In [1]:

```
import pandas as pd
import matplotlib.pyplot as plt
```

In [2]:

```
# reading dataset
ds = pd.read_csv('culturefood.csv')
```

CityWise Analysis:

In [3]:

```
ds['City'] = ds['City'].str.strip() #removing unwanted character from data
```

In [4]:

```
#Making labels consistent throughout
ds.loc[ds['City'] == 'Kharghar, Navi Mumbai' , 'City'] = 'Navi Mumbai'
ds.loc[ds['City'] == 'Mumbai, MH' , 'City'] = 'Mumbai'
ds.loc[ds['City'].str.lower() == 'pune' , 'City'] = 'Pune'
ds.loc[ds['City'].str.lower() == 'new delhi', 'City'] = 'New Delhi'
ds.loc[ds['City'].str.lower() == 'navi mumbai', 'City'] = 'Navi Mumbai'
ds.loc[ds['City'].str.lower() == 'bengaluru', 'City'] = 'Bengaluru'
ds.loc[ds['City'].str.contains('Kalyan'), 'City'] = 'Thane'
ds.loc[ds['City'].str.contains('Ahmedabad'), 'City'] = 'Ahmedabad'
```

Number of Entries from each cities:

In [5]:

```
print("City names and Number of entries from each: ")
city_frame = pd.DataFrame(data = {'Names': ds.City.unique() ,
                                  'NumEntries': [ds.City.value_counts()[str(data)] for data in ds.C
ty.unique()] })
city_frame['Percentage:'] = [data/city_frame['NumEntries'].sum()*100 for data in city_frame['NumEnt
ries']]
print(city_frame.sort_values('NumEntries', ascending=False).to_string(index=False))
```

```
City names and Number of entries from each:
         Names  NumEntries  Percentage:
          Pune          44    16.000000
        Mumbai          30    10.909091
       Jodhpur          21     7.636364
     Ahmedabad          19     6.909091
     Bangalore          14     5.090909
     Hyderabad          13     4.727273
         Delhi          12     4.363636
       Chennai           8     2.909091
         Thane           8     2.909091
     New Delhi           8     2.909091
     Bengaluru           7     2.545455
        Nagpur           6     2.181818
   Navi Mumbai           6     2.181818
       Gurgaon           6     2.181818
         Surat           5     1.818182
        Jaipur           4     1.454545
      Vadodara           4     1.454545
    Chandigarh           4     1.454545
        Indore           4     1.454545
       Kolkata           4     1.454545
      Ludhiana           3     1.090909
        Nashik           2     0.727273
      New York           2     0.727273
         Latur           2     0.727273
        Rajkot           2     0.727273
     Kathmandu           2     0.727273
```

```
        Bhopal           2     0.727273
       Manipal           2     0.727273
         jaipur          1     0.363636
     East Blue           1     0.363636
        Barmer           1     0.363636
          Vapi           1     0.363636
     Melbourne           1     0.363636
        JAIPUR           1     0.363636
           Goa           1     0.363636
         Noida           1     0.363636
        Kuwait           1     0.363636
    Ahmednagar           1     0.363636
    Port Blair           1     0.363636
     Osmanabad           1     0.363636
       Udaipur           1     0.363636
        Ranchi           1     0.363636
      Varanasi           1     0.363636
     Rishikesh           1     0.363636
     Kalimpong           1     0.363636
     Jalandhar           1     0.363636
   Bhubaneswar           1     0.363636
         Vizag           1     0.363636
     Greenwich           1     0.363636
      Gurugram           1     0.363636
 Visakhapatnam           1     0.363636
      Bhusaval           1     0.363636
      Brampton           1     0.363636
      Bhusawal           1     0.363636
         Ajmer           1     0.363636
   Chicago Usa           1     0.363636
     Faridabad           1     0.363636
     Kharagpur           1     0.363636
 VISAKHAPATNAM           1     0.363636
```

How do People Manage thier meals:

In [47]:

```
meals = pd.DataFrame(data = {'Cook' : [ ds['managingmeals'].str.contains('A cook who prepares them
for me').value_counts()[1] ],
                             'Outside': [ ds['managingmeals'].str.contains('Eat Outside \(order onl
ne or go out to dine\)').value_counts()[1] ],
                            'Ownmeals' : [ ds['managingmeals'].str.contains('Cook my own meals').va
ue_counts()[1] ] } )
print("Number of Entries:")
print(meals)
```

```
Number of Entries:
   Cook  Outside  Ownmeals
0   148       79       112
```

In [43]:

```
mealpercentage = pd.DataFrame(data = {'Preference': ['Cook' , 'Outside' , 'Ownmeals'] ,'percentage'
: [  meals['Cook']/(meals['Cook'] + meals['Outside'] + meals['Ownmeals'] ) * 100 ,
                                    meals['Outside']/(meals['Cook'] + meals['Outside'] +
meals['Ownmeals'] ) * 100 ,
                                    meals['Ownmeals']/(meals['Cook'] + meals['Outside'] +
meals['Ownmeals'] ) * 100 ]})
```

In [40]:

```
mealpercentage
```

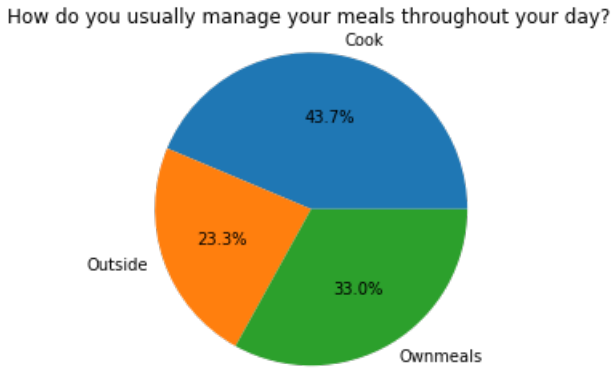
Out[40]:

	Preference	percentage
0	Cook	0 43.657817 dtype: float64
1	Outside	0 23.303835 dtype: float64
2	Ownmeals	0 33.038348 dtype: float64

Preference percentage

In [46]:

```
plt.pie(mealpercentage['percentage'],labels=mealpercentage['Preference'],autopct='%1.1f%%',radius = 0.1)
plt.title('How do you usually manage your meals throughout your day?')
plt.axis('equal')
#plt.tight_layout()
plt.show()
```



City Wise meals management:

In [53]:

```
listcities = ['Pune', 'Mumbai', 'Delhi']
```

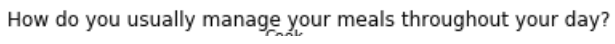
In [61]:

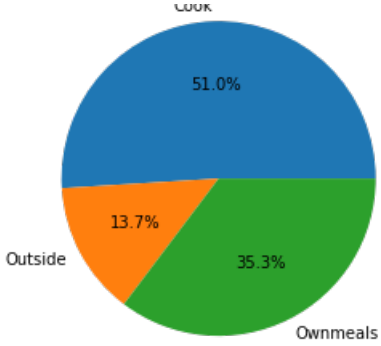
```
def labelwisepie(labelname , column):
    print("For ", labelname , ": ")
    mealscity = ds.loc[ds[column] == labelname, 'managingmeals']
    meals = pd.DataFrame(data = {'Cook' : [ mealscity.str.contains('A cook who prepares them for me').value_counts()[1] ],
                                 'Outside': [ mealscity.str.contains('Eat Outside \(order online or go out to dine\)').value_counts()[1] ],
                                'Ownmeals' : [ mealscity.str.contains('Cook my own meals').value_counts()[1] ] } )
    print("Number of Entries:")
    print(meals)
    mealpercentage = pd.DataFrame(data = {'Preference': ['Cook' , 'Outside' , 'Ownmeals'] ,'percentage': [  meals['Cook']/(meals['Cook'] + meals['Outside'] + meals['Ownmeals'] ) * 100 ,
                                        meals['Outside']/(meals['Cook'] + meals['Outside'] + meals['Ownmeals'] ) * 100 ,
                                        meals['Ownmeals']/(meals['Cook'] + meals['Outside'] + meals['Ownmeals'] ) * 100 ]})
    plt.pie(mealpercentage['percentage'],labels=mealpercentage['Preference'],autopct='%1.1f%%',radius = 0.1)
    plt.title('How do you usually manage your meals throughout your day?')
    plt.axis('equal')
    #plt.tight_layout()
    plt.show()
```

In [62]:

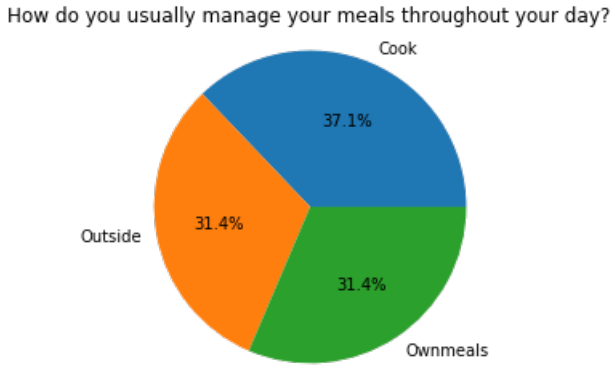
```
column = 'City'
for data in listcities:
    labelwisepie(data , column)
```

```
For  Pune :
Number of Entries:
   Cook  Outside  Ownmeals
0    26        7        18
```

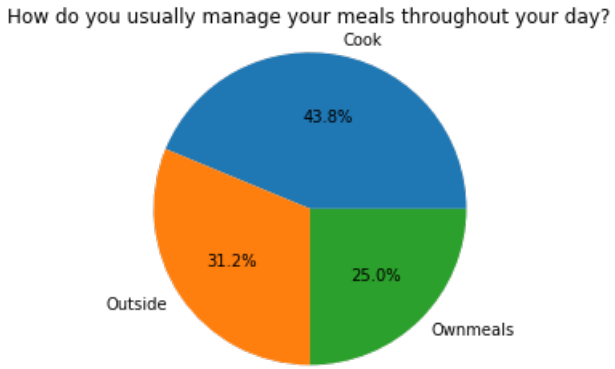





```
For  Mumbai :
Number of Entries:
   Cook  Outside  Ownmeals
0    13       11        11
```




```
For  Delhi :
Number of Entries:
   Cook  Outside  Ownmeals
0     7        5         4
```



Meal Management according to relationship status:

In [67]:

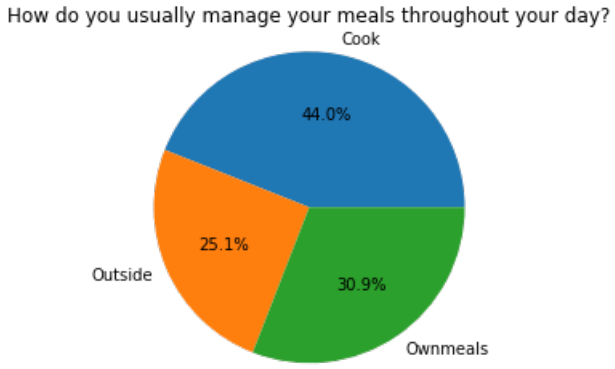
```
liststatus = ['Single','Married','Prefer not to say'] #not using relationship since only one entry found
```

In [68]:

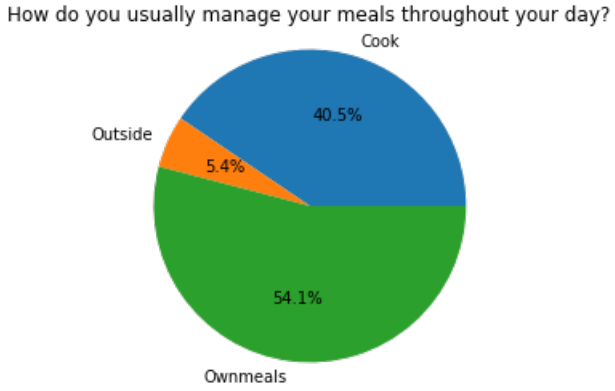
```
for data in liststatus:
    labelwisepie(data , 'MaritalStatus')
```

```
For  Single :
Number of Entries:
   Cook  Outside  Ownmeals
```

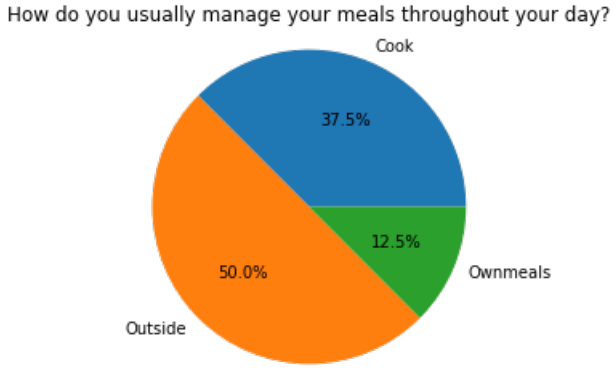
```
0    128        73         90
```




```
For  Married :
Number of Entries:
   Cook  Outside  Ownmeals
0    15        2        20
```




```
For  Prefer not to say :
Number of Entries:
   Cook  Outside  Ownmeals
0     3        4         1
```



Meal Management with respect to age:

In [81]:

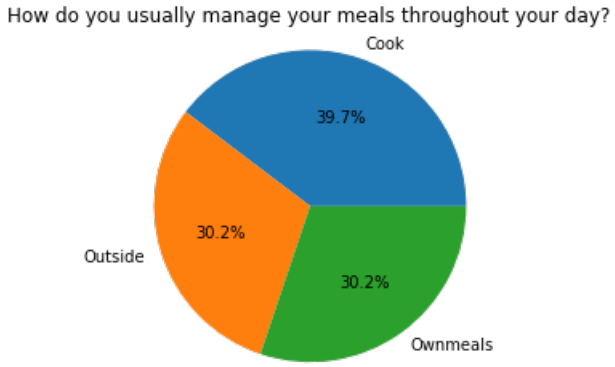
```
def labelwisepieage(labelname , column):
    print("For ", labelname , ": ")
    if labelname == '25to30':
        ages = ds.loc[ds['Age'] > 25 ,['Age', 'managingmeals'] ]
        mealscity = ages.loc[ages['Age'] < 30 , 'managingmeals' ]
    if labelname == 'lessthan25':
        mealscity = ds.loc[ds['Age'] < 25 ,'managingmeals' ]
    if labelname == 'morethan30':
        mealscity = ds.loc[ds['Age'] > 30 ,'managingmeals' ]
```

```
    meals = pd.DataFrame(data = {'Cook' : [ mealscity.str.contains('A cook who prepares them for me
').value_counts()[1] ],
                                 'Outside': [ mealscity.str.contains('Eat Outside \(order online or
go out to dine\)').value_counts()[1] ],
                                'Ownmeals' : [ mealscity.str.contains('Cook my own meals').value_co
nts()[1] ] } )
    print("Number of Entries:")
    print(meals)
    mealpercentage = pd.DataFrame(data = {'Preference': ['Cook' , 'Outside' , 'Ownmeals'] ,'percent
age': [  meals['Cook']/(meals['Cook'] + meals['Outside'] + meals['Ownmeals'] ) * 100 ,
                                          meals['Outside']/(meals['Cook'] + meals['Outside'] + meals
Ownmeals'] ) * 100 ,
                                          meals['Ownmeals']/(meals['Cook'] + meals['Outside'] + meal
'Ownmeals'] ) * 100 ]})
    plt.pie(mealpercentage['percentage'],labels=mealpercentage['Preference'],autopct='%1.1f%%',radi
us = 0.1)
    plt.title('How do you usually manage your meals throughout your day?')
    plt.axis('equal')
    #plt.tight_layout()
    plt.show()
```

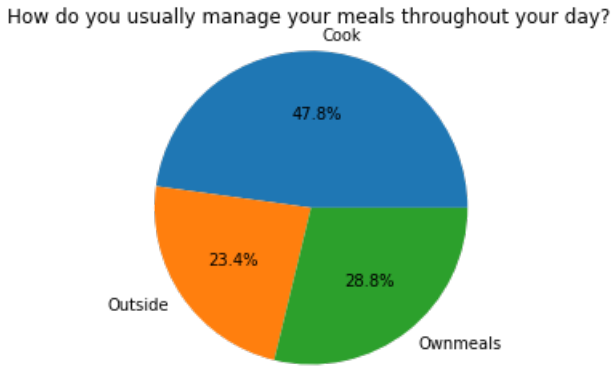
In [82]:

```
Agegroup = ['25to30', 'lessthan25', 'morethan30']
for data in Agegroup:
    labelwisepieage(data , 'Age')
```

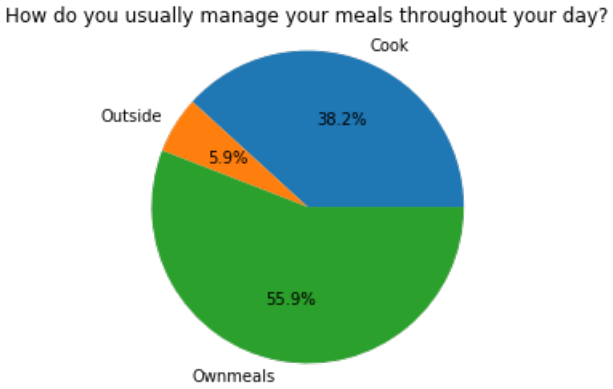
```
For  25to30 :
Number of Entries:
   Cook  Outside  Ownmeals
0    25       19        19
```




```
For  lessthan25 :
Number of Entries:
   Cook  Outside  Ownmeals
0    98       48        59
```




```
For  morethan30 :
Number of Entries:
   Cook  Outside  Ownmeals
0    13        2        19
```



Meal Management with respect to employment status:

In [83]:

```
ds.CurrentEmploymentStatus.unique()
```

Out[83]:

```
array(['Self employed', 'Employed full time', 'Student', 'Freelancer',
       'Employed part time', 'Unemployed', 'MC :) ', 'Home maker',
       'Homemaker', 'Homemaker '], dtype=object)
```

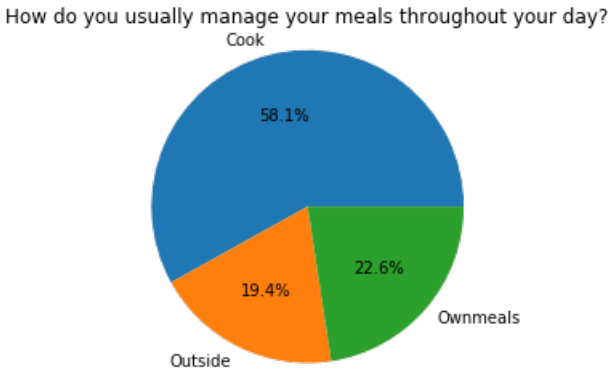
In [85]:

```
employmentstatus = ['Self employed' ,'Employed full time', 'Student', 'Freelancer', 'Employed part time', 'Unemployed']
```

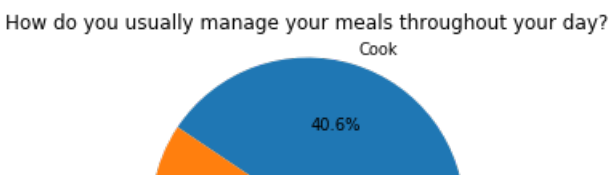
In [86]:

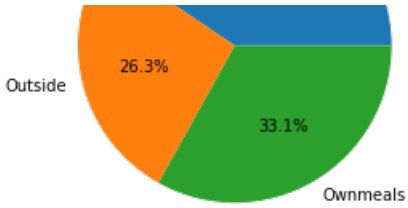
```
for data in employmentstatus:
    labelwisepie(data , 'CurrentEmploymentStatus')
```

```
For  Self employed :
Number of Entries:
   Cook  Outside  Ownmeals
0    18        6         7
```

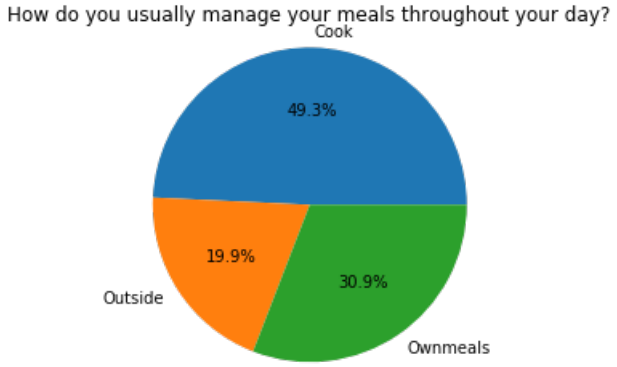



```
For  Employed full time :
Number of Entries:
   Cook  Outside  Ownmeals
0    54       35        44
```

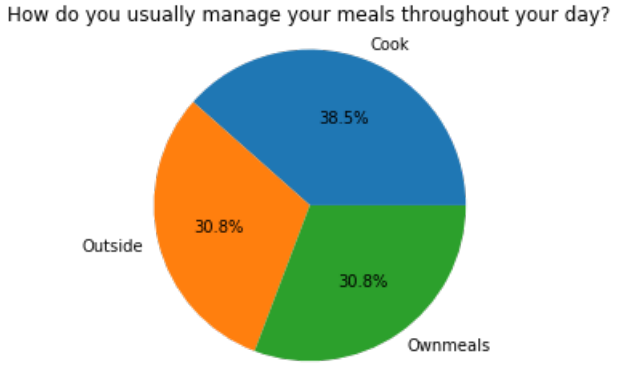




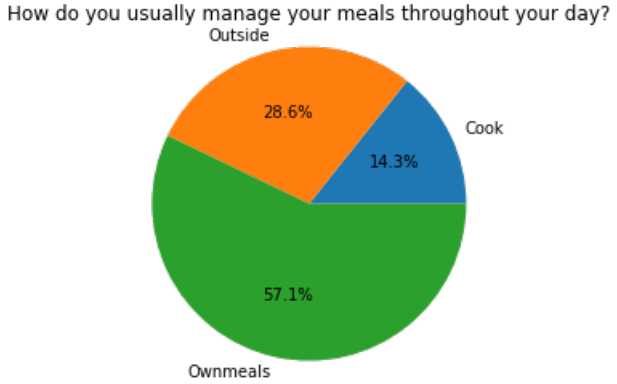

```
For  Student :
Number of Entries:
   Cook  Outside  Ownmeals
0    67       27        42
```




```
For  Freelancer :
Number of Entries:
   Cook  Outside  Ownmeals
0     5        4         4
```

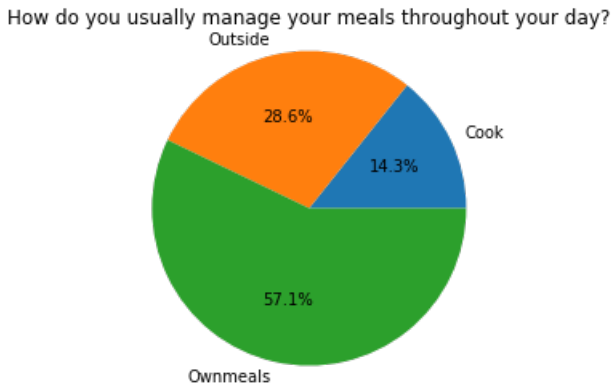



```
For  Employed part time :
Number of Entries:
   Cook  Outside  Ownmeals
0     1        2         4
```




```
For  Unemployed :
```

```
Number of Entries:
   Cook  Outside  Ownmeals
0     2        4         8
```



In [ ]: