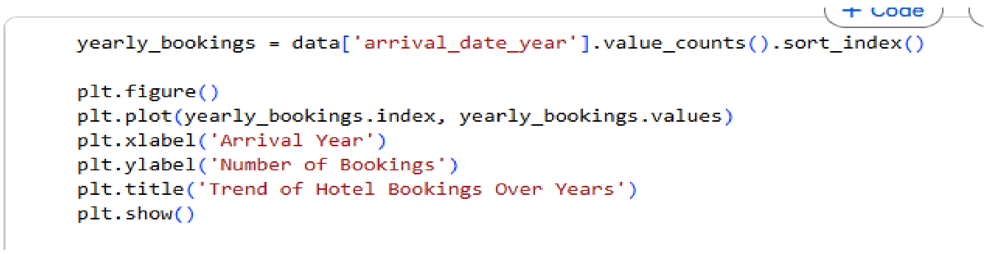
```
plt.figure()
data['hotel'].value_counts().plot(kind='bar')
plt.xlabel('Hotel Type')
plt.ylabel('Number of Bookings')
plt.title('Booking Distribution by Hotel Type')
plt.show()
```



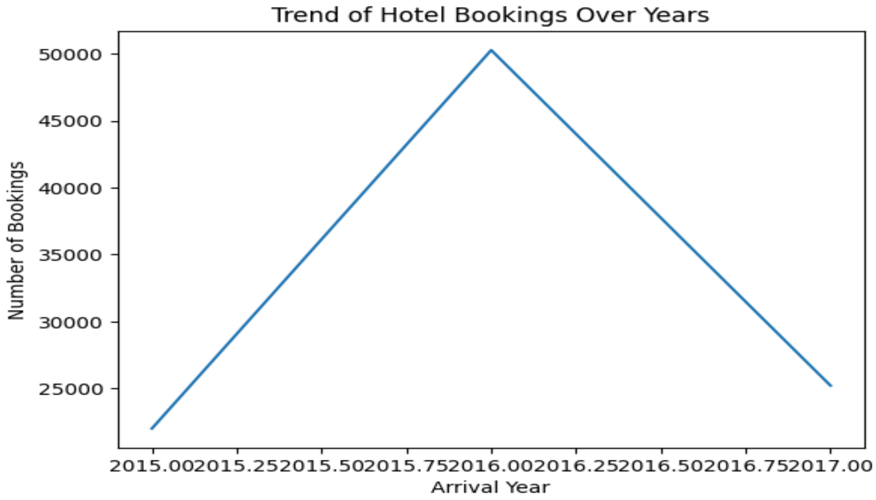


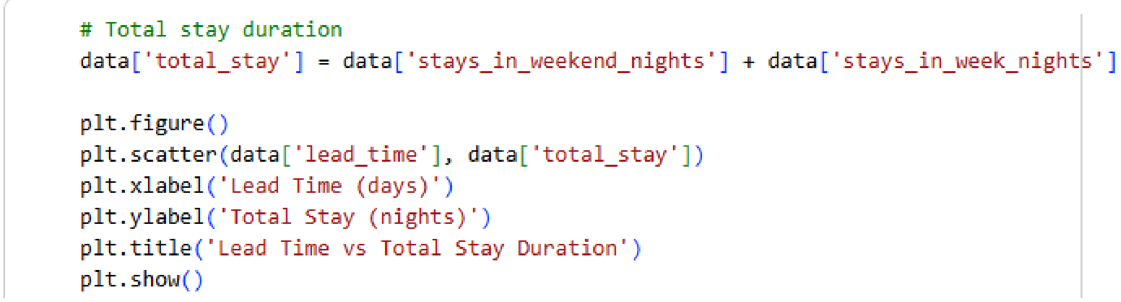
```
plt.figure()
plt.hist(data['lead_time'], bins=30)
plt.xlabel('Lead Time (days)')
plt.ylabel('Number of Bookings')
plt.title('Distribution of Lead Time')
plt.show()
```



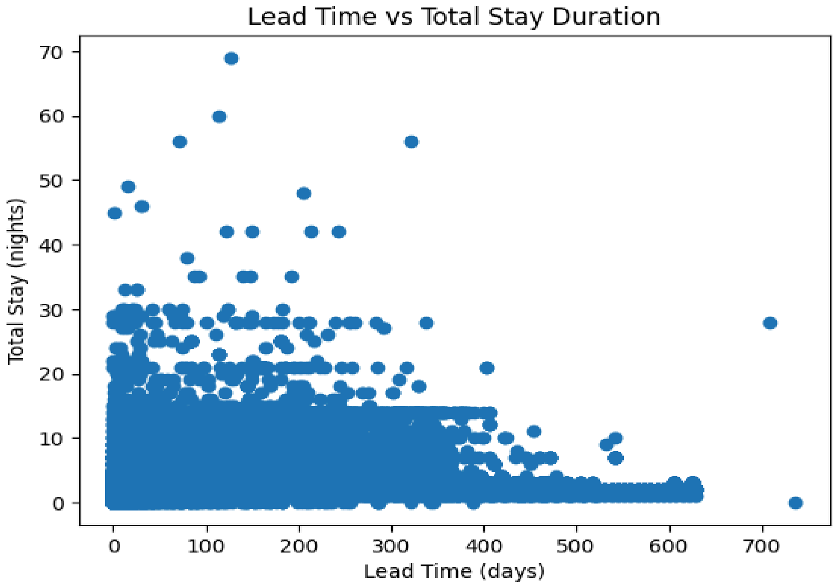


```
yearly_bookings = data['arrival_date_year'].value_counts().sort_index()

plt.figure()
plt.plot(yearly_bookings.index, yearly_bookings.values)
plt.xlabel('Arrival Year')
plt.ylabel('Number of Bookings')
plt.title('Trend of Hotel Bookings Over Years')
plt.show()
```





```
# Total stay duration
data['total_stay'] = data['stays_in_weekend_nights'] + data['stays_in_week_nights']

plt.figure()
plt.scatter(data['lead_time'], data['total_stay'])
plt.xlabel('Lead Time (days)')
plt.ylabel('Total Stay (nights)')
plt.title('Lead Time vs Total Stay Duration')
plt.show()
```


Lead Time vs Total Stay Duration
Total Stay (nights)
Lead Time (days)