

In [15]:

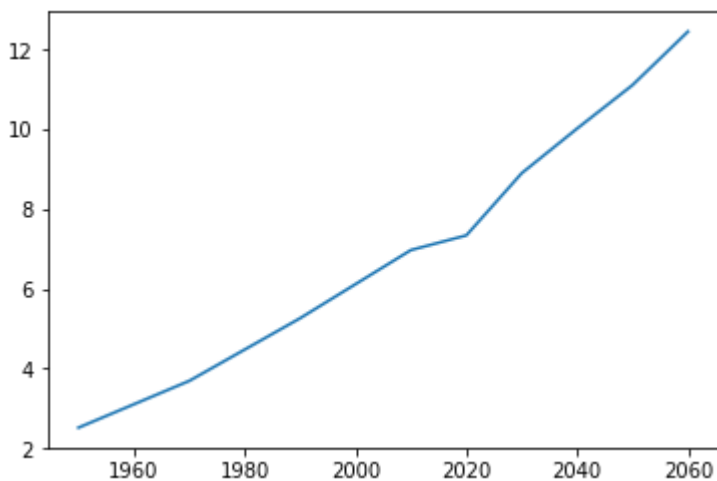
```
year = [1950, 1970, 1990, 2010, 2020, 2030, 2040, 2050, 2060]  
pop = [2.519, 3.692, 5.263, 6.972, 7.34, 8.9, 10.02, 11.11, 12.45]
```

```
# Print the last item from year and pop  
print(year[-1])  
print(pop[-1])
```

```
2060  
12.45
```

In [16]:

```
# Import matplotlib.pyplot as plt  
import matplotlib.pyplot as plt  
  
# Make a line plot: year on the x-axis, pop on the y-axis  
plt.plot(year, pop)  
  
# Display the plot with plt.show()  
plt.show()
```



In [17]:

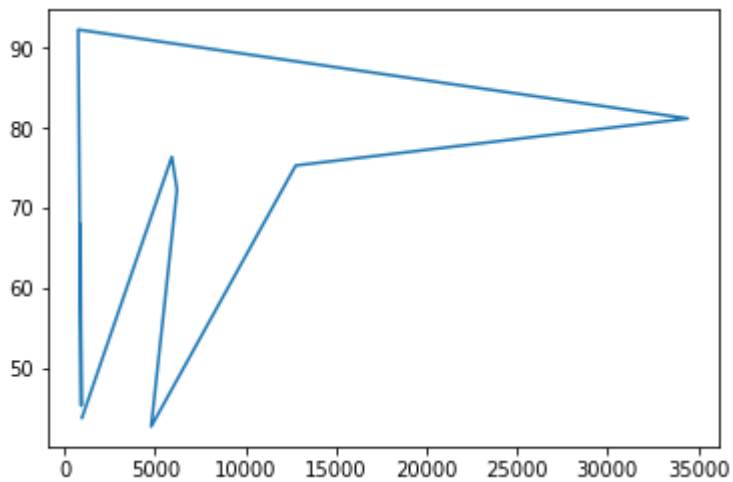
```
gdp_cap = [974.5, 5937.02, 6223.36, 4797.23, 12779.37, 34435.36, 765.34, 934.45, 892.34]
life_exp = [43.8, 76.4, 72.3, 42.7, 75.32, 81.2, 92.3, 45.3, 67.9]
# Print the last item of gdp_cap and life_exp
print(gdp_cap[-1])
print(life_exp[-1])

# Make a line plot, gdp_cap on the x-axis, life_exp on the y-axis
plt.plot(gdp_cap, life_exp)

# Display the plot
plt.show()
```

892.34

67.9

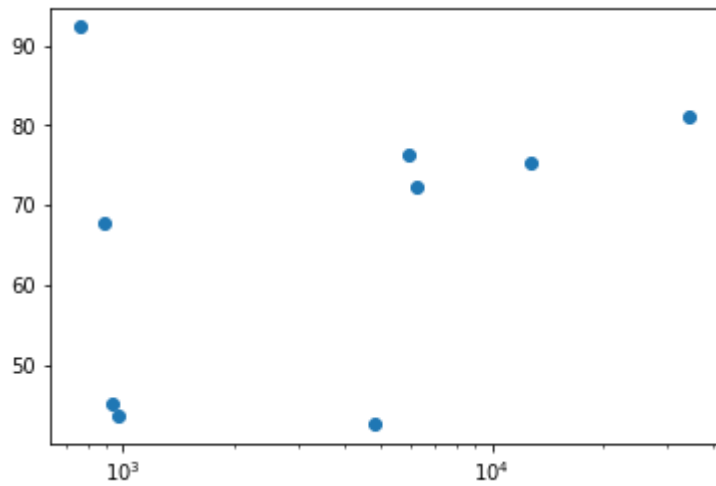


In [18]:

```
# Change the line plot below to a scatter plot
plt.scatter(gdp_cap, life_exp)

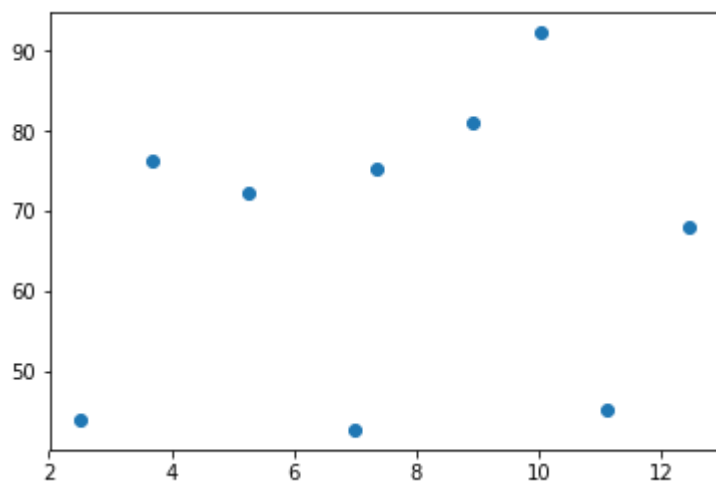
# Put the x-axis on a logarithmic scale
plt.xscale('log')

# Show plot
plt.show()
```



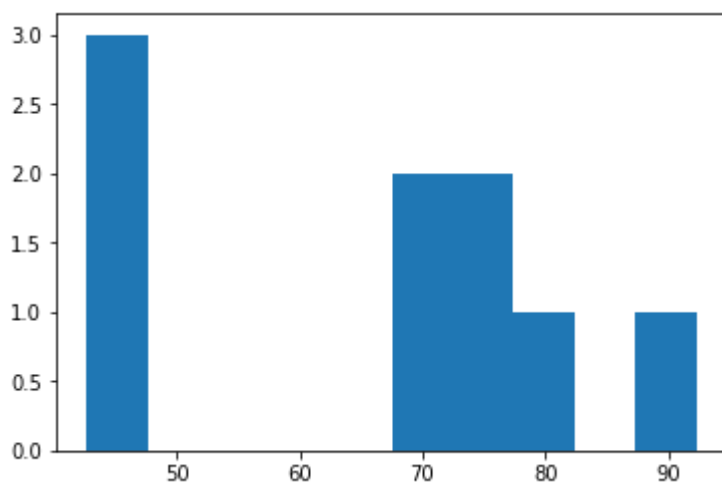
In [19]:

```
# Build Scatter plot  
plt.scatter(pop,life_exp)  
  
# Show plot  
plt.show()
```



In [20]:

```
# Create histogram of life_exp data  
plt.hist(life_exp)  
  
# Display histogram  
plt.show()
```



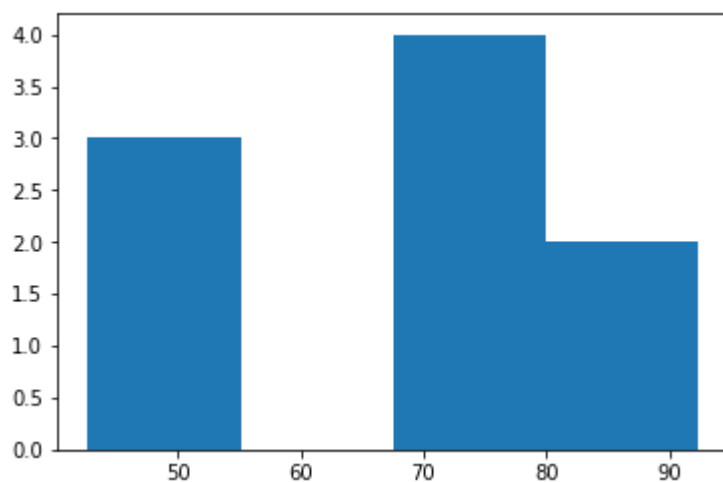
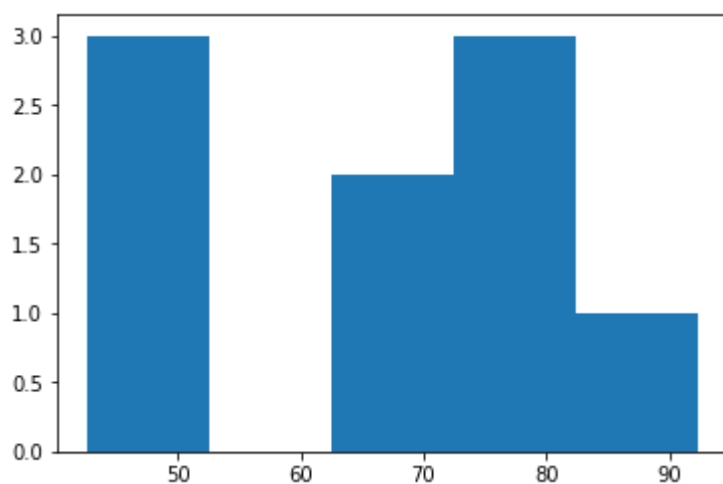
In [26]:

```
# Build histogram with 5 bins
plt.hist(life_exp,bins=5)

# Show and clean up plot
plt.show()
plt.clf()

# Build histogram with 4 bins
plt.hist(life_exp,bins=4)

# Show and clean up again
plt.show()
plt.clf()
```



<Figure size 432x288 with 0 Axes>

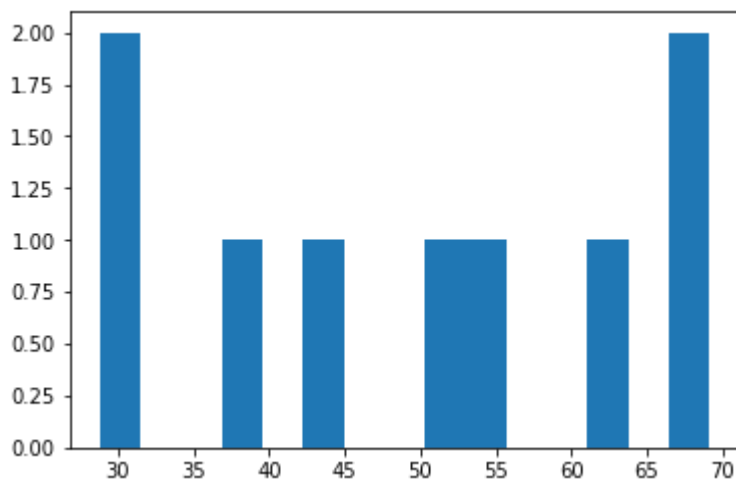
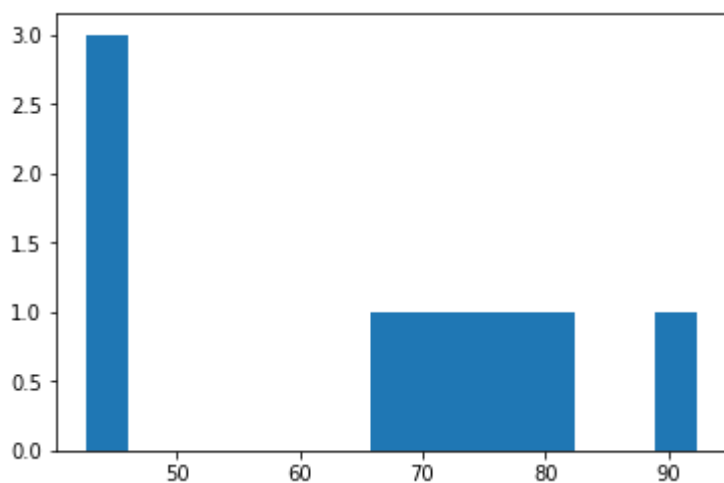
In [25]:

```
life_exp1950 = [28.8, 55.23, 43.08, 30.02, 62.48, 69.12, 66.8, 50.94, 37.48]
# Histogram of life_exp, 15 bins
plt.hist(life_exp, bins=15)

# Show and clear plot
plt.show()
plt.clf()

# Histogram of life_exp1950, 15 bins
plt.hist(life_exp1950, bins=15)

# Show and clear plot again
plt.show()
plt.clf()
```



<Figure size 432x288 with 0 Axes>

In [27]:

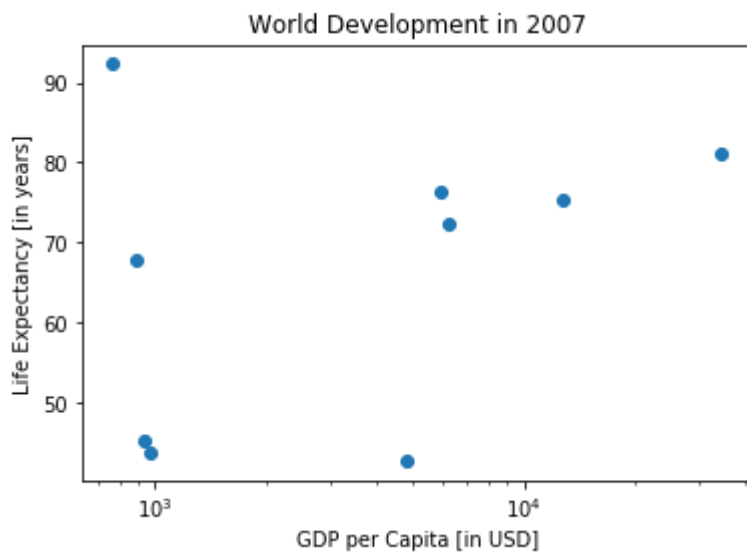
```
# Basic scatter plot, log scale
plt.scatter(gdp_cap, life_exp)
plt.xscale('log')

# Strings
xlab = 'GDP per Capita [in USD]'
ylab = 'Life Expectancy [in years]'
title = 'World Development in 2007'

# Add axis labels
plt.xlabel(xlab)
plt.ylabel(ylab)

# Add title
plt.title(title)

# After customizing, display the plot
plt.show()
```



In [28]:

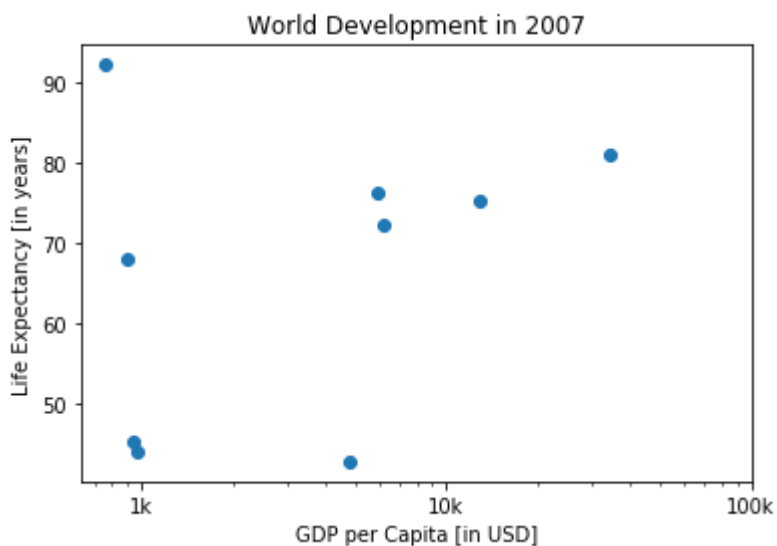
```
# Scatter plot
plt.scatter(gdp_cap, life_exp)

# Previous customizations
plt.xscale('log')
plt.xlabel('GDP per Capita [in USD]')
plt.ylabel('Life Expectancy [in years]')
plt.title('World Development in 2007')

# Definition of tick_val and tick_lab
tick_val = [1000, 10000, 100000]
tick_lab = ['1k', '10k', '100k']

# Adapt the ticks on the x-axis
plt.xticks(tick_val, tick_lab)

# After customizing, display the plot
plt.show()
```



In [29]:

```
# Import numpy as np
import numpy as np

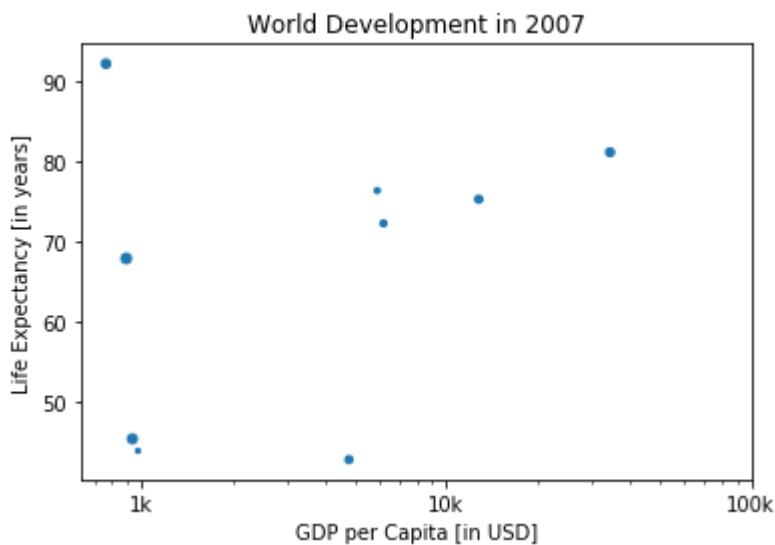
# Store pop as a numpy array: np_pop
np_pop = np.array(pop)

# Double np_pop
np_pop = np_pop*2

# Update: set s argument to np_pop
plt.scatter(gdp_cap, life_exp, s = np_pop)

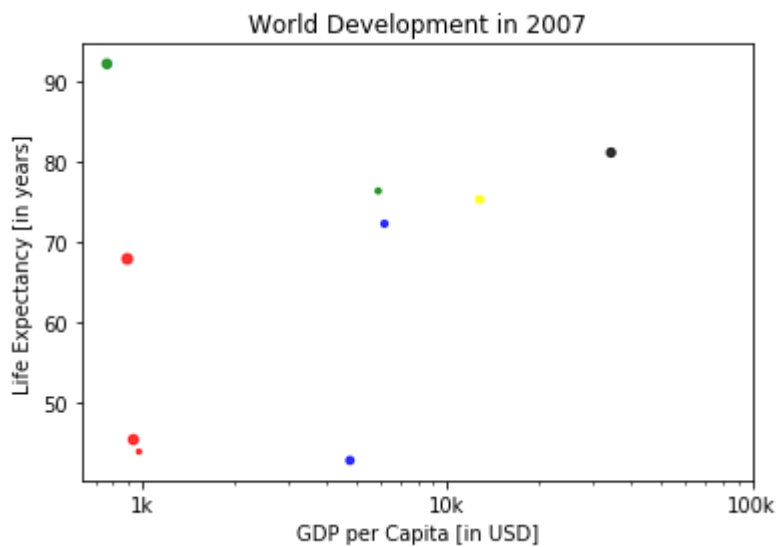
# Previous customizations
plt.xscale('log')
plt.xlabel('GDP per Capita [in USD]')
plt.ylabel('Life Expectancy [in years]')
plt.title('World Development in 2007')
plt.xticks([1000, 10000, 100000],['1k', '10k', '100k'])

# Display the plot
plt.show()
```



In [31]:

```
col = ['red', 'green', 'blue', 'blue', 'yellow', 'black', 'green', 'red', 'red']  
# Specify c and alpha inside plt.scatter()  
plt.scatter(x = gdp_cap, y = life_exp, s = np.array(pop) * 2, c=col, alpha=0.8)  
  
# Previous customizations  
plt.xscale('log')  
plt.xlabel('GDP per Capita [in USD]')  
plt.ylabel('Life Expectancy [in years]')  
plt.title('World Development in 2007')  
plt.xticks([1000, 10000, 100000], ['1k', '10k', '100k'])  
  
# Show the plot  
plt.show()
```



In [34]:

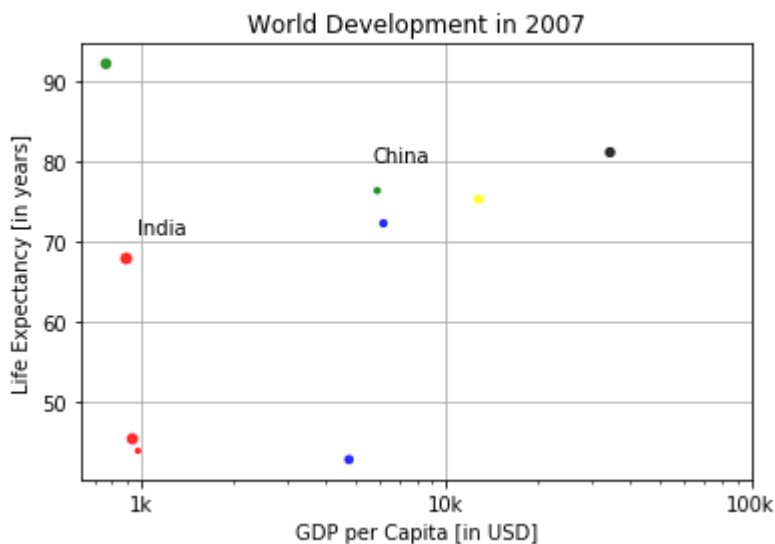
```
# Scatter plot
plt.scatter(x = gdp_cap, y = life_exp, s = np.array(pop) * 2, c = col, alpha = 0.8)

# Previous customizations
plt.xscale('log')
plt.xlabel('GDP per Capita [in USD]')
plt.ylabel('Life Expectancy [in years]')
plt.title('World Development in 2007')
plt.xticks([1000,10000,100000], ['1k', '10k', '100k'])

# Additional customizations
plt.text(970, 71, 'India')
plt.text(5700, 80, 'China')

# Add grid() call
plt.grid(True)

# Show the plot
plt.show()
```



In []: