DS_SKILLING-4

ID:190030900
NAME:K BABY SAHITHI

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

```
import warnings
warnings.filterwarnings("ignore")
```

```
pd.options.display.float_format = '{:.5f}'.format
pd.options.display.max_columns=None
pd.options.display.max_rows=None
np.random.seed(100)
```

```
day=pd.read_csv("day.csv")
day.head()
```

	instant	dteday	season	yr	mnth	holiday	weekday	workingday	weathersit	temp
0	1	1/1/2011	1	0	1	0	6	0	2	0.34417
1	2	1/2/2011	1	0	1	0	0	0	2	0.36348
2	3	1/3/2011	1	0	1	0	1	1	1	0.19636
3	4	1/4/2011	1	0	1	0	2	1	1	0.20000
4	5	1/5/2011	1	0	1	0	3	1	1	0.22696

```
hour=pd.read_csv("hour.csv")
hour.head()
```

	instant	dteday	season	yr	mnth	hr	holiday	weekday	workingday	weathersit	
0	1	1/1/2011	1	0	1	0	0	6	0	1	0.2
1	2	1/1/2011	1	0	1	1	0	6	0	1	0.2
2	3	1/1/2011	1	0	1	2	0	6	0	1	0.2
3	4	1/1/2011	1	0	1	3	0	6	0	1	0.2
4	5	1/1/2011	1	0	1	4	0	6	0	1	0.2

```
day.isna().sum
```

674	False	False	False	False	False	False	False	False
675	False	False	False	False	False	False	False	False
676	False	False	False	False	False	False	False	False
677	False	False	False	False	False	False	False	False
678	False	False	False	False	False	False	False	False
679	False	False	False	False	False	False	False	False
680	False	False	False	False	False	False	False	False
681	False	False	False	False	False	False	False	False
682	False	False	False	False	False	False	False	False
683	False	False	False	False	False	False	False	False
684	False	False	False	False	False	False	False	False
685	False	False	False	False	False	False	False	False
686	False	False	False	False	False	False	False	False
687	False	False	False	False	False	False	False	False
688	False	False	False	False	False	False	False	False
689	False	False	False	False	False	False	False	False
690	False	False	False	False	False	False	False	False
691	False	False	False	False	False	False	False	False
692	False	False	False	False	False	False	False	False
693	False	False	False	False	False	False	False	False
694	False	False	False	False	False	False	False	False
695	False	False	False	False	False	False	False	False
696	False	False	False	False	False	False	False	False
697	False	False	False	False	False	False	False	False
698	False	False	False	False	False	False	False	False
699	False	False	False	False	False	False	False	False
700	False	False	False	False	False	False	False	False
701	False	False	False	False	False	False	False	False
702	False	False	False	False	False	False	False	False
703	False	False	False	False	False	False	False	False
704	False	False	False	False	False	False	False	False
705	False	False	False	False	False	False	False	False
706	False	False	False	False	False	False	False	False
707	False	False	False	False	False	False	False	False
708	False	False	False	False	False	False	False	False
709	False	False	False	False	False	False	False	False
710	False	False	False	False	False	False	False	False
711	False	False	False	False	False	False	False	False
712	False	False	False	False	False	False	False	False
713	False	False	False	False	False	False	False	False
714	False	False	False	False	False	False	False	False
715	False	False	False	False	False	False	False	False
716	False	False	False	False	False	False	False	False
717	False	False	False	False	False	False	False	False
718	False	False	False	False	False	False	False	False
719	False	False	False	False	False	False	False	False
720	False	False	False	False	False	False	False	False
721	False	False	False	False	False	False	False	False
722	False	False	False	False	False	False	False	False
723	False	False	False	False	False	False	False	False
724	False	False	False	False	False	False	False	False
725	False	False	False	False	False	False	False	False
726	False	False	False	False	False	False	False	False
727	False	False	False	False	False	False	False	False

```
728       False  False  False  False      False   False       False  False
729       False  False  False  False      False   False       False  False
730       False  False  False  False      False   False       False  False  >
```

```
hour.isna().sum()
```

```
instant       0
dteday        0
season        0
yr            0
mnth          0
hr            0
holiday       0
weekday       0
workingday    0
weathersit    0
temp          0
atemp         0
hum           0
windspeed     0
casual        0
registered    0
cnt           0
dtype: int64
```

```
hour.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 17379 entries, 0 to 17378
Data columns (total 17 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   instant     17379 non-null  int64  
 1   dteday      17379 non-null  object 
 2   season      17379 non-null  int64  
 3   yr          17379 non-null  int64  
 4   mnth        17379 non-null  int64  
 5   hr          17379 non-null  int64  
 6   holiday     17379 non-null  int64  
 7   weekday     17379 non-null  int64  
 8   workingday  17379 non-null  int64  
 9   weathersit  17379 non-null  int64  
 10  temp        17379 non-null  float64
 11  atemp       17379 non-null  float64
 12  hum         17379 non-null  float64
 13  windspeed   17379 non-null  float64
 14  casual      17379 non-null  int64  
 15  registered  17379 non-null  int64  
 16  cnt         17379 non-null  int64  
dtypes: float64(4), int64(12), object(1)
memory usage: 2.3+ MB
```

```
day.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 731 entries, 0 to 730
Data columns (total 16 columns):
 #   Column      Non-Null Count  Dtype  
---  ------      --------------  -----  
 0   instant     731 non-null    int64  
 1   dteday      731 non-null    object 
 2   season      731 non-null    int64  
 3   yr          731 non-null    int64  
 4   mnth        731 non-null    int64  
 5   holiday     731 non-null    int64  
 6   weekday     731 non-null    int64  
 7   workingday  731 non-null    int64  
 8   weathersit  731 non-null    int64  
 9   temp        731 non-null    float64
 10  atemp       731 non-null    float64
 11  hum         731 non-null    float64
 12  windspeed   731 non-null    float64
 13  casual      731 non-null    int64  
 14  registered  731 non-null    int64  
 15  cnt         731 non-null    int64  
dtypes: float64(4), int64(11), object(1)
memory usage: 91.5+ KB
```

```
day['temp'] = day['temp']*41
hour['temp']= hour['temp']*41

day['atemp'] = day['atemp']*50
hour['atemp']= hour['atemp']*50

day['hum'] = day['hum']*100
hour['hum']= hour['hum']*100

day['windspeed'] = day['windspeed']*67
hour['windspeed']= hour['windspeed']*67
```

```
day.head()
```

	instant	dteday	season	yr	mnth	holiday	weekday	workingday	weathersit	tem
0	1	1/1/2011	1	0	1	0	6	0	2	14.1108
1	2	1/2/2011	1	0	1	0	0	0	2	14.9026
2	3	1/3/2011	1	0	1	0	1	1	1	8.0509
3	4	1/4/2011	1	0	1	0	2	1	1	8.2000
4	5	1/5/2011	1	0	1	0	3	1	1	9.3052

```
day.describe().T
```

	count	mean	std	min	25%	50%	75
instant	731.00000	366.00000	211.16581	1.00000	183.50000	366.00000	548.5000
season	731.00000	2.49658	1.11081	1.00000	2.00000	3.00000	3.0000
yr	731.00000	0.50068	0.50034	0.00000	0.00000	1.00000	1.0000
mnth	731.00000	6.51984	3.45191	1.00000	4.00000	7.00000	10.0000
holiday	731.00000	0.02873	0.16715	0.00000	0.00000	0.00000	0.0000
weekday	731.00000	2.99726	2.00479	0.00000	1.00000	3.00000	5.0000
workingday	731.00000	0.68399	0.46523	0.00000	0.00000	1.00000	1.0000
weathersit	731.00000	1.39535	0.54489	1.00000	1.00000	1.00000	2.0000
temp	731.00000	20.31078	7.50509	2.42435	13.82042	20.43165	26.8720
atemp	731.00000	23.71770	8.14806	3.95348	16.89213	24.33665	30.4301
hum	731.00000	62.78941	14.24291	0.00000	52.00000	62.66670	73.0208
windspeed	731.00000	12.76258	5.19236	1.50024	9.04165	12.12533	15.6253
casual	731.00000	848.17647	686.62249	2.00000	315.50000	713.00000	1096.0000
registered	731.00000	3656.17237	1560.25638	20.00000	2497.00000	3662.00000	4776.5000
cnt	731.00000	4504.34884	1937.21145	22.00000	3152.00000	4548.00000	5956.0000

```
col = ['season','yr','mnth','yr','holiday','weekday', 'workingday','weathersit']


def change_dtype(data,col):
    for i in col:
        if i in data.columns.to_list():
            data[i] = data[i].astype('category')



for i in col:
    print("Name of {} col".format(i))
    print("No. of NUnique",hour[i].nunique())
    print("Unique Values",hour[i].unique())
    print('*'*30)
    print("")
    print("")
```

```
Name of season col
No. of NUnique 4
Unique Values [1 2 3 4]
******************************


Name of yr col
```

```
No. of NUnique 2
Unique Values [0 1]
******************************


Name of mnth col
No. of NUnique 12
Unique Values [ 1  2  3  4  5  6  7  8  9 10 11 12]
******************************


Name of yr col
No. of NUnique 2
Unique Values [0 1]
******************************


Name of holiday col
No. of NUnique 2
Unique Values [0 1]
******************************


Name of weekday col
No. of NUnique 7
Unique Values [6 0 1 2 3 4 5]
******************************


Name of workingday col
No. of NUnique 2
Unique Values [0 1]
******************************


Name of weathersit col
No. of NUnique 4
Unique Values [1 2 3 4]
******************************
```

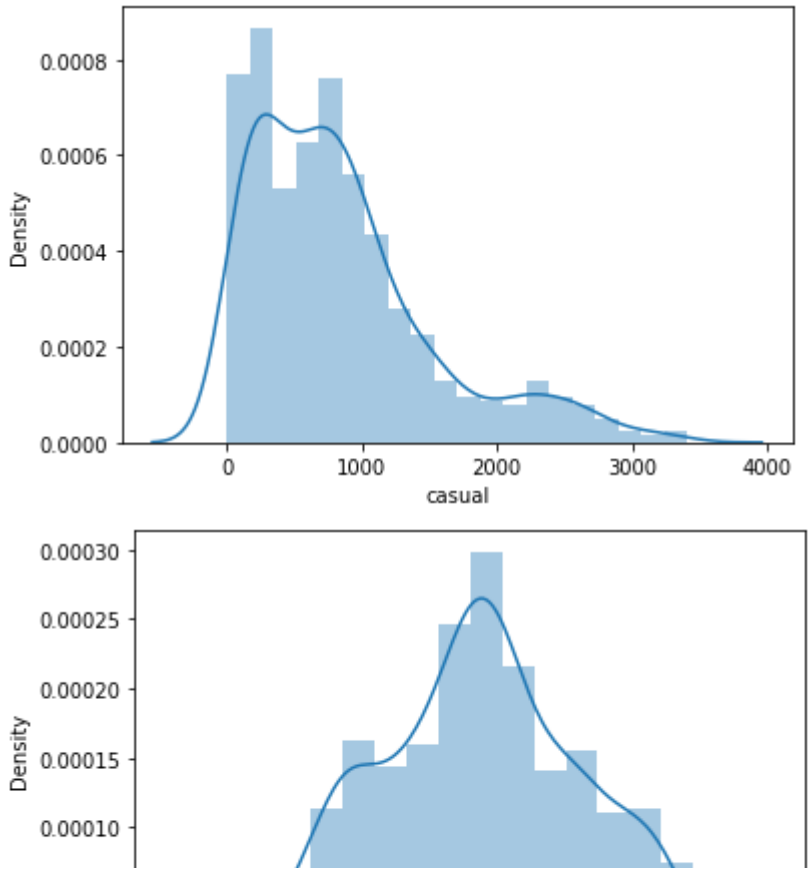
```
change_dtype(day, col)
change_dtype(hour, col)


for i in col:
    print("Name of {} col".format(i))
    print("No. of NUnique",hour[i].nunique())
    print("Unique Values",hour[i].unique())
    print('*'*30)
    print("")
    print("")
```

```
Name of season col
No. of NUnique 4
Unique Values [1, 2, 3, 4]
Categories (4, int64): [1, 2, 3, 4]
******************************


Name of yr col
No. of NUnique 2
Unique Values [0, 1]
Categories (2, int64): [0, 1]
******************************


Name of mnth col
No. of NUnique 12
Unique Values [1, 2, 3, 4, 5, ..., 8, 9, 10, 11, 12]
Length: 12
Categories (12, int64): [1, 2, 3, 4, ..., 9, 10, 11, 12]
******************************


Name of yr col
No. of NUnique 2
Unique Values [0, 1]
Categories (2, int64): [0, 1]
******************************


Name of holiday col
No. of NUnique 2
Unique Values [0, 1]
Categories (2, int64): [0, 1]
******************************


Name of weekday col
No. of NUnique 7
Unique Values [6, 0, 1, 2, 3, 4, 5]
Categories (7, int64): [6, 0, 1, 2, 3, 4, 5]
******************************


Name of workingday col
No. of NUnique 2
Unique Values [0, 1]
Categories (2, int64): [0, 1]
******************************


Name of weathersit col
No. of NUnique 4
Unique Values [1, 2, 3, 4]
Categories (4, int64): [1, 2, 3, 4]
******************************
```

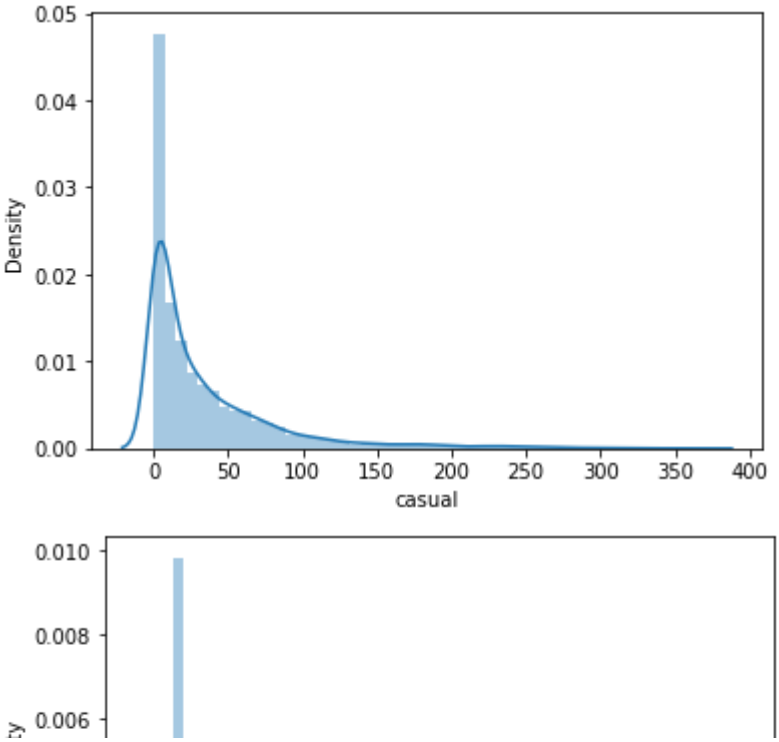
```
def drop_instant(data):
    data.drop(['instant'], axis=1, inplace=True)


drop_instant(day)
drop_instant(hour)



hour.describe().T
```

	count	mean	std	min	25%	50%	75%	
hr	17379.00000	11.54675	6.91441	0.00000	6.00000	12.00000	18.00000	2
temp	17379.00000	20.37647	7.89480	0.82000	13.94000	20.50000	27.06000	4
atemp	17379.00000	23.78876	8.59251	0.00000	16.66500	24.24000	31.06000	5
hum	17379.00000	62.72288	19.29298	0.00000	48.00000	63.00000	78.00000	10
windspeed	17379.00000	12.73654	8.19680	0.00000	7.00150	12.99800	16.99790	5
casual	17379.00000	35.67622	49.30503	0.00000	4.00000	17.00000	48.00000	36
registered	17379.00000	153.78687	151.35729	0.00000	34.00000	115.00000	220.00000	88
cnt	17379.00000	189.46309	181.38760	1.00000	40.00000	142.00000	281.00000	97

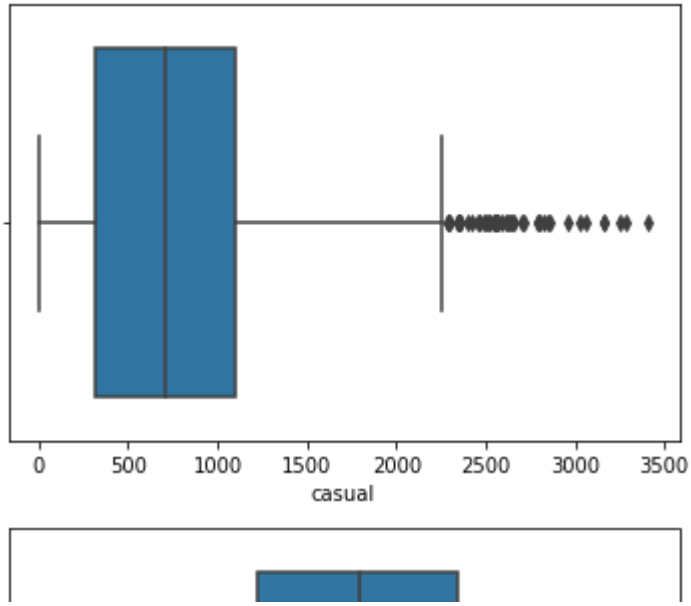
```
for i in day.select_dtypes(include='int'):
  sns.distplot(day[i])
  plt.show()
```



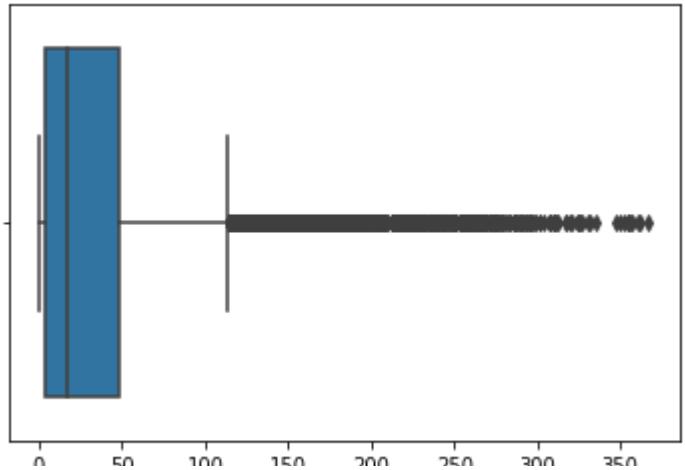

```
for i in day.select_dtypes(include='int'):
  sns.distplot(hour[i])
  plt.show()
```



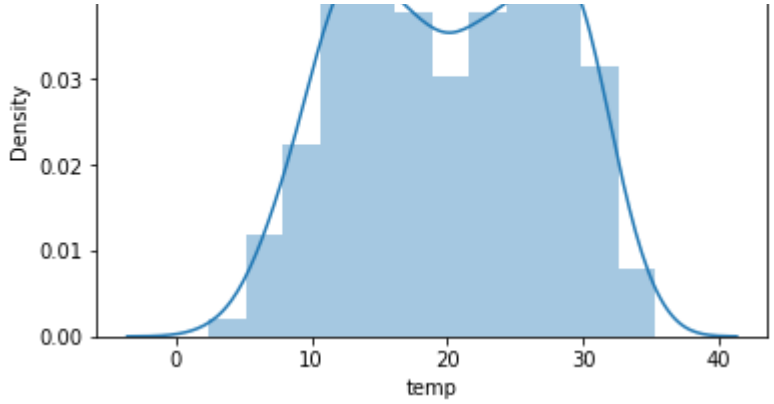

```
for i in day.select_dtypes(include='int'):
  sns.boxplot(day[i])
  plt.show()
```

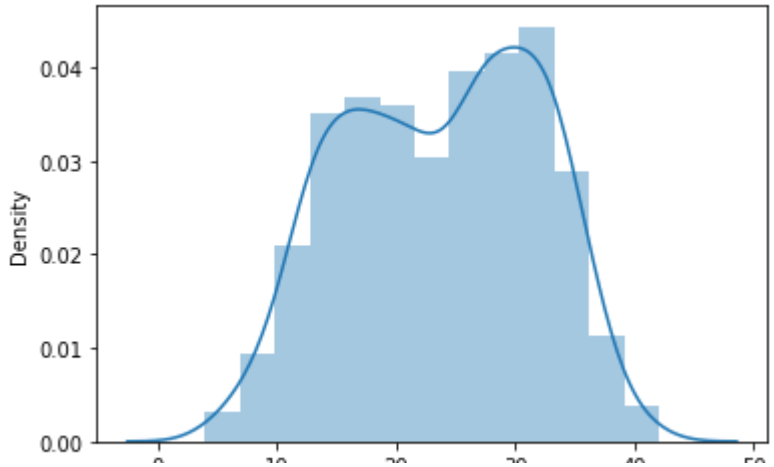



```
for i in day.select_dtypes(include='int'):
  sns.boxplot(hour[i])
  plt.show()
```



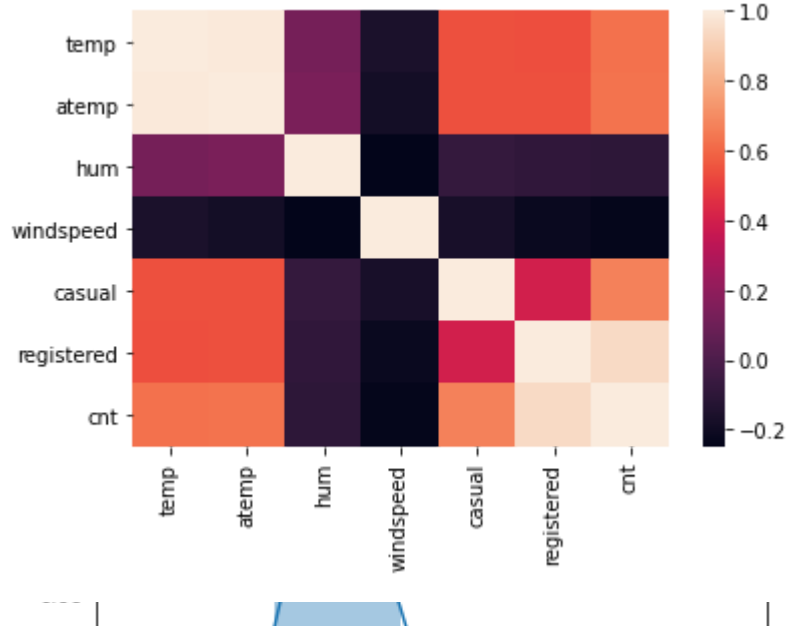

```
for i in day.select_dtypes(include='float'):
  sns.distplot(day[i])
  plt.show()
```






```
sns.heatmap(day.corr())
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f5b358c1240>
```




```
day.corr()['cnt']
```

```
temp          0.62749
atemp         0.63107
hum          -0.10066
windspeed    -0.23454
casual        0.67280
registered    0.94552
```

```
cnt           1.00000
Name: cnt, dtype: float64
```

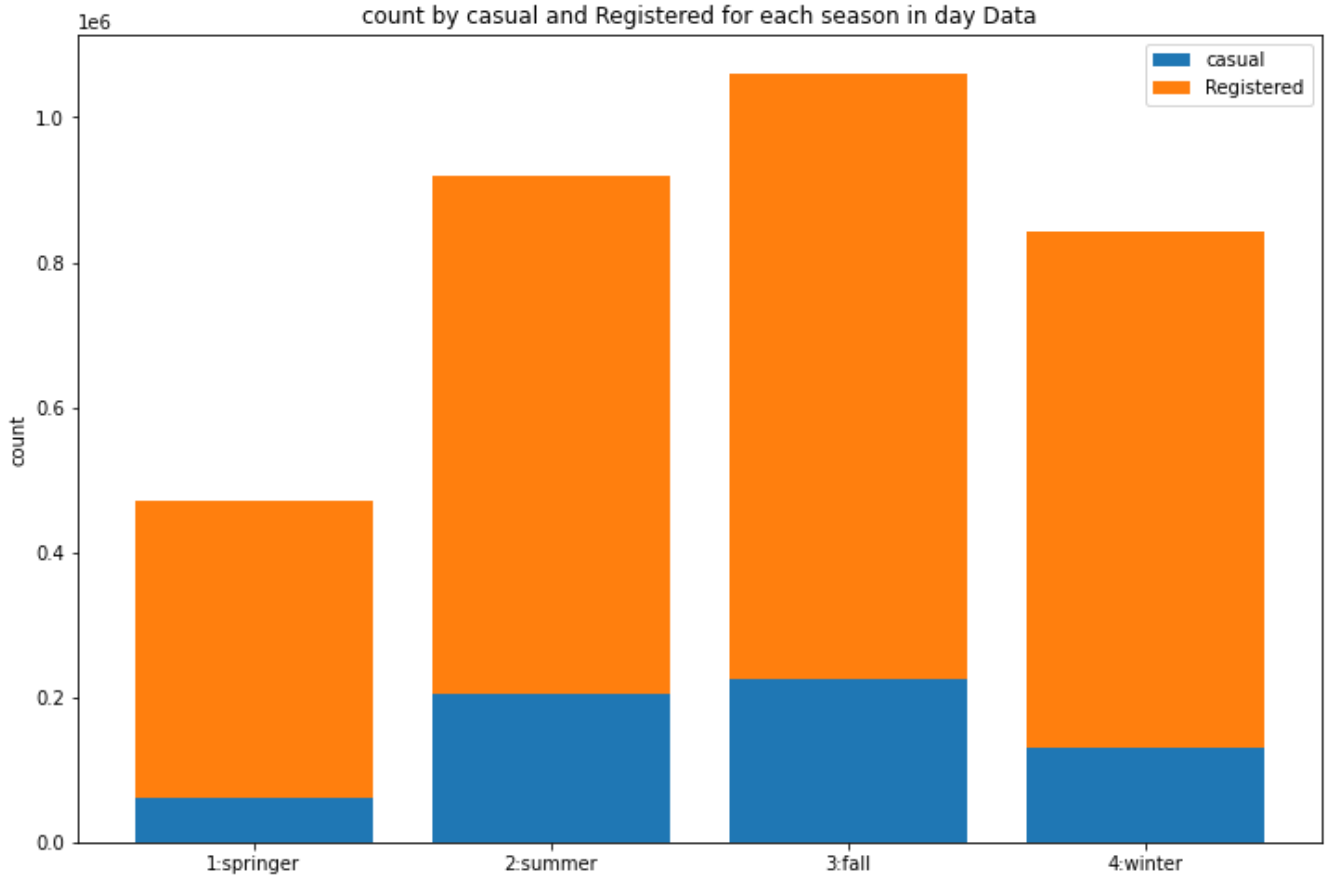
```
hour.corr()['cnt']
```

```
hr            0.39407
temp          0.40477
atemp         0.40093
hum          -0.32291
windspeed     0.09323
casual        0.69456
registered    0.97215
cnt           1.00000
Name: cnt, dtype: float64
```

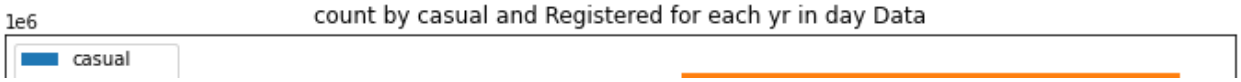
```
day.head()
```

	dteday	season	yr	mnth	holiday	weekday	workingday	weathersit	temp	ate
0	1/1/2011	1	0	1	0	6	0	2	14.11085	18.181
1	1/2/2011	1	0	1	0	0	0	2	14.90260	17.686
2	1/3/2011	1	0	1	0	1	1	1	8.05092	9.470
3	1/4/2011	1	0	1	0	2	1	1	8.20000	10.606
4	1/5/2011	1	0	1	0	3	1	1	9.30524	11.463

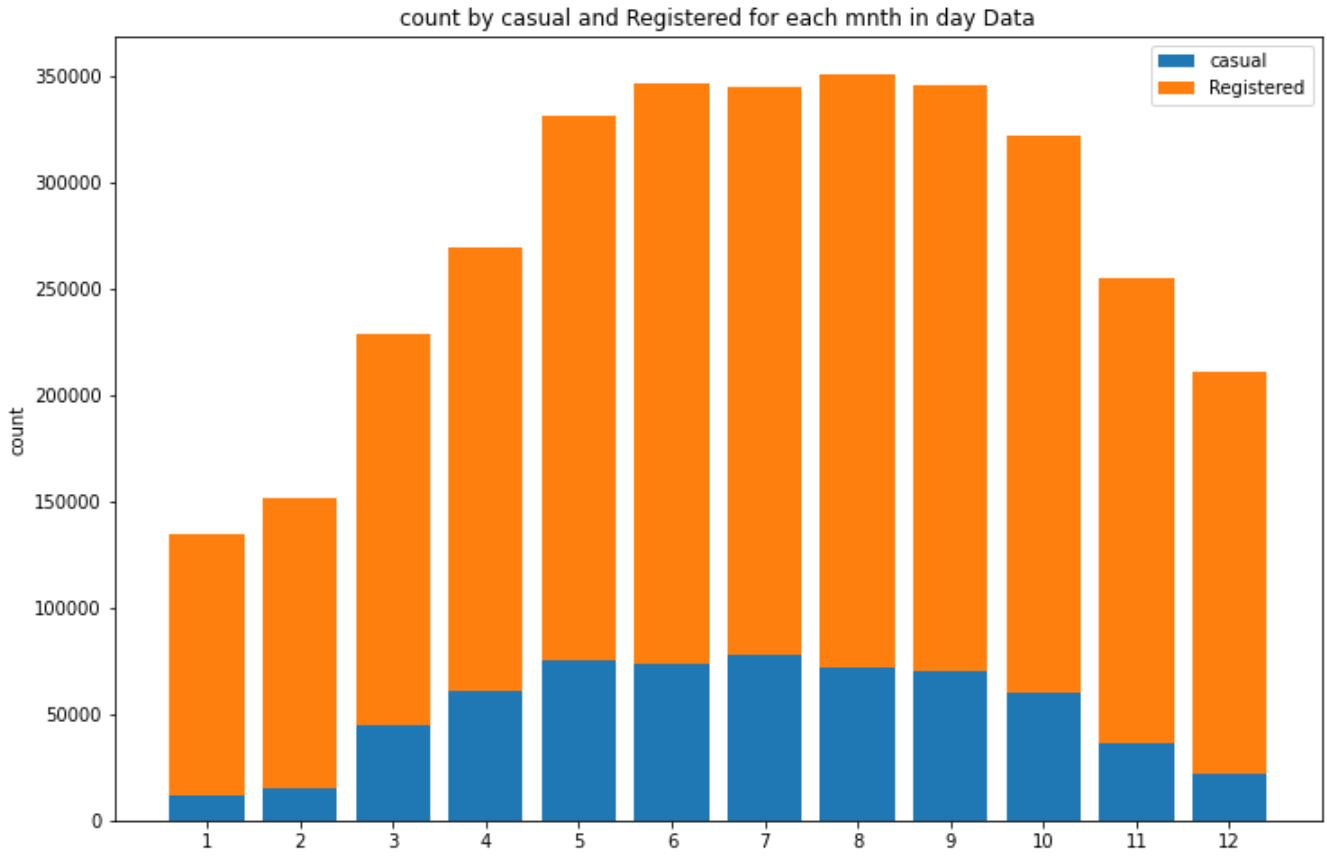
```
def get_df_name(df):
    name=[x for x in globals() if globals()[x] is df][0]
    return name


def plot_stack_bar_chart(data,col,name):
    plt.figure(figsize=(12,8))
    p1 =plt.bar(data[col].unique(),
                   data.groupby([col])['casual'].sum())

    p2 =plt.bar(data[col].unique(),
                data.groupby([col])['registered'].sum(),
                 bottom =data.groupby([col])['casual'].sum())

    plt.ylabel('count')
    plt.title("count by casual and Registered for each {} in {} Data".format(col, get_df_name
    plt.xticks(data[col].unique(),name)
    plt.legend((p1[0],p2[0]), ('casual','Registered'))



plot_stack_bar_chart(day, 'season', ('1:springer','2:summer','3:fall','4:winter'))
```




```
plot_stack_bar_chart(day, 'yr',('2011','2012'))
```

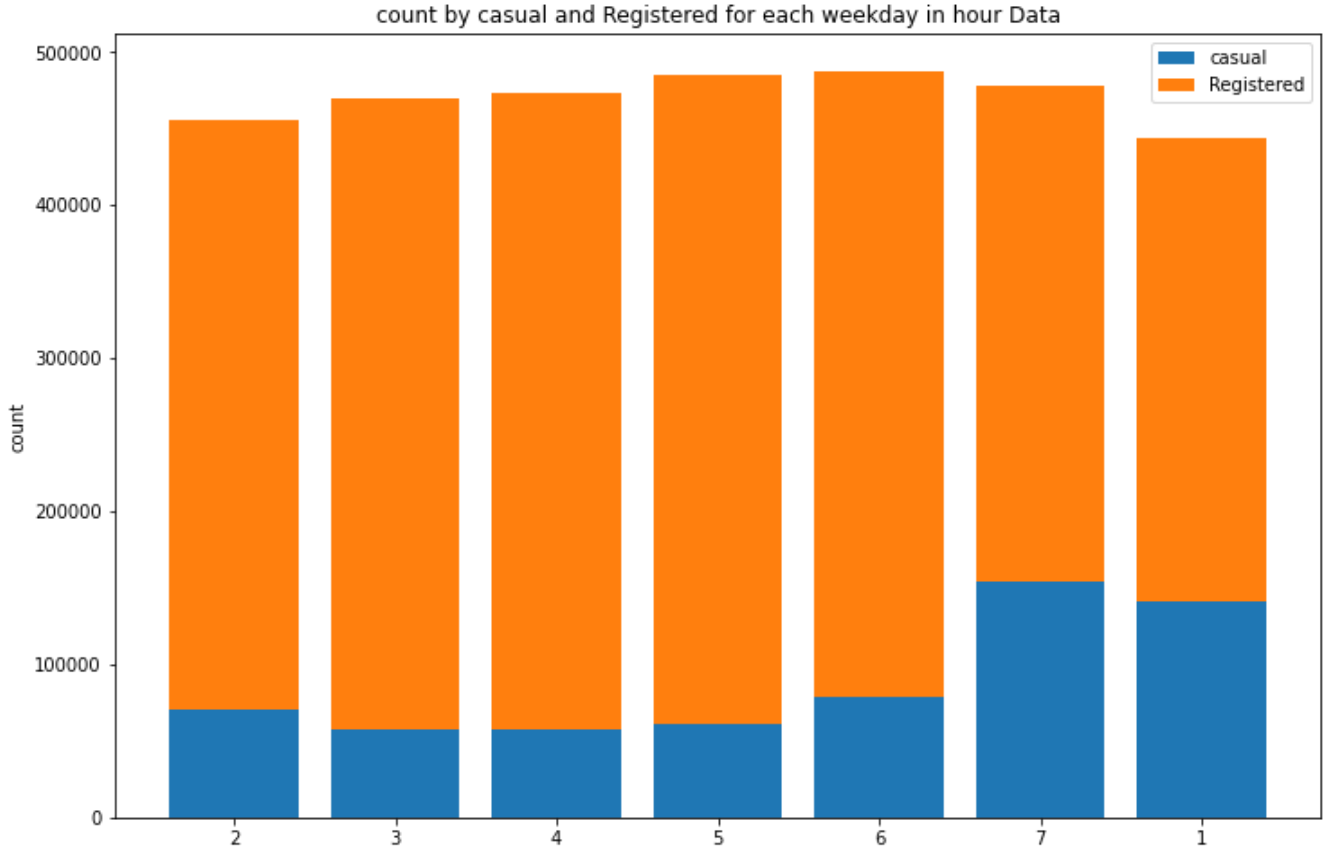



```
plot_stack_bar_chart(day, 'mnth', [str(i) for i in day['mnth'].unique()])
```

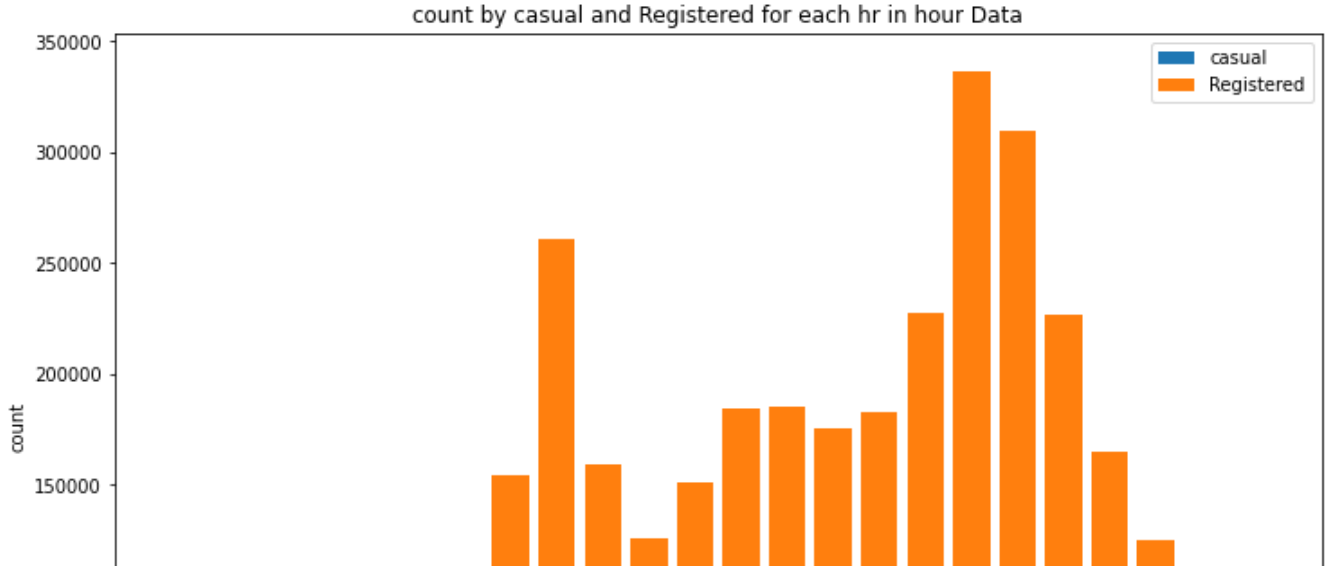



```
plot_stack_bar_chart(day, 'holiday',('Yes','No'))
```

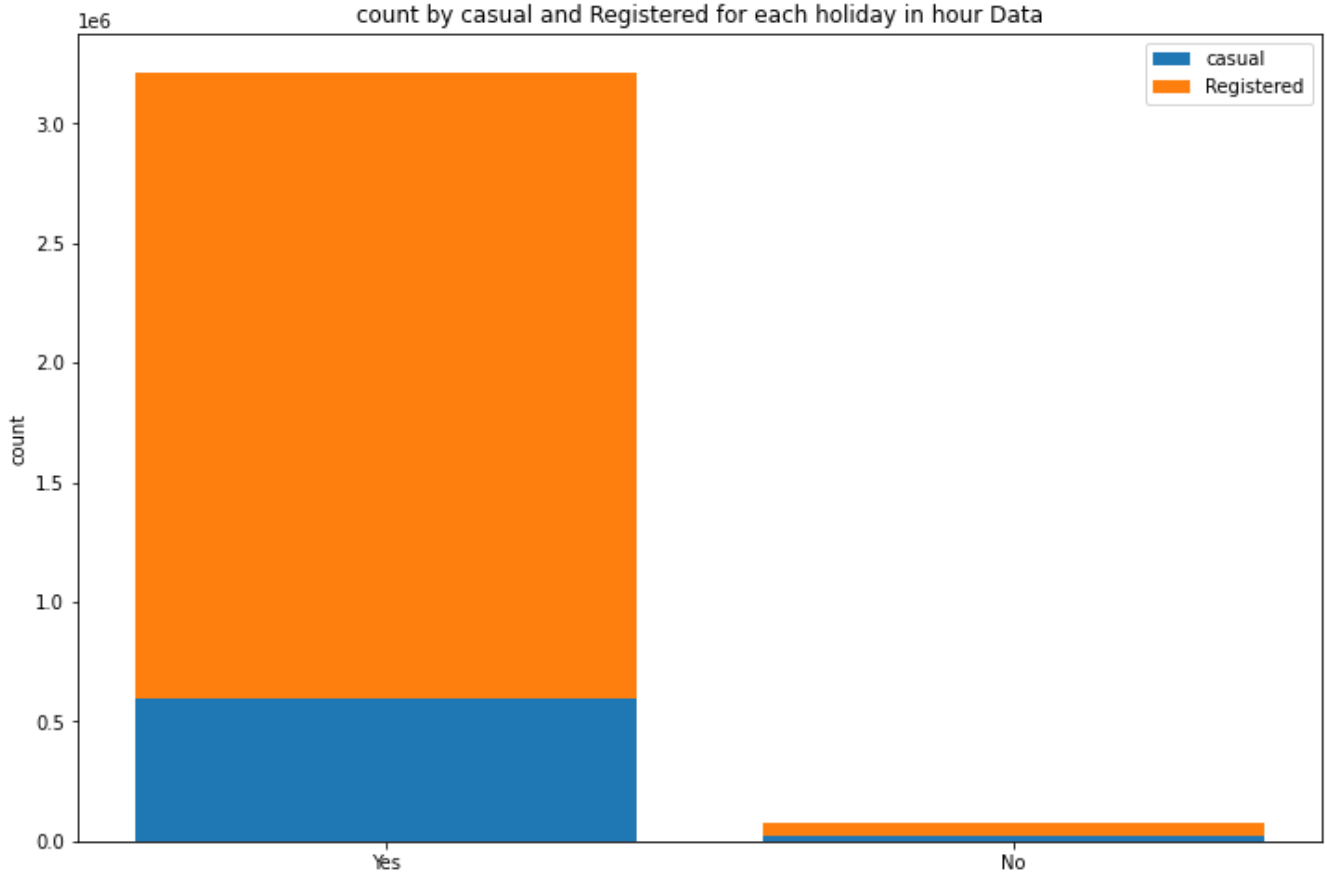

```
plot_stack_bar_chart(hour, 'weekday', [str(i) for i in day['mnth'].unique()])
```



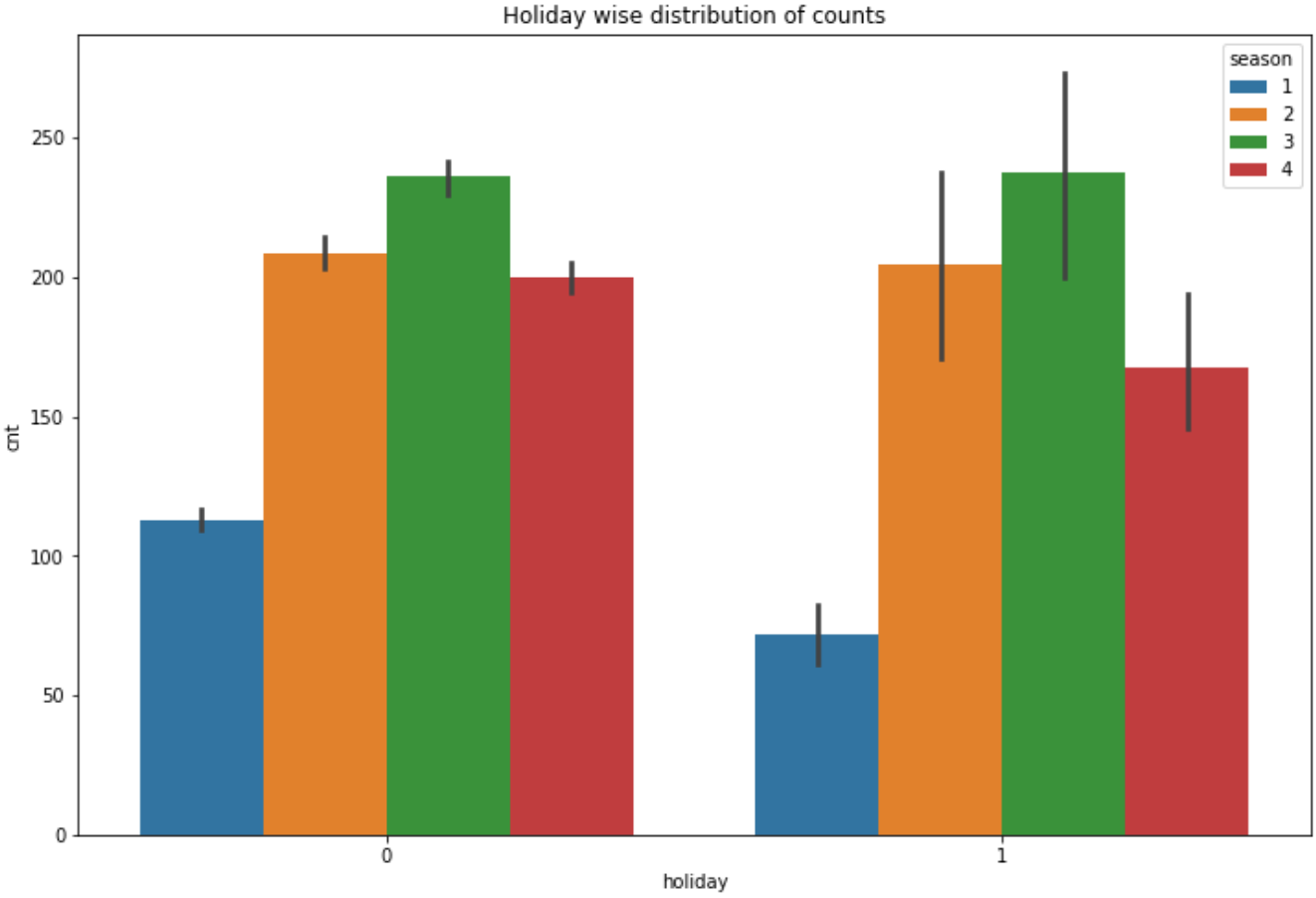

```
plot_stack_bar_chart(hour, 'hr', [str(i) for i in day['mnth'].unique()])
```



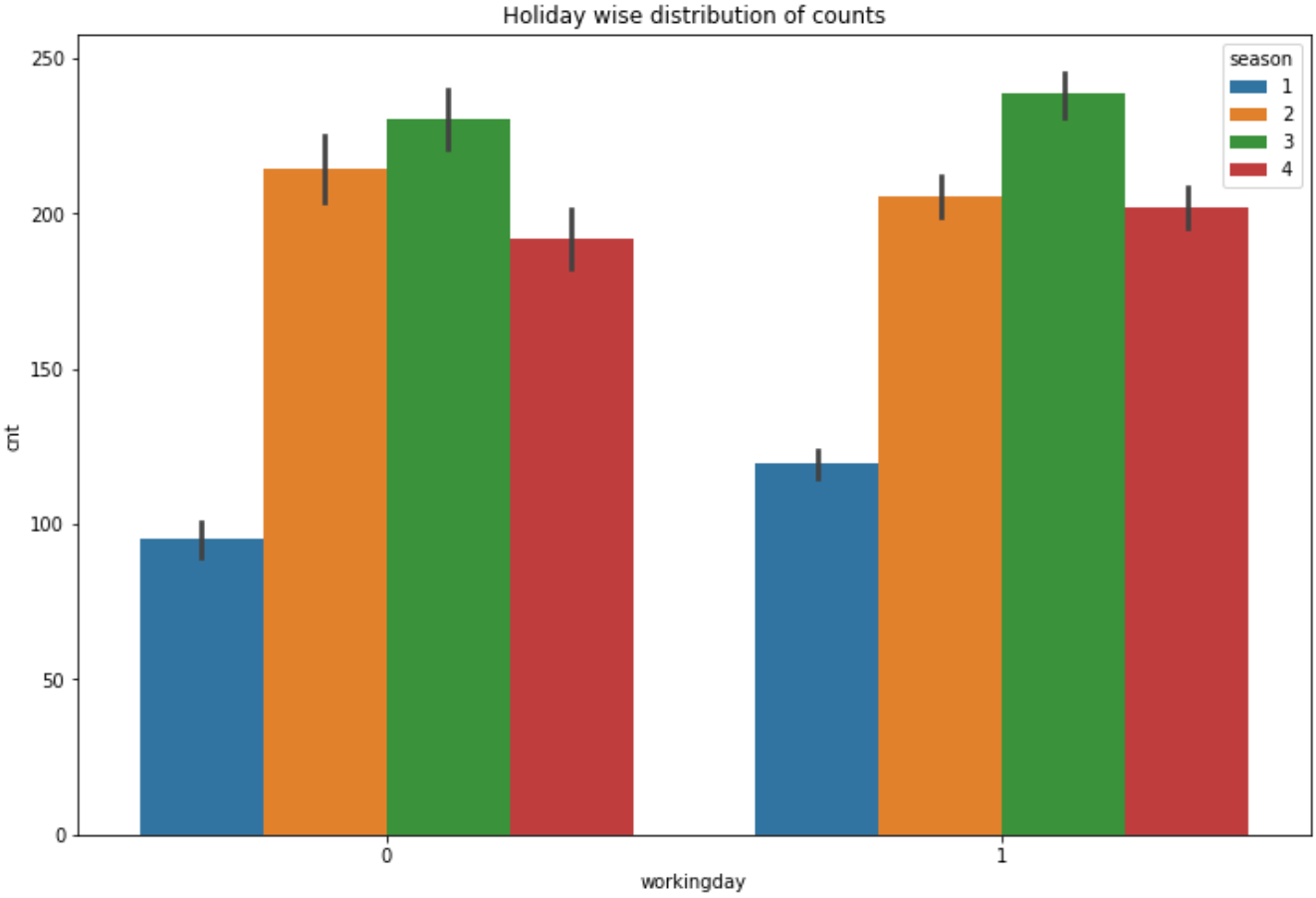

```
plot_stack_bar_chart(hour, 'holiday',('Yes','No'))
```




```
plt.figure(figsize=(12,8))
sns.barplot(x = hour['holiday'], y = hour['cnt'],hue = hour['season'])
plt.title('Holiday wise distribution of counts')
plt.show()
```




```
plt.figure(figsize=(12,8))
sns.barplot(x = hour['workingday'], y = hour['cnt'],hue = hour['season'])
plt.title('Holiday wise distribution of counts')
plt.show()
```


Holiday wise distribution of counts
season
1
2
3
4
cnt
250
200
150
100
50
0
0
1
workingday