In [11]:
```
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
```

In [2]:
```
df = pd.read_csv('Downloads/Salaries.csv')
```

In [3]:
```
df
```

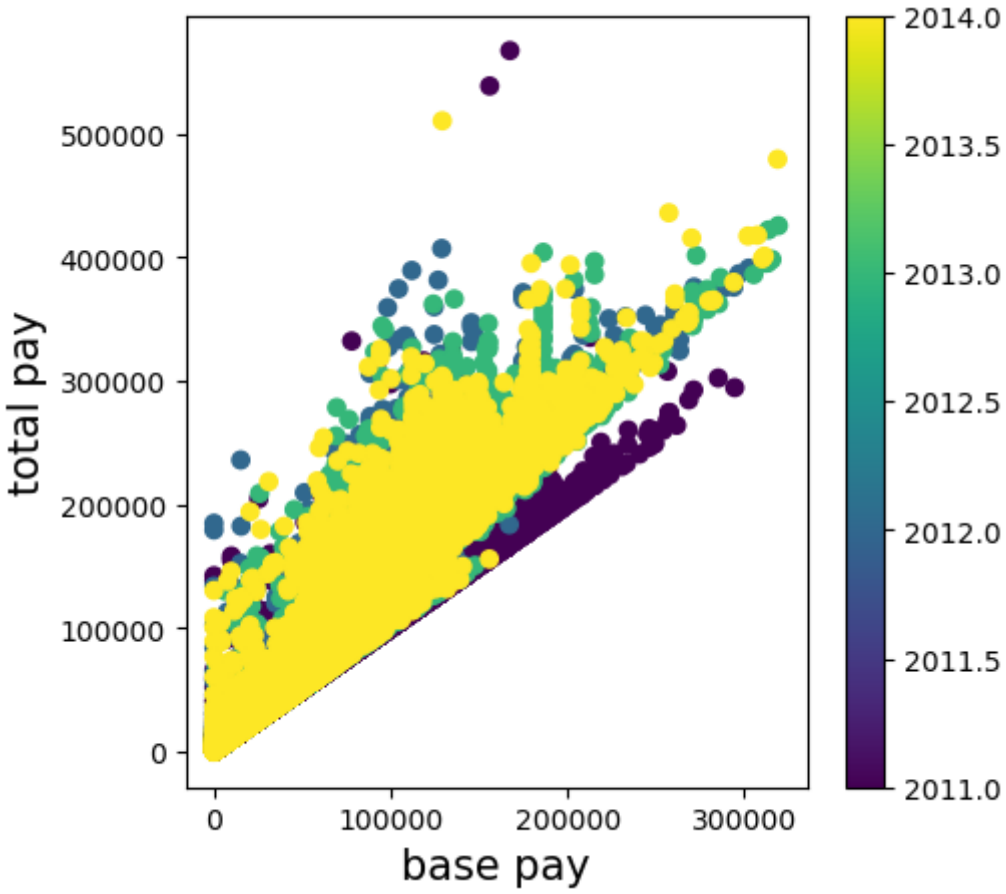
Out[3]:

	Id	EmployeeName	JobTitle	BasePay	OvertimePay	OtherPay	Benefits	T
0	1	NATHANIEL FORD	GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY	167411.18	0.00	400184.25	NaN	56
1	2	GARY JIMENEZ	CAPTAIN III (POLICE DEPARTMENT)	155966.02	245131.88	137811.38	NaN	53
2	3	ALBERT PARDINI	CAPTAIN III (POLICE DEPARTMENT)	212739.13	106088.18	16452.60	NaN	33
3	4	CHRISTOPHER CHONG	WIRE ROPE CABLE MAINTENANCE MECHANIC	77916.00	56120.71	198306.90	NaN	33
4	5	PATRICK GARDNER	DEPUTY CHIEF OF DEPARTMENT, (FIRE DEPARTMENT)	134401.60	9737.00	182234.59	NaN	32
...	...	...	...	...	...	...	...	
148649	148650	Roy I Tillery	Custodian	0.00	0.00	0.00	0.0	
148650	148651	Not provided	Not provided	NaN	NaN	NaN	NaN	
148651	148652	Not provided	Not provided	NaN	NaN	NaN	NaN	
148652	148653	Not provided	Not provided	NaN	NaN	NaN	NaN	
148653	148654	Joe Lopez	Counselor, Log Cabin Ranch	0.00	0.00	-618.13	0.0	

148654 rows × 13 columns

In [4]:
```
basepay=df['BasePay']
totalpay=df['TotalPayBenefits']
year=df['Year']
plt.figure(figsize=(5,5))
plt.scatter(basepay,totalpay,c = year)
plt.colorbar()
plt.xlabel('base pay',fontsize=15)
```

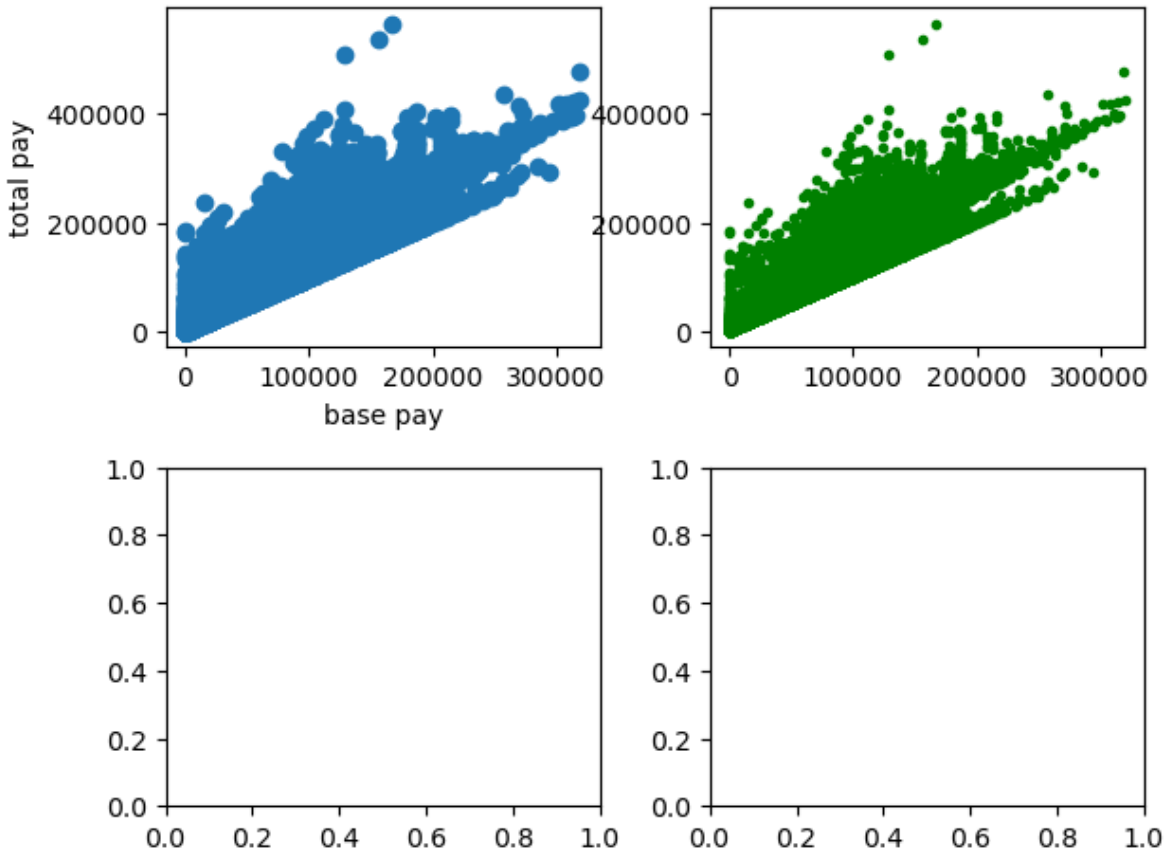
```
plt.ylabel('total pay',fontsize=15)
plt.show()
plt.tight_layout()
```




```
<Figure size 640x480 with 0 Axes>
```

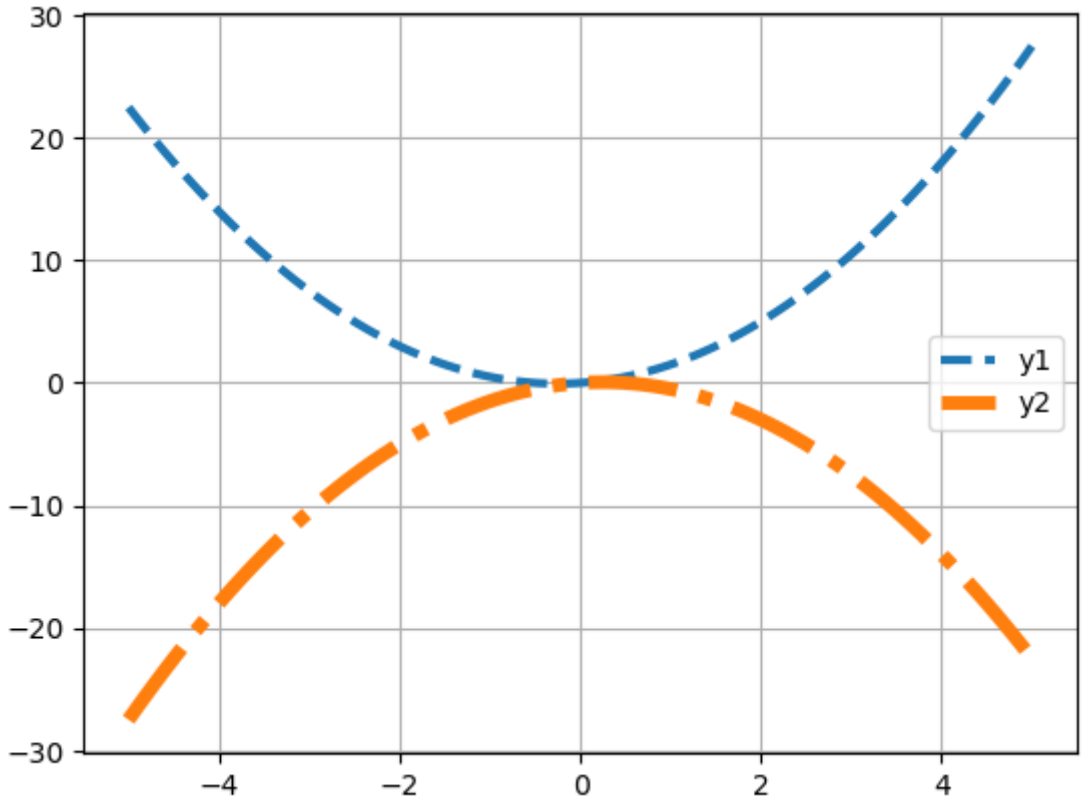
In [5]:
```
fig,axis=plt.subplots(2,2)
axis[0,0].scatter(basepay,totalpay)
axis[0,0].set_xlabel("base pay")
axis[0,0].set_ylabel("total pay")
plt.tight_layout()


axis[0,1].plot(basepay,totalpay,'.',color='green')
```

Out[5]:
```
[<matplotlib.lines.Line2D at 0x1acaccef610>]
```

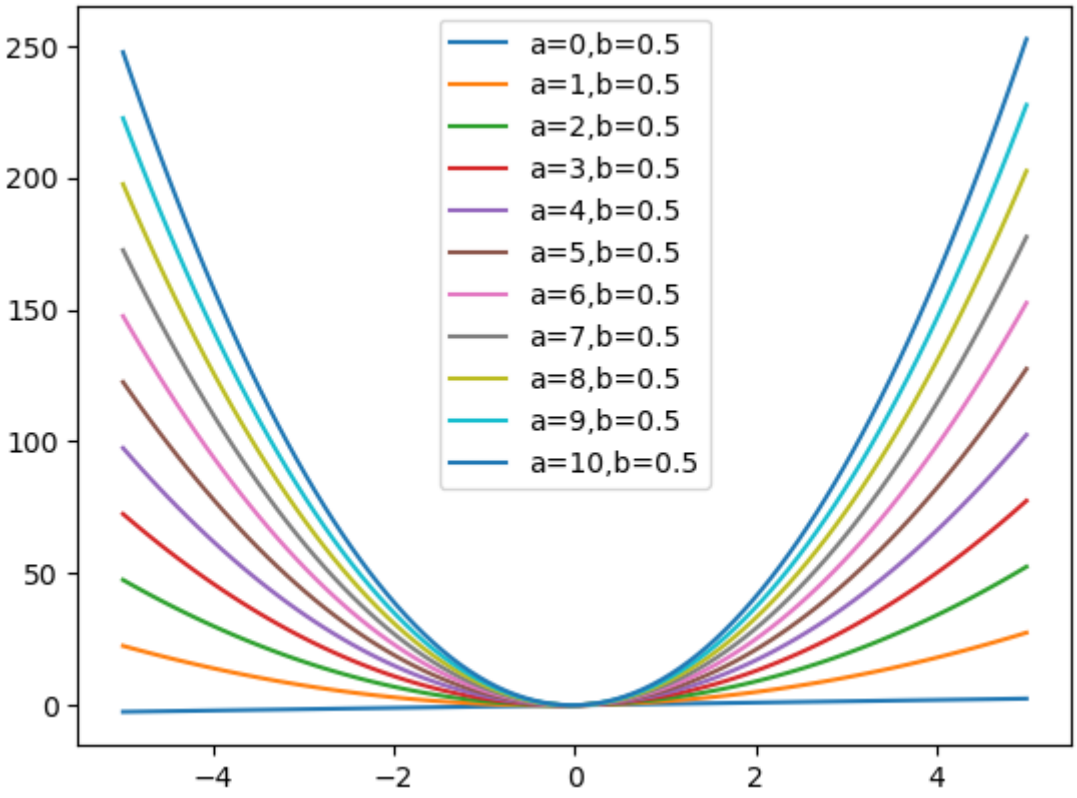


In [ ]:

In [6]:
```
def f(x,a,b):
    return a*x**2+b*x


x=np.linspace(-5,5,100)

a,b=1,0.5
y1 = f(x,a,b)
a,b=-1,0.5
y2 = f(x,a,b)

plt.plot(x,y1,label='y1',ls='--',lw=3)
plt.plot(x,y2,label='y2',ls='-.',lw=5)
plt.legend()
plt.grid()
plt.show()
```

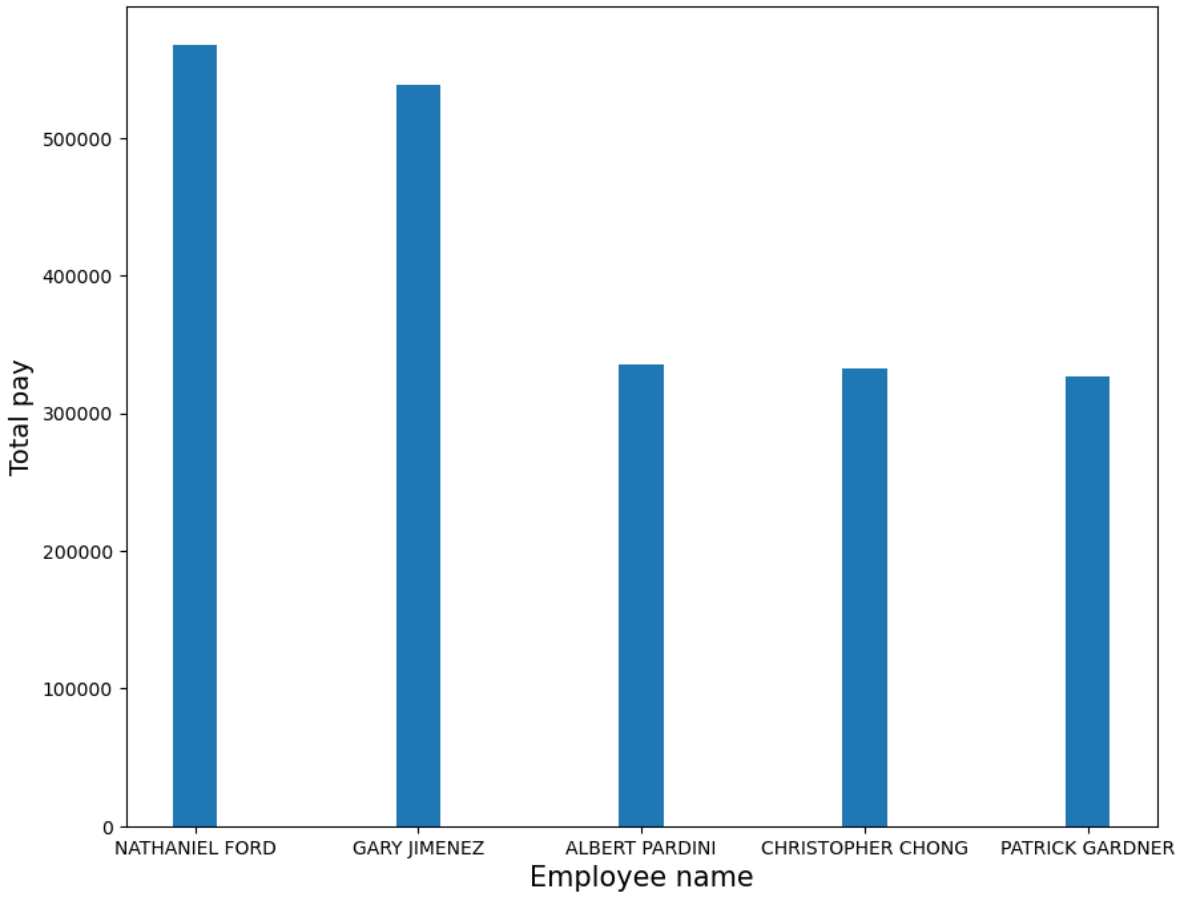


In [7]:
```
for i in range(11):
    y=f(x,i,b)
    plt.plot(x,y,label='a={},b={}'.format(i,0.5))
    plt.legend()
plt.show()
```

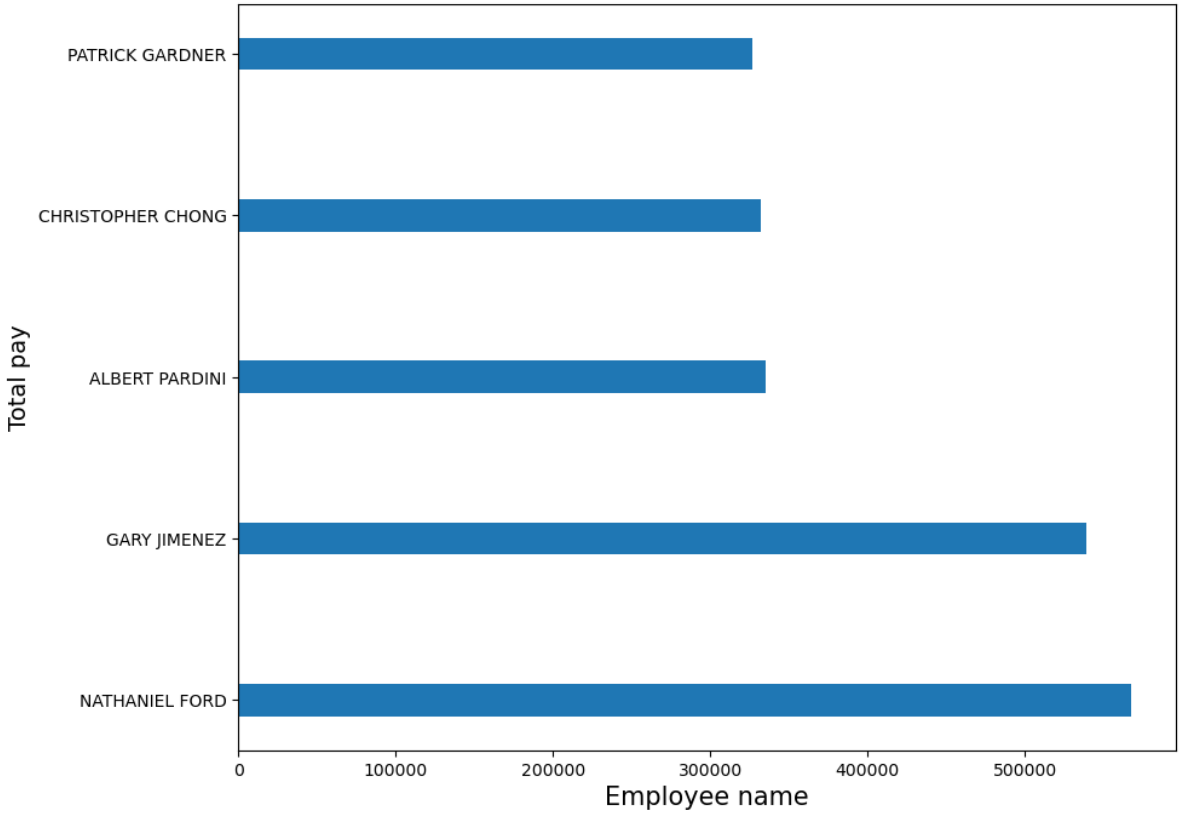


In [8]:
```
name=df['EmployeeName'].head(5)
pay=df['TotalPayBenefits'].head(5)
plt.figure(figsize=(10,8))
plt.bar(name,pay,width=0.2)
plt.xlabel('Employee name',fontsize=15)
```

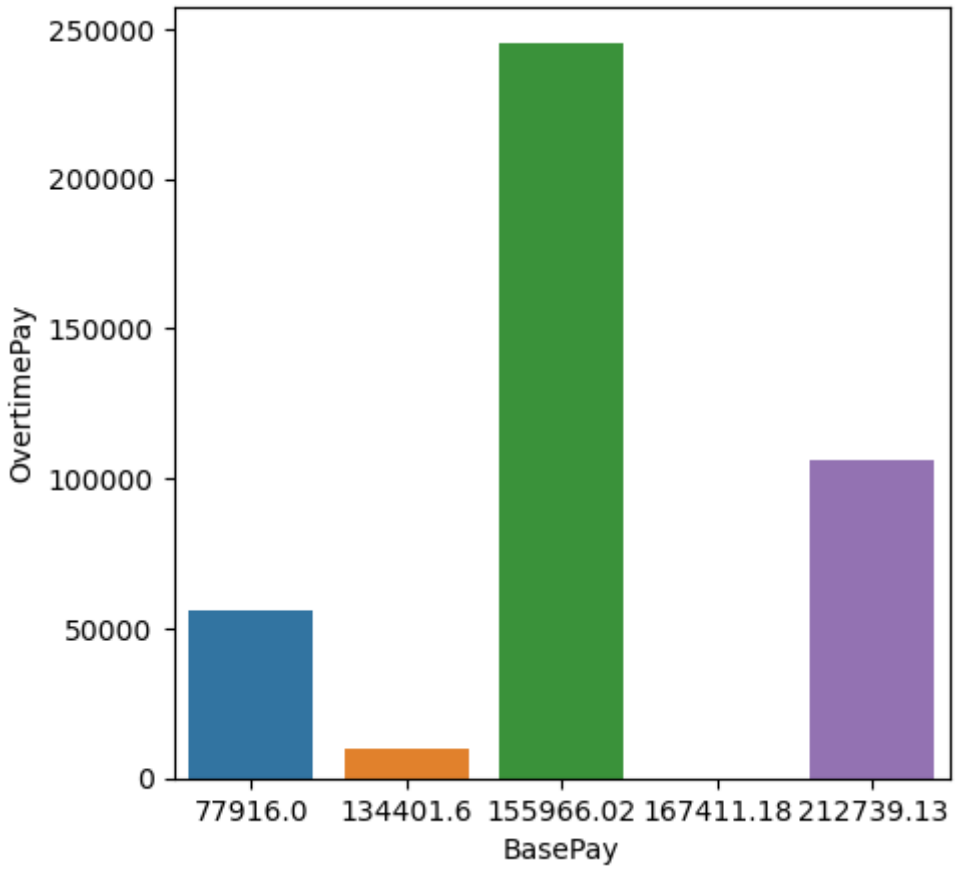
```
plt.ylabel('Total pay',fontsize=14)
plt.show()
```



In [9]:
```
name=df['EmployeeName'].head(5)
pay=df['TotalPayBenefits'].head(5)
plt.figure(figsize=(10,8))
plt.barh(name,pay,height=0.2)
plt.xlabel('Employee name',fontsize=15)
plt.ylabel('Total pay',fontsize=14)
plt.show()
```



In [17]:
```
plt.figure(figsize=(5,5))
#sns.barplot(data = df)
sns.barplot(x='BasePay',y='OvertimePay',data = df.head(5))
plt.show()
```

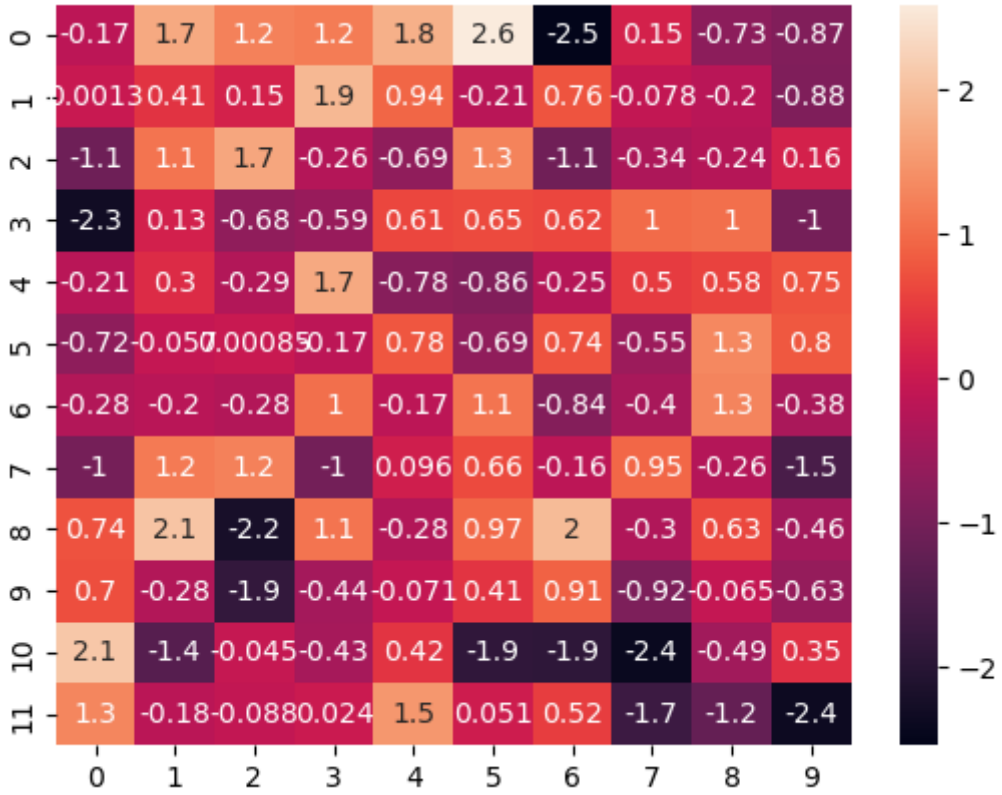


In [ ]:

In [ ]:
```
#x=n.random.randint(12,10)
#x
```

In [25]:
```
x=np.random.randn(12,10)
x.shape
sns.heatmap(x,annot=True)
```

Out[25]:
```
<Axes: >
```



In [ ]:
```
```

In [12]:
```
hoteldf = pd.read_csv('Downloads/Hotel Reservations.csv')
```

In [13]:
```
hoteldf
```

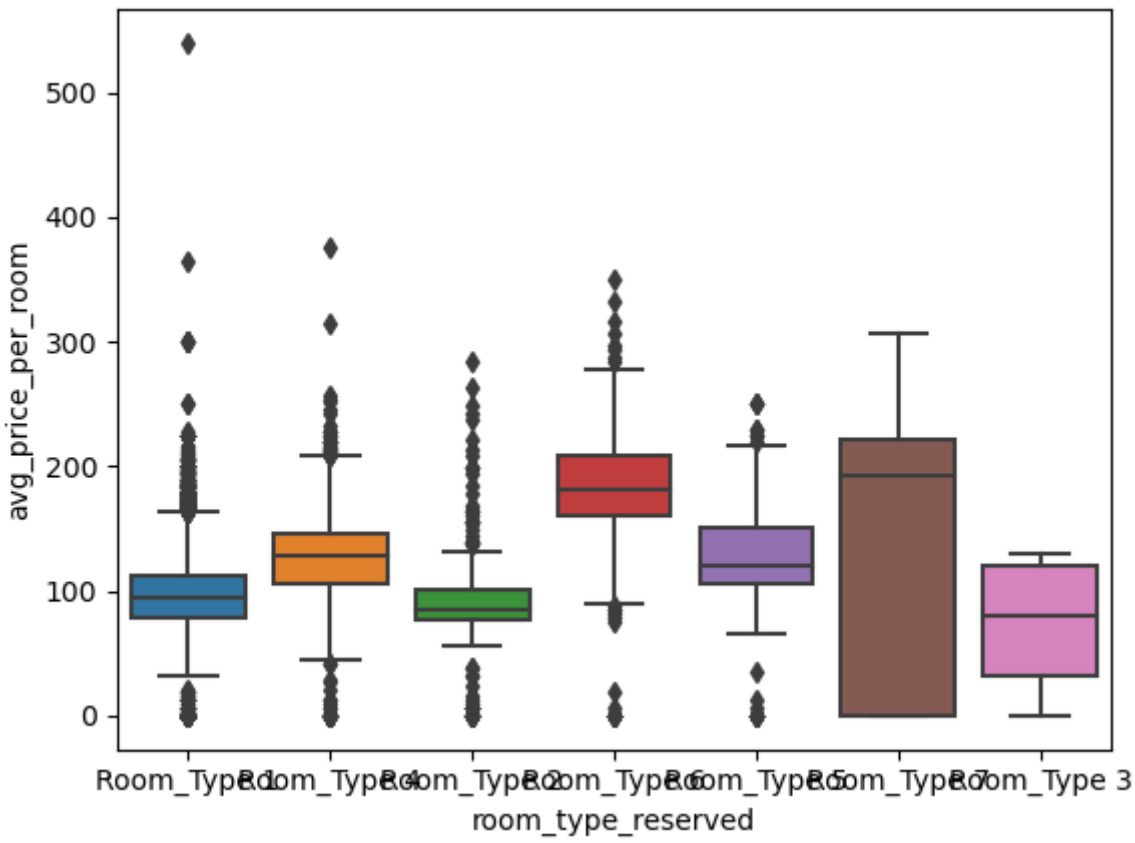
Out[13]:

	Booking_ID	no_of_adults	no_of_children	no_of_weekend_nights	no_of_week_nights	type_
0	INN00001	2	0	1	2	
1	INN00002	2	0	2	3	
2	INN00003	1	0	2	1	
3	INN00004	2	0	0	2	
4	INN00005	2	0	1	1	
...	...	...	...	...	...	
36270	INN36271	3	0	2	6	
36271	INN36272	2	0	1	3	
36272	INN36273	2	0	2	6	
36273	INN36274	2	0	0	3	
36274	INN36275	2	0	1	2	

36275 rows × 19 columns

In [14]:
```
sns.boxplot(x='room_type_reserved',y='avg_price_per_room',data=hoteldf)
```

Out[14]:
```
<Axes: xlabel='room_type_reserved', ylabel='avg_price_per_room'>
```



In [21]:
```
hoteldf.query("room_type_reserved == 'Room_Type 1'")
```

Out[21]:

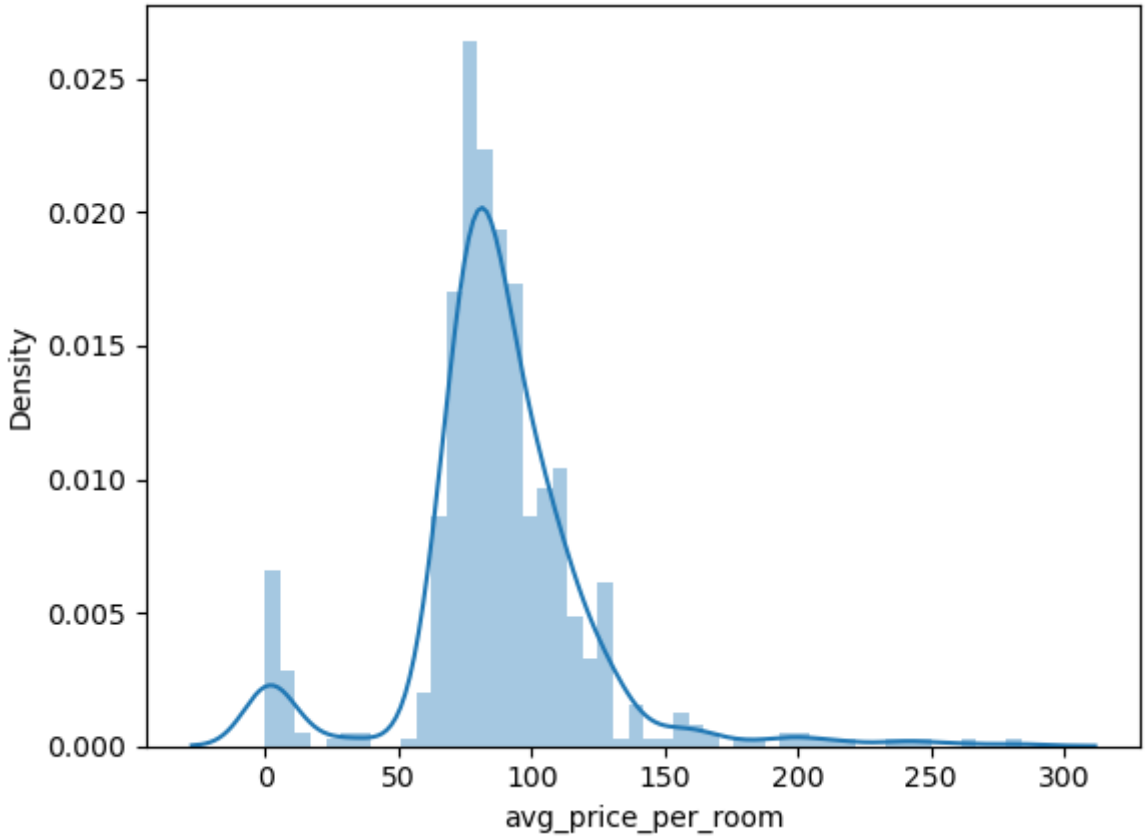
	Booking_ID	no_of_adults	no_of_children	no_of_weekend_nights	no_of_week_nights	type_
0	INN00001	2	0	1	2	
1	INN00002	2	0	2	3	
2	INN00003	1	0	2	1	
3	INN00004	2	0	0	2	
4	INN00005	2	0	1	1	
...	...	...	...	...	...	...
36268	INN36269	1	0	0	3	
36271	INN36272	2	0	1	3	
36272	INN36273	2	0	2	6	
36273	INN36274	2	0	0	3	
36274	INN36275	2	0	1	2	

28130 rows × 19 columns

In [27]:
```
room1=hoteldf.query("room_type_reserved=='Room_Type 2'")
sns.distplot(room1['avg_price_per_room'],kde=True)
```

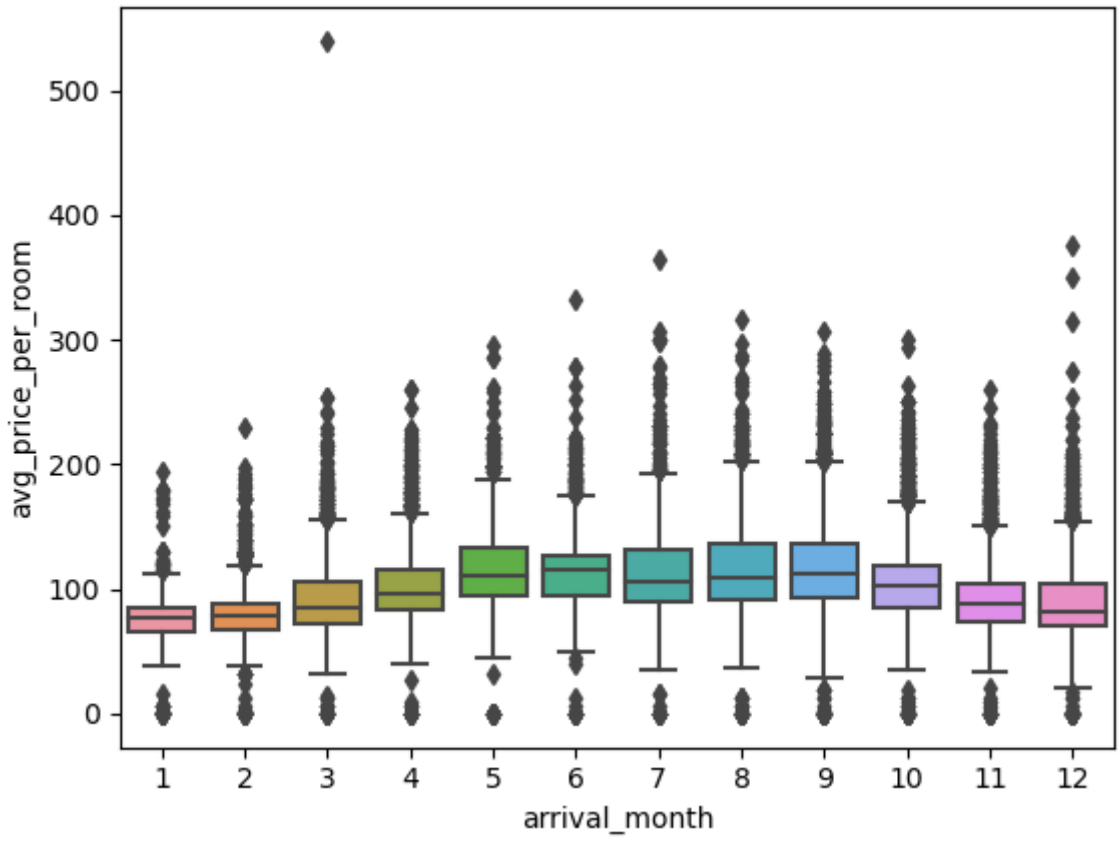
```
C:\Users\Sony Vaio\AppData\Local\Temp\ipykernel_10036\1346883958.py:2: UserWarnin
g:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(room1['avg_price_per_room'],kde=True)
```

Out[27]:
```
<Axes: xlabel='avg_price_per_room', ylabel='Density'>
```



In [28]:
```
sns.boxplot(x='arrival_month',y='avg_price_per_room',data=hoteldf)
```

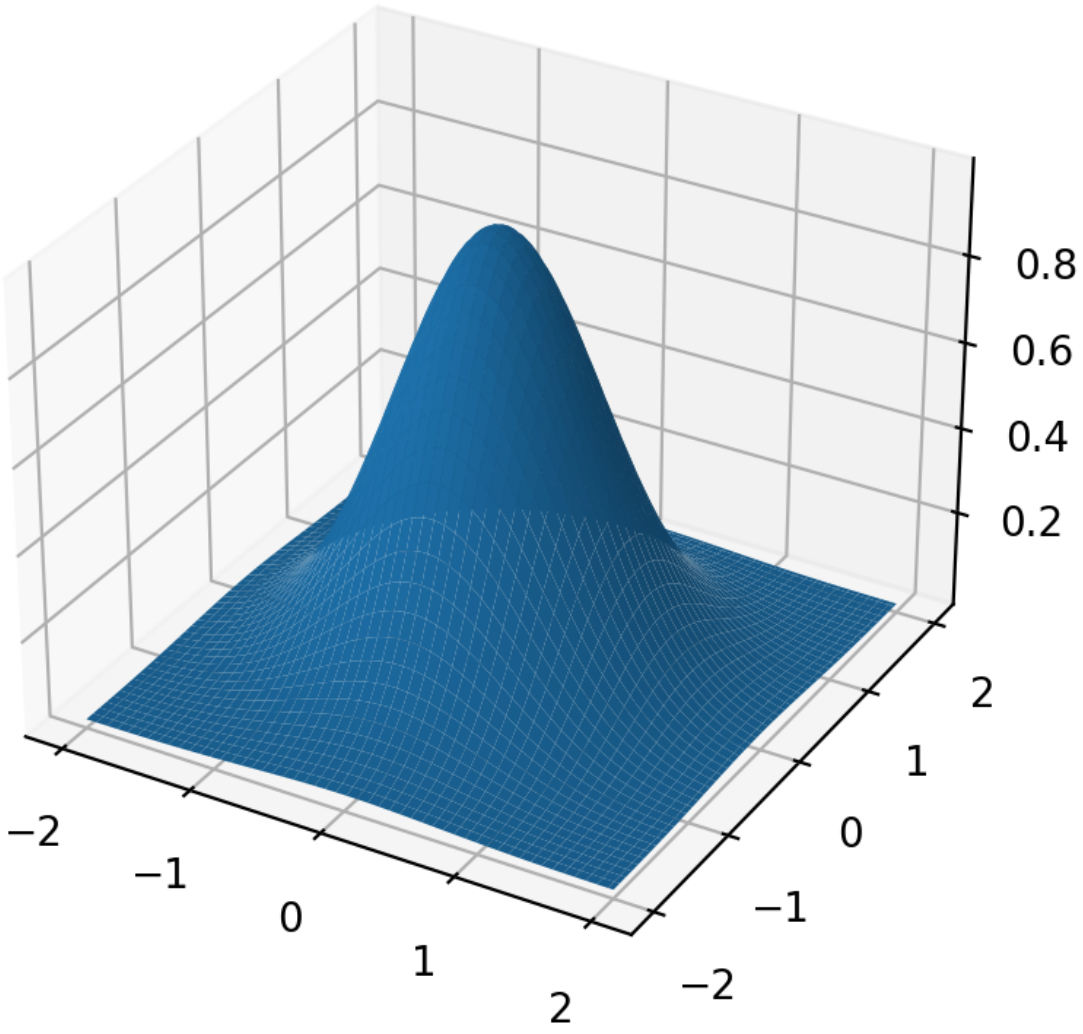
Out[28]:
```
<Axes: xlabel='arrival_month', ylabel='avg_price_per_room'>
```



In [45]:
```
def fun(x,y):
    #return x**2+y**2
    #return np.exp(x**2)*np.sin(y**2)
    return np.exp(-x**2-y**2)
```

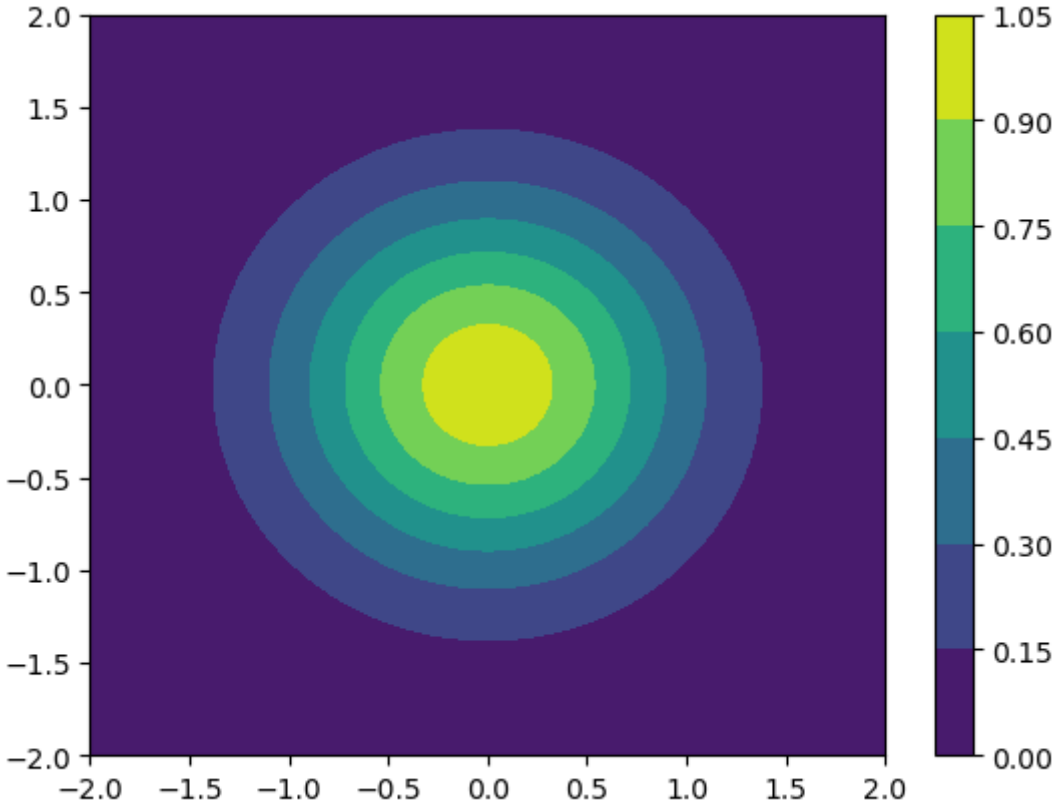
```
y=np.linspace(-2,2,100)
x=np.linspace(-2,2,100)
x,y=np.meshgrid(x,y)
z=fun(x,y)
print(z)
fig=plt.figure(dpi=200)
ax=plt.axes(projection='3d')
#plt.plot(x,y,z)
ax.plot_surface(x,y,z)
plt.show()
```

```
[[0.00033546 0.00039366 0.00046045 ... 0.00046045 0.00039366 0.00033546]
 [0.00039366 0.00046196 0.00054034 ... 0.00054034 0.00046196 0.00039366]
 [0.00046045 0.00054034 0.00063202 ... 0.00063202 0.00054034 0.00046045]
 ...
 [0.00046045 0.00054034 0.00063202 ... 0.00063202 0.00054034 0.00046045]
 [0.00039366 0.00046196 0.00054034 ... 0.00054034 0.00046196 0.00039366]
 [0.00033546 0.00039366 0.00046045 ... 0.00046045 0.00039366 0.00033546]]
```



In [49]:

```
#plt.contour(x,y,z)
plt.contourf(x,y,z)
plt.colorbar()
plt.savefig('contour.png',dpi=200)
plt.show()
```


2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
−2.0
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
2.0
1.05
0.90
0.75
0.60
0.45
0.30
0.15
0.00

In [ ]: