In [1]:
```
import pandas as pd
```

In [2]:
```
df = pd.read_csv(r"C:/Users/Kusuma Reddy/Desktop/bikeshare.csv")
```

In [3]:
```
print(df)
```

```
       instant      dteday  season  yr  mnth  hr  holiday  weekday  \
0            1  2011-01-01       1   0     1   0        0        6
1            2  2011-01-01       1   0     1   1        0        6
2            3  2011-01-01       1   0     1   2        0        6
3            4  2011-01-01       1   0     1   3        0        6
4            5  2011-01-01       1   0     1   4        0        6
...        ...         ...     ...  ..   ...  ..      ...      ...
17374    17375  2012-12-31       1   1    12  19        0        1
17375    17376  2012-12-31       1   1    12  20        0        1
17376    17377  2012-12-31       1   1    12  21        0        1
17377    17378  2012-12-31       1   1    12  22        0        1
17378    17379  2012-12-31       1   1    12  23        0        1

       workingday  weathersit  temp   atemp   hum  windspeed  casual  \
0               0           1  0.24  0.2879  0.81     0.0000       3
1               0           1  0.22  0.2727  0.80     0.0000       8
2               0           1  0.22  0.2727  0.80     0.0000       5
3               0           1  0.24  0.2879  0.75     0.0000       3
4               0           1  0.24  0.2879  0.75     0.0000       0
...           ...         ...   ...     ...   ...        ...     ...
17374           1           2  0.26  0.2576  0.60     0.1642      11
17375           1           2  0.26  0.2576  0.60     0.1642       8
17376           1           1  0.26  0.2576  0.60     0.1642       7
17377           1           1  0.26  0.2727  0.56     0.1343      13
17378           1           1  0.26  0.2727  0.65     0.1343      12

       registered  cnt
0              13   16
1              32   40
2              27   32
3              10   13
4               1    1
...           ...  ...
17374         108  119
17375          81   89
17376          83   90
17377          48   61
17378          37   49

[17379 rows x 17 columns]
```

In [4]:
```
df.head()
```

Out[4]:

	instant	dteday	season	yr	mnth	hr	holiday	weekday	workingday	weathersit	temp	ate
0	1	2011-01-01	1	0	1	0	0	6	0	1	0.24	0.28

	instant	dteday	season	yr	mnth	hr	holiday	weekday	workingday	weathersit	temp	ate
1	2	2011-01-01	1	0	1	1	0	6	0	1	0.22	0.27
2	3	2011-01-01	1	0	1	2	0	6	0	1	0.22	0.27
3	4	2011-01-01	1	0	1	3	0	6	0	1	0.24	0.28

In [5]:
```
df.head(10)
```

Out[5]:

	instant	dteday	season	yr	mnth	hr	holiday	weekday	workingday	weathersit	temp	ate
0	1	2011-01-01	1	0	1	0	0	6	0	1	0.24	0.28
1	2	2011-01-01	1	0	1	1	0	6	0	1	0.22	0.27
2	3	2011-01-01	1	0	1	2	0	6	0	1	0.22	0.27
3	4	2011-01-01	1	0	1	3	0	6	0	1	0.24	0.28
4	5	2011-01-01	1	0	1	4	0	6	0	1	0.24	0.28
5	6	2011-01-01	1	0	1	5	0	6	0	2	0.24	0.25
6	7	2011-01-01	1	0	1	6	0	6	0	1	0.22	0.27
7	8	2011-01-01	1	0	1	7	0	6	0	1	0.20	0.25
8	9	2011-01-01	1	0	1	8	0	6	0	1	0.24	0.28
9	10	2011-01-01	1	0	1	9	0	6	0	1	0.32	0.34

In [6]:
```
df.tail()
```

Out[6]:

	instant	dteday	season	yr	mnth	hr	holiday	weekday	workingday	weathersit	temp
17374	17375	2012-12-31	1	1	12	19	0	1	1	2	0.26
17375	17376	2012-12-31	1	1	12	20	0	1	1	2	0.26
17376	17377	2012-12-31	1	1	12	21	0	1	1	1	0.26
17377	17378	2012-12-31	1	1	12	22	0	1	1	1	0.26
17378	17379	2012-12-31	1	1	12	23	0	1	1	1	0.26

In [7]:
```
df.tail(10)
```

Out[7]:

	instant	dteday	season	yr	mnth	hr	holiday	weekday	workingday	weathersit	temp
17369	17370	2012-12-31	1	1	12	14	0	1	1	2	0.28
17370	17371	2012-12-31	1	1	12	15	0	1	1	2	0.28
17371	17372	2012-12-31	1	1	12	16	0	1	1	2	0.26
17372	17373	2012-12-31	1	1	12	17	0	1	1	2	0.26
17373	17374	2012-12-31	1	1	12	18	0	1	1	2	0.26

	instant	dteday	season	yr	mnth	hr	holiday	weekday	workingday	weathersit	temp
17374	17375	2012-12-31	1	1	12	19	0	1	1	2	0.26
17375	17376	2012-12-31	1	1	12	20	0	1	1	2	0.26
17376	17377	2012-12-31	1	1	12	21	0	1	1	1	0.26
17377	17378	2012-12-31	1	1	12	22	0	1	1	1	0.26

In [8]:
```
df.describe()
```

Out[8]:

	instant	season	yr	mnth	hr	holiday	weekda
count	17379.0000	17379.000000	17379.000000	17379.000000	17379.000000	17379.000000	17379.0000
mean	8690.0000	2.501640	0.502561	6.537775	11.546752	0.028770	3.0036
std	5017.0295	1.106918	0.500008	3.438776	6.914405	0.167165	2.0057
min	1.0000	1.000000	0.000000	1.000000	0.000000	0.000000	0.0000
25%	4345.5000	2.000000	0.000000	4.000000	6.000000	0.000000	1.0000
50%	8690.0000	3.000000	1.000000	7.000000	12.000000	0.000000	3.0000
75%	13034.5000	3.000000	1.000000	10.000000	18.000000	0.000000	5.0000
max	17379.0000	4.000000	1.000000	12.000000	23.000000	1.000000	6.0000

In [9]:
```
df.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
```

In [11]:
```
df.shape
```

Out[11]:
```
(17379, 17)
```

In [12]:
```
df['instant'].unique()
```

Out[12]:
```
array([    1,     2,     3, ..., 17377, 17378, 17379], dtype=int64)
```

In [13]:
```
df['instant'].nunique()
```

Out[13]:
```
17379
```

In [14]:
```
df[["holiday","weekday"]]
```

Out[14]:

	holiday	weekday
0	0	6
1	0	6
2	0	6
3	0	6
4	0	6
...	...	...
17374	0	1
17375	0	1
17376	0	1
17377	0	1
17378	0	1

17379 rows × 2 columns

In [15]:
```
df.loc[:,["holiday","weekday"]]
```

Out[15]:

	holiday	weekday
0	0	6
1	0	6
2	0	6
3	0	6
4	0	6
...	...	...

	holiday	weekday
17374	0	1
17375	0	1
17376	0	1
17377	0	1
17378	0	1

In [16]:
```
df.iloc[:,:2]
```

Out[16]:

	instant	dteday
0	1	2011-01-01
1	2	2011-01-01
2	3	2011-01-01
3	4	2011-01-01
4	5	2011-01-01
...	...	...
17374	17375	2012-12-31
17375	17376	2012-12-31
17376	17377	2012-12-31
17377	17378	2012-12-31
17378	17379	2012-12-31

17379 rows × 2 columns

In [37]:
```
df.dtypes
```

Out[37]:
```
instant         int64
dteday         object
season          int64
yr              int64
mnth            int64
hr              int64
holiday         int64
weekday         int64
workingday      int64
weathersit      int64
temp          float64
atemp         float64
hum           float64
windspeed     float64
casual          int64
registered      int64
```

```
cnt             int64
```

In [38]:
```
df = df.drop(['weathersit','atemp','casual'], axis = 1)
df.head(5)
```

Out[38]:

	instant	dteday	season	yr	mnth	hr	holiday	weekday	workingday	temp	hum	windspee
0	1	2011-01-01	1	0	1	0	0	6	0	0.24	0.81	0.
1	2	2011-01-01	1	0	1	1	0	6	0	0.22	0.80	0.
2	3	2011-01-01	1	0	1	2	0	6	0	0.22	0.80	0.
3	4	2011-01-01	1	0	1	3	0	6	0	0.24	0.75	0.
4	5	2011-01-01	1	0	1	4	0	6	0	0.24	0.75	0.

In [39]:
```
df = df.rename(columns={"instant":"Instant", "dteday":"Date", "season":"Season", "yr"
df.head(5)
```

Out[39]:

	Instant	Date	Season	Year	Month	Hour	Holiday	weekday	workingday	temp	hum	wi
0	1	2011-01-01	1	0	1	0	0	6	0	0.24	0.81	
1	2	2011-01-01	1	0	1	1	0	6	0	0.22	0.80	
2	3	2011-01-01	1	0	1	2	0	6	0	0.22	0.80	
3	4	2011-01-01	1	0	1	3	0	6	0	0.24	0.75	
4	5	2011-01-01	1	0	1	4	0	6	0	0.24	0.75	

In [41]:
```
df.shape
```

Out[41]:
```
(17379, 14)
```

In [42]:
```
df.count()
```

Out[42]:
```
Instant       17379
Date          17379
Season        17379
Year          17379
Month         17379
Hour          17379
Holiday       17379
weekday       17379
workingday    17379
temp          17379
hum           17379
windspeed     17379
registered    17379
cnt           17379
dtype: int64
```

In [43]:
```
df =df.drop_duplicates()
df.head()
```

Out[43]:

	Instant	Date	Season	Year	Month	Hour	Holiday	weekday	workingday	temp	hum	wi
0	1	2011-01-01	1	0	1	0	0	6	0	0.24	0.81	
1	2	2011-01-01	1	0	1	1	0	6	0	0.22	0.80	
2	3	2011-01-01	1	0	1	2	0	6	0	0.22	0.80	
3	4	2011-01-01	1	0	1	3	0	6	0	0.24	0.75	
4	5	2011-01-01	1	0	1	4	0	6	0	0.24	0.75	

In [44]:
```
df.count()
```

Out[44]:
```
Instant       17379
Date          17379
Season        17379
Year          17379
Month         17379
Hour          17379
Holiday       17379
weekday       17379
workingday    17379
temp          17379
hum           17379
windspeed     17379
registered    17379
cnt           17379
dtype: int64
```

In [45]:
```
print(df.isnull().sum())
```

```
Instant       0
Date          0
Season        0
Year          0
Month         0
Hour          0
Holiday       0
weekday       0
workingday    0
temp          0
hum           0
windspeed     0
registered    0
cnt           0
dtype: int64
```

In [46]:
```
df = df.dropna()
df.count()
```

Out[46]:
```
Instant       17379
```

```
Date          17379
Season        17379
Year          17379
Month         17379
Hour          17379
Holiday       17379
weekday       17379
workingday    17379
temp          17379
hum           17379
windspeed     17379
registered    17379
cnt           17379
```

In [47]:
```
print(df.isnull().sum())
```

```
Instant       0
Date          0
Season        0
Year          0
Month         0
Hour          0
Holiday       0
weekday       0
workingday    0
temp          0
hum           0
windspeed     0
registered    0
cnt           0
dtype: int64
```

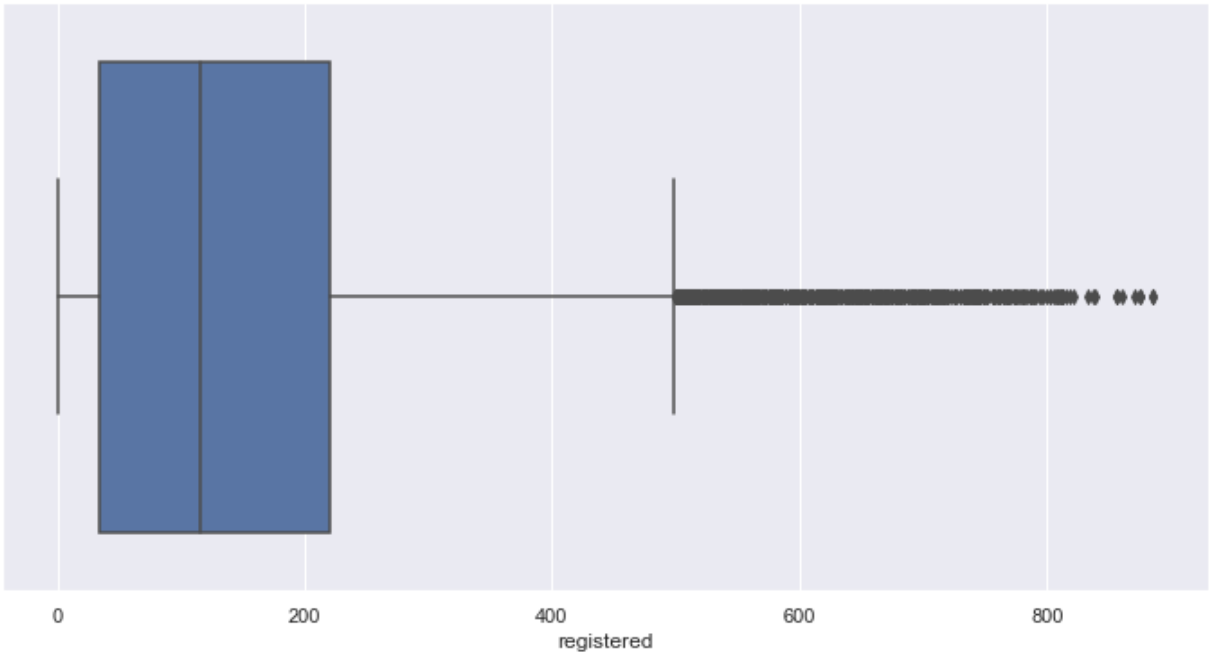
In [48]:
```
import seaborn as sns
```

In [49]:
```
sns.boxplot(x = df['Instant'])
```

Out[49]:
```
<AxesSubplot:xlabel='Instant'>
```

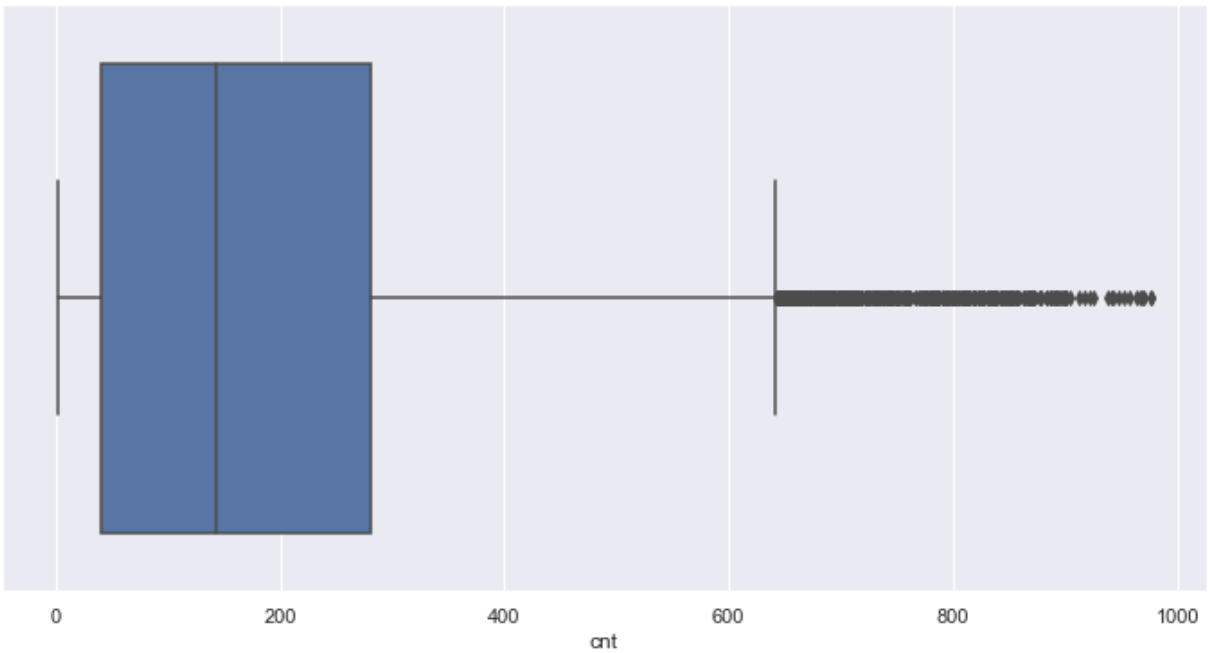
In [50]:
```
sns.boxplot(x = df['registered'])
```

Out[50]:
```
<AxesSubplot:xlabel='registered'>
```



In [51]:
```
sns.boxplot(x = df['cnt'])
```

Out[51]:
```
<AxesSubplot:xlabel='cnt'>
```



In [54]:
```
Q1 = df.quantile(0.25)
Q3 = df.quantile(0.75)
IQR = Q3-Q1
print(IQR)
```

```
Instant       8689.0000
Season           1.0000
Year             1.0000
Month            6.0000
Hour            12.0000
Holiday          0.0000
weekday          4.0000
workingday       1.0000
temp             0.3200
hum              0.3000
windspeed        0.1492
registered     186.0000
cnt            241.0000
dtype: float64
```

In [56]:
```
df = df[~((df < (Q1 - 1.5 * IQR)) |(df > (Q3 + 1.5 * IQR))).any(axis=1)]
df.shape
```

```
C:\Users\KUSUMA~1\AppData\Local\Temp/ipykernel_15268/4147643536.py:1: FutureWarning:
Automatic reindexing on DataFrame vs Series comparisons is deprecated and will raise
ValueError in a future version.  Do `left, right = left.align(right, axis=1, copy=Fal
se)` before e.g. `left == right`
  df = df[~((df < (Q1 - 1.5 * IQR)) |(df > (Q3 + 1.5 * IQR))).any(axis=1)]
```

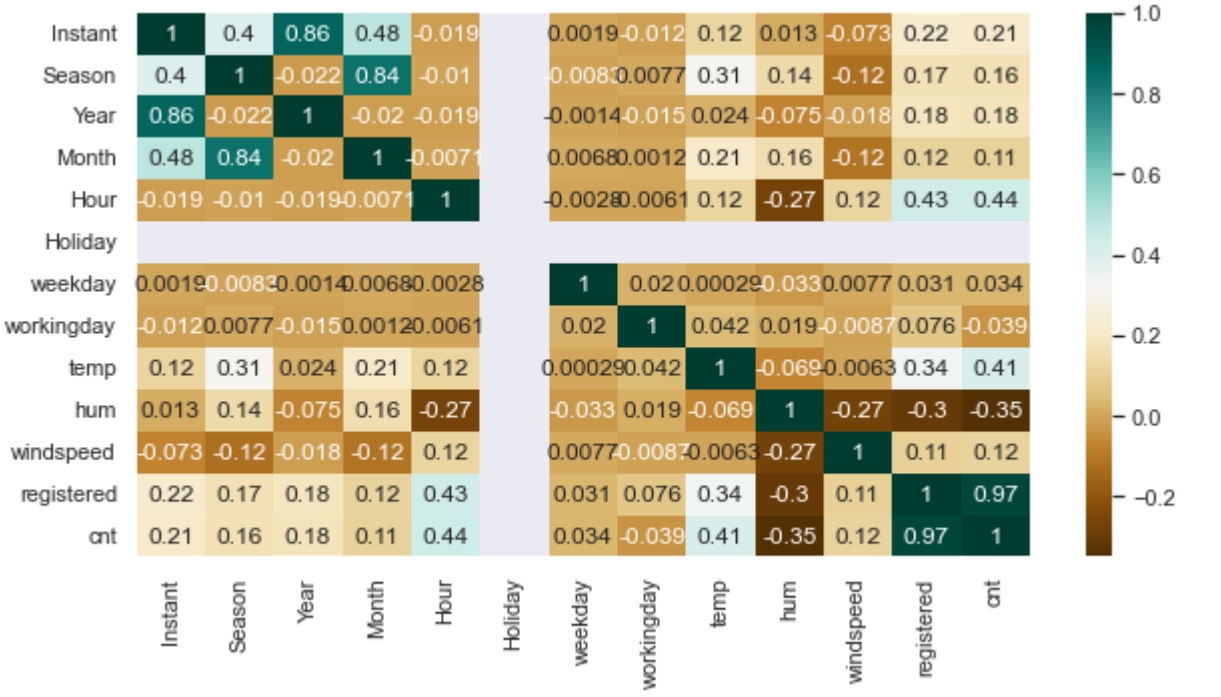
Out[56]:
```
(15774, 14)
```

In [58]:
```
plt.figure(figsize=(10,5))
c= df.corr()
sns.heatmap(c,cmap="BrBG",annot=True)
c
```

Out[58]:

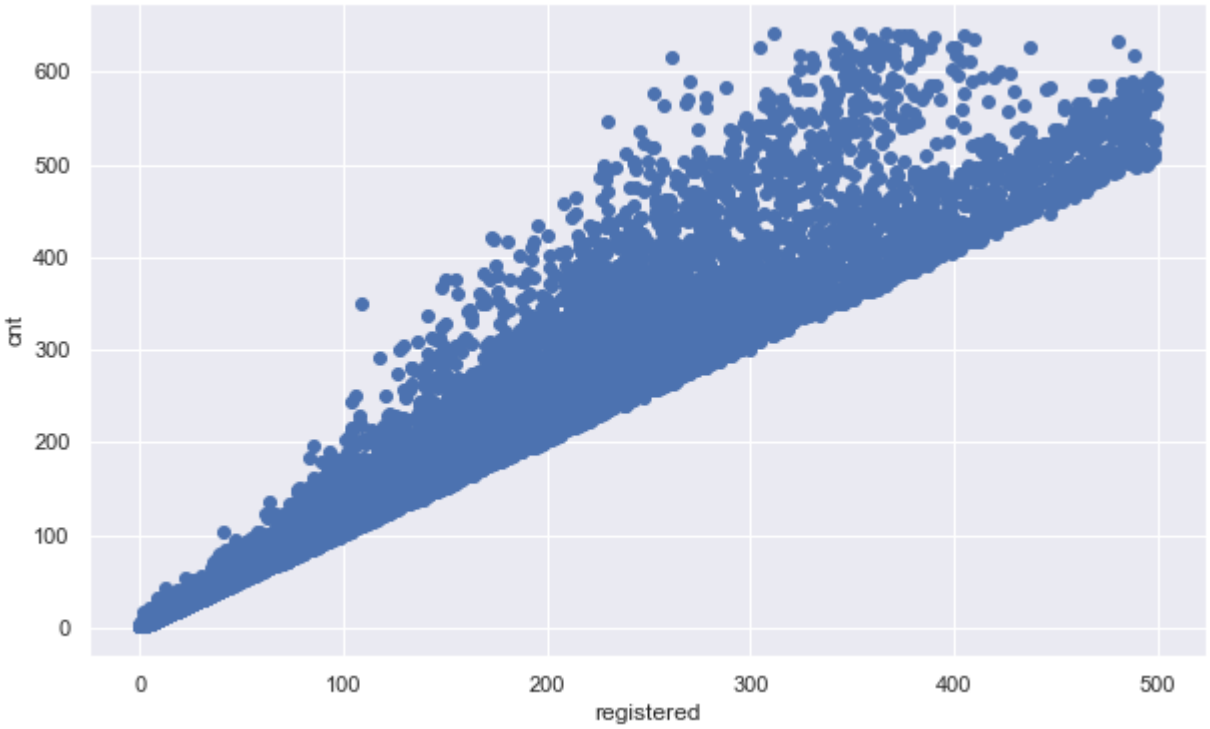
	Instant	Season	Year	Month	Hour	Holiday	weekday	workingday	
Instant	1.000000	0.398446	0.864740	0.483401	-0.018782	NaN	0.001905	-0.012489	(
Season	0.398446	1.000000	-0.022491	0.835559	-0.009985	NaN	-0.008327	0.007685	(
Year	0.864740	-0.022491	1.000000	-0.019604	-0.019158	NaN	-0.001387	-0.015421	(
Month	0.483401	0.835559	-0.019604	1.000000	-0.007058	NaN	0.006844	0.001177	(
Hour	-0.018782	-0.009985	-0.019158	-0.007058	1.000000	NaN	-0.002787	-0.006146	(
Holiday	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
weekday	0.001905	-0.008327	-0.001387	0.006844	-0.002787	NaN	1.000000	0.020382	(
workingday	-0.012489	0.007685	-0.015421	0.001177	-0.006146	NaN	0.020382	1.000000	(
temp	0.124348	0.312877	0.024309	0.205316	0.124201	NaN	0.000288	0.041590	'
hum	0.013224	0.141629	-0.074721	0.155951	-0.267004	NaN	-0.032515	0.018656	-(

	Instant	Season	Year	Month	Hour	Holiday	weekday	workingday	
windspeed	-0.073100	-0.123763	-0.017714	-0.116399	0.124979	NaN	0.007664	-0.008651	-(
registered	0.216010	0.165263	0.178330	0.119911	0.431387	NaN	0.031450	0.076477	(



In [60]:

```
fig, ax = plt.subplots(figsize=(10,6))
ax.scatter(df['registered'], df['cnt'])
ax.set_xlabel('registered')
ax.set_ylabel('cnt')
plt.show()
```



In [ ]: