1. Read the visit_data.csv file.
2. Check for duplicates.
3. Check if any essential column contains NaN.
4. Get rid of the outliers.
5. Report the size difference.
6. Create a box plot to check for outliers. Get rid of any outliers.

In [25]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

%matplotlib inline
```

1. Read the visit_data.csv file.

In [26]:
```
df = pd.read_csv("visit_data.csv")
```

In [27]:
```
df.head()
```

Out[27]:

	id	first_name	last_name	email	gender	ip_address	
0	1	Sonny	Dahl	sdahl0@mysql.com	Male	135.36.96.183	1
1	2	NaN	NaN	dhoovart1@hud.gov	NaN	237.165.194.143	
2	3	Gar	Armal	garmal2@technorati.com	NaN	166.43.137.224	
3	4	Chiarra	Nulty	cnulty3@newyorker.com	NaN	139.98.137.108	1
4	5	NaN	NaN	sleaver4@elegantthemes.com	NaN	46.117.117.27	2

1. Check for duplicates.

In [28]:
```
print("First name is duplictaed - {}".format(any(df.first_name.duplicated
()))
print("Last name is duplictaed - {}".format(any(df.last_name.duplicated
()))
print("Email is duplictaed - {}".format(any(df.email.duplicated())))
```

```
First name is duplictaed - True
Last name is duplictaed - True
Email is duplictaed - False
```

1. Check if any essential column contains NaN.

In [29]:
```
print("The column Email contains NaN - %r " % df.email.isnull().values.any())
print("The column IP Address contains NaN - %s " % df.ip_address.isnull().values.any())
print("The column Visit contains NaN - %s " % df.visit.isnull().values.any())
```

```
The column Email contains NaN - False
The column IP Address contains NaN - False
The column Visit contains NaN - True
```

1. Get rid of the outliers.

In [30]:
```
print('Max number of visits' ,df['visit'].max())
print('Min number of visits' ,df['visit'].min())
print('Avg number of visits' ,  df['visit'].mean())
print('std number of visits' ,df['visit'].std())

outlier_upper_bound = df['visit'].mean() + df['visit'].std()
outlier_lower_bound = df['visit'].mean() - df['visit'].std()

print('outlier upper bound' ,outlier_upper_bound)
print('outlier lower bound' ,outlier_lower_bound  )
```

```
Max number of visits 2998.0
Min number of visits 1.0
Avg number of visits 1497.976386036961
std number of visits 838.959459554409
outlier upper bound 2336.93584559137
outlier lower bound 659.016926482552
```

In [31]:
```
size_prev = df.shape
df_noOutlier = df[(df['visit'] <= outlier_upper_bound) & (df['visit'] >= outlier_lower_bound)]
size_after = df_noOutlier.shape
```

1. Report the size difference.

In [32]:
```
# Notice how parameterized format is used and then the indexing is working inside the quote marks
print("The size of previous data was - {prev[0]} rows and the size of the new one is - {after[0]} rows".
      format(prev=size_prev, after=size_after))
```

```
The size of previous data was - 1000 rows and the size of the new one is - 578 rows
```

1. Create a box plot to check for outliers. Get rid of any outliers.

In [24]:
```
plt.boxplot(df.visit, notch=True)
```

Out[24]:
```
{'whiskers': [<matplotlib.lines.Line2D at 0x62bce30>,
  <matplotlib.lines.Line2D at 0x51d70d0>],
 'caps': [<matplotlib.lines.Line2D at 0x51d7350>,
  <matplotlib.lines.Line2D at 0x51d75d0>],
 'boxes': [<matplotlib.lines.Line2D at 0x62bcbb0>],
 'medians': [<matplotlib.lines.Line2D at 0x51d7870>],
 'fliers': [<matplotlib.lines.Line2D at 0x51d7af0>],
 'means': []}
```