Name: Mahad Hameed

Roll no: 20K-0338

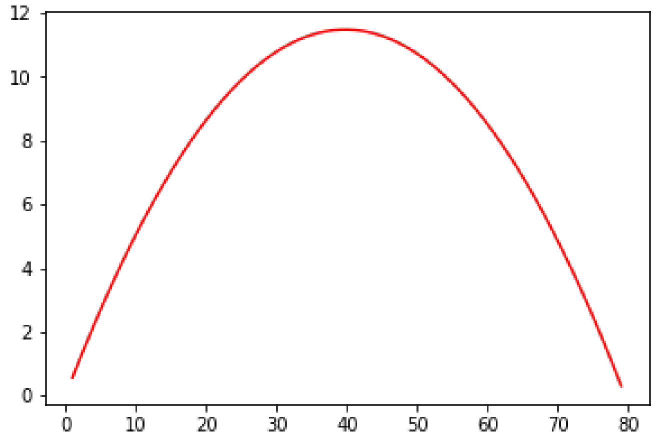
Projectile Motion without making Custom Functions

The equation of Projectile's trajectory is

$$y = tan\theta . x - \frac{g}{2.u^2.cos^2\theta} . x^2$$

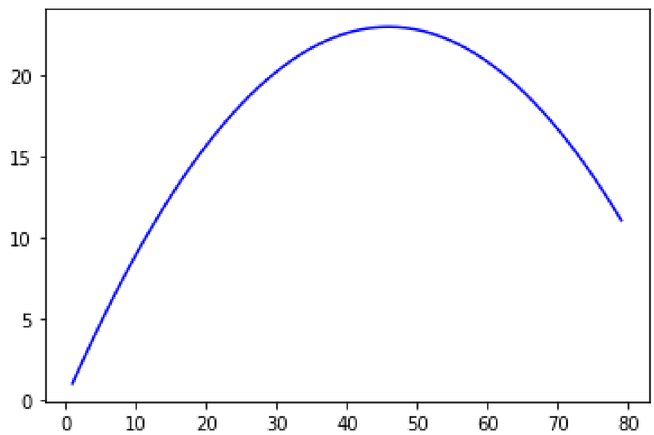
```
import matplotlib.pyplot as plt
import numpy as np
import math
x = np.arange(1,80, 1)
g = 9.8
v0 = 30
theta = math.radians(30)
# The equation of Projectile's trajectory is :
y = x* math.tan(theta)-(x**2 * g)/(2 * v0**2 * (math.cos(theta)**2))
plt.plot(x,y,'r')
```

```
[<matplotlib.lines.Line2D at 0x7f69032dfba8>]
```




```
theta1 = math.radians(45)
y1 = x* math.tan(theta1)-(x**2 * g)/(2 * v0**2 * (math.cos(theta1)**2))
plt.plot(x,y1,'b')
```

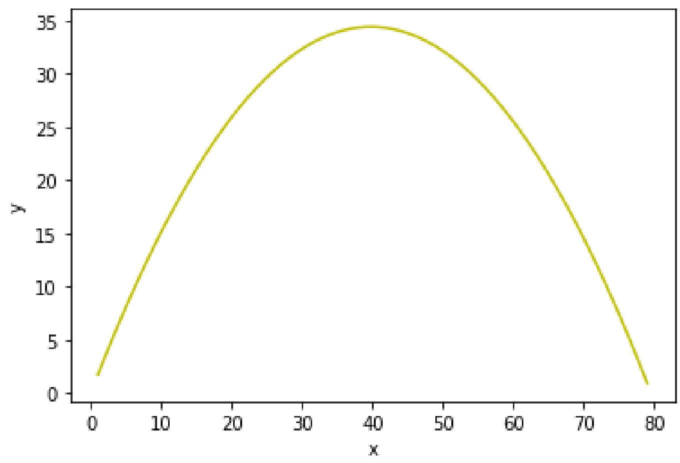
```
[<matplotlib.lines.Line2D at 0x7f69032f90b8>]
```



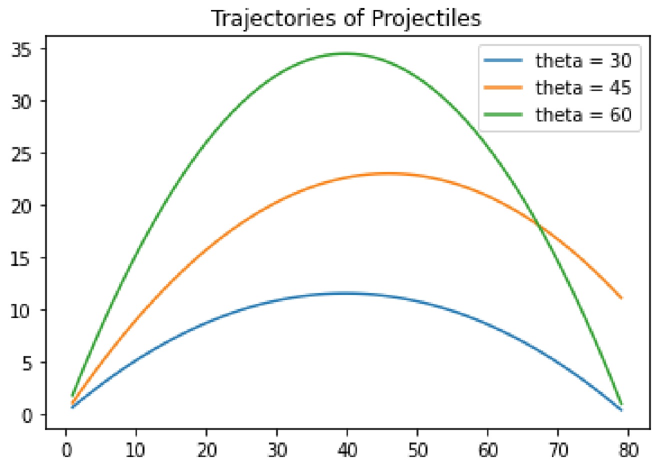

```
theta2 =math.radians(60)
y2 = x * math.tan(theta2)-(x**2 * g)/(2 * v0**2 * (math.cos(theta2)**2))
plt.plot(x,y2,'g')
```

```
plt.xlabel('x')
plt.ylabel('y')
```

```
Text(0, 0.5, 'y')
```



```
ax = plt.subplot(111)
ax.plot(x, y, label='theta = 30')
ax.plot(x, y1, label='theta = 45')
ax.plot(x, y2, label='theta = 60')
plt.title('Trajectories of Projectiles')
ax.legend()
plt.show()
```



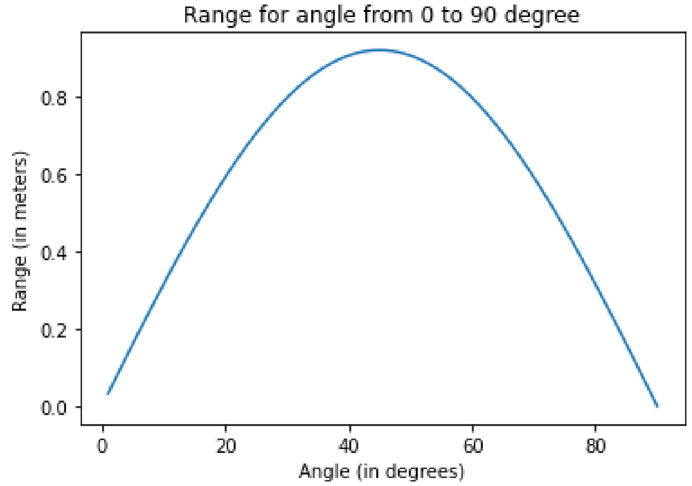
Range and Height of projectile

Range=$v^2 . \frac{sin2\theta}{g}$

Height=$v^2 . \frac{sin^2\theta}{2g}$

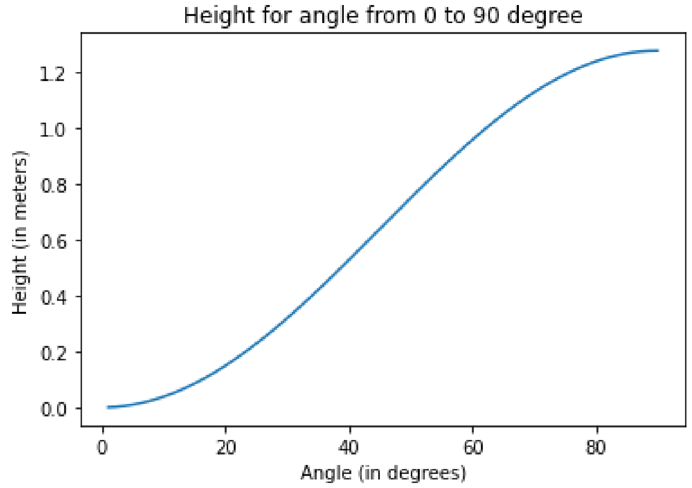
```
import math
def Range(angle):
 v = 3
 R = v**2 *(np.sin(np.radians(2*angle)))/ g
 return R
angle = np.arange(1, 90, 0.01)
plt.plot(angle, Range(angle))
plt.xlabel('Angle (in degrees)')
plt.ylabel('Range (in meters)')
plt.title ('Range for angle from 0 to 90 degree')
```

```
Text(0.5, 1.0, 'Range for angle from 0 to 90 degree')
```




```
def Height(angle):
 v = 5
 return v**2 *(np.sin(np.radians(angle))**2)/ (g*2)
angle = np.arange(1, 90, 0.01)
plt.plot(angle,Height(angle))
plt.xlabel('Angle (in degrees)')
plt.ylabel('Height (in meters)')
plt.title ('Height for angle from 0 to 90 degree')
```

```
Text(0.5, 1.0, 'Height for angle from 0 to 90 degree')
```



Projectile motion by making custom functions

```
import math
def Range(angle,in_vel):
 R = in_vel**2 *(np.sin(np.radians(2*angle)))/ g
 return R
# Height of projectile
def Height(angle,in_vel):
 return in_vel**2 *(np.sin(np.radians(angle))**2)/ (g*2)
Height(45,3), Range(45,3)
```

```
(0.22959183673469383, 0.9183673469387754)
```