solar

June 25, 2020

Peacock Solar Project: Exploratory Data Analysis

We have the **Customer DataSet** provided by the company. This dataset contains only Likert and Categorical data

Operations Performed on This DataSet: 1. Importing required libraries 2. General overview of the dataset 3. Visualizing data for better understanding 4. Creating an algorithm for plotting Ratings Data against Conditions 5. Visualizing data using this algorithm 6. Conclusion and summary of our findings

Table of Contents: 1. Importing required Libraries and DataSet 2. Description of the DataSet 3. Analysing missing values and then plotting a heatmap for visualization 4. Plotting a Correlational Heatmap for the data 5. Counting number of projects in individual city and plotting graphs for visualization 6. Counting ratings given by the customers that can affect their purchase decisions and plotting visualizations 7. Sorting cities based on considerable number of projects and suitable type of house customers live in 8. Analysing and visualizing customers based on these categories: a. Average Monthly Income b. Type of House c. Active EMI d. Percentage of Roof for Solar Installation e. Home Loan f. Type of Organization g. Expectations on Saving Electricity Bill h. Maximum Investment in Solar Technology i. Payment Method 9. Creating a custom algorithm for plotting data against required conditions 10. Plotting data using this algorithm and getting valuable insights from it 11. Conclusion, List of cities for expanding business and Summary of findings

Importing required libraries

[1]:
```
#importing required libraries
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

Importing the Dataset

Here we import our cleaned dataset

[2]:
```
#creating a DataFrame for storing the data set
df_solar = pd.read_csv('final_solar_dataset.csv')
```

Data Description

Show Data Head

This displays the upper 10 rows of our dataset

[3]:
```
#displaying initial 10 rows of the data set
df_solar.head(10)
```

[3]:

	organization_type	profession	type_of_home	number_of_floors	rooms	build_additional_floor	percentage_of_roof	home_loan	two_wheelers	four_wheelers	…	concerns_on_outlook_of_the_house
0	Business	Marketing	Independent house or Villa	3	3	No	<25%	No	2	1	…	1
1	Private	Housewifr	Housewife	1	4	No	26-50%	No	2	1	…	2
2	Government	REVENUE TALATI	Independent house or Villa	2	3	No	<25%	No	2	1	…	1
3	Private	Working as an sales executive	Apartment or Flat	1	4	No	<25%	Yes	1	2	…	4
4	Government	Professor	Independent house or Villa	1	5	No	26-50%	No	0	0	…	3
5	Business	Shop keeper	Independent house or Villa	2	4	No	26-50%	No	0	1	…	3
6	Private	Engineer	Independent house or Villa	2	5	Yes	<25%	Yes	3	2	…	4
7	Student	Student	Independent house or Villa	1	4	No	26-50%	Yes	0	1	…	2
8	Student	Student	Independent house or Villa	1	4	No	26-50%	Yes	2	2	…	3
9	Student	Student	Independent house or Villa	1	3	No	<25%	No	1	0	…	2

```
5              1  …                                1
6              0  …                                1
7              1  …                                2
8              0  …                                4
9              1  …                                2

  not_understanding_solar_systems not_knowing_good_solar_installer  \
0                               1                                  1
1                               2                                  2
2                               1                                  1
3                               2                                  2
4                               2                                  3
5                               2                                  3
6                               1                                  1
7                               1                                  2
8                               4                                  2
9                               2                                  2

  non_suitable_rooftop  expectation_on_saving_electricity_bill  \
0                    1                              10% - 20%
1                    2                              10% - 20%
2                    1                              41% - 50%
3                    4                              21% - 30%
4                    2                              31% - 40%
5                    1                              21% - 30%
6                    1                              21% - 30%
7                    2                              21% - 30%
8                    2                              21% - 30%
9                    2                              10% - 20%

   maximum_investment_in_solar         solar_companies_you_know       city  \
0             Less than 50,000                          Namaste  Hyderabad
1             Less than 50,000                       Don'T Know     Nashik
2             Less than 50,000                             Surya     Rajkot
3             Less than 50,000                         Kirloskar     Panvel
4             Less than 50,000                     Peacock Solar    Lucknow
5             Less than 50,000                               Nil       Kota
6             Less than 50,000                               Sun   Kolhapur
7             Less than 50,000  I Don'T  Know Any Solar Company       Kota
8             Less than 50,000                      Walgreen Co.       Kota
9             Less than 50,000                      I Don'T Know       Kota

   state  know_any_solar_installation_companies
0    NaN                                   Yes
1    NaN                                    No
2    NaN                                   Yes
3    NaN                                    No
```

```
4    NaN                                    No
5    NaN                                    No
6    NaN                                   Yes
7    NaN                                    No
8    NaN                                    No
9    NaN                                    No

[10 rows x 38 columns]
```

Show Data Tail

This displays the bottom 10 rows of our dataset

[4]:
```
#displaying ending 10 rows of the data set
df_solar.tail(10)
```

[4]:
```
     organization_type  profession                type_of_home  \
1811           Private    Engineer                Flat(On Rent)
1812           Private    Engineer  Independent house or Villa
1813           Private    Engineer            Apartment or Flat
1814           Private    Engineer  Independent house or Villa
1815           Private    Engineer                Flat(On Rent)
1816           Private    Engineer  Independent house or Villa
1817           Private    Engineer                Flat(On Rent)
1818          Business     Student  Independent house or Villa
1819           Private  Consultant            Apartment or Flat
1820           Private    Engineer                Flat(On Rent)

      number_of_floors  rooms  build_additional_floor percentage_of_roof  \
1811                 5      3                      No              26-50%
1812                 2      4                      No              26-50%
1813                 2      4                      No              51-75%
1814                 2      4                      No              51-75%
1815                 2      4                      No              26-50%
1816                 1      5                      No              26-50%
1817                 1      2                      No                <25%
1818                 2      4                     Yes                <25%
1819                 5      3                      No              26-50%
1820                 5      5                      No                <25%

     home_loan  two_wheelers  four_wheelers  …  \
1811        No             0              0  …
1812       Yes             2              0  …
1813       Yes             1              0  …
1814       Yes             1              1  …
1815        No             1              0  …
1816        No             2              1  …
1817        No             2              2  …
```

```
1818        No             2             2  …
1819        No             2             1  …
1820        No             0             2  …

     concerns_on_outlook_of_the_house not_understanding_solar_systems  \
1811                                3                               3
1812                                3                               2
1813                                1                               2
1814                                1                               2
1815                                3                               3
1816                                1                               1
1817                                4                               2
1818                                1                               1
1819                                3                               3
1820                                1                               1

     not_knowing_good_solar_installer non_suitable_rooftop  \
1811                                3                    3
1812                                5                    3
1813                                3                    4
1814                                3                    3
1815                                1                    1
1816                                1                    1
1817                                3                    4
1818                                1                    4
1819                                3                    3
1820                                1                    4

      expectation_on_saving_electricity_bill  maximum_investment_in_solar  \
1811                                31% - 40%            Less than 50,000
1812                                21% - 30%            Less than 50,000
1813                                21% - 30%              1 Lac - 2 Lacs
1814                                41% - 50%            Less than 50,000
1815                                31% - 40%            Less than 50,000
1816                                41% - 50%              1 Lac - 2 Lacs
1817                                21% - 30%               50,000- 1 Lac
1818                                31% - 40%            Less than 50,000
1819                                31% - 40%            Less than 50,000
1820                                31% - 40%               50,000- 1 Lac

       solar_companies_you_know       city  state  \
1811                       None    Auraiya    NaN
1812                      Bosch    Belgaum    NaN
1813                  Jal Vayu  Bangalore    NaN
1814  Selco Solar Lights Pvt Lmt  Bangalore    NaN
1815   Selco Solar Light Pvt Ltd  Bangalore    NaN
1816                      Tesla     Nagpur    NaN
```

```
1817                         NaN   Varanasi    NaN
1818               Peacock Solar     Jaipur    NaN
1819                        Tata  Bangalore    NaN
1820                    Solarpwr  Bangalore    NaN

      know_any_solar_installation_companies
1811                                    No
1812                                    No
1813                                    No
1814                                    No
1815                                    No
1816                                   Yes
1817                                    No
1818                                   Yes
1819                                    No
1820                                    No

[10 rows x 38 columns]
```

Show Data Shape and Size

[5]:
```
#displaying the shape and size of the DataFrame
print('Shape: ', df_solar.shape)
print('Size : ', df_solar.size)
```

```
Shape:  (1821, 38)
Size :  69198
```

Show Data Column Names

[6]:
```
#displaying all the column names
df_solar.columns
```

[6]:
```
Index(['organization_type', 'profession', 'type_of_home', 'number_of_floors',
       'rooms ', 'build_additional_floor', 'percentage_of_roof', 'home_loan',
       'two_wheelers', 'four_wheelers', 'payment_method', 'active_emi',
       'average_monthly_income', 'power_backup_source', 'solar_awareness',
       'power_backup', 'increase_in_electricity_prices',
       'saving_on_the_electricity_bill', 'life_span_of_solar_system',
       'interest_in_new_technology', 'environment_friendly', 'status_symbol',
       'earning_from_empty_roof', 'investment_payback',
       'not_getting_sufficient_money_for_extra_electricity',
       'lack_of_government_incentives', 'lack_of_appropriate_loan_option',
       'high_cost_of_solar_power_systems', 'concerns_on_outlook_of_the_house',
       'not_understanding_solar_systems', 'not_knowing_good_solar_installer',
       'non_suitable_rooftop', 'expectation_on_saving_electricity_bill',
       'maximum_investment_in_solar', 'solar_companies_you_know', 'city',
       'state', 'know_any_solar_installation_companies'],
```

```
      dtype='object')
```

Show statistical analysis for our data

Let's display count, unique, top, frequency, mean, std, min, 25%, 50%, 75% and max of each column in the dataset.

[7]:
```
df_solar.describe(include = "all")
```

[7]:

	organization_type	profession	type_of_home	\
count	1809	1815	1820	
unique	112	295	49	
top	Private	Engineer	Independent house or Villa	
freq	853	659	1017	
mean	NaN	NaN	NaN	
std	NaN	NaN	NaN	
min	NaN	NaN	NaN	
25%	NaN	NaN	NaN	
50%	NaN	NaN	NaN	
75%	NaN	NaN	NaN	
max	NaN	NaN	NaN	

	number_of_floors	rooms	build_additional_floor	\
count	1821.000000	1821.000000	1821	
unique	NaN	NaN	27	
top	NaN	NaN	No	
freq	NaN	NaN	1465	
mean	2.455244	3.596925	NaN	
std	1.370731	1.162638	NaN	
min	1.000000	1.000000	NaN	
25%	1.000000	3.000000	NaN	
50%	2.000000	4.000000	NaN	
75%	3.000000	5.000000	NaN	
max	5.000000	5.000000	NaN	

	percentage_of_roof	home_loan	two_wheelers	four_wheelers	…	\
count	1821	1821	1821.000000	1821.000000	…	
unique	4	2	NaN	NaN	…	
top	26-50%	No	NaN	NaN	…	
freq	733	1451	NaN	NaN	…	
mean	NaN	NaN	1.427238	0.709500	…	
std	NaN	NaN	0.928075	0.832488	…	
min	NaN	NaN	0.000000	0.000000	…	
25%	NaN	NaN	1.000000	0.000000	…	
50%	NaN	NaN	1.000000	1.000000	…	
75%	NaN	NaN	2.000000	1.000000	…	
max	NaN	NaN	4.000000	4.000000	…	

	concerns_on_outlook_of_the_house	not_understanding_solar_systems	\
count	1821.000000	1821.000000	
unique	NaN	NaN	
top	NaN	NaN	
freq	NaN	NaN	
mean	2.689731	2.212521	
std	1.259622	1.240514	
min	1.000000	1.000000	
25%	2.000000	1.000000	
50%	3.000000	2.000000	
75%	4.000000	3.000000	
max	5.000000	5.000000	

	not_knowing_good_solar_installer	non_suitable_rooftop	\
count	1821.000000	1821.000000	
unique	NaN	NaN	
top	NaN	NaN	
freq	NaN	NaN	
mean	2.585942	2.517847	
std	1.276542	1.320729	
min	1.000000	1.000000	
25%	1.000000	1.000000	
50%	3.000000	2.000000	
75%	3.000000	3.000000	
max	5.000000	5.000000	

	expectation_on_saving_electricity_bill	maximum_investment_in_solar	\
count	1821	1821	
unique	5	4	
top	21% - 30%	Less than 50,000	
freq	565	1167	
mean	NaN	NaN	
std	NaN	NaN	
min	NaN	NaN	
25%	NaN	NaN	
50%	NaN	NaN	
75%	NaN	NaN	
max	NaN	NaN	

	solar_companies_you_know	city	state	\
count	1699	1821	0.0	
unique	580	220	NaN	
top	Peacock Solar	Bangalore	NaN	
freq	271	177	NaN	
mean	NaN	NaN	NaN	
std	NaN	NaN	NaN	
min	NaN	NaN	NaN	

```
25%                          NaN         NaN    NaN
50%                          NaN         NaN    NaN
75%                          NaN         NaN    NaN
max                          NaN         NaN    NaN

        know_any_solar_installation_companies
count                                    1821
unique                                      2
top                                        No
freq                                     1360
mean                                      NaN
std                                       NaN
min                                       NaN
25%                                       NaN
50%                                       NaN
75%                                       NaN
max                                       NaN

[11 rows x 38 columns]
```

Some interesting insights from above statistics

1. Most of the people work as Private Engineer
2. Most of the people own Independent house or Villa
3. 733 people agree to utilize 26% to 50% of their roof tops for solar panel installation
4. Most of the solar panel installations were in Bangalore i.e 177
5. 271 people already know Peacock Solar

Show Data Types

[8]:
```
df_solar.dtypes
```

[8]:
```
organization_type                    object
profession                           object
type_of_home                         object
number_of_floors                      int64
rooms                                 int64
build_additional_floor               object
percentage_of_roof                   object
home_loan                            object
two_wheelers                          int64
four_wheelers                         int64
payment_method                       object
active_emi                           object
average_monthly_income               object
power_backup_source                  object
solar_awareness                       int64
power_backup                          int64
```

```
increase_in_electricity_prices                          int64
saving_on_the_electricity_bill                          int64
life_span_of_solar_system                               int64
interest_in_new_technology                              int64
environment_friendly                                    int64
status_symbol                                           int64
earning_from_empty_roof                                 int64
investment_payback                                      int64
not_getting_sufficient_money_for_extra_electricity      int64
lack_of_government_incentives                           int64
lack_of_appropriate_loan_option                         int64
high_cost_of_solar_power_systems                        int64
concerns_on_outlook_of_the_house                        int64
not_understanding_solar_systems                         int64
not_knowing_good_solar_installer                        int64
non_suitable_rooftop                                    int64
expectation_on_saving_electricity_bill                 object
maximum_investment_in_solar                            object
solar_companies_you_know                               object
city                                                   object
state                                                 float64
know_any_solar_installation_companies                  object
dtype: object
```

Here we are displaying the total number of missing values in the columns

[9]:
```
print(df_solar.isnull().sum())
```

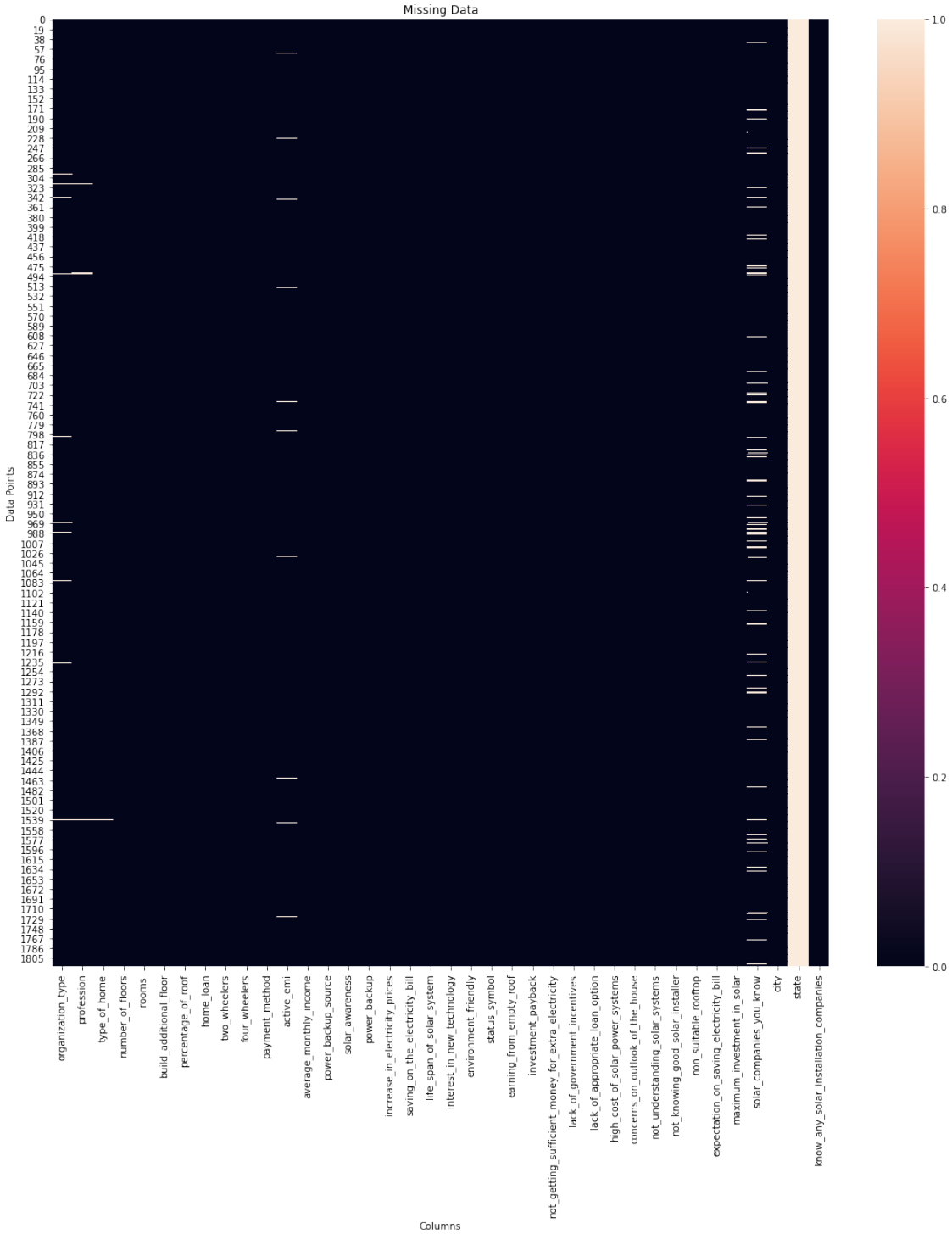
```
organization_type                                       12
profession                                               6
type_of_home                                             1
number_of_floors                                         0
rooms                                                    0
build_additional_floor                                   0
percentage_of_roof                                       0
home_loan                                                0
two_wheelers                                             0
four_wheelers                                            0
payment_method                                           0
active_emi                                              24
average_monthly_income                                   0
power_backup_source                                      0
solar_awareness                                          0
power_backup                                             0
increase_in_electricity_prices                           0
saving_on_the_electricity_bill                           0
life_span_of_solar_system                                0
interest_in_new_technology                               0
```

```
environment_friendly                                    0
status_symbol                                           0
earning_from_empty_roof                                 0
investment_payback                                      0
not_getting_sufficient_money_for_extra_electricity      0
lack_of_government_incentives                           0
lack_of_appropriate_loan_option                         0
high_cost_of_solar_power_systems                        0
concerns_on_outlook_of_the_house                        0
not_understanding_solar_systems                         0
not_knowing_good_solar_installer                        0
non_suitable_rooftop                                    0
expectation_on_saving_electricity_bill                  0
maximum_investment_in_solar                             0
solar_companies_you_know                              122
city                                                    0
state                                                1821
know_any_solar_installation_companies                   0
dtype: int64
```

Now we display a heatmap for the same missing values

[10]:
```
#displaying a heatmap for missing data in the dataset
fig, ax = plt.subplots(figsize=(18, 18))
sns.heatmap(df_solar.isnull()).set(title = 'Missing Data', xlabel = 'Columns',␣
 ↪ylabel = 'Data Points')
```

[10]:
```
[Text(141.0, 0.5, 'Data Points'),
 Text(0.5, 140.09375, 'Columns'),
 Text(0.5, 1.0, 'Missing Data')]
```



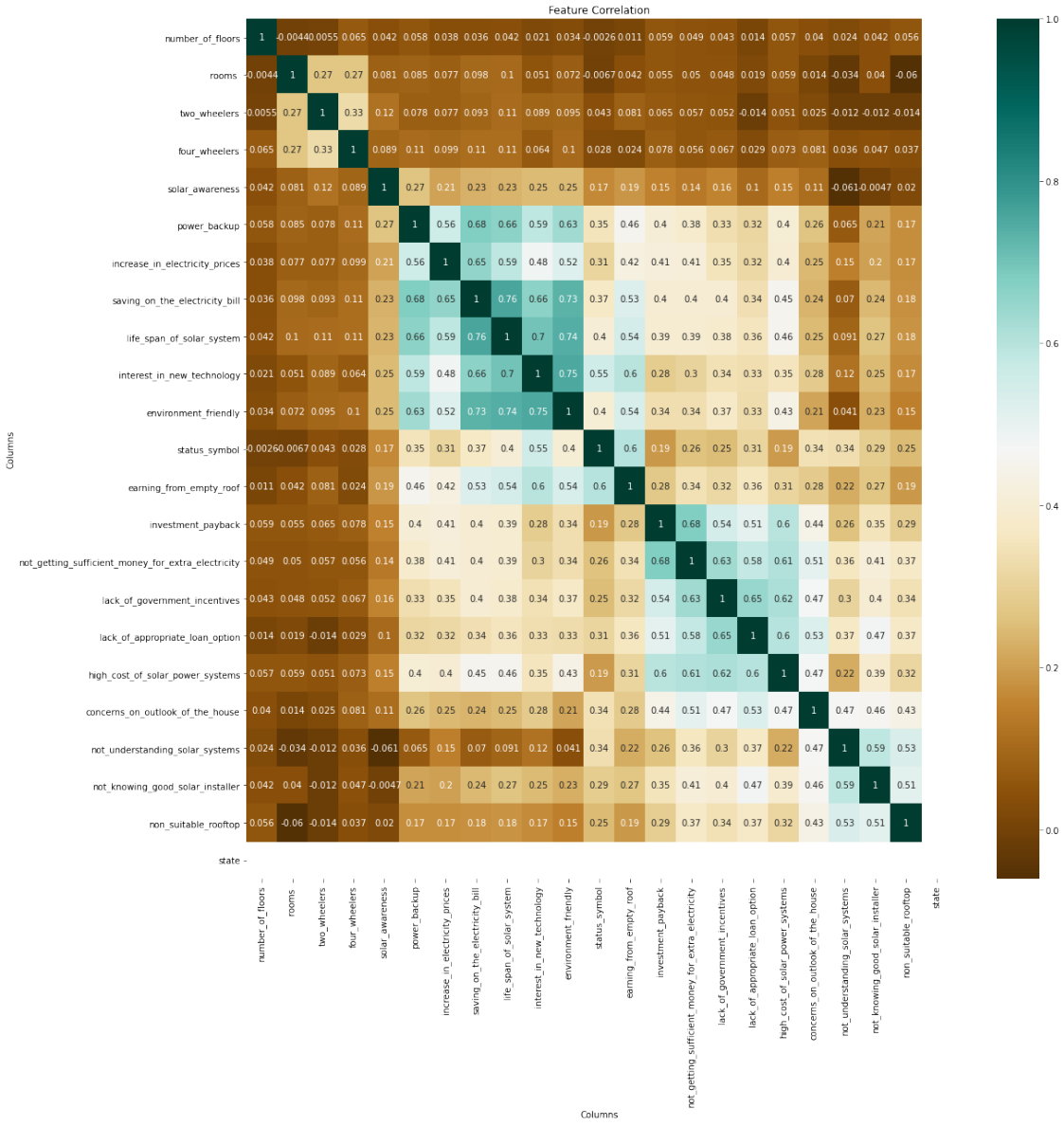
From the above heatmap we an see that we have very few columns with missing data points.

The columns **organization_type**, **average_monthly_income**, **solar_companies_you_know** and **state** have missing values These columns have some

missing values: 1. Organization Type 2. Average Monthly Income 3. So

Now we display a correlational heatmap

[11]:
```
#creating a correlation heap map
fig, ax = plt.subplots(figsize = (18, 18))
sns.heatmap(df_solar.corr(), annot = True, ax = ax, cmap = "BrBG").set(title =␣
↪"Feature Correlation",
                                                                       xlabel =␣
↪"Columns",
                                                                       ylabel =␣
↪"Columns")
#displaying the heatmap
plt.show()
```



Some insights from the above Heatmap (Attributes with correlation above 0.50)

There is relatively stronger correlation between these factors

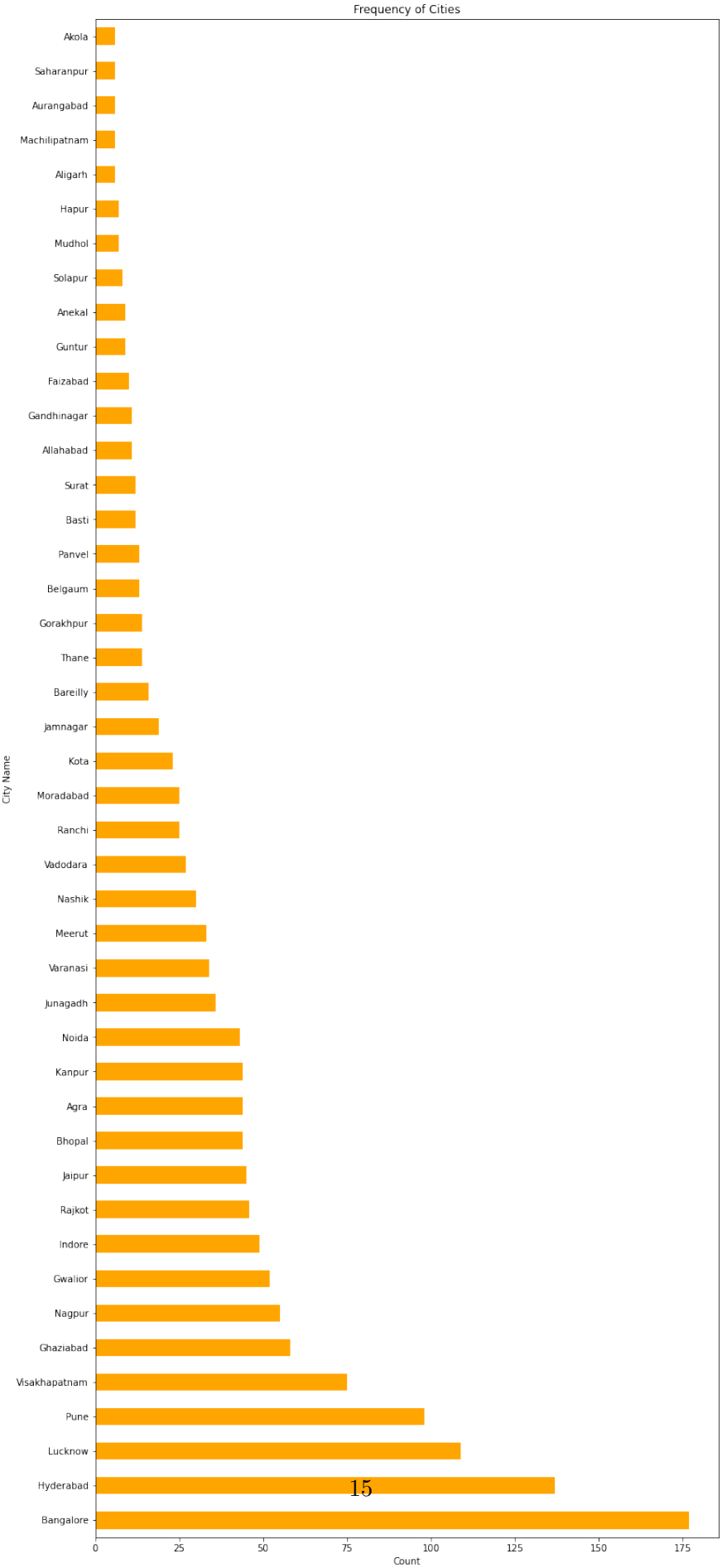
1. Expected correlation between **saving__on__electricity__bill** and **lifespan__of__solar__system**
2. Expected correlation between **interest__in__new__technology** and **environment__friendly**
3. Expected correlation between **lifespan__of__solar__system** and **interest__in__new__technology**
4. Unxpected correlation between **saving__on__electricity__bill** and **envionment__friendly**
5. Unxpected correlation between **lifespan__of__solar__system** and **environment__friendly**

Counting number of projects in individual cities

[12]:
```
df_solar['city'].value_counts()
```

[12]:
```
Bangalore        177
Hyderabad        137
Lucknow          109
Pune              98
Visakhapatnam     75
                 ...
Kodinar            1
Pen                1
Hathras            1
Bikaner            1
Tenali             1
Name: city, Length: 220, dtype: int64
```

[13]:
```
prob = df_solar['city'].value_counts(normalize = False)
threshold = 5
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'barh', color = "orange", figsize = (12, 30))
plt.title('Frequency of Cities')
plt.ylabel('City Name')
plt.xlabel('Count')
plt.xticks(rotation = 0)
plt.show()
```


Frequency of Cities
Akola
Saharanpur
Aurangabad
Machilipatnam
Aligarh
Hapur
Mudhol
Solapur
Anekal
Guntur
Faizabad
Gandhinagar
Allahabad
Surat
Basti
Panvel
Belgaum
Gorakhpur
Thane
Bareilly
Jamnagar
Kota
Moradabad
Ranchi
Vadodara
Nashik
Meerut
Varanasi
Junagadh
Noida
Kanpur
Agra
Bhopal
Jaipur
Rajkot
Indore
Gwalior
Nagpur
Ghaziabad
Visakhapatnam
Pune
Lucknow
Hyderabad
Bangalore
City Name
0
25
50
75
100
125
150
175
Count

Now we have some factors that might affect purchase decisions of the customers

We have two types of factors, 1. Encouraging factors (These are the factors which encourage the customers to install solar plants at their homes)

a. solar_awareness
b. power_backup
c. increase_in_electricity_prices
d. saving_on_the_electricity_bill
e. life_span_of_solar_system
f. interest_in_new_technology
g. environment_friendly
h. status_symbol
i. earning_from_empty_roof

The values of each factor lies on a scale of 1 to 5 where, 1 means least important and 5 means most important

Example: If someone has 5 in their solar_awareness, it means they are aware about solar technology and its uses.

It means the values closer to 5 are important for us

2. Discouraging factors (These are the factors which discourage the customers to install solar

a. investment_payback
b. not_getting_sufficient_money_for_extra_electricity
c. lack_of_government_incentives
d. lack_of_appropriate_loan_option
e. high_cost_of_solar_power_systems
f. concerns_on_outlook_of_the_house
g. not_understanding_solar_systems
h. not_knowing_good_solar_installer
i. non_suitable_rooftop

The values of each factor lies on a scale of 1 to 5 where, 1 means least important and 5 means most important

Example: If someone has 5 in their investment_payback, it means they are least likely to invest in solar power plant and they are not sure if the solar plant will repay a fair amount.

It means the values closer to 1 are important for us

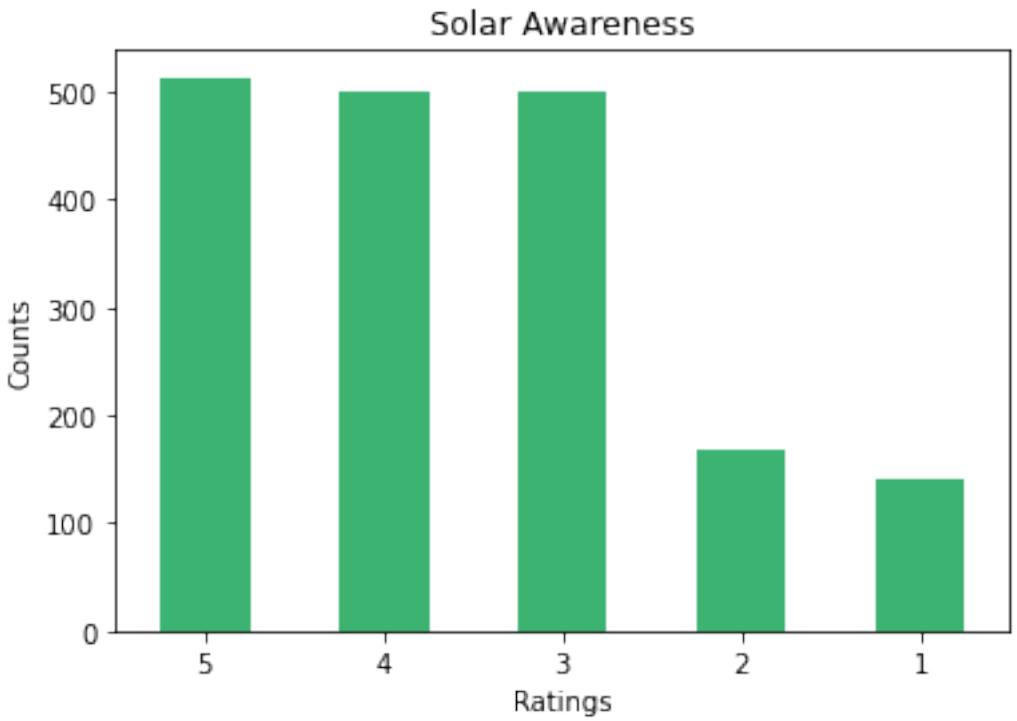
Now let us count and plot values for some of the encouraging factors

The values which are closer to 5 are better as compared to the values which are closer to 1

[14]:
```
#counting the counts of distinct values in solar_awareness column
df_solar['solar_awareness'].value_counts()
```

[14]:
```
5    513
4    499
3    499
2    169
1    141
Name: solar_awareness, dtype: int64
```

[15]:
```
#barplot to visualize the value counts of distinct values in colar_awareness␣
↪column
prob = df_solar['solar_awareness'].value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Solar Awareness')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

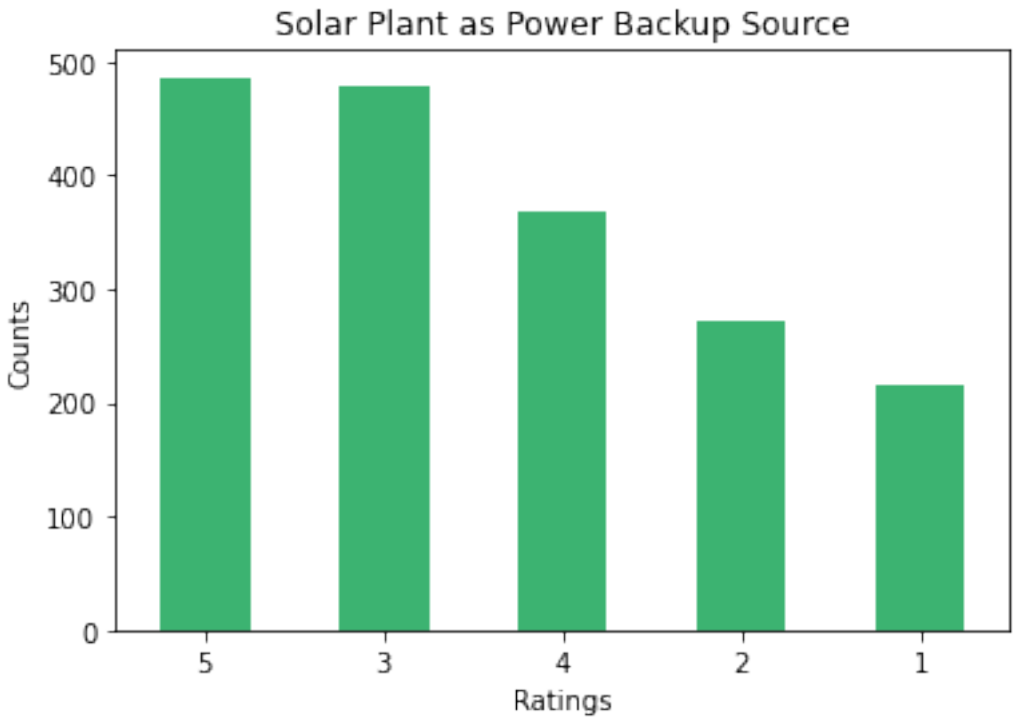
Most people have rated 5, 4, and 3 for *Solar Awareness* which means that they are quite aware of Solar Technology.

---

[16]:
```
df_solar['power_backup'].value_counts()
```

[16]:
```
5    486
3    479
4    368
2    272
1    216
Name: power_backup, dtype: int64
```

[17]:
```
#barplot to visualize the value counts of distinct values in power_backup column
prob = df_solar['power_backup'].value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Solar Plant as Power Backup Source')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

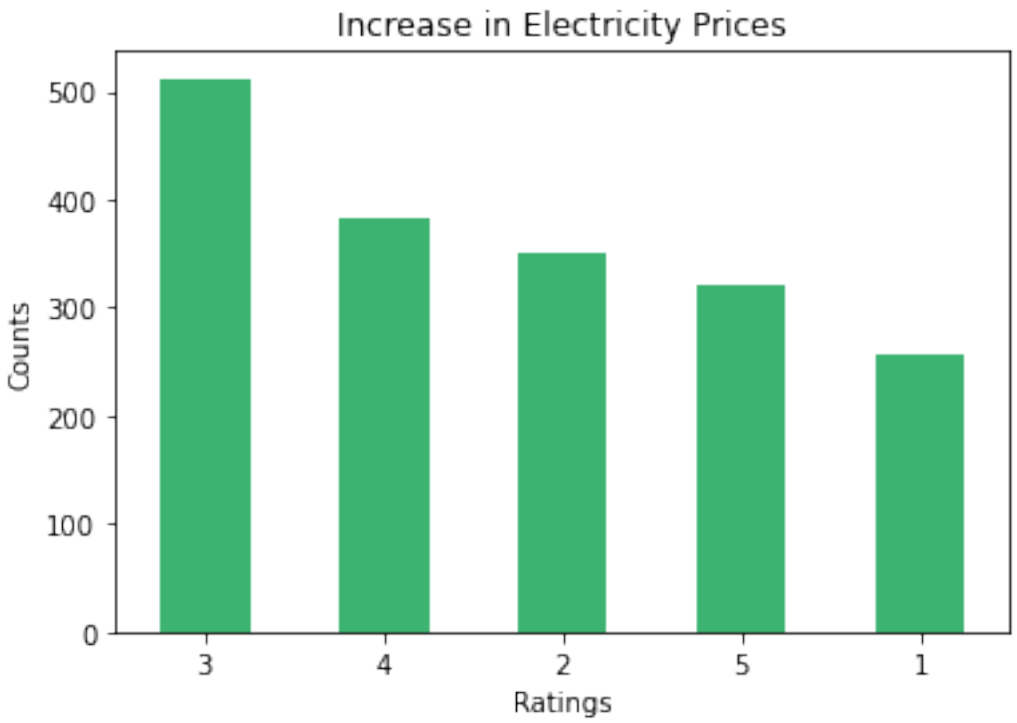
most people have rated 5, 3, and 4 for *Solar Plant as a Power Backup Source* which means people consider Solar Technology as a reliable source for power backup.

---

[18]:
```
df_solar['increase_in_electricity_prices'].value_counts()
```

[18]:
```
3    512
4    382
2    350
5    321
1    256
Name: increase_in_electricity_prices, dtype: int64
```

[19]:
```
#barplot to visualize the value counts of distinct values in␣
↪increase_in_electricity_prices column
prob = df_solar['increase_in_electricity_prices'].value_counts(normalize =␣
↪False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Increase in Electricity Prices')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

Most people 3, 4, and 2 for *Increase in Electricity Prices* which means people would invest in Solar tEchnology if there is an increase in electricity prices.

---

[20]:
```
df_solar['saving_on_the_electricity_bill'].value_counts()
```

[20]:
```
3    529
5    449
4    428
2    251
1    164
Name: saving_on_the_electricity_bill, dtype: int64
```

[21]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪saving_on_the_electricity_bill column
prob = df_solar['saving_on_the_electricity_bill'].value_counts(normalize =␣
 ↪False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
```

```
plt.ylabel('Counts')
plt.title('Saving on Electricity Bill')
plt.xticks(rotation = 0)
plt.show()
```

From the above barplot, we can see that

Most people have rated 3, 5, and 4 for *Saving on Electricity Bill* which means people would invest in Solar Technology for saving on electricity bill.

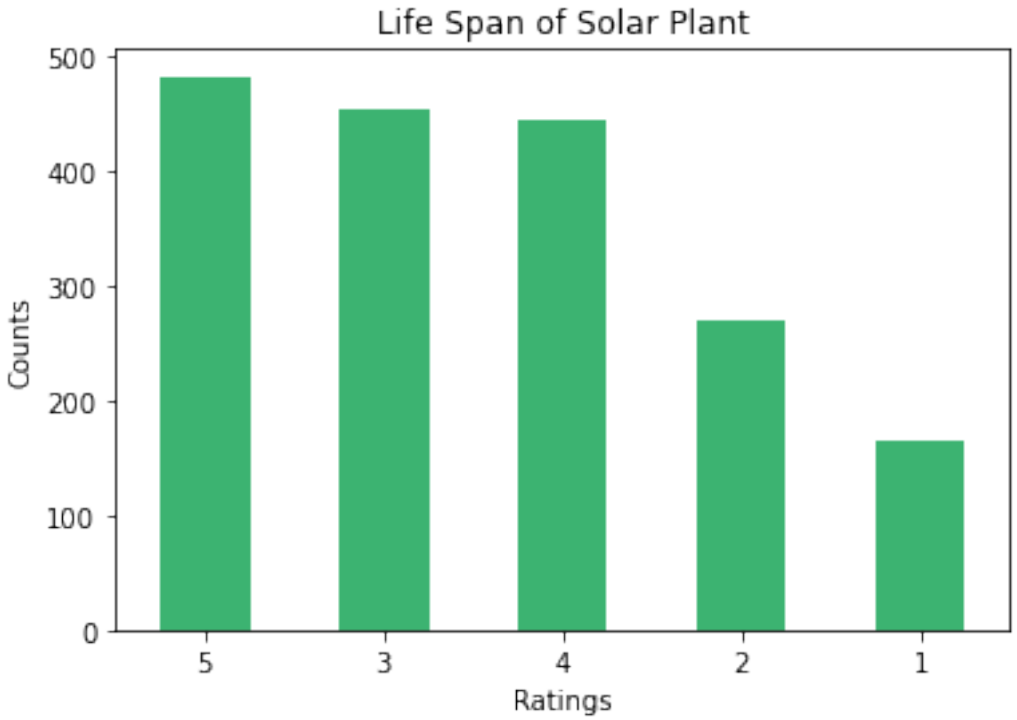
---

[22]:
```
df_solar['life_span_of_solar_system'].value_counts()
```

[22]:
```
5    483
3    455
4    446
2    271
1    166
Name: life_span_of_solar_system, dtype: int64
```

[23]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪life_span_of_solar_system column
prob = df_solar['life_span_of_solar_system'].value_counts(normalize = False)
threshold = 0.02
```

```
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Life Span of Solar Plant')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

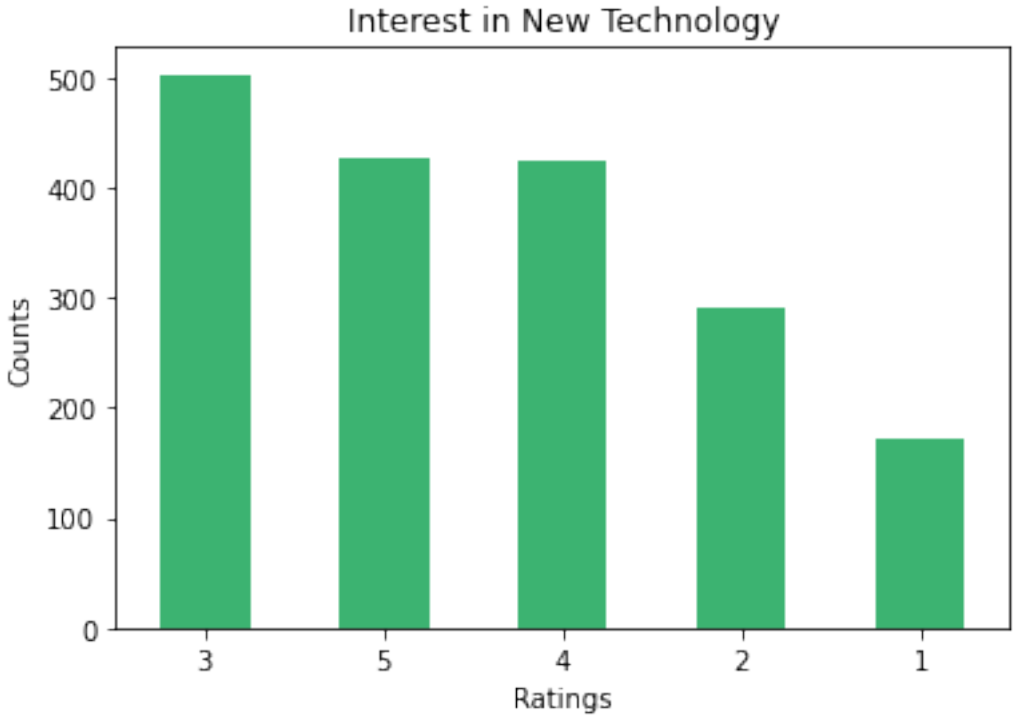
Most people have rated 5, 3, and 4 for *Life Span of Solar Plant* which means people will consider investing in Solar Technology if it has a longer life span.

---

[24]:
```
df_solar['interest_in_new_technology'].value_counts()
```

[24]:
```
3    503
5    428
4    425
2    292
1    173
Name: interest_in_new_technology, dtype: int64
```

[25]:
```
#barplot to visualize the value counts of distinct values in␣
↪interest_in_new_technology column
prob = df_solar['interest_in_new_technology'].value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Interest in New Technology')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

Most people have rated 3, 5, and 4 for *Interest n New Technology* which means they might invest in Solar technology if they are interested enough in it.

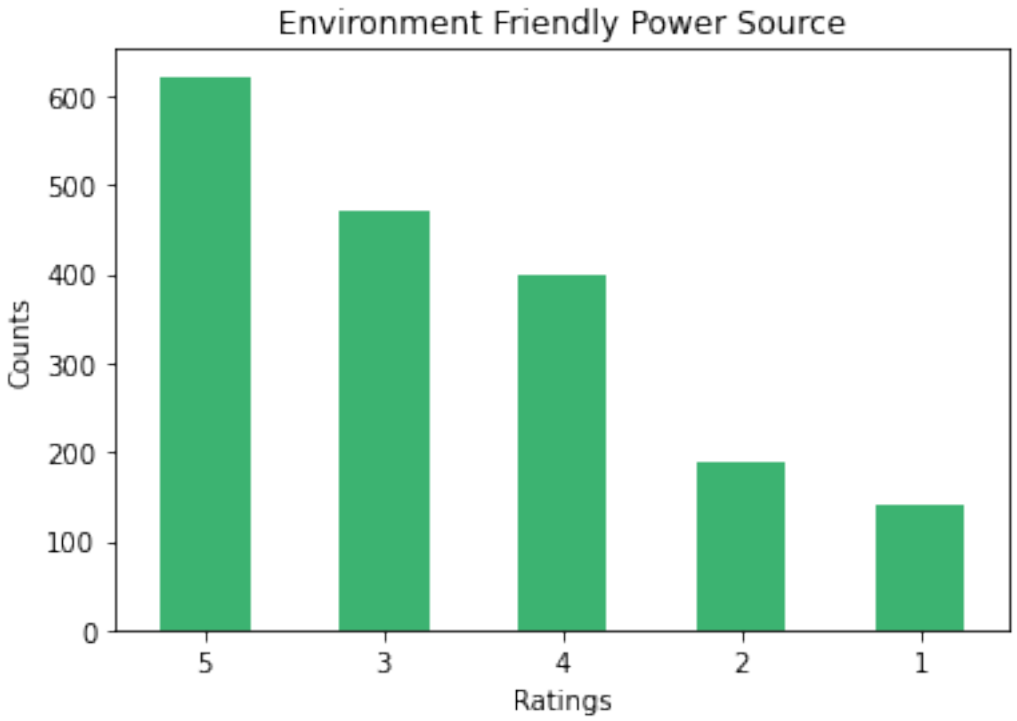
---

[26]:
```
df_solar['environment_friendly'].value_counts()
```

[26]:
```
5    622
3    471
4    399
```

```
2    188
1    141
Name: environment_friendly, dtype: int64
```

[27]:
```
#barplot to visualize the value counts of distinct values in␣
↪environment_friendly column
prob = df_solar['environment_friendly'].value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Environment Friendly Power Source')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

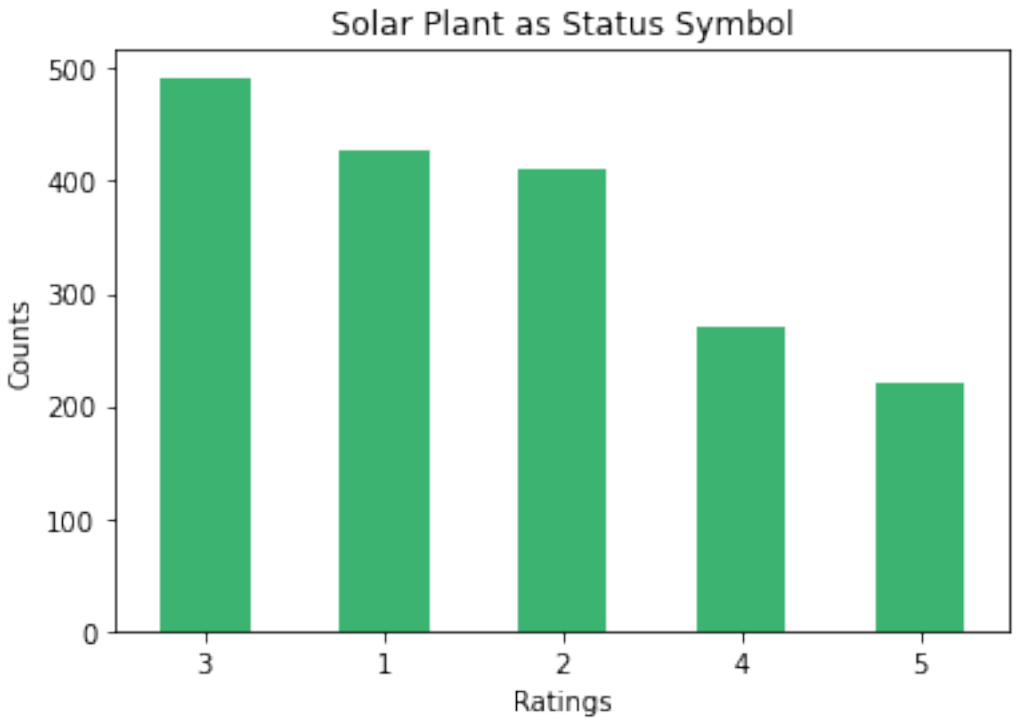
Most people have rated 5, 3, and 4 for *Solar Plant as an Environment Friendly Power Source* which means people tend to invest in Renewable and Environment Friendly Power Source

---

[28]:
```
df_solar['status_symbol'].value_counts()
```

[28]:
```
3    492
1    428
2    411
4    270
5    220
Name: status_symbol, dtype: int64
```

[29]:
```
#barplot to visualize the value counts of distinct values in status_symbol␣
↪column
prob = df_solar['status_symbol'].value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Solar Plant as Status Symbol')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

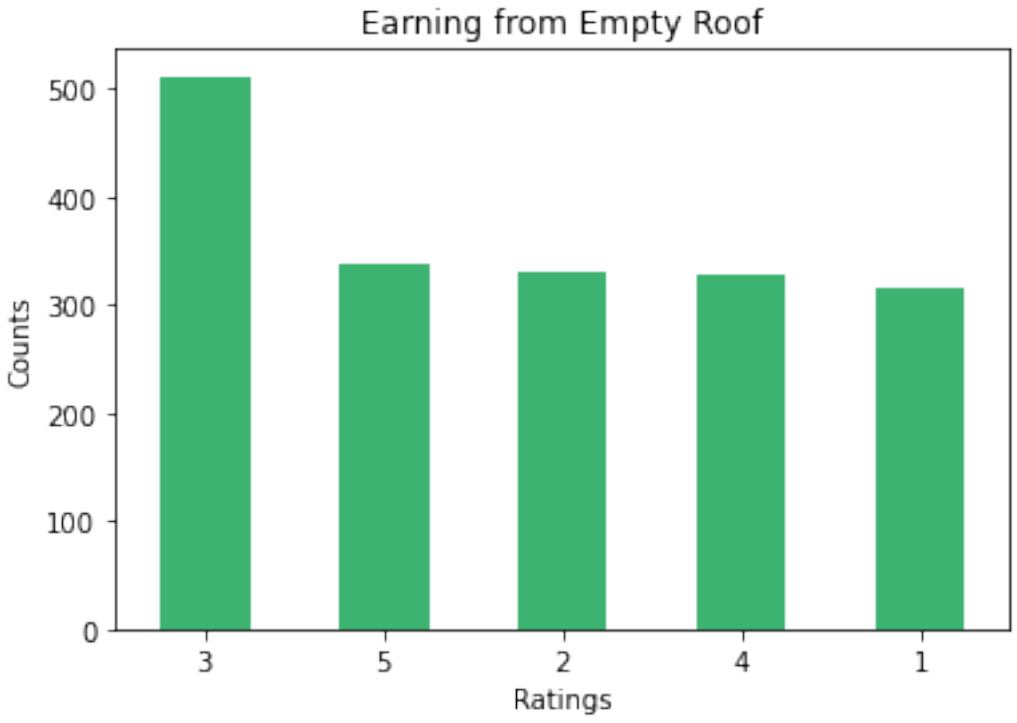
Most people have rated 3, 1, and 2 for *Solar Plant as Status Symbol* which means people do not tend to invest in Solar technology for it being a mere status symbol.

---

[30]:
```
df_solar['earning_from_empty_roof'].value_counts()
```

[30]:
```
3    511
5    338
2    329
4    328
1    315
Name: earning_from_empty_roof, dtype: int64
```

[31]:
```
#barplot to visualize the value counts of distinct values in␣
↪earning_from_empty_roof column
prob = df_solar['earning_from_empty_roof'].value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Earning from Empty Roof')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

Most people have rated 3, 5, and 2 for *Earning From Empty Rooftop* which means that customers consider this factor important while investing in solar technology.

Now let us count and plot values for some of the discouraging factors

The values which are closer to 1 are better as compared to the values which are closer to 5

[32]:
```
df_solar['investment_payback'].value_counts()
```

[32]:
```
3    544
1    339
2    329
5    305
4    304
Name: investment_payback, dtype: int64
```

[33]:
```
#barplot to visualize the value counts of distinct values in investment_payback␣
↪column
prob = df_solar['investment_payback'].value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Investment Payback')
plt.xticks(rotation = 0)
plt.show()
```

From the above bar plot, we can see that

Most people have rated 3, 1, and 2 for *Investment Payback* which means this factor doesnot have a major impact on customer's purchase decision

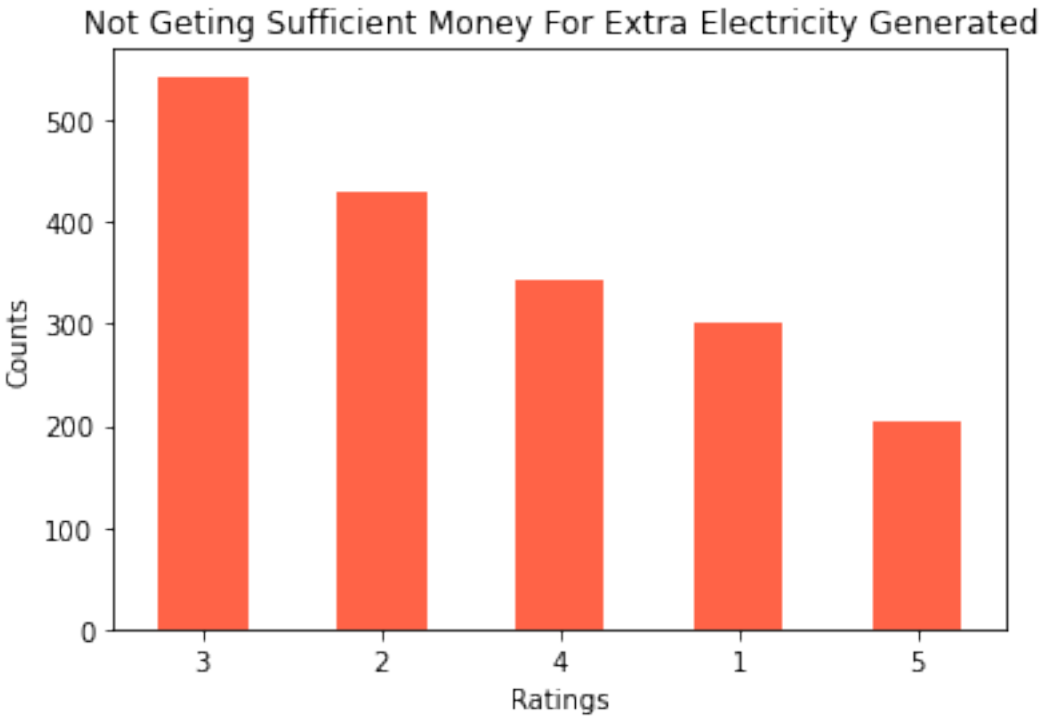
---

[34]:
```
df_solar['not_getting_sufficient_money_for_extra_electricity'].value_counts()
```

[34]:
```
3    543
2    430
4    342
1    302
5    204
Name: not_getting_sufficient_money_for_extra_electricity, dtype: int64
```

[35]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪not_getting_sufficient_money_for_extra_electricity column
prob = df_solar['not_getting_sufficient_money_for_extra_electricity'].
 ↪value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
```

```
plt.ylabel('Counts')
plt.title('Not Geting Sufficient Money For Extra Electricity Generated')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

Most people have rated 3, 2, and 4 for *Not getting Money For Extra Electricity Generated* which means this factor might have an impact on customer's purchase decision

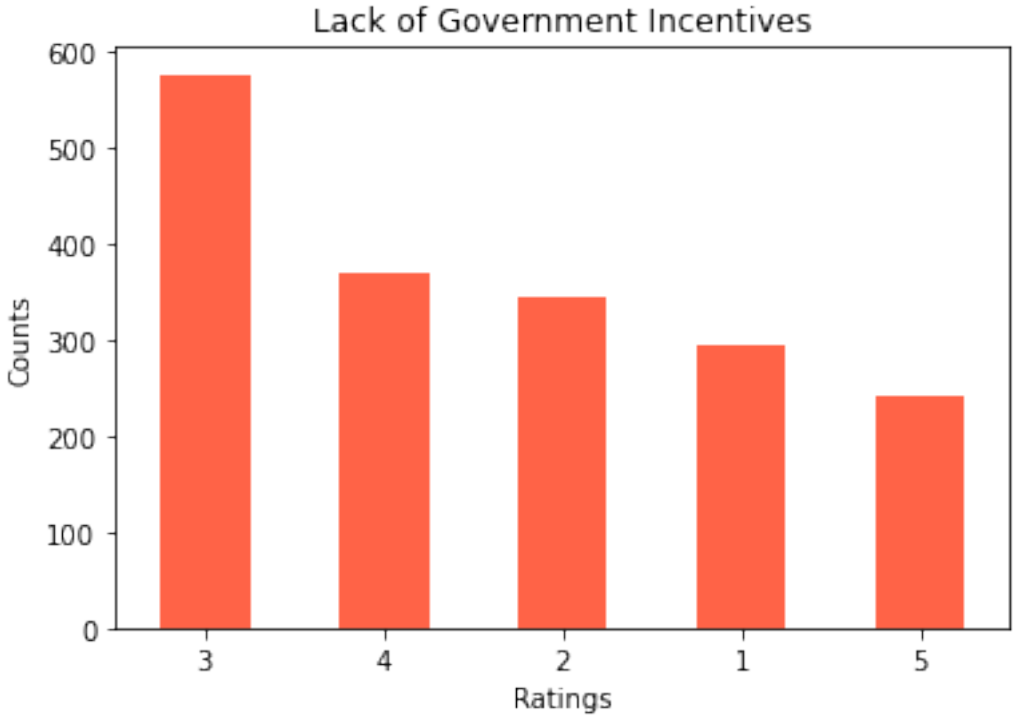
---

[36]:
```
df_solar['lack_of_government_incentives'].value_counts()
```

[36]:
```
3    575
4    370
2    343
1    293
5    240
Name: lack_of_government_incentives, dtype: int64
```

[37]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪lack_of_government_incentives column
prob = df_solar['lack_of_government_incentives'].value_counts(normalize = False)
```

```
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Lack of Government Incentives')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

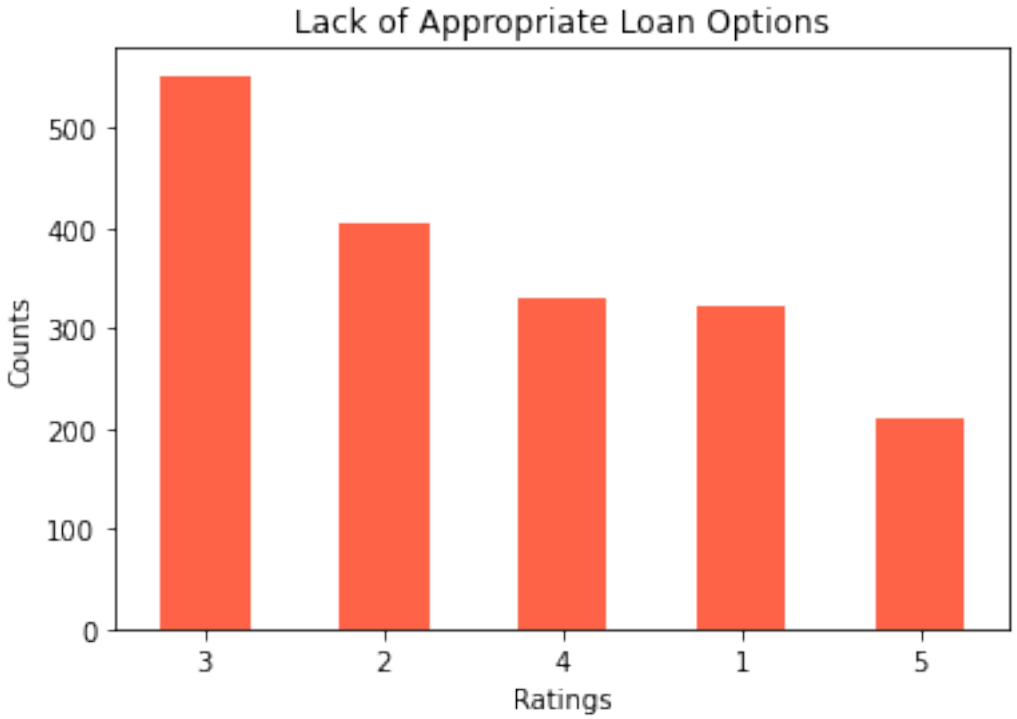
Most people have rated 3, 4, and 2 for *Lack of government incentives* which means most people are not affected by givernment incentives for renewable energy

---

[38]:
```
df_solar['lack_of_appropriate_loan_option'].value_counts()
```

[38]:
```
3    552
2    405
4    331
1    323
5    210
Name: lack_of_appropriate_loan_option, dtype: int64
```

[39]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪lack_of_appropriate_loan_option column
prob = df_solar['lack_of_appropriate_loan_option'].value_counts(normalize =␣
 ↪False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Lack of Appropriate Loan Options')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

Most people have rated 3, 2, and 4 for *Lack of Appropriate Loan Options* which means this factor doesnot have a major impact on customer's purchase decision

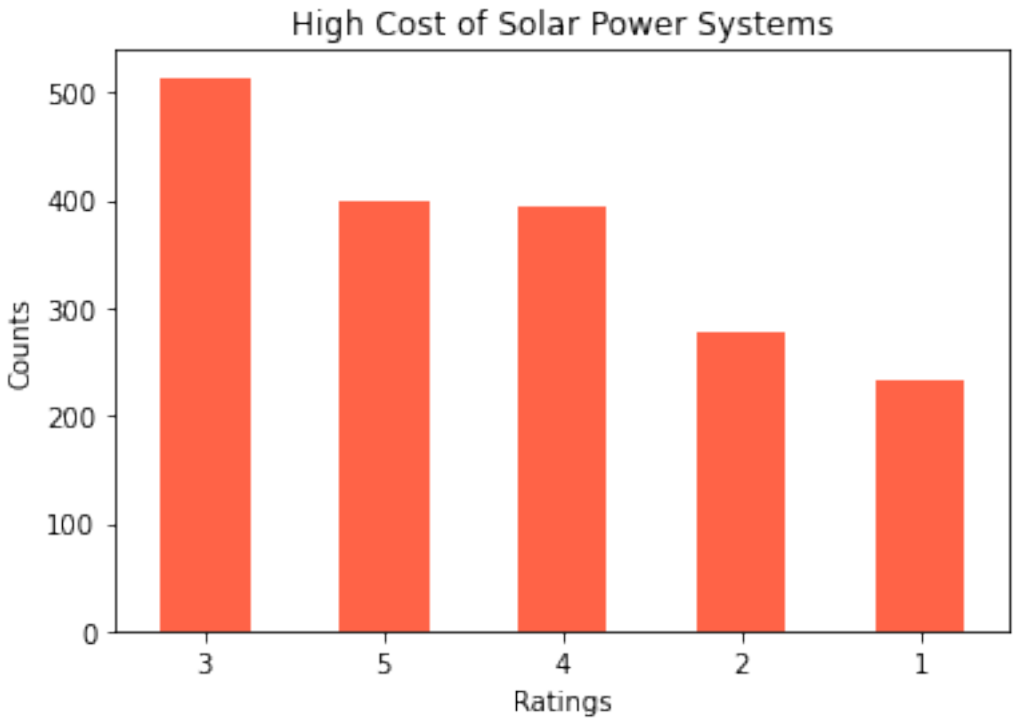
---

[40]:
```
df_solar['high_cost_of_solar_power_systems'].value_counts()
```

[40]:
```
3    514
5    400
```

```
4    395
2    278
1    234
Name: high_cost_of_solar_power_systems, dtype: int64
```

[41]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪high_cost_of_solar_power_systems column
prob = df_solar['high_cost_of_solar_power_systems'].value_counts(normalize =␣
 ↪False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('High Cost of Solar Power Systems')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

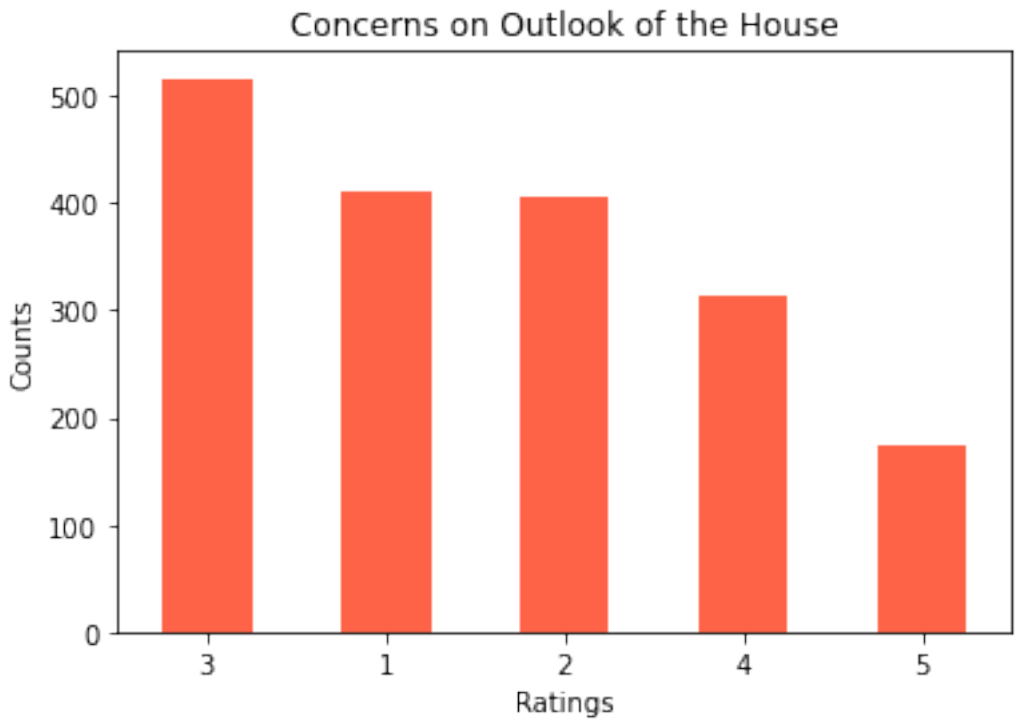
Most people have rated 3, 5, and 4 for *High Cost of Solar Power Plants* which means this factor can have a major impact on customer's purchase decision

---

[42]:
```
df_solar['concerns_on_outlook_of_the_house'].value_counts()
```

[42]:
```
3    516
1    411
2    406
4    313
5    175
Name: concerns_on_outlook_of_the_house, dtype: int64
```

[43]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪concerns_on_outlook_of_the_house column
prob = df_solar['concerns_on_outlook_of_the_house'].value_counts(normalize =␣
 ↪False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Concerns on Outlook of the House')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

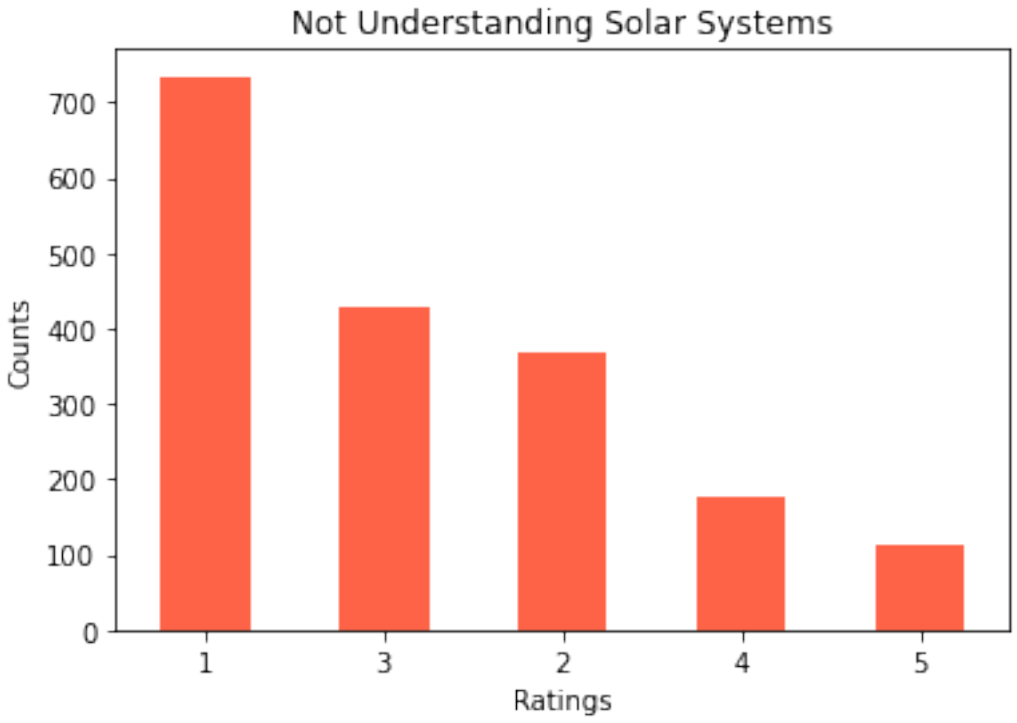
Most people have rated 3, 1, and 2 for *Concerns on Outlook of the House* which means this factor doesnot have a major impact on customer's purchase decision

---

[44]:
```
df_solar['not_understanding_solar_systems'].value_counts()
```

[44]:
```
1    734
3    429
2    368
4    178
5    112
Name: not_understanding_solar_systems, dtype: int64
```

[45]:
```
#barplot to visualize the value counts of distinct values in␣
↪not_understanding_solar_systems column
prob = df_solar['not_understanding_solar_systems'].value_counts(normalize =␣
↪False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Not Understanding Solar Systems')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

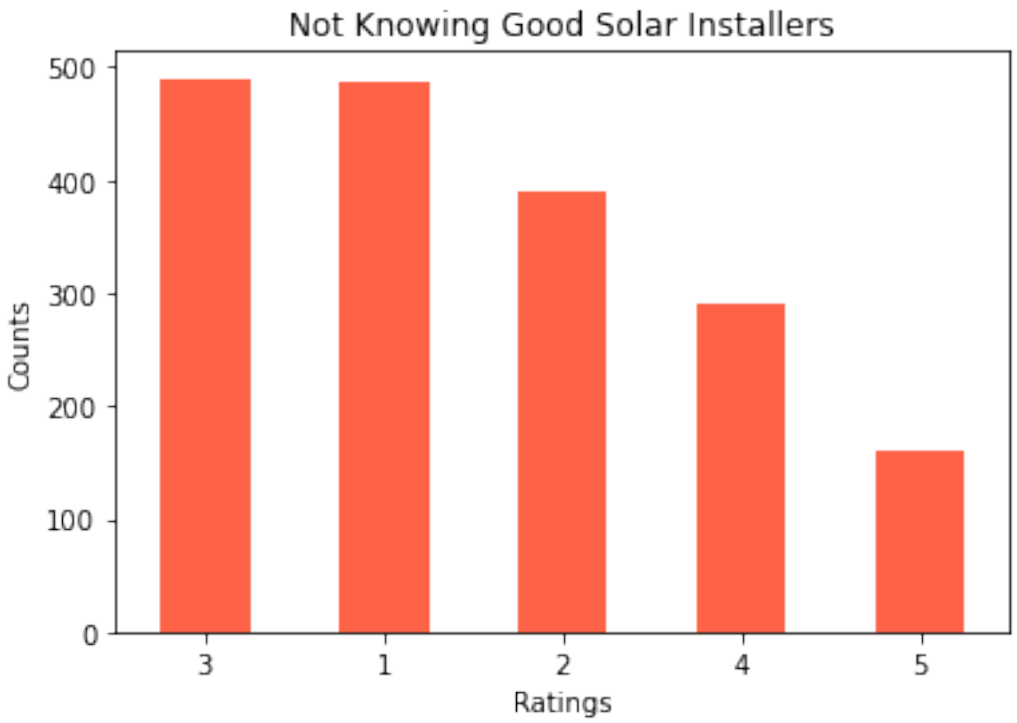
Most people have rated 1, 3, and 2 for *Not Understanding Solar Systems* which means this factor doesnot have a major impact on customer's purchase decision

---

[46]:
```
df_solar['not_knowing_good_solar_installer'].value_counts()
```

[46]:
```
3    490
1    488
2    391
4    291
5    161
Name: not_knowing_good_solar_installer, dtype: int64
```

[47]:
```
#barplot to visualize the value counts of distinct values in␣
↪not_knowing_good_solar_installer column
prob = df_solar['not_knowing_good_solar_installer'].value_counts(normalize =␣
↪False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Not Knowing Good Solar Installers')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

Most people have rated 3, 1, and 2 for *Not Knowing Good Solar Installers* which means this factor doesnot have a major impact on customer's purchase decision

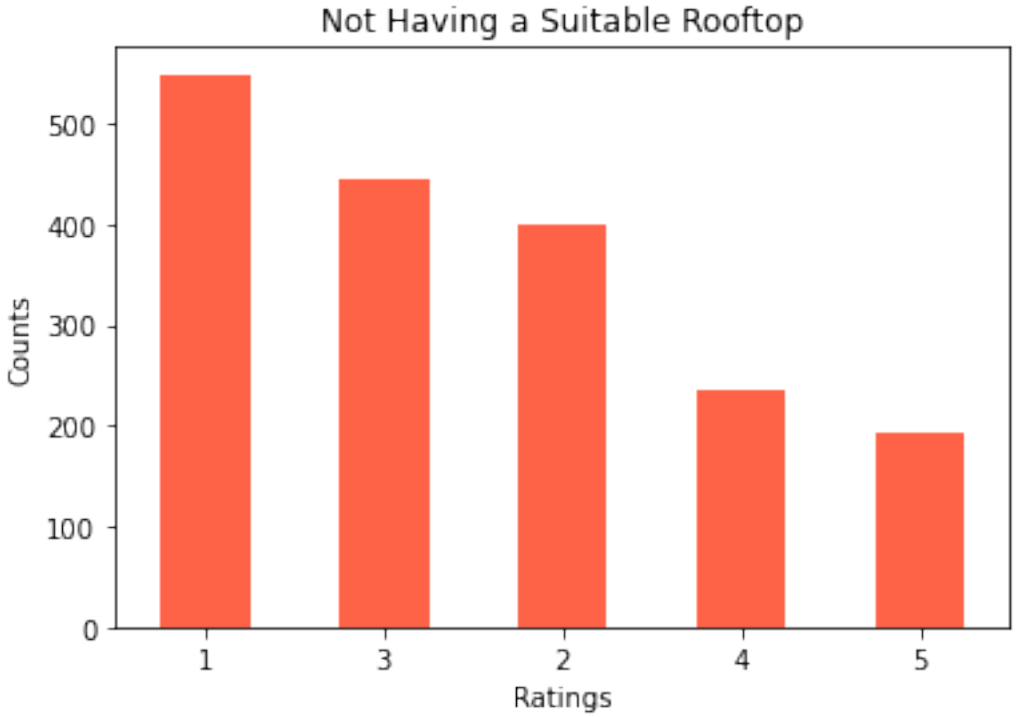
---

[48]:
```
df_solar['non_suitable_rooftop'].value_counts()
```

[48]:
```
1    549
3    446
2    399
4    235
5    192
Name: non_suitable_rooftop, dtype: int64
```

[49]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪non_suitable_rooftop column
prob = df_solar['non_suitable_rooftop'].value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
plt.ylabel('Counts')
```

```
plt.title('Not Having a Suitable Rooftop')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

Most people have rated 1, 3, and 2 for *Not Having a Suitable Rooftop* which means this factor doesnot have a major impact on customer's purchase decision

---

Now let us analyze Type of House that the customers own

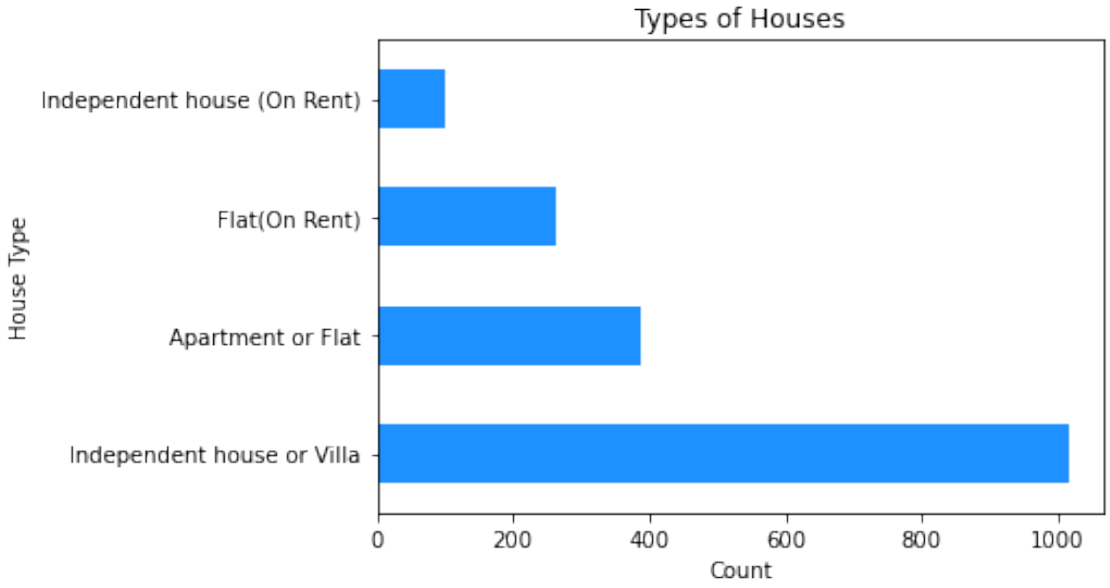
[50]:
```
df_solar['type_of_home'].value_counts()
```

[50]:
```
Independent house or Villa          1017
Apartment or Flat                    388
Flat(On Rent)                        262
Independent house (On Rent)           99
None                                   5
House                                  3
Company provided                       2
Student                                2
PG                                     2
No house                               1
```

```
Job                                                 1
Company quarter                                     1
At my mom's house                                   1
house                                               1
Study                                               1
Hostelite                                           1
Nothinh                                             1
Room on rent                                        1
Paying guest                                        1
Hostel                                              1
Hostel life                                         1
Floor                                               1
With parents.                                       1
ORGANISATION  PROVIDED QUARTER.                     1
Dont own any                                        1
Rent house                                          1
company quarters                                    1
Independent house,Under construction                1
Quartersc                                           1
No own house                                        1
Quarter                                             1
Government accomodation                             1
Village                                             1
Company quarters                                    1
Nothing i own                                       1
Hut                                                 1
Calling house                                       1
Paying guest                                        1
Company Quarter                                     1
My parents house                                    1
Housewife                                           1
Company House                                       1
Company Quarters                                    1
House                                               1
COMPANY PROVIDED QUARTER                            1
3floors building                                    1
Father's hause                                      1
Owns nothing                                        1
Row house                                           1
Name: type_of_home, dtype: int64
```

[51]:
```
prob = df_solar['type_of_home'].value_counts(normalize = False)
threshold = 5
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'barh', color = "dodgerblue")
plt.title('Types of Houses')
```

```
plt.ylabel('House Type')
plt.xlabel('Count')
plt.xticks(rotation = 0)
plt.show()
```



We can see that maximum people own an Independent house or Villa

---

Now we will sort out the cities with considerable number of projects for expanding business.

[52]:
```
#creating a list of cities with sales more than 10
cities = ['Bangalore', 'Hyderabad', 'Lucknow', 'Pune', 'Visakhapatnam',␣
 ↪'Ghaziabad', 'Nagpur',
          'Gwalior', 'Indore', 'Rajkot', 'Jaipur', 'Kanpur', 'Agra', 'Bhopal',␣
 ↪'Noida', 'Junagadh',
          'Varanasi', 'Meerut', 'Nashik', 'Vadodara', 'Moradabad', 'Ranchi',␣
 ↪'Kota', 'Jamnagar',
          'Bareilly', 'Thane', 'Gorakhpur', 'Belgaum', 'Panvel', 'Basti',␣
 ↪'Surat']

#creating a list of houses which are suitable for installing a solar plant
houses = ['Independent house or Villa', 'Apartment or Flat', 'Flat(On Rent)',␣
 ↪'Independent house (On Rent)']

#creating a new dataframe with above conditions
df_sorted_city = df_solar[df_solar.city.isin(cities) & df_solar.type_of_home.
 ↪isin(houses)]
```

```
#displaying the total row counts of new dataframe
print("Total rows count: ", len(df_sorted_city.index))
```

```
Total rows count:  1378
```

[53]:
```
df_sorted_city.head(5)
```

[53]:
```
  organization_type                      profession  \
0          Business                       Marketing
2        Government                  REVENUE TALATI
3           Private  Working as an sales executive
4        Government                       Professor
5          Business                     Shop keeper

                type_of_home  number_of_floors  rooms   \
0  Independent house or Villa                 3       3
2  Independent house or Villa                 2       3
3           Apartment or Flat                 1       4
4  Independent house or Villa                 1       5
5  Independent house or Villa                 2       4

  build_additional_floor percentage_of_roof home_loan  two_wheelers  \
0                     No               <25%        No             2
2                     No               <25%        No             2
3                     No               <25%       Yes             1
4                     No             26-50%        No             0
5                     No             26-50%        No             0

   four_wheelers  … concerns_on_outlook_of_the_house  \
0              1  …                                 1
2              1  …                                 1
3              2  …                                 4
4              0  …                                 3
5              1  …                                 1

  not_understanding_solar_systems not_knowing_good_solar_installer  \
0                               1                                1
2                               1                                1
3                               2                                2
4                               2                                3
5                               2                                3

  non_suitable_rooftop  expectation_on_saving_electricity_bill  \
0                    1                              10% - 20%
2                    1                              41% - 50%
3                    4                              21% - 30%
```

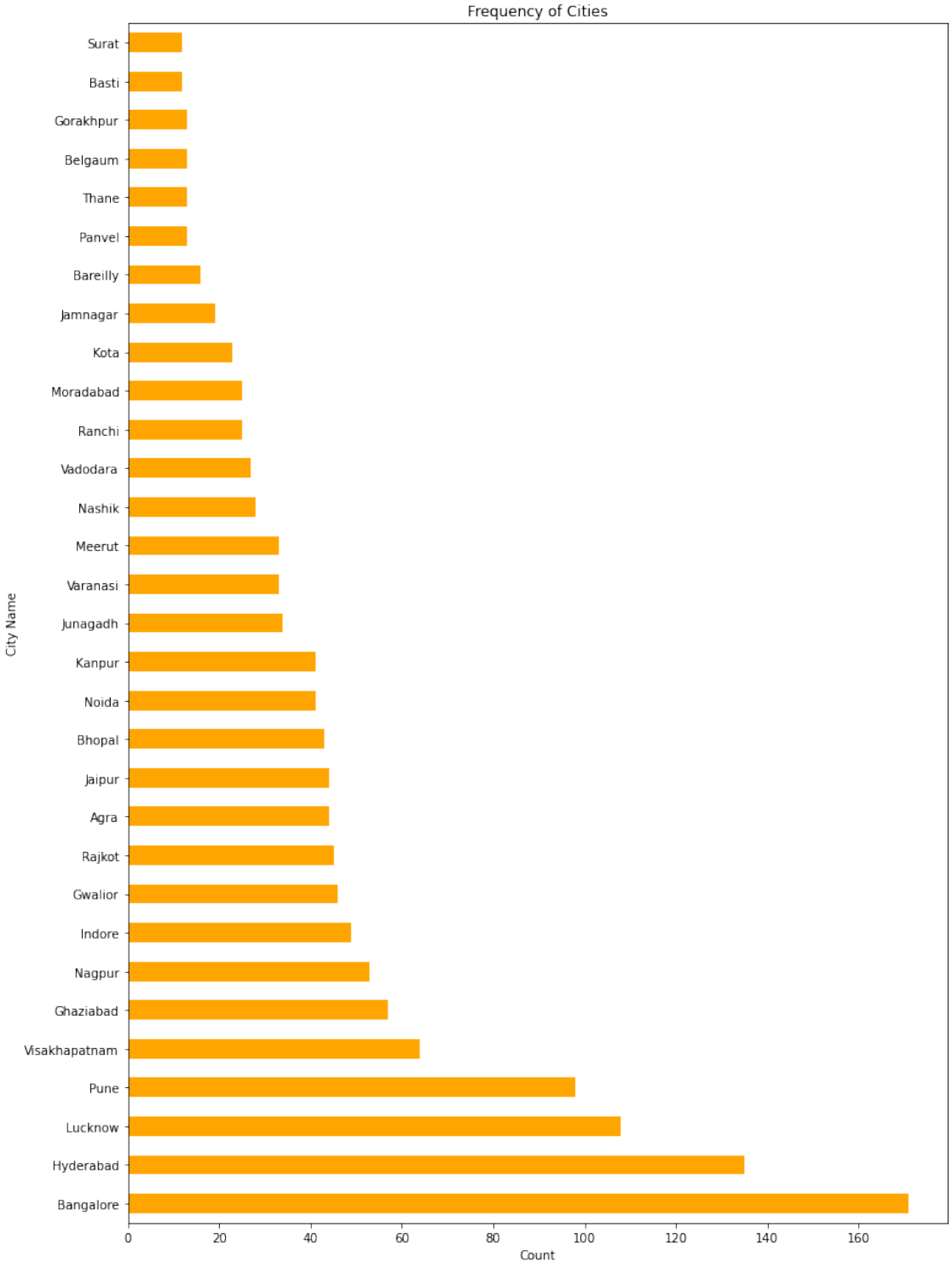
```
4                    2                              31% - 40%
5                    1                              21% - 30%

   maximum_investment_in_solar  solar_companies_you_know       city  state  \
0             Less than 50,000                  Namaste  Hyderabad    NaN
2             Less than 50,000                    Surya     Rajkot    NaN
3             Less than 50,000                Kirloskar     Panvel    NaN
4             Less than 50,000            Peacock Solar    Lucknow    NaN
5             Less than 50,000                      Nil       Kota    NaN

   know_any_solar_installation_companies
0                                    Yes
2                                    Yes
3                                     No
4                                     No
5                                     No

[5 rows x 38 columns]
```

[54]:
```
df_sorted_city['city'].value_counts()
```

[54]:
```
Bangalore        171
Hyderabad        135
Lucknow          108
Pune              98
Visakhapatnam     64
Ghaziabad         57
Nagpur            53
Indore            49
Gwalior           46
Rajkot            45
Agra              44
Jaipur            44
Bhopal            43
Noida             41
Kanpur            41
Junagadh          34
Varanasi          33
Meerut            33
Nashik            28
Vadodara          27
Ranchi            25
Moradabad         25
Kota              23
Jamnagar          19
Bareilly          16
Panvel            13
```

```
Thane              13
Belgaum            13
Gorakhpur          13
Basti              12
Surat              12
Name: city, dtype: int64
```

[55]:
```
prob = df_sorted_city['city'].value_counts(normalize = False)
threshold = 5
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'barh', color = "orange", figsize = (12, 18))
plt.title('Frequency of Cities')
plt.ylabel('City Name')
plt.xlabel('Count')
plt.xticks(rotation = 0)
plt.show()
```



Now we have some factors that might affect purchase decisions of the customers

We have two types of factors,

1. Encouraging factors (These are the factors which encourage the customers to install solar pl

    a. solar_awareness
    b. power_backup
    c. increase_in_electricity_prices
    d. saving_on_the_electricity_bill
    e. life_span_of_solar_system
    f. interest_in_new_technology
    g. environment_friendly
    h. status_symbol
    i. earning_from_empty_roof

The values of each factor lies on a scale of 1 to 5 where, 1 means least important and 5 means most important

Example: If someone has 5 in their solar_awareness, it means they are aware about solar technology and its uses.

It means the values closer to 5 are important for us

2. Discouraging factors (These are the factors which discourage the customers to install solar

    a. investment_payback
    b. not_getting_sufficient_money_for_extra_electricity
    c. lack_of_government_incentives
    d. lack_of_appropriate_loan_option
    e. high_cost_of_solar_power_systems
    f. concerns_on_outlook_of_the_house
    g. not_understanding_solar_systems
    h. not_knowing_good_solar_installer
    i. non_suitable_rooftop

The values of each factor lies on a scale of 1 to 5 where, 1 means least important and 5 means most important

Example: If someone has 5 in their investment_payback, it means they are least likely to invest in solar power plant and they are not sure if the solar plant will repay a fair amount.

It means the values closer to 1 are important for us

Now let us count and plot values for some of the encouraging factors

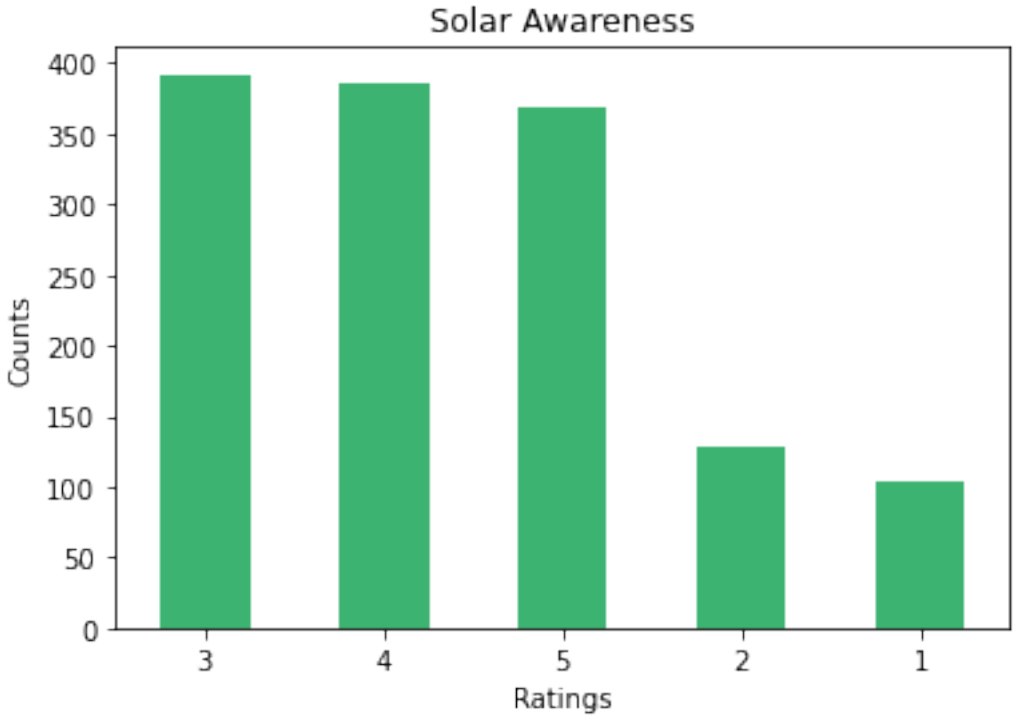
The values which are closer to 5 are better as compared to the values which are closer to 1

[56]:
```
#counting the counts of distinct values in solar_awareness column
df_sorted_city['solar_awareness'].value_counts()
```

[56]:
```
3    392
4    385
```

```
5    369
2    129
1    103
Name: solar_awareness, dtype: int64
```

[57]:
```
#barplot to visualize the value counts of distinct values in colar_awareness column
prob = df_sorted_city['solar_awareness'].value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Solar Awareness')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

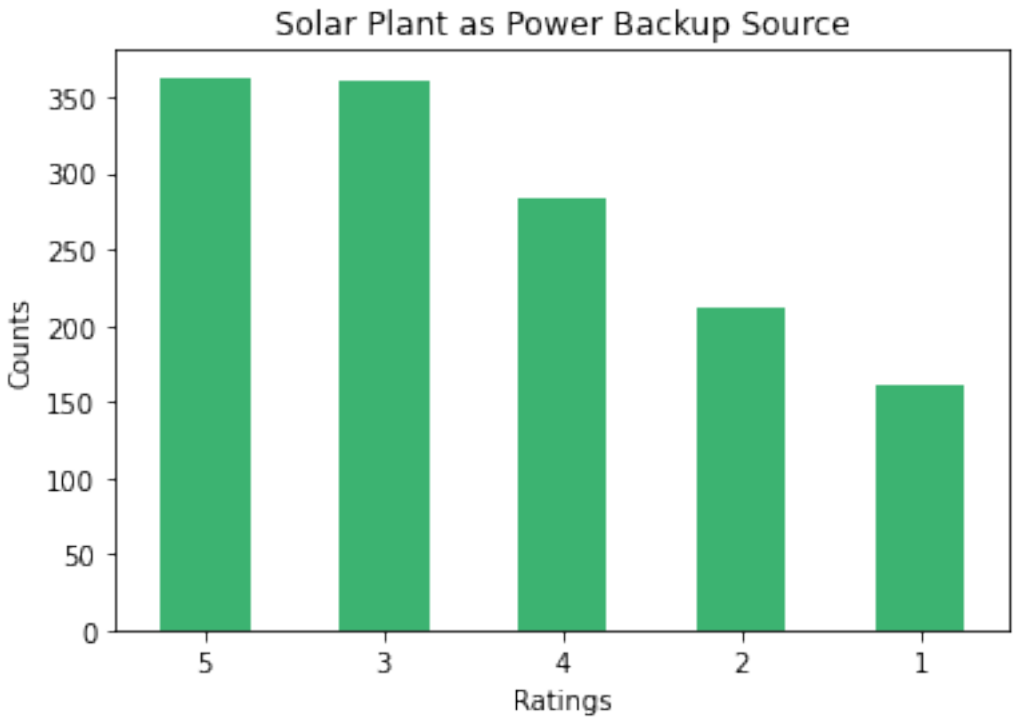
Most people have rated 3, 4, and 5 for *Solar Awareness* which means that they are quite aware of Solar Technology.

---

[58]:
```
df_sorted_city['power_backup'].value_counts()
```

[58]:
```
5    363
3    360
4    283
2    211
1    161
Name: power_backup, dtype: int64
```

[59]:
```
#barplot to visualize the value counts of distinct values in power_backup column
prob = df_sorted_city['power_backup'].value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Solar Plant as Power Backup Source')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

most people have rated 3, 5, and 4 for *Solar Plant as a Power Backup Source* which

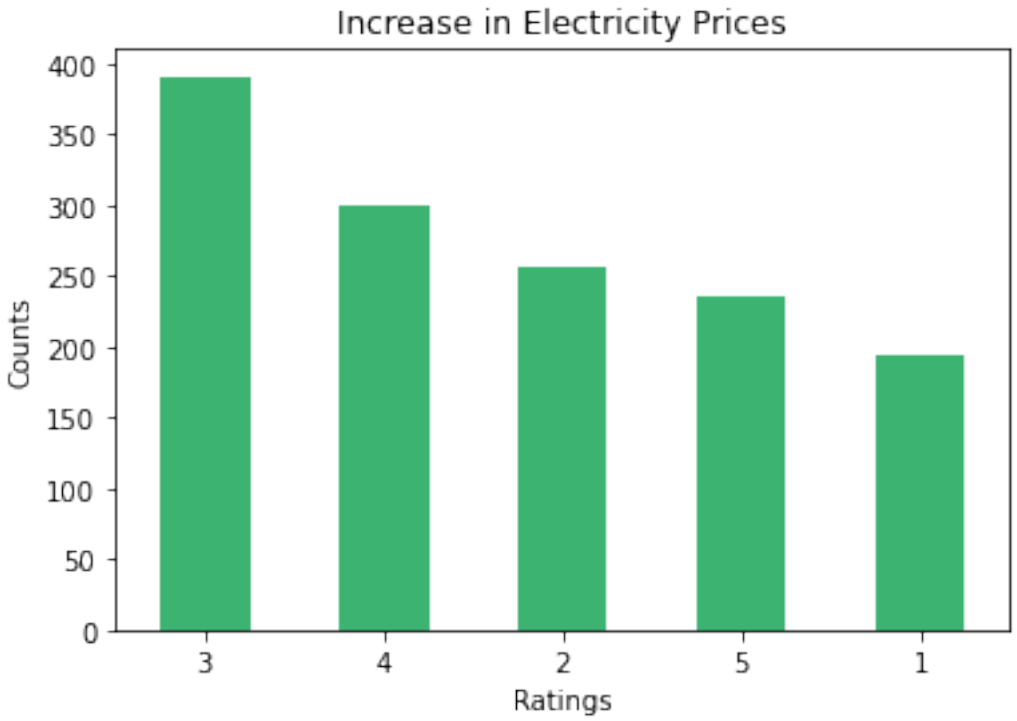
means people consider Solar Technology as a reliable source for power backup.

---

[60]:
```
df_sorted_city['increase_in_electricity_prices'].value_counts()
```

[60]:
```
3    391
4    300
2    257
5    236
1    194
Name: increase_in_electricity_prices, dtype: int64
```

[61]:
```
#barplot to visualize the value counts of distinct values in␣
↪increase_in_electricity_prices column
prob = df_sorted_city['increase_in_electricity_prices'].value_counts(normalize␣
↪= False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Increase in Electricity Prices')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

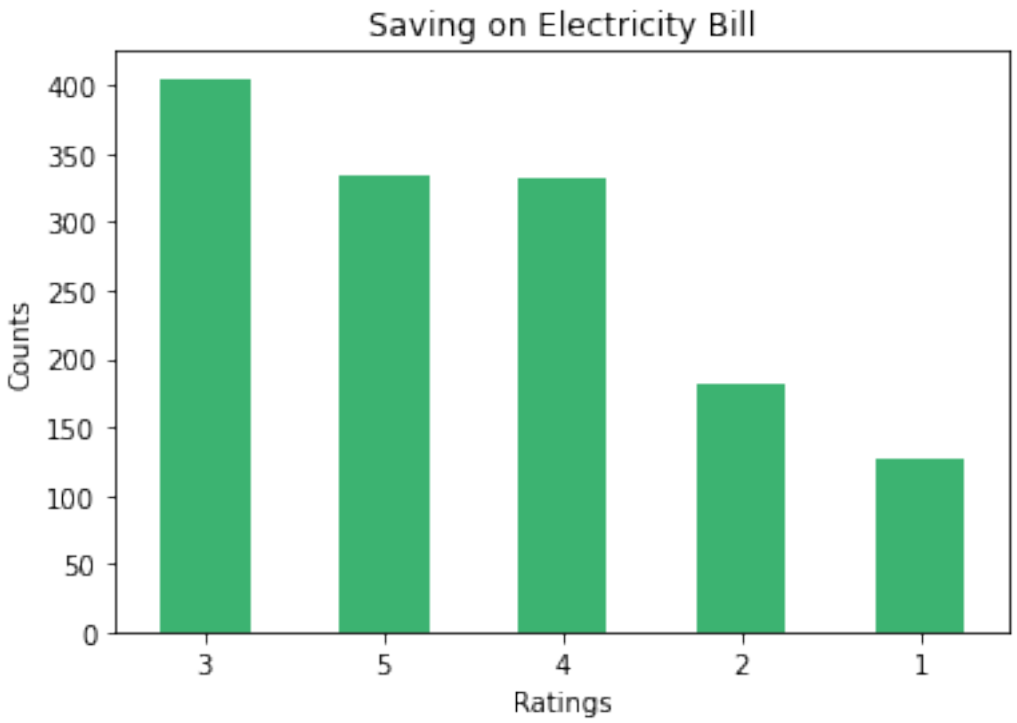
Most people 4, 3, and 2 for *Increase in Electricity Prices* which means people would invest in Solar tEchnology if there is an increase in electricity prices.

---

[62]:
```
df_sorted_city['saving_on_the_electricity_bill'].value_counts()
```

[62]:
```
3    405
5    334
4    332
2    181
1    126
Name: saving_on_the_electricity_bill, dtype: int64
```

[63]:
```
#barplot to visualize the value counts of distinct values in␣
↪saving_on_the_electricity_bill column
prob = df_sorted_city['saving_on_the_electricity_bill'].value_counts(normalize␣
↪= False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Saving on Electricity Bill')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

Most people have rated 3, 5, and 4 for *Saving on Electricity Bill* which means people would invest in Solar Technology for saving on electricity bill.

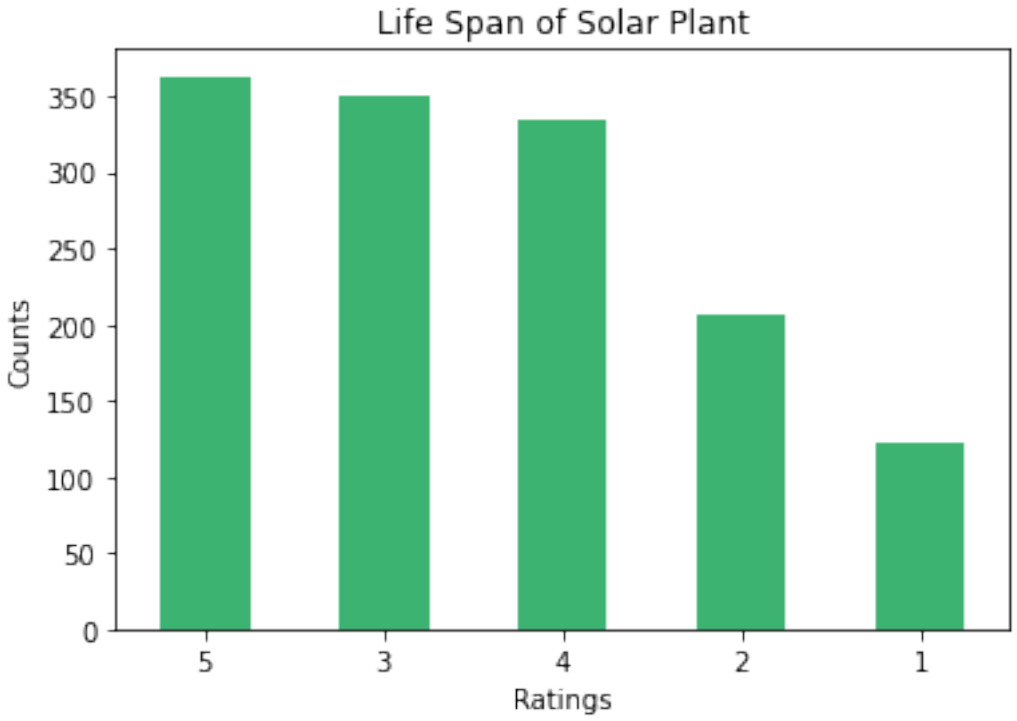
---

[64]:
```
df_sorted_city['life_span_of_solar_system'].value_counts()
```

[64]:
```
5    363
3    350
4    335
2    207
1    123
Name: life_span_of_solar_system, dtype: int64
```

[65]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪life_span_of_solar_system column
prob = df_sorted_city['life_span_of_solar_system'].value_counts(normalize =␣
 ↪False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
```

```
plt.ylabel('Counts')
plt.title('Life Span of Solar Plant')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

Most people have rated 5, 3, and 4 for *Life Span of Solar Plant* which means people will consider investing in Solar Technology if it has a longer life span.

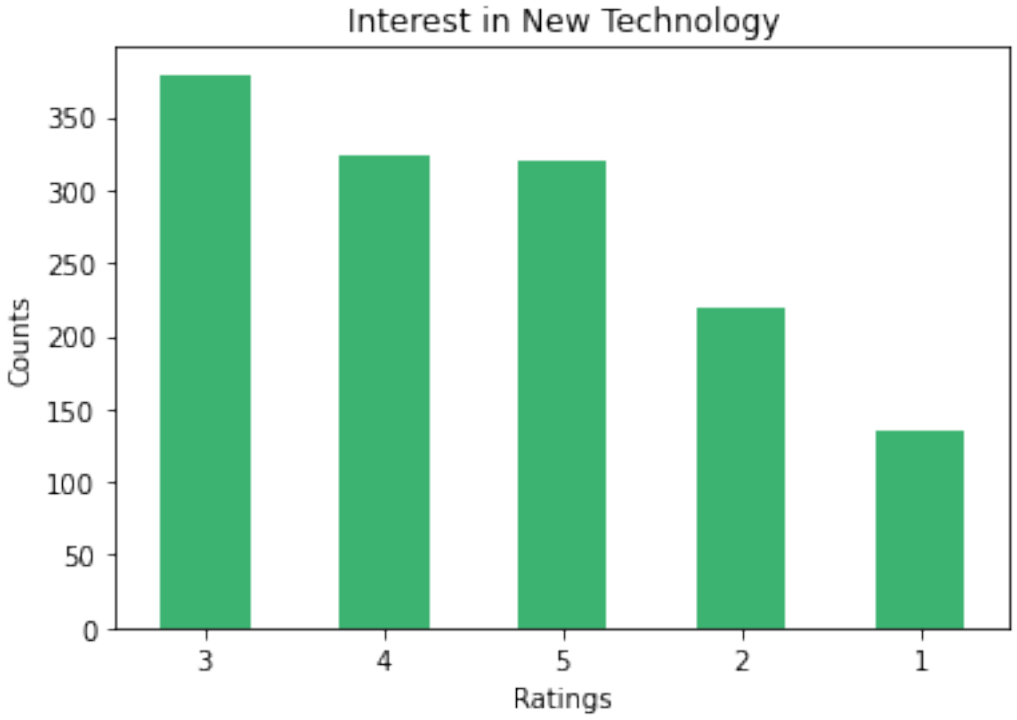
---

[66]:
```
df_sorted_city['interest_in_new_technology'].value_counts()
```

[66]:
```
3    379
4    324
5    320
2    220
1    135
Name: interest_in_new_technology, dtype: int64
```

[67]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪interest_in_new_technology column
prob = df_sorted_city['interest_in_new_technology'].value_counts(normalize =␣
 ↪False)
```

```
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Interest in New Technology')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

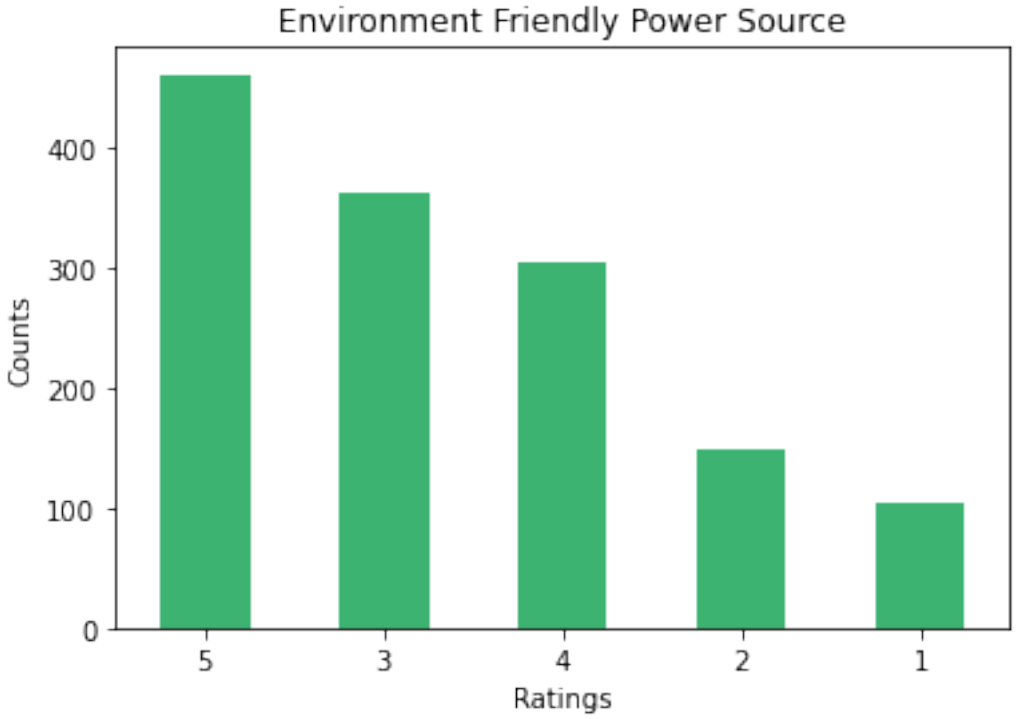
Most people have rated 3, 4, and 5 for *Interest n New Technology* which means they might invest in Solar technology if they are interested enough in it.

---

[68]:
```
df_sorted_city['environment_friendly'].value_counts()
```

[68]:
```
5    460
3    362
4    304
2    149
1    103
Name: environment_friendly, dtype: int64
```

[69]:
```
#barplot to visualize the value counts of distinct values in␣
↪environment_friendly column
prob = df_sorted_city['environment_friendly'].value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Environment Friendly Power Source')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

Most people have rated 5, 3, and 4 for *Solar Plant as an Environment Friendly Power Source* which means people tend to invest in Renewable and Environment Friendly Power Source

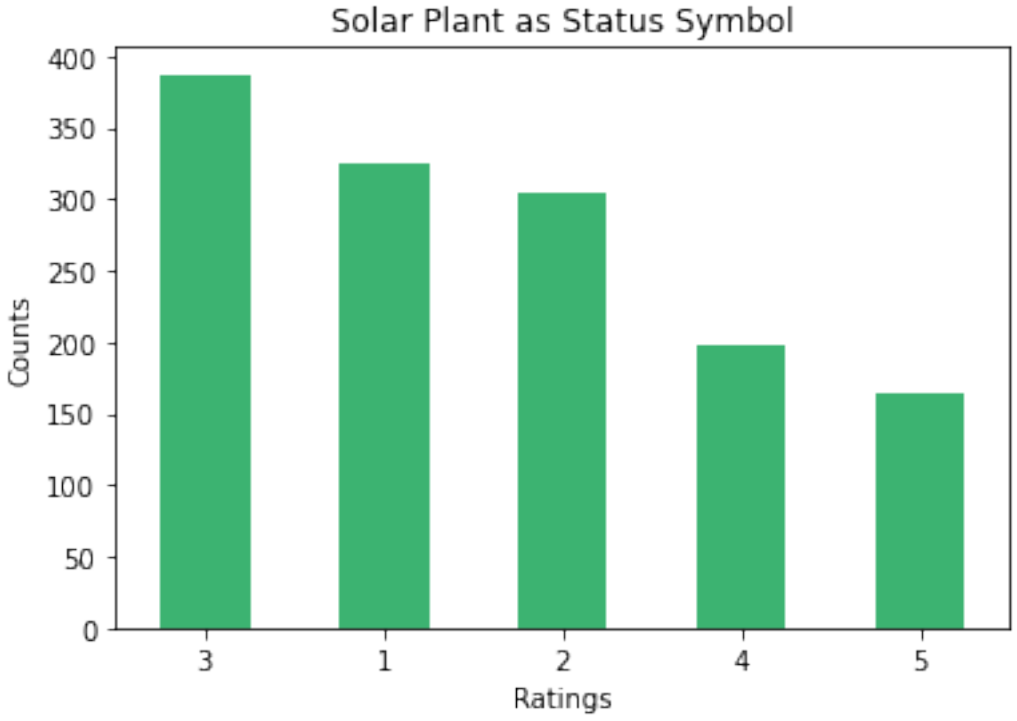
---

[70]:
```
df_sorted_city['status_symbol'].value_counts()
```

[70]:
```
3    387
1    324
```

```
2    305
4    198
5    164
Name: status_symbol, dtype: int64
```

[71]:
```
#barplot to visualize the value counts of distinct values in status_symbol␣
↪column
prob = df_sorted_city['status_symbol'].value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Solar Plant as Status Symbol')
plt.xticks(rotation = 0)
plt.show()
```



From the above barplot, we can see that

Most people have rated 3, 1, and 2 for *Solar Plant as Status Symbol* which means people do not tend to invest in Solar technology for it being a mere status symbol.

---

[72]:
```
df_sorted_city['earning_from_empty_roof'].value_counts()
```

[72]:
```
3    390
5    252
2    249
1    245
4    242
Name: earning_from_empty_roof, dtype: int64
```

[73]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪earning_from_empty_roof column
prob = df_sorted_city['earning_from_empty_roof'].value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color = 'mediumseagreen')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Earning from Empty Roof')
plt.xticks(rotation = 0)
plt.show()
```

From the above barplot, we can see that

Most people have rated 3, 5, and 2 for *Earning From Empty Rooftop* which means that customers consider this factor important while investing in solar technology.

Now let us count and plot values for some of the discouraging factors

The values which are closer to 1 are better as compared to the values which are closer to 5

[74]:
```
df_sorted_city['investment_payback'].value_counts()
```

[74]:
```
3    420
1    261
2    253
5    230
4    214
Name: investment_payback, dtype: int64
```

[75]:
```
#barplot to visualize the value counts of distinct values in investment_payback␣
↪column
prob = df_sorted_city['investment_payback'].value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Investment Payback')
plt.xticks(rotation = 0)
plt.show()
```

From the above bar plot, we can see that

Most people have rated 3, 2, and 1 for *Investment Payback* which means this factor doesnot have a major impact on customer's purchase decision

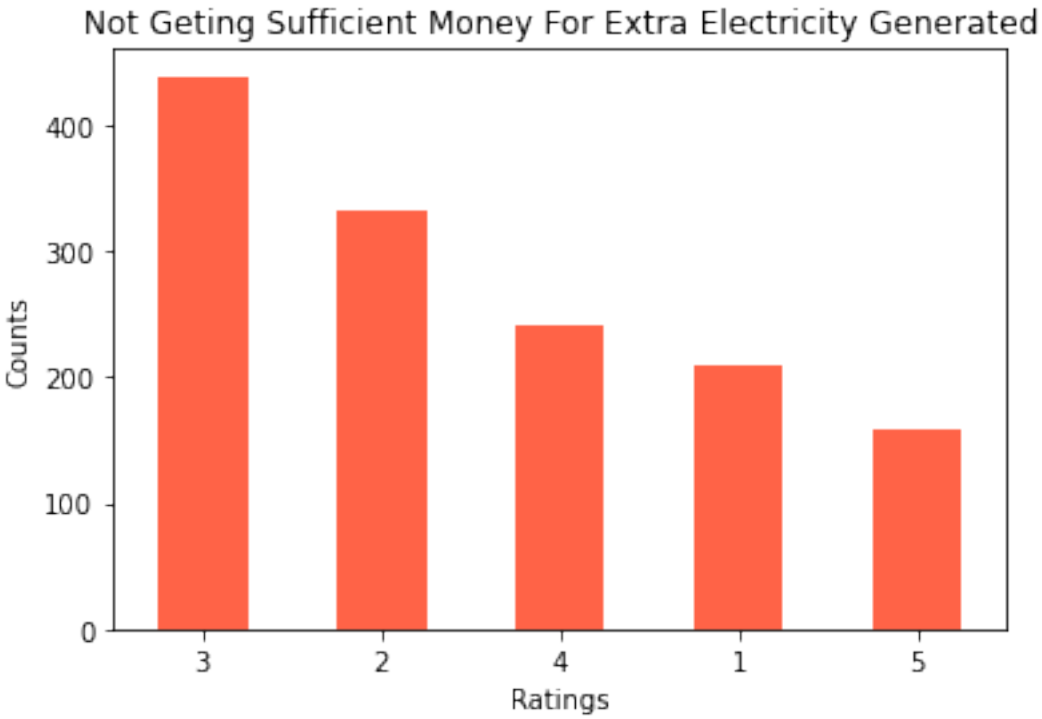
---

[76]:
```
df_sorted_city['not_getting_sufficient_money_for_extra_electricity'].
 ↪value_counts()
```

[76]:
```
3    438
2    332
4    241
1    209
5    158
Name: not_getting_sufficient_money_for_extra_electricity, dtype: int64
```

[77]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪not_getting_sufficient_money_for_extra_electricity column
prob = df_sorted_city['not_getting_sufficient_money_for_extra_electricity'].
 ↪value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
```

```
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Not Geting Sufficient Money For Extra Electricity Generated')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

Most people have rated 3, 2, and 4 for *Not getting Money For Extra Electricity Generated* which means this factor might have an impact on customer's purchase decision

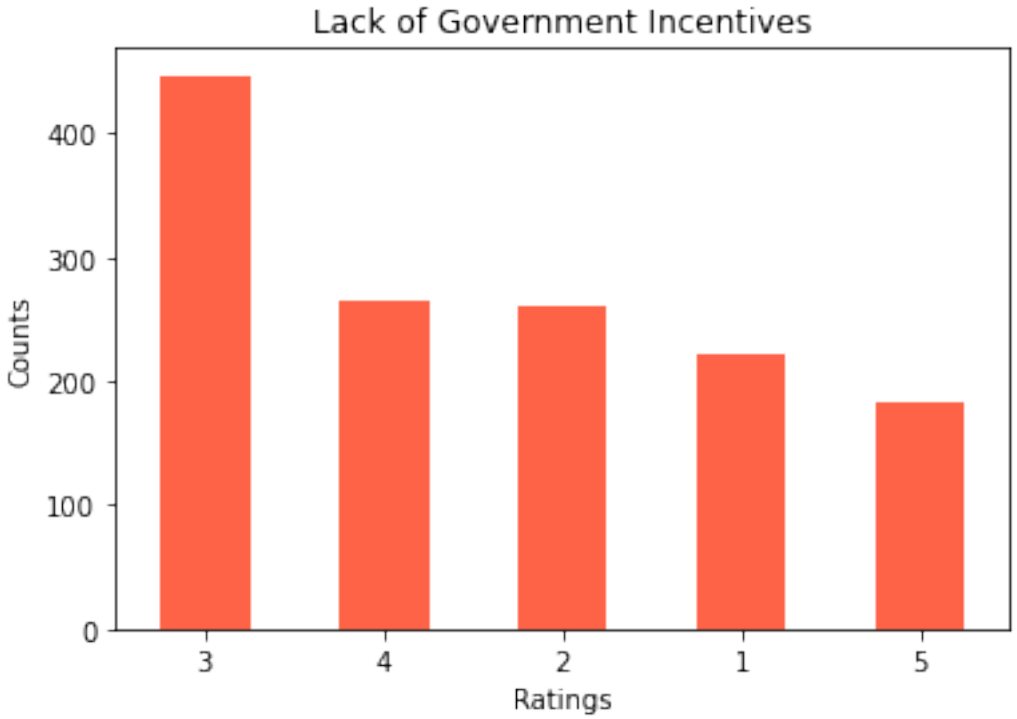
---

[78]:
```
df_sorted_city['lack_of_government_incentives'].value_counts()
```

[78]:
```
3    447
4    266
2    260
1    222
5    183
Name: lack_of_government_incentives, dtype: int64
```

[79]:
```
#barplot to visualize the value counts of distinct values in␣
↪lack_of_government_incentives column
```

```
prob = df_sorted_city['lack_of_government_incentives'].value_counts(normalize =␣
 ↪False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Lack of Government Incentives')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

Most people have rated 3, 4, and 2 for *Lack of government incentives* which means most people are not affected by givernment incentives for renewable energy

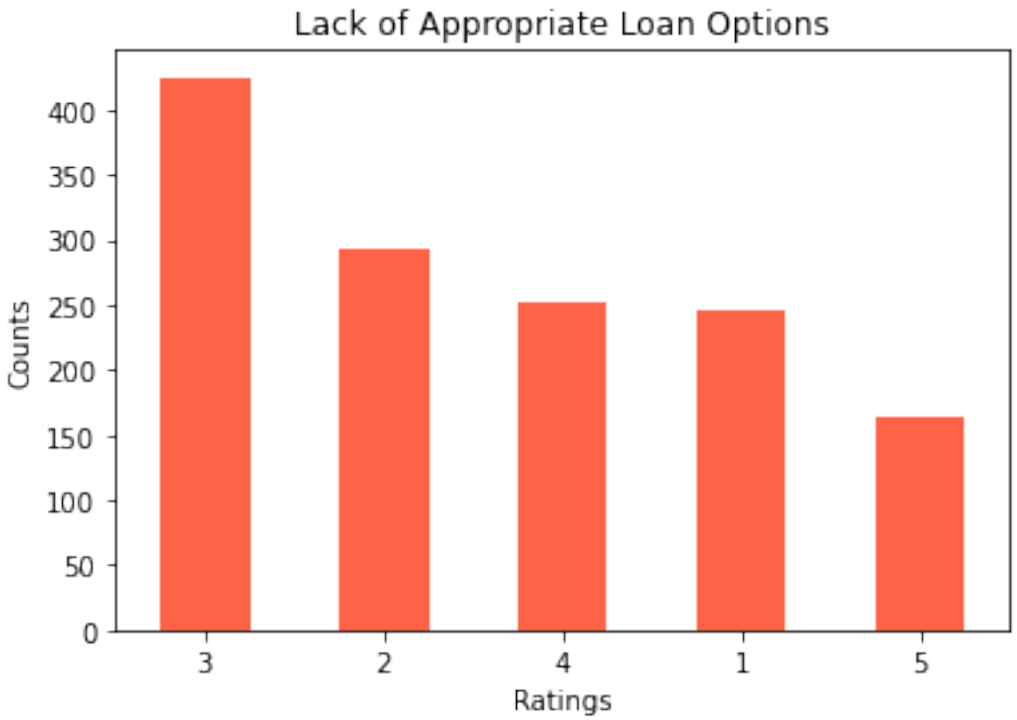
---

[80]:
```
df_sorted_city['lack_of_appropriate_loan_option'].value_counts()
```

[80]:
```
3    425
2    293
4    251
1    246
```

```
5    163
Name: lack_of_appropriate_loan_option, dtype: int64
```

[81]:
```
#barplot to visualize the value counts of distinct values in␣
↪lack_of_appropriate_loan_option column
prob = df_sorted_city['lack_of_appropriate_loan_option'].value_counts(normalize␣
↪= False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Lack of Appropriate Loan Options')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

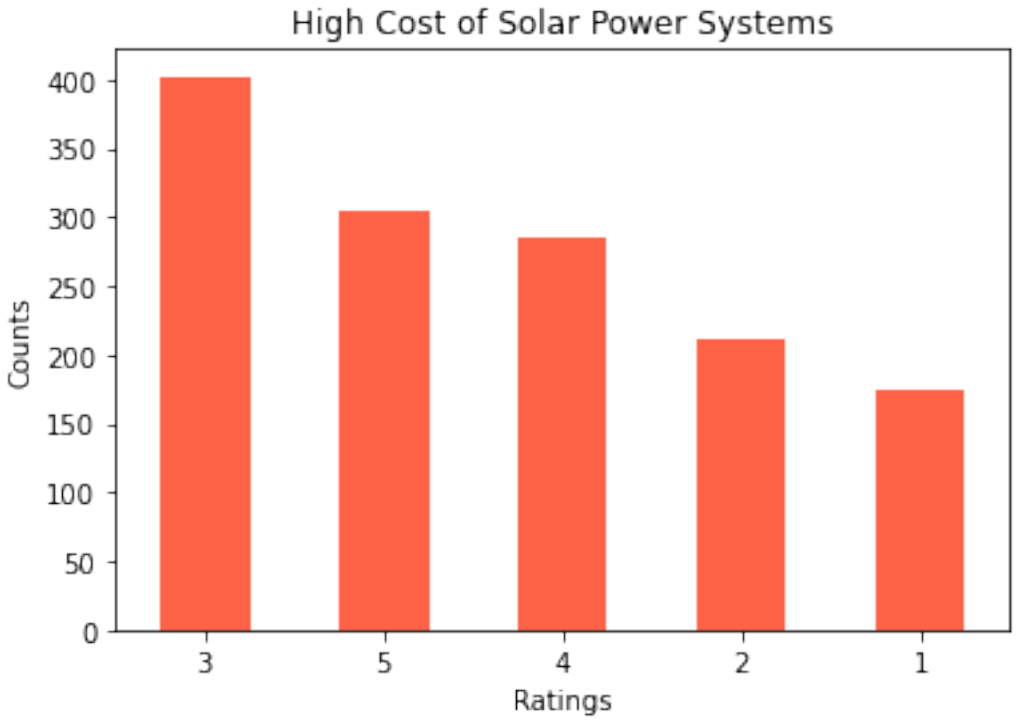
Most people have rated 3, 2, and 4 for *Lack of Appropriate Loan Options* which means this factor doesnot have a major impact on customer's purchase decision

---

[82]:
```
df_sorted_city['high_cost_of_solar_power_systems'].value_counts()
```

[82]:
```
3    402
5    305
4    285
2    211
1    175
Name: high_cost_of_solar_power_systems, dtype: int64
```

[83]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪high_cost_of_solar_power_systems column
prob = df_sorted_city['high_cost_of_solar_power_systems'].
 ↪value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('High Cost of Solar Power Systems')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

Most people have rated 3, 5, and 4 for *High Cost of Solar Power Plants* which means

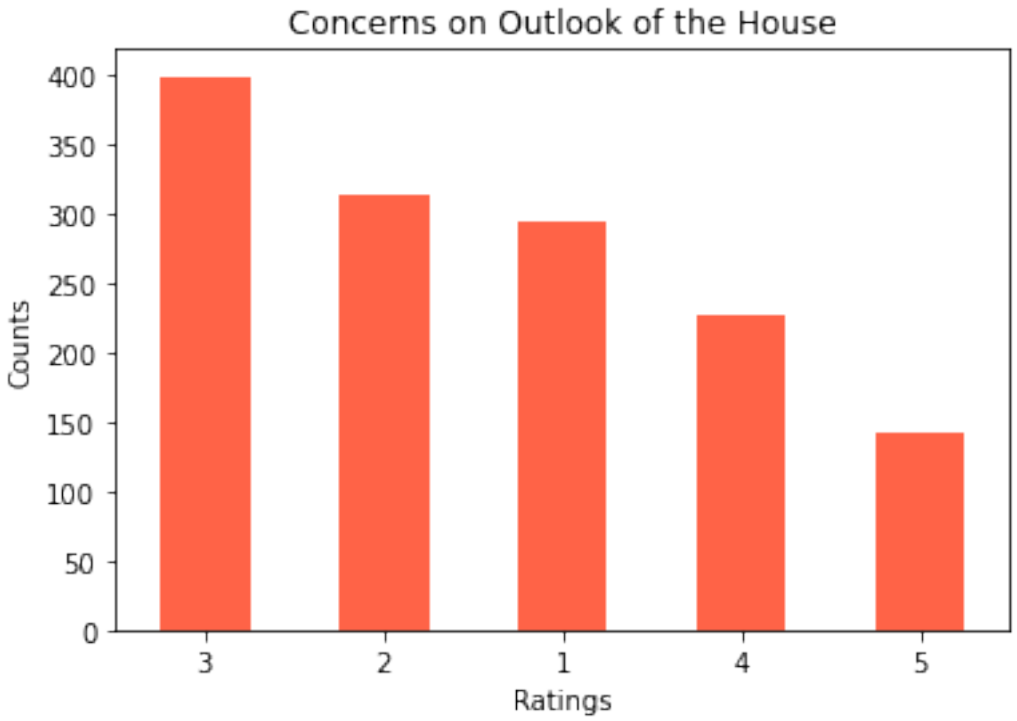
this factor can have a major impact on customer's purchase decision

---

[84]:
```
df_sorted_city['concerns_on_outlook_of_the_house'].value_counts()
```

[84]:
```
3    399
2    314
1    295
4    227
5    143
Name: concerns_on_outlook_of_the_house, dtype: int64
```

[85]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪concerns_on_outlook_of_the_house column
prob = df_sorted_city['concerns_on_outlook_of_the_house'].
 ↪value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Concerns on Outlook of the House')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

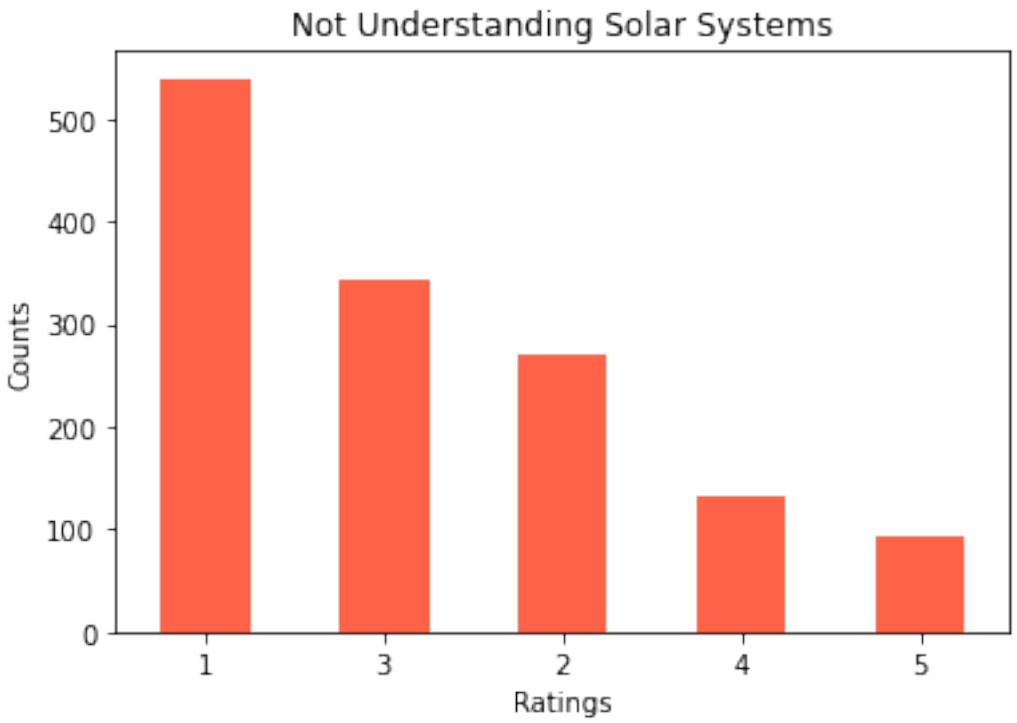
Most people have rated 3, 2, and 1 for *Concerns on Outlook of the House* which means this factor doesnot have a major impact on customer's purchase decision

---

[86]:
```
df_sorted_city['not_understanding_solar_systems'].value_counts()
```

[86]:
```
1    539
3    343
2    271
4    133
5     92
Name: not_understanding_solar_systems, dtype: int64
```

[87]:
```
#barplot to visualize the value counts of distinct values in␣
↪not_understanding_solar_systems column
prob = df_sorted_city['not_understanding_solar_systems'].value_counts(normalize␣
↪= False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Not Understanding Solar Systems')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

Most people have rated 1, 3, and 2 for *Not Understanding Solar Systems* which means this factor doesnot have a major impact on customer's purchase decision

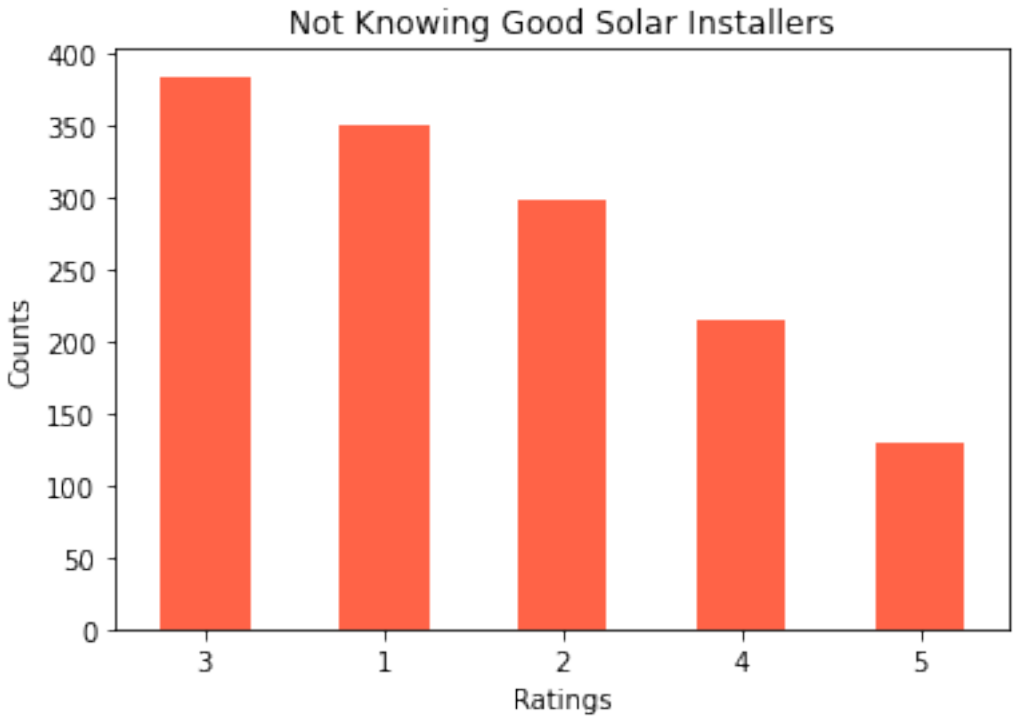
---

[88]:
```
df_sorted_city['not_knowing_good_solar_installer'].value_counts()
```

[88]:
```
3    384
1    351
2    298
4    215
5    130
Name: not_knowing_good_solar_installer, dtype: int64
```

[89]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪not_knowing_good_solar_installer column
prob = df_sorted_city['not_knowing_good_solar_installer'].
 ↪value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
```

```
plt.ylabel('Counts')
plt.title('Not Knowing Good Solar Installers')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

Most people have rated 3, 1, and 2 for *Not Knowing Good Solar Installers* which means this factor doesnot have a major impact on customer's purchase decision

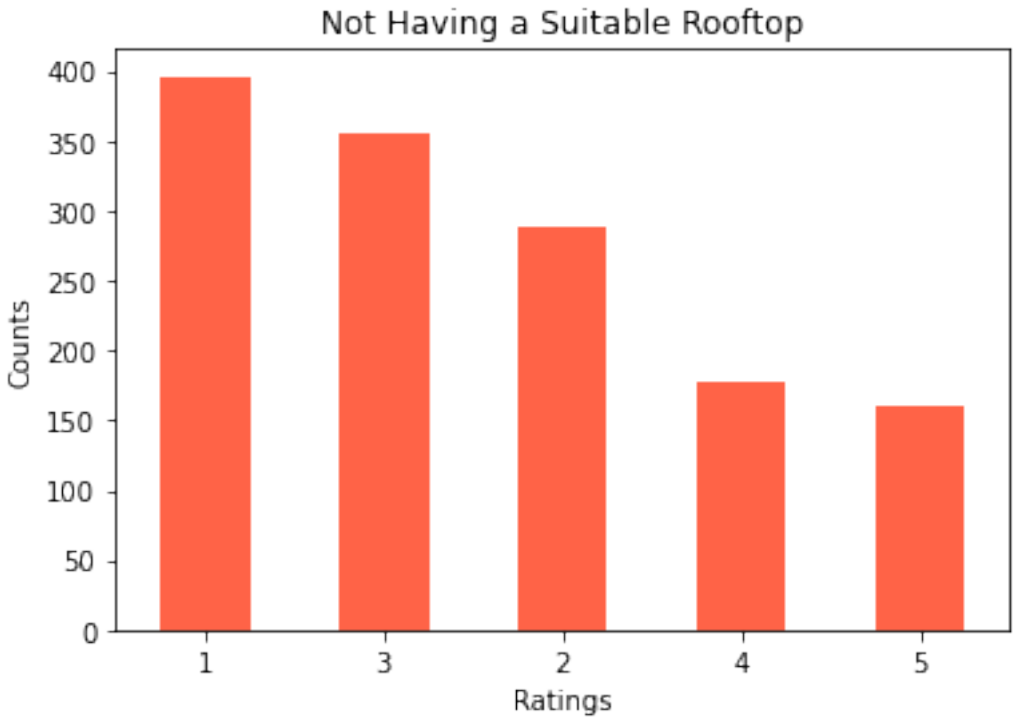
---

[90]:
```
df_sorted_city['non_suitable_rooftop'].value_counts()
```

[90]:
```
1    396
3    356
2    289
4    177
5    160
Name: non_suitable_rooftop, dtype: int64
```

[91]:
```
#barplot to visualize the value counts of distinct values in␣
 ↪non_suitable_rooftop column
prob = df_sorted_city['non_suitable_rooftop'].value_counts(normalize = False)
threshold = 0.02
```

```
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'bar', color='tomato')
plt.xlabel('Ratings')
plt.ylabel('Counts')
plt.title('Not Having a Suitable Rooftop')
plt.xticks(rotation = 0)
plt.show()
```



From the above bar plot, we can see that

Most people have rated 1, 3, and 2 for *Not Having a Suitable Rooftop* which means this factor doesnot have a major impact on customer's purchase decision

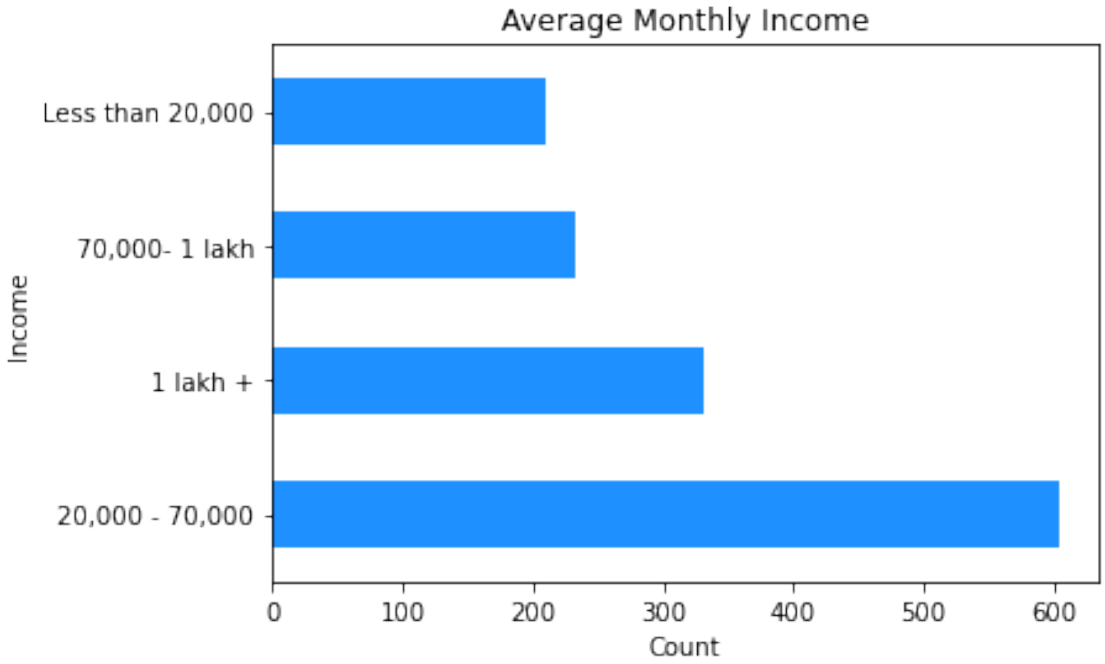
---

Now let us analyze the Average Monthly Income of the customers

[92]:
```
df_sorted_city['average_monthly_income'].value_counts()
```

[92]:
```
20,000 - 70,000      604
1 lakh +             332
70,000- 1 lakh       232
Less than 20,000     210
Name: average_monthly_income, dtype: int64
```

[93]:
```
#barplot to visialize the average monthly income of our customers
prob = df_sorted_city['average_monthly_income'].value_counts(normalize = False)
threshold = 0.02
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'barh', color = "dodgerblue")
plt.title('Average Monthly Income')
plt.ylabel('Income')
plt.xlabel('Count')
plt.xticks(rotation = 0)
plt.show()
```



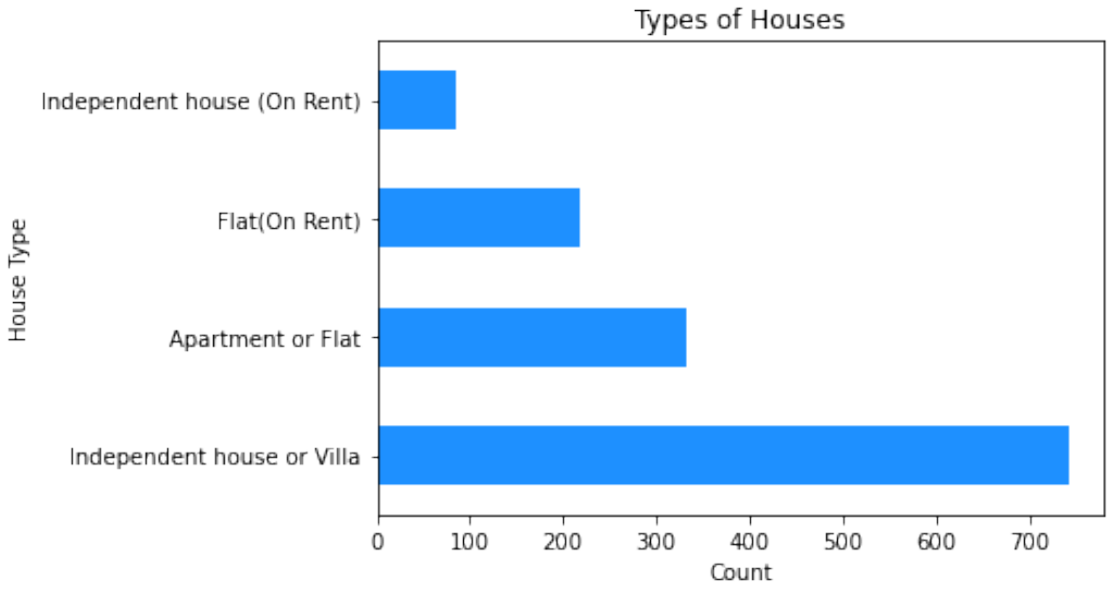
We can see that maximum people have Monthly Income between 20,000 to 70,000

Now let us analyze Type of House that the customers own

[94]:
```
df_sorted_city['type_of_home'].value_counts()
```

[94]:
```
Independent house or Villa     744
Apartment or Flat              332
Flat(On Rent)                  218
Independent house (On Rent)     84
Name: type_of_home, dtype: int64
```

[95]:
```
prob = df_sorted_city['type_of_home'].value_counts(normalize = False)
threshold = 5
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'barh', color = "dodgerblue")
plt.title('Types of Houses')
plt.ylabel('House Type')
plt.xlabel('Count')
plt.xticks(rotation = 0)
plt.show()
```



We can see that maximum people own an Independent house or Villa

---

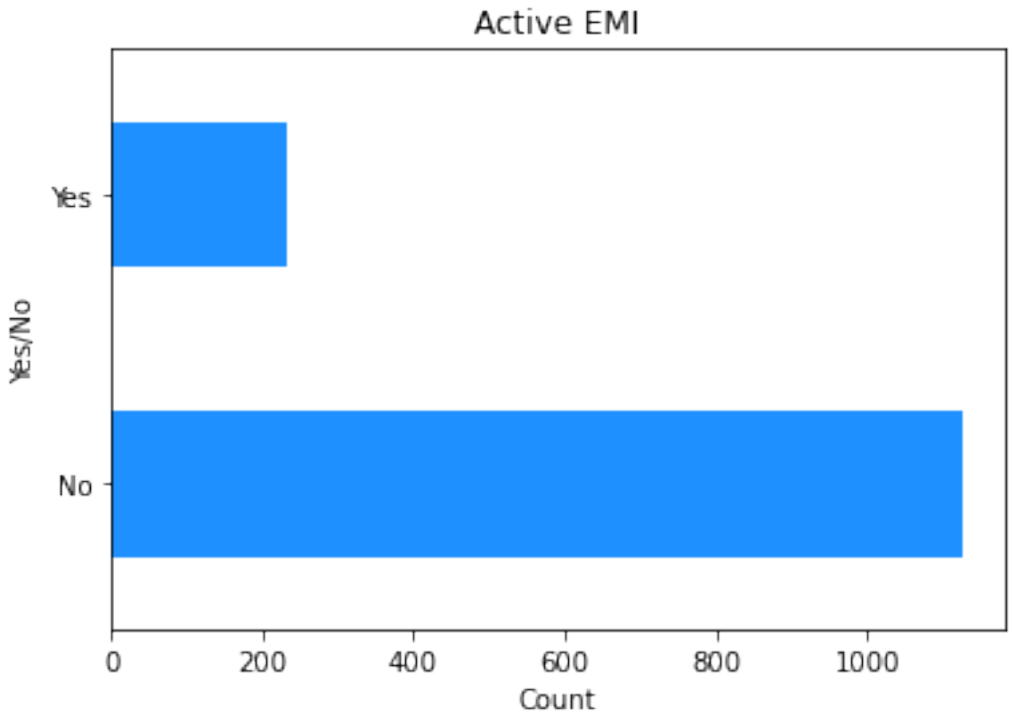
Now let us analyze the customers who have an Active EMI

[96]:
```
df_sorted_city['active_emi'].value_counts()
```

[96]:
```
No     1127
Yes     232
Name: active_emi, dtype: int64
```

[97]:
```
prob = df_sorted_city['active_emi'].value_counts(normalize = False)
threshold = 5
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'barh', color = "dodgerblue")
plt.title('Active EMI')
```

```
plt.ylabel('Yes/No')
plt.xlabel('Count')
plt.xticks(rotation = 0)
plt.show()
```



We can see that maximum people donot have an Active EMI

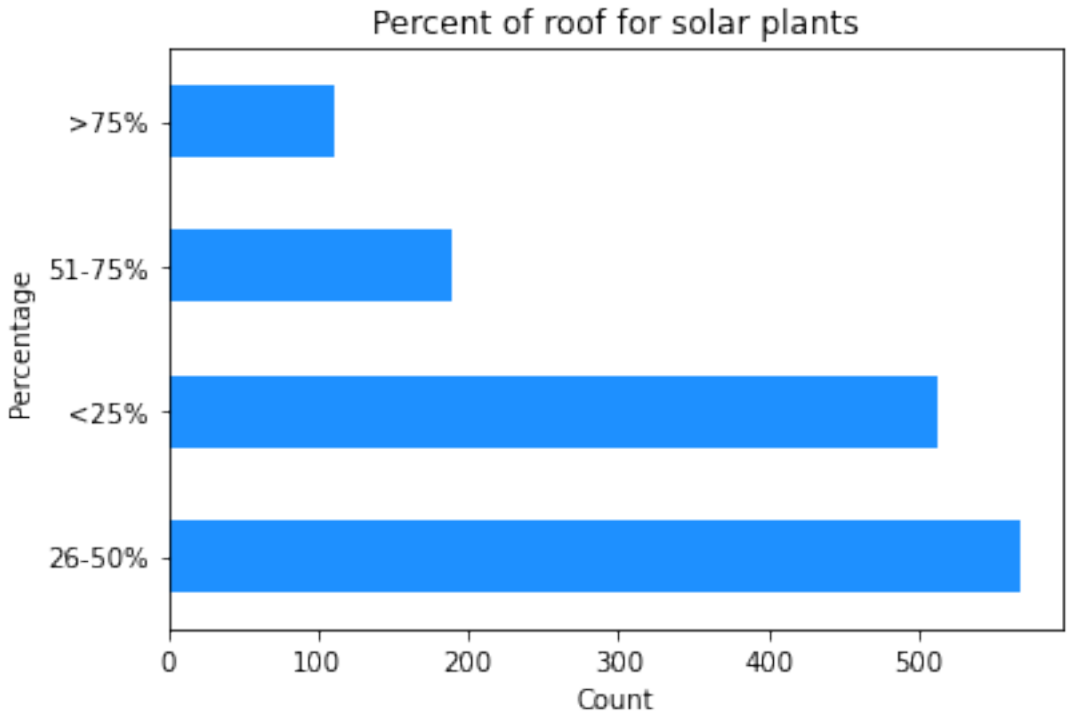
Now let us analyze Percentage of Roof that the customers are ready for utilizing for solar installation

[98]:
```
df_sorted_city['percentage_of_roof'].value_counts()
```

[98]:
```
26-50%    567
<25%      512
51-75%    189
>75%      110
Name: percentage_of_roof, dtype: int64
```

[99]:
```
prob = df_sorted_city['percentage_of_roof'].value_counts(normalize = False)
threshold = 5
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'barh', color = "dodgerblue")
plt.title('Percent of roof for solar plants')
```

```
plt.ylabel('Percentage')
plt.xlabel('Count')
plt.xticks(rotation = 0)
plt.show()
```



We can see that maximum people are comfortable with utilizing 26% - 50% of their roof for installing solar plant

---

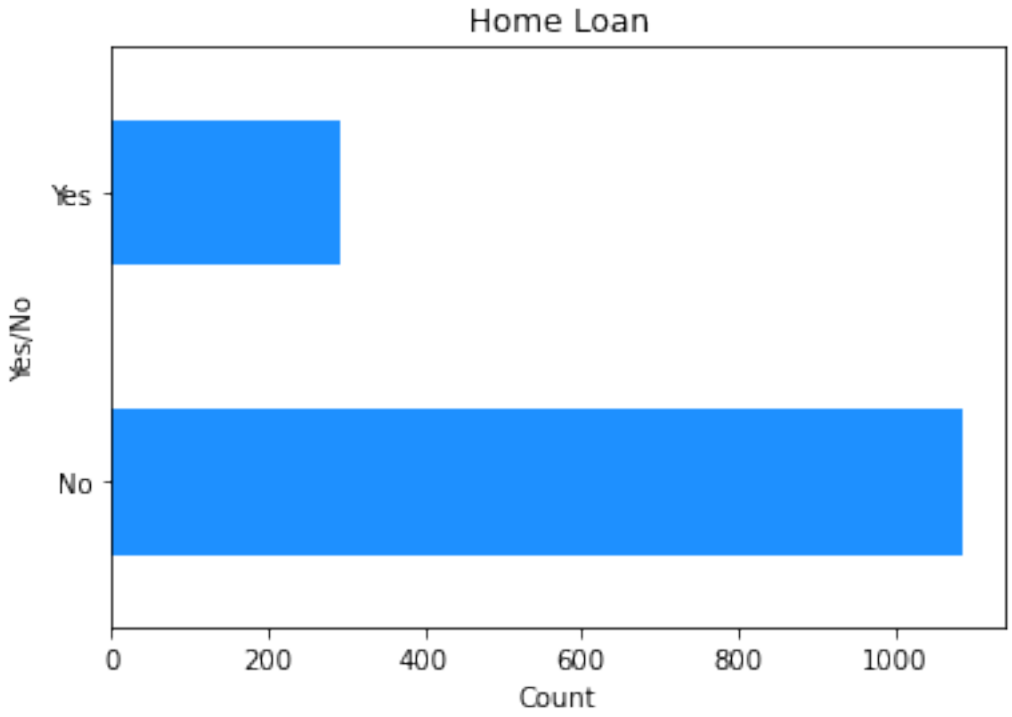
Now let us analyze the customers who have taken Home Loan

[100]:
```
df_sorted_city['home_loan'].value_counts()
```

[100]:
```
No     1087
Yes     291
Name: home_loan, dtype: int64
```

[101]:
```
prob = df_sorted_city['home_loan'].value_counts(normalize = False)
threshold = 5
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'barh', color = "dodgerblue")
plt.title('Home Loan')
plt.ylabel('Yes/No')
```

```
plt.xlabel('Count')
plt.xticks(rotation = 0)
plt.show()
```



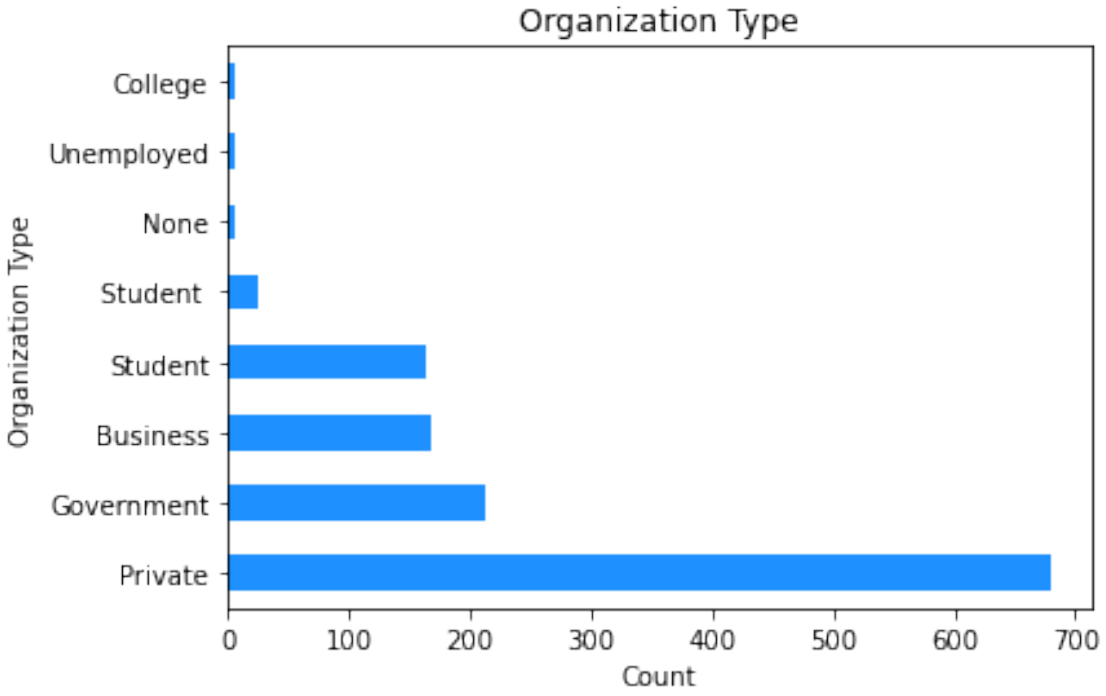
We can see that maximum people donot have a Home Loan

Now let us analyze Type of Organization that the customers work at

[102]:
```
df_sorted_city['organization_type'].value_counts()
```

[102]:
```
Private             680
Government          212
Business            169
Student             163
Student              25
                   ...
No working            1
Iam studying          1
MNC                   1
Not working now       1
STUDENT               1
Name: organization_type, Length: 89, dtype: int64
```

[103]:
```
prob = df_sorted_city['organization_type'].value_counts(normalize = False)
threshold = 5
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'barh', color = "dodgerblue")
plt.title('Organization Type')
plt.ylabel('Organization Type')
plt.xlabel('Count')
plt.xticks(rotation = 0)
plt.show()
```



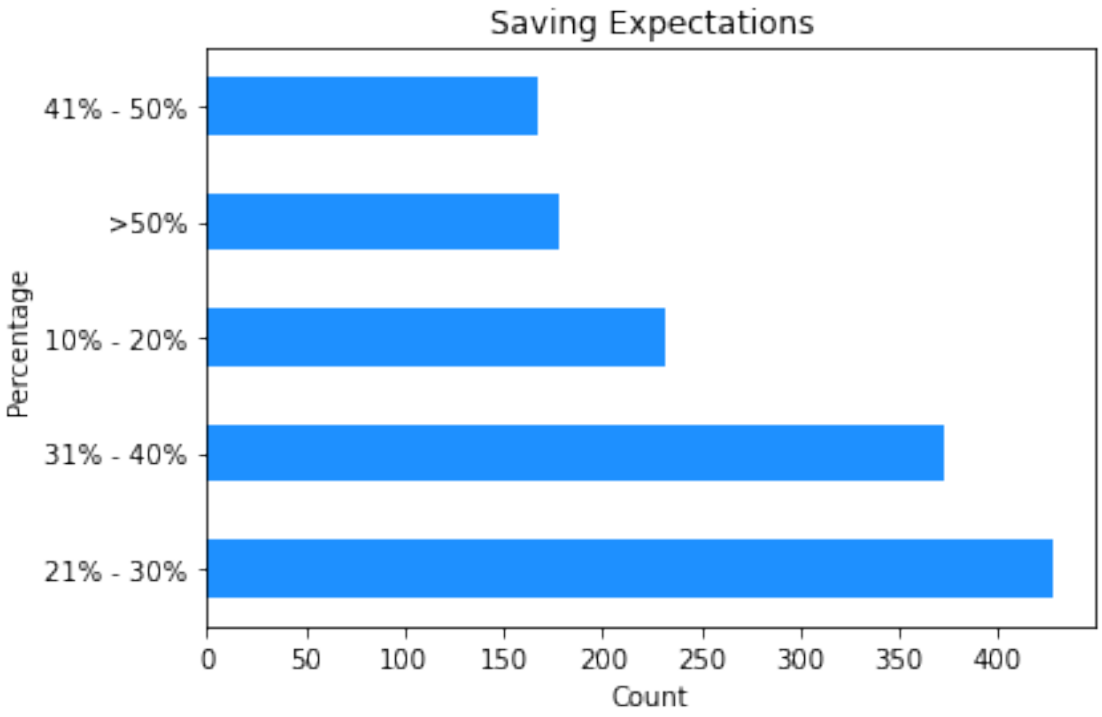
We can see that maximum people have a Private Job

Now let us analyze Expectations of Customers on Saving Electricity Bill

[104]:
```
df_sorted_city['expectation_on_saving_electricity_bill'].value_counts()
```

[104]:
```
21% - 30%    428
31% - 40%    373
10% - 20%    232
>50%         178
41% - 50%    167
Name: expectation_on_saving_electricity_bill, dtype: int64
```

[105]:
```
prob = df_sorted_city['expectation_on_saving_electricity_bill'].
 ↪value_counts(normalize = False)
threshold = 5
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'barh', color = "dodgerblue")
plt.title('Saving Expectations')
plt.ylabel('Percentage')
plt.xlabel('Count')
plt.xticks(rotation = 0)
plt.show()
```



We can see that maximum people expect to save 21% - 30% on Electricity Bill after installing Solar Plant

---

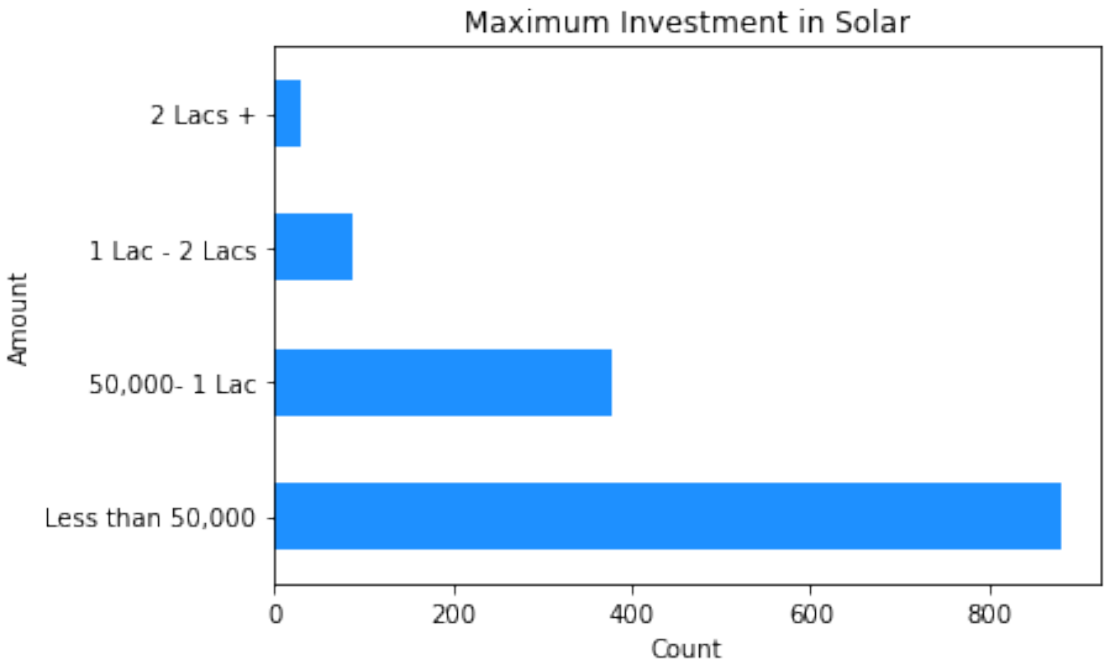
Now let us analyze Maximum Investment that the Customers are Ready to make

[106]:
```
df_sorted_city['maximum_investment_in_solar'].value_counts()
```

[106]:
```
Less than 50,000    881
50,000- 1 Lac       377
1 Lac - 2 Lacs       89
2 Lacs +             31
```

```
Name: maximum_investment_in_solar, dtype: int64
```

[107]:
```
prob = df_sorted_city['maximum_investment_in_solar'].value_counts(normalize =␣
↪False)
threshold = 5
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'barh', color = "dodgerblue")
plt.title('Maximum Investment in Solar')
plt.ylabel('Amount')
plt.xlabel('Count')
plt.xticks(rotation = 0)
plt.show()
```



We can see that maximum people would invest Less than 50,000 in Solar Technology

---

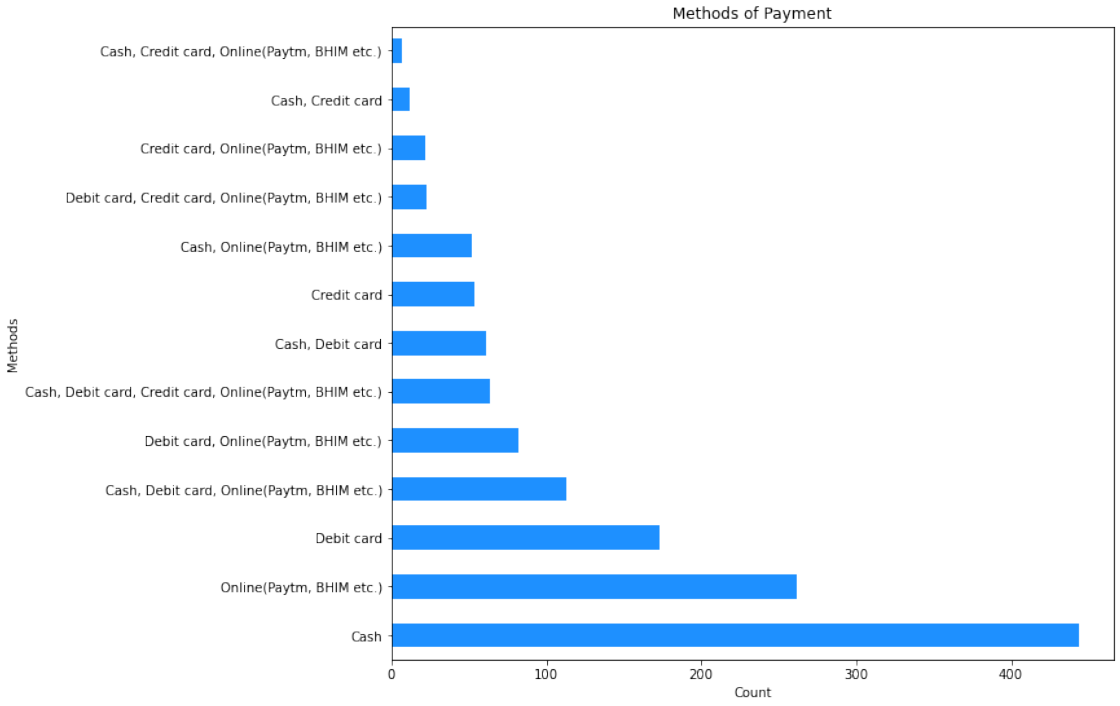
Now let us analyze Payment Method that the customers used

[108]:
```
df_sorted_city['payment_method'].value_counts()
```

[108]:
```
Cash                                          444
Online(Paytm, BHIM etc.)                      262
Debit card                                    173
Cash, Debit card, Online(Paytm, BHIM etc.)    113
```

```
Debit card, Online(Paytm, BHIM etc.)                          82
Cash, Debit card, Credit card, Online(Paytm, BHIM etc.)       64
Cash, Debit card                                              61
Credit card                                                   54
Cash, Online(Paytm, BHIM etc.)                                52
Debit card, Credit card, Online(Paytm, BHIM etc.)             23
Credit card, Online(Paytm, BHIM etc.)                         22
Cash, Credit card                                             12
Cash, Credit card, Online(Paytm, BHIM etc.)                    7
Debit card, Credit card                                        5
Cash, Debit card, Credit card                                  4
Name: payment_method, dtype: int64
```

[109]:
```
prob = df_sorted_city['payment_method'].value_counts(normalize = False)
threshold = 5
mask = prob > threshold
prob = prob.loc[mask]
prob.plot(kind = 'barh', color = "dodgerblue", figsize = (10, 9))
plt.title('Methods of Payment')
plt.ylabel('Methods')
plt.xlabel('Count')
plt.xticks(rotation = 0)
plt.show()
```



We can see that maximum people are comfortable in making Payment with Cash

Creating a custom algorithm for plotting data against conditions

[110]:
```
##########################################################################################
##########################################################################################
#----------------------------------------------------------------------------------------
#------------------------------------------ Algorithm for plotting our data␣
↪------------------------------------------#
#----------------------------------------------------------------------------------------
##########################################################################################
##########################################################################################

#import seaborn as sns
#import pandas as pd
#import numpy as np
#import matplotlib.pyplot as plt
import matplotlib.patches as patches
import math as math
from pylab import *
import io, os, sys, types
pd.options.mode.chained_assignment = None  # default='warn'
sns.set_style("whitegrid")
mpl.rc("savefig", dpi=150)

def␣
↪PlotLikertOverConditions(tb,nPoint,customLikertRange=None,tb2=None,customLikertRange2=None)
↪
    # This functions gets a table of questions and their responces in likert␣
↪scale (1:positive N:negative) as coulmns,
    # as well as a another column indicating the condition of the responce
    # This function can also get another table for general questions after all␣
↪conditions


    Qs=tb.columns.tolist()
    CustomLikertLabels_orderd_by_y_axis=[]

    df=tb.copy(deep=True)
    likert_colors= sns.color_palette("coolwarm", nPoint)
    likert_colors=[(1.0,1.0,1.0)]+likert_colors


    font1 = {'family': 'sans-serif','color':  'white','weight': 'normal','size':
↪ 10,}
    font2 = {'family': 'sans-serif','color':  'grey','weight': 'normal','size':␣
↪8,}
```

```
    LikertRange=[1,2,3,4,5]
    fig, ax = plt.subplots(1, 1,figsize=(10,8))


    ␣
 ↪##-------------------------------------------------------------------------------------
    ##Seperate into conditions and count the scores for each
    df_conds=[]
    middles_all=[]
#     df.loc[:,'condition']=df['condition'].astype(np.int32)

    barwidth=0.2
    def SHIFT(N,i):
        return float(i)*barwidth*1.0-(float(N)-1)*barwidth


    conds=[]
    for cond in df['condition'].unique():
        conds.append(cond)
        temp=df[df['condition']==cond]
        temp.drop('condition', axis=1,inplace=True)
        temp=pd.DataFrame(temp.stack())
        temp=pd.DataFrame(temp.unstack(0))


        g= lambda x,y: x.loc[y] if y in x.index else 0
        temp2=temp.copy(deep=True)
        for q in range(1,len(Qs)):
            for i in range(1,nPoint+1):
    #             print 'there was %s of %s in Q %s'%(g(temp.loc[Qs[q],:].
 ↪value_counts(),i),i,q)
                temp2.loc[Qs[q], i]=g(temp.loc[Qs[q],:].value_counts(),i)

        temp2.drop(0, axis=1,inplace=True)
        temp2.columns = temp2.columns.droplevel(level=1)
#         print temp2
        df_conds.append( temp2)
        middles_all.append(temp2[LikertRange[:len(LikertRange)//2]].
 ↪sum(axis=1)+temp2[len(LikertRange)//2+1]*.5)


    ␣
 ↪##-------------------------------------------------------------------------------------
    ## general questions
    df_conds_generals=[]
    middles_all_generals=[]
    ## here we add general questions as well if there is any!
```

```
    if type(tb2)==pd.core.frame.DataFrame:
        temp=tb2.copy(deep=True)
        temp2=tb2.T.copy(deep=True)

        for gQ, g_col in temp.iteritems():
            g= lambda x,y: x.loc[y] if y in x.index else 0
            for i in range(1,nPoint+1):
                temp2.loc[gQ, i]=g(g_col.astype(np.int32,inplace=True).
 ↪value_counts(),i)

        temp2=temp2.loc[:,range(1,nPoint+1)]
#         print temp2
        df_conds_generals= temp2
        middles_all_generals=temp2[LikertRange[:len(LikertRange)//2]].
 ↪sum(axis=1)+temp2[len(LikertRange)//2+1]*.5



 ↪##-----------------------------------------------------------------------------------------

    ## add shift column to each table
    if len(np.array(middles_all_generals))==0:
        longest= max(map(max,  np.array(middles_all)) )
    else:
        longest= max(max(map(max,  np.array(middles_all)) ),max(np.
 ↪array(middles_all_generals)) )



    patches_already_moved=[]
    for cond,df_c in enumerate(df_conds):

        df_c.insert(0, '', (middles_all[cond] - longest).abs())
        complete_longest=int(longest+(df_c[:].sum(axis=1).max()-longest))#in␣
 ↪our case is 16

        patch_handles = []

        patch_handles.append(df_c.plot.barh(ax=ax,stacked=True,␣
 ↪color=likert_colors, legend=False,

 ↪width=barwidth,edgecolor='white'))#,alpha=1.0-(float(cond)/len(df_conds))*0.7

        shift=SHIFT(len(df_conds),cond)

        for j in range(len(patch_handles)):
```

```
            for i, p in enumerate(patch_handles[j].get_children()):


                if type(p)==(matplotlib.patches.Rectangle):

                    if p.get_height()==barwidth and not (p in␣
 ↪patches_already_moved):
#                         print (p in patches_already_moved),

                        p.set_xy((p.get_x(),p.get_y()+shift))


                        if p.get_width()>1 and p.get_facecolor()[0:3]!
 ↪=likert_colors[0]:#p.get_facecolor()!=(1.0, 1.0, 1.0, 1.0):
#                             if cond % 2 == 0:
#                                 p.set_hatch('\ '*cond)
#                             else:
#                                 p.set_hatch('/ '*cond)

                            patch_handles[j].text(
                                p.get_x()+p.get_width()/2.,
                                p.get_y()+ p.get_height() /(len(Qs)-1),
                                "{0:.0f}%".format(p.get_width()/(len(tb)/
 ↪len(tb['condition'].unique())) * 100),
                                ha="center",
                                fontdict=font1)#.set_zorder(-1)



        patches_already_moved=patches_already_moved+patch_handles[j].
 ↪get_children()

    yticks=list(ax.get_yticks())
#     print customLikertRange
    CustomLikertLabels_orderd_by_y_axis=[customLikertRange[key] if␣
 ↪(customLikertRange!=None and customLikertRange.get(key))
                                         else ['very low','very high']
                                         for key in (ax.get_yticks()+1)]


    if type(tb2)==pd.core.frame.DataFrame:
        CustomLikertLabels_orderd_by_y_axis=[customLikertRange2[key] if␣
 ↪(customLikertRange2!=None and customLikertRange2.get(key))
                                         else ['very low','very high']
                                         for key in␣
 ↪range(1,len(df_conds_generals)+1)][::-1]+CustomLikertLabels_orderd_by_y_axis
```

```
        ## Plotting general questions
        def SHIFT2(i):
            i=i+0.1
            extra=0.5
            return -1.3 -(i*(2*(barwidth)+extra))

        df_conds_generals.insert(0, '', (middles_all_generals - longest).abs())
        for i in range(0,len(df_conds_generals)):
            y=SHIFT2(i-0.1)
            yticks=[y]+ yticks
            y=y+barwidth/2.0
            ax.plot([-5,df_conds_generals.iloc[i,0]],[y,y],linestyle=':',␣
↪color='grey', alpha=.2,linewidth=1)



        patch_handles = []
        patch_handles.append(df_conds_generals.plot.barh(ax=ax,stacked=True,␣
↪color=likert_colors, legend=False,
                                                width=barwidth,edgecolor='white'))

        for j in range(len(patch_handles)):
            for i, p in enumerate(patch_handles[j].get_children()):


                if type(p)==(matplotlib.patches.Rectangle):

                    if p.get_height()==barwidth and not (p in␣
↪patches_already_moved):
                        shift=SHIFT2(p.get_y())

                        p.set_xy((p.get_x(),shift))
                        if p.get_width()>1 and p.get_facecolor()[0:3]!
↪=likert_colors[0]:#p.get_facecolor()!=(1.0, 1.0, 1.0, 1.0):

                            patch_handles[j].text(
                                p.get_x()+p.get_width()/2.,
                                shift+ p.get_height() /(len(Qs)-1),
                                "{0:.0f}%".format(p.get_width()/(len(tb)/
↪len(tb['condition'].unique())) * 100),
                                ha="center",
                                fontdict=font1)#.set_zorder(-1)
```

```
        patches_already_moved=patches_already_moved+patch_handles[j].
↪get_children()


    z = ax.axvline(longest, linestyle='-', color='black', alpha=.5,linewidth=1)
    z.set_zorder(-1)
#     print longest


    plt.xlim(-5, complete_longest+5)
    ymin=-1*len(df_conds_generals)-1
    plt.ylim(ymin,len(Qs)-1.5)

    xvalues = range(0, complete_longest,10)
    xlabels = []#[str(x-longest) for x in xvalues]
    plt.xticks(xvalues, xlabels)
    plt.xlabel('Percentage', fontsize=12,horizontalalignment='left')
    ax.xaxis.set_label_coords(float(longest)/(complete_longest+5),-0.01)

    general_Qs=[] if  len(df_conds_generals)==0 else df_conds_generals.index.
↪values.tolist() #+['']
    ylabels =general_Qs +Qs[1:]

    plt.yticks(yticks, ylabels)

    for tick in ax.yaxis.get_major_ticks():
        tick.label.set_fontsize(12)


    ## adding condition indicators on the y axis
    for cond,df_c in enumerate(df_conds):
        shift=SHIFT(len(df_conds),cond)
        for row in  range(0,len(df_c)):

            y=row+shift
            x=ax.get_xlim()[0]+0.5
#             x=ax.get_xlim()[0]+1.3
            ax.text(
                x,
                y-barwidth/4.0,
#                 str(cond),
#                 'C ' + str(cond+1),
                conds[cond],
                ha="center",
                fontdict=font2)
            ax.plot([x+0.7,df_c.iloc[row,0]],[y,y],linestyle=':', color='grey',␣
↪alpha=.2,linewidth=1)
```

```
    plt.grid('off')
    ax.spines['left'].set_visible(False)
    ax.spines['top'].set_visible(False)
    ax.spines['right'].set_visible(False)
#     print yticks
    ## adding Likert range legend
    for i,y_tick in enumerate(yticks):

        v=CustomLikertLabels_orderd_by_y_axis[i]
        x=-12

        y=yticks[i]-0.4
        ax.text(x,y, v[0],fontsize = 8,zorder = 6, color =␣
↪'white',horizontalalignment='right',
                    bbox={'edgecolor':'none','facecolor':likert_colors[1],␣
↪'alpha':1.0, 'pad':2})

        middle_colors=likert_colors[1:-1]
        for ci,c in enumerate(middle_colors):
            x=x+0.3
            ax.text(x,y,' ',fontsize = 8,zorder = 6, color =␣
↪'white',horizontalalignment='right',
                    bbox={'edgecolor':'none','facecolor':middle_colors[ci],␣
↪'alpha':1.0, 'pad':2})


        ax.text(x+0.2,y,v[1],fontsize = 8,zorder = 6, color =␣
↪'white',horizontalalignment='left',
                    bbox={'edgecolor':'none','facecolor':likert_colors[-1],␣
↪'alpha':1.0, 'pad':2})
    plt.tight_layout()
    plt.savefig('example.png')
    plt.show()
```

0.0.1 Plots using our algorithm

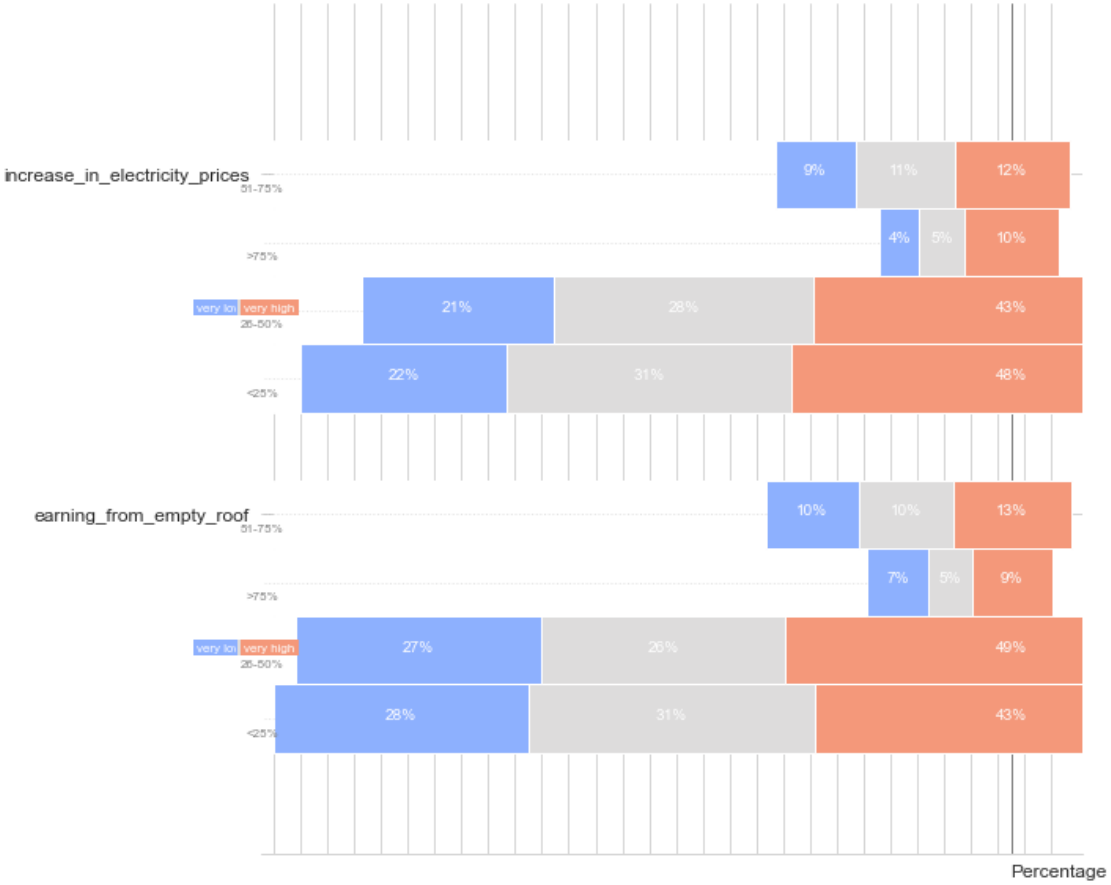
1. Percentage of Roof
2. Earning From Empty Roof
3. Increase in Electricity Prices

[111]:
```
df_plot_1 = pd.DataFrame([df_sorted_city.percentage_of_roof,
                          df_sorted_city.earning_from_empty_roof,
```

```
                                df_sorted_city.increase_in_electricity_prices]).
↪transpose()
```

[112]:
```
df_plot_1 = df_plot_1.rename(columns = {'percentage_of_roof': 'condition'})
```

[113]:
```
PlotLikertOverConditions(df_plot_1, 3)
```



Some insights from the above plots are:

(only considering highest percentage of people from each category)

1. Percentage of Roof vs Ratings for Increase in Electricity Prices

13% of the customers would consider utilizing **51% - 75% area of their roof** for installing solar plant in case of **Increase in Electricity Prices**.

10% of the customers would consider utilizing **greater than 75% area of their roof** for installing solar plant in case of **Increase in Electricity Prices**.

44% of the customers would consider utilizing **26% - 50% area of their roof** for installing solar plant in case of **Increase in Electricity Prices**.

47% of the customers would consider utilizing **less than 25% area of their roof** for installing solar plant in case of **Increasing Electricity Prices**.

2. Percentage of Roof vs Ratings for Earning from Empty Roof vs Ratings

13% of the customers would consider utilizing **51% - 75% area of their roof** for installing solar plants if there is enough scope for **Earning From Empty Roof**.

9% of the customers would consider utilizing **greater than 75% of their roof** for installing solar plants if there is enough scope for **Earning From Empty Roof**.

50% of the customers would consider utilizing **26% - 50% of their roof** for installing solar plants if there is enough scope for **Earning From Empty Roof**.

43% of the customers would consider utilizing **less than 25% of their roof** for installing solar plants if there is enough scope for **Earning From Empty Roof**.

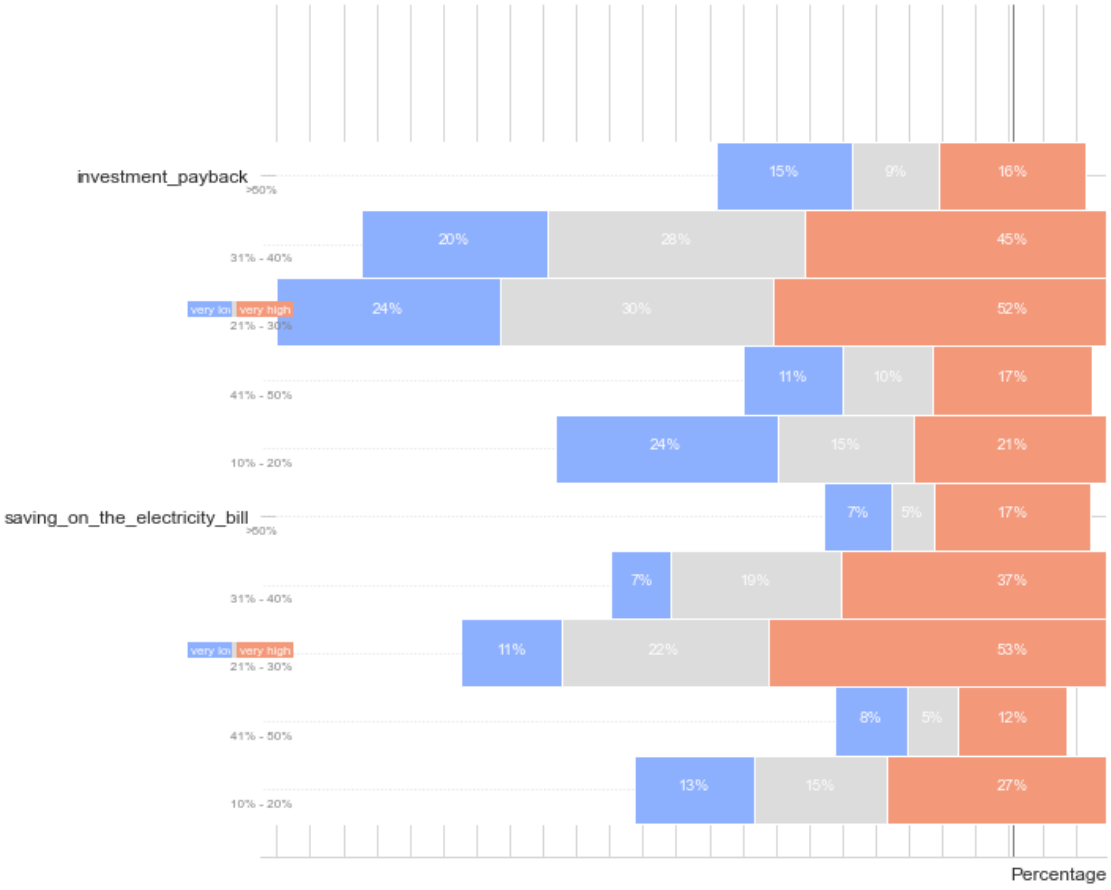
0.0.2 Plots using our algorithm

1. Expectation on Saving Electricity Bill
2. Saving on Electricity Bill
3. Investment Payback

[114]:
```
df_plot_2 = pd.DataFrame([df_sorted_city.expectation_on_saving_electricity_bill,
                          df_sorted_city.saving_on_the_electricity_bill,
                          df_sorted_city.investment_payback]).transpose()
```

[115]:
```
df_plot_2 = df_plot_2.rename(columns =␣
↪{'expectation_on_saving_electricity_bill': 'condition'})
```

[116]:
```
PlotLikertOverConditions(df_plot_2, 3)
```



Some insights from the above plots are:

(only considering highest percentage of people from each category)

1. Expected Percentage of Saving on Electricity Bill vs Ratings for Investment Payback

16% of the customers who expect to save **more than 50% on electricity bill** would consider guranteed **Investment Payback** as an important factor for investing in solar technology.

46% of the customers who expect to save between **31% - 40% on electricity bill** would consider guranteed **Investment Payback** as an important factor for investing in solar technology.

53% of the customers who expect to save **21% - 30% on electricity bill** would consider guranteed **Investment Payback** as an important factor for investing in solar technology.

18% of the customers who expect to save **41% - 50% on electricity bill** would consider guranteed **Investment Payback** as an important factor for investing in solar technology.

22% of the customers who expect to save **10% - 20% on electricity bill** would consider guranteed **Investment Payback** as an important factor for investing in solar technology.

2. Expected Percentage of Saving on Electricity Bill vs Ratings for Saving on Electricity Bill

18% of the customers who expect to save **more than 50% on electricity bill** consider **Saving on Electricity Bill** as an important factor for investing in solar technology.

37% of the customers who expect to save **31% - 40% on electricity bill** consider **Saving on Electricity Bill** as an important factor for investing in solar technology.

51% of the customers who expect to save **21% - 30% on electricity bill** consider **Saving on Electricity Bill** as an important factor for investing in solar technology.

13% of the customers who expect to save **41% - 50% on electricity bill** consider **Saving on Electricity Bill** as an important factor for investing in solar technology.

28% of the customers who expect to save **10% - 20% on electricity bill** consider **Saving on Electricity Bill** as an important factor for investing in solar technology.

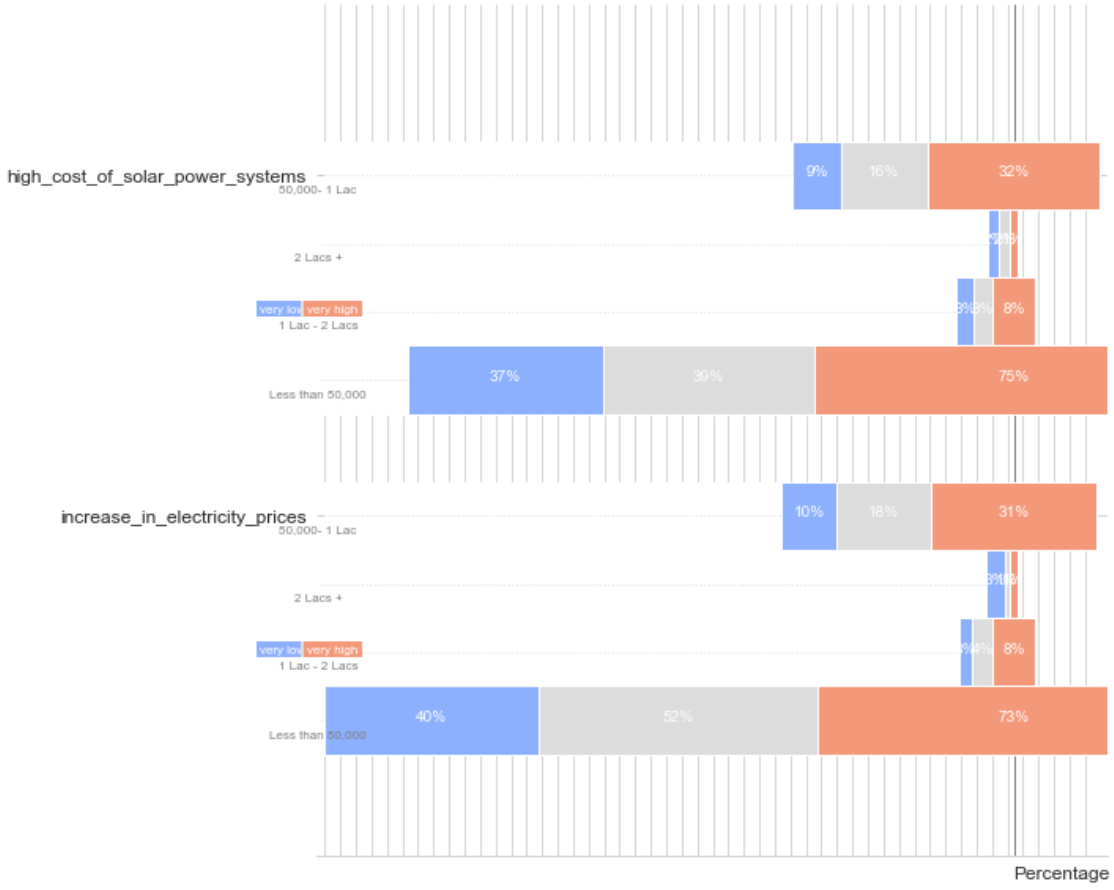
0.0.3 Plots using our algorithm

1. Maximum Investment in Solar
2. Increase in Electricity Prices
3. High Cost of Solar Power Systems

[117]:
```
df_plot_3 = pd.DataFrame([df_sorted_city.maximum_investment_in_solar,
                          df_sorted_city.increase_in_electricity_prices,
                          df_sorted_city.high_cost_of_solar_power_systems]).
↪transpose()
```

[118]:
```
df_plot_3 = df_plot_3.rename(columns = {'maximum_investment_in_solar':␣
↪'condition'})
```

[119]:
```
PlotLikertOverConditions(df_plot_3, 3)
```



Some insights from the above plots are:

(only considering highest percentage of people from each category)

1. Maximum Investment in Solar vs Ratings for Increase in Electricity Prices

33% of the customers would **invest 50,000 - 1 Lac** in Solar Technology but consider **High Cost of Solar Power Systems** as a factor for not investing in solar technology.

8% of the customers would **invest 1 Lac - 2 Lacs** in Solar Technology but consider **High Cost of Solar Power Systems** as an factor for not investing in solar technology.

74% of the customers would **invest less than 50,000** in Solar Technology but consider **High Cost of Solar Power Systems** as a factor for not investing in solar technology.

2. Maximum Investment in Solar vs Ratings for High Cost of Solar Power Systems

32% of the customers would **invest 50,000 - 1 Lac** in Solar Technology and consider **Increase in Electricity Prices** for their investment decision.

8% of the customers would **invest 1 Lac - 2 Lacs** in Solar Technology and consider **Increase in Electricity Prices** for their investment decision.

72% of the customers would **invest less than 50,000** in Solar Technology and consider **Increase in Electricity Prices** for their investment decision.

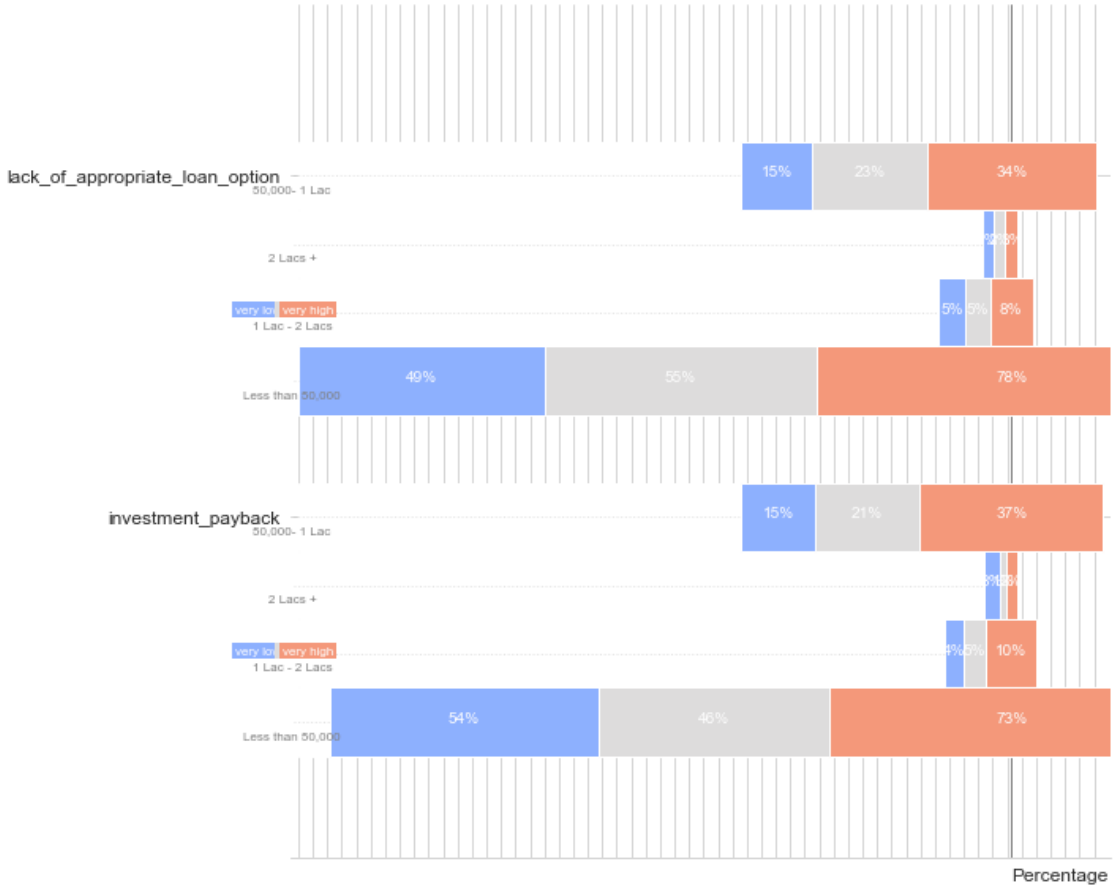
0.0.4 Plot for

1. Maximum Investment in Solar
2. Investment Payback
3. Lack of Appropriate Loan Options

[120]:
```
df_plot_4 = pd.DataFrame([df_sorted_city.maximum_investment_in_solar,
                          df_sorted_city.investment_payback,
                          df_sorted_city.lack_of_appropriate_loan_option]).
↪transpose()
```

[121]:
```
df_plot_4 = df_plot_4.rename(columns = {'maximum_investment_in_solar':␣
↪'condition'})
```

[122]:
```
PlotLikertOverConditions(df_plot_4, 3)
```



Some insights from the above plots are:

(only considering highest percentage of people from each category)

1. Maximum Investment in Solar vs Ratings for Lack of Appropriate Loan Options

34% of the customers with **maximum investment between 50,000 - 1 Lac** in Solar Technology will tend to show less interest in Solar Technology if there are no **Appropriate Loan Options**.

9% of the customers with **maximum investment between 1 Lac - 2 Lac** in Solar Technology will tend to show less interest in Solar Technology if there are no **Appropriate Loan Options**.

79% of the customers with **less than 50,000** in Solar Technology will tend to show less interest in Solar Technology if there are no **Appropriate Loan Options**.

2. Maximum Investment in Solar vs Ratings for Investment Payback

38% of the customers would **invest 50,000 - 1 Lac** in Solar Technology if there is guranteed **Investment Payback**.

11% of the customers would **invest 1 Lac - 2 Lac** in Solar Technology if there is guranteed **Investment Payback**.

38% of the customers would **invest less than 50,000** in Solar Technology if there is guranteed **Investment Payback**.

Conclusion

From our Exploratory Data Analysis on the DataSet we have found these insights:

Cities for Expanding Business

These cities have given maximum business to the company

S. No.	City	State	Projects Completed
1	Bangalore	Karnataka	171
2	Hyderabad	Telangana	135
3	Lucknow	Uttar Pradesh	108
4	Pune	Maharashtra	98
5	Visakhapatnam	Andhra Pradesh	64
6	Ghaziabad	Uttar Pradesh	57
7	Nagpur	Maharashtra	53
8	Gwalior	Madhya Pradesh	46
9	Indore	Madhya Pradesh	49
10	Rajkot	Gujarat	45
11	Jaipur	Rajasthan	44
12	Kanpur	Uttar Pradesh	41
13	Agra	Uttar Pradesh	44
14	Bhopal	Madhya Pradesh	43
15	Noida	Uttar Pradesh	41
16	Junagadh	Gujarat	34

S. No.	City	State	Projects Completed
17	Varanasi	Uttar Pradesh	33
18	Meerut	Uttar Pradesh	33
19	Nashik	Maharashtra	28
20	Vadodara	Gujarat	27
21	Moradabad	Uttar Pradesh	25
22	Ranchi	Jharkhand	25
23	Kota	Rajasthan	23
24	Jamnagar	Gujarat	19
25	Bareilly	Uttar Pradesh	16
26	Thane	Maharashtra	13
27	Gorakhpur	Uttar Pradesh	13
28	Belgaum	Karnataka	13
29	Panvel	Maharashtra	13
30	Basti	Uttar Pradesh	12
31	Surat	Gujarat	12

Summary of Findings

```
1. Peacock Solar has already completed a good number of projects in these cities
2. Customers are comfortale in making payment through these mediums:
    a. Cash
    b. Online
    c. Debit / Credit Card
3. Maximum customers donot have a Homeloan
4. Maximum customers donot have an Active EMI
5. Almost all the customers in these cities:
    a. Are aware about Solar Technology
    b. Consider Solar Plants as a Source for Power Backup
    c. Would invest in Solar Technology due to Increase in Electricity Prices
    d. Would invest in Solar Technology for Saving on the Electricity Bill
    e. Do not consider Life Span of Solar Plant as a factor for not investing in Solar Technolo
    f. Show Interest in new Technology
    g. Would invest in Solar Technology for being Environment Friendly
    h. Would not invest in Solar Technology for just Status Symbol
    i. Would consider investing in Solar Technology if it has a scope for Earning from Empty Ro
```

[ ]: