In [227]:
```
## q1
def calculation(x,y):
    add=x+y
    sub=x-y
    return(add,sub)
calculation(1,3)
```

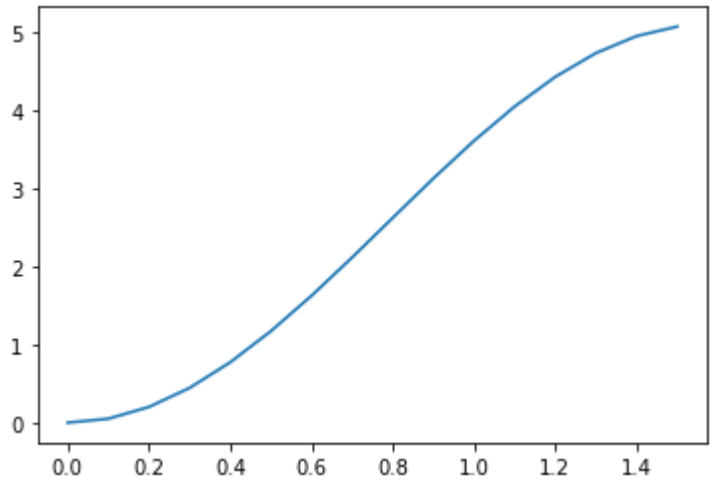
Out[227]: (4, -2)

In [238]:
```
## q2
import numpy as np
def projectile(v,a):
    b=np.radians(a)
    maximumHeight=((v*v) * ((np.sin(b)*np.sin(b))/(2*9.81)))
    maximumRange=v*v*(np.sin(2*b))/9.81
    return(maximumHeight,maximumRange)

projectile(10,45)
```

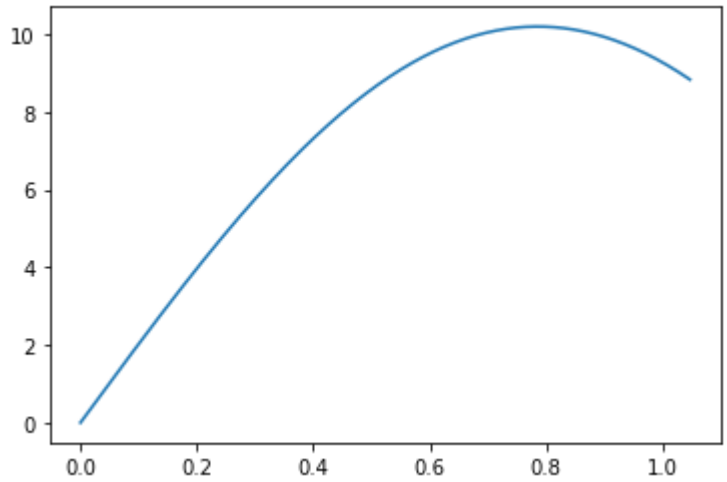
Out[238]: (2.5484199796126408, 10.19367991845056)

In [248]:
```
import matplotlib.pyplot as plt
import numpy as np
x= np.arange(0,np.pi/2,0.01 )
y=maximumHeight=((10*10) * ((np.sin(x)*np.sin(x))/(2*9.81)))
plt.plot(x,y)
plt.show()
print("graph for height variation")
```



graph for heighht variation

In [254]:
```
import matplotlib.pyplot as plt
import numpy as np
x= np.arange(0,np.pi/3,0.0001 )
y=maximumRange=10*10*(np.sin(2*x))/9.81
plt.plot(x,y)
plt.show()
print("graph for range variation")
```

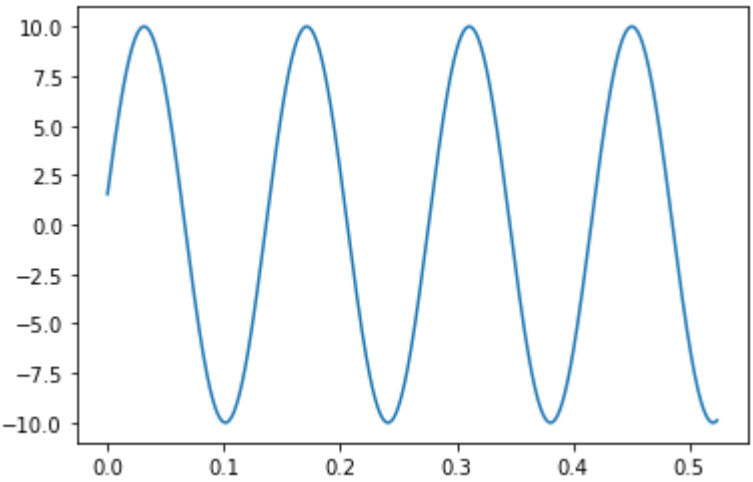



```
graph for range variation
```

In [235]:
```
##q3
import numpy as np
def SHM(amp,angfreq,phi,t):
    angfreq=np.radians(angfreq)
    phi=np.radians(phi)
    x=amp*np.cos((angfreq*t)+phi)
    v=-amp*angfreq*(np.sin((angfreq*t)+phi))
    a=amp*angfreq*angfreq*(np.cos((angfreq*t)+phi))
    return(x,v,a)
SHM(10,45,30,4)
```

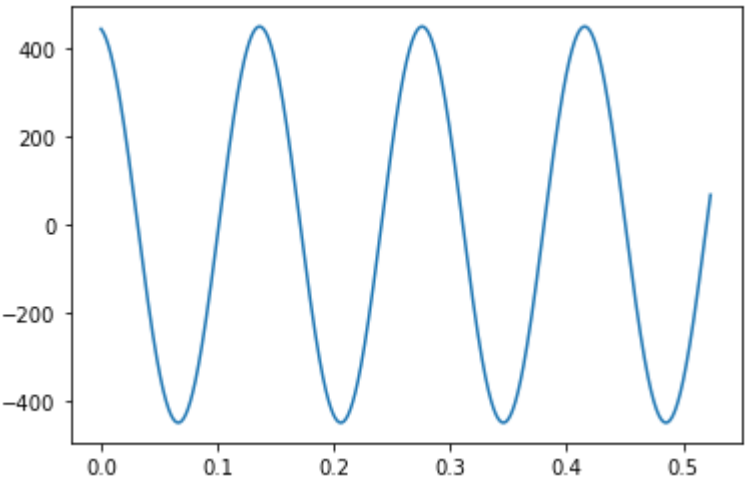
Out[235]:
```
(-8.660254037844389, 3.926990816987239, -5.342080085403803)
```

In [263]:
```
import matplotlib.pyplot as plt
import numpy as np
x= np.arange(0,np.pi/6,0.0001 )
y=10*np.cos((45*x)+30)
plt.plot(x,y)
plt.show()
print("displacement varying with time ")
```



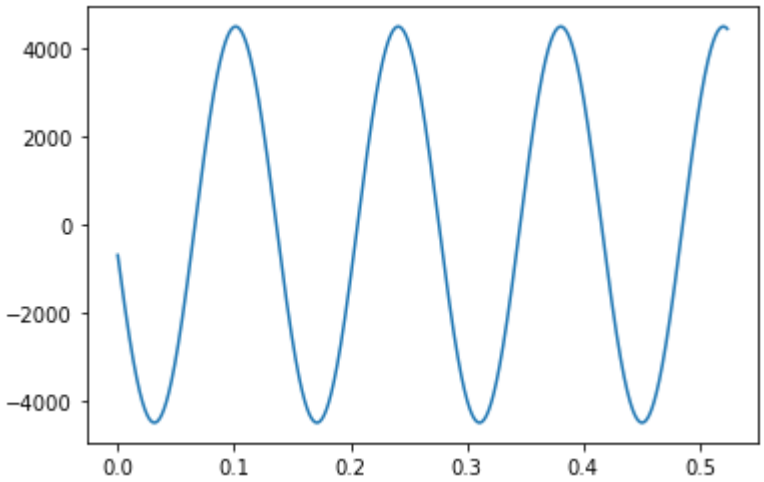
displacement varying with time

In [261]:
```
import matplotlib.pyplot as plt
import numpy as np
x= np.arange(0,np.pi/6,0.0001 )
y=-10*45*(np.sin((45*x)+30))
plt.plot(x,y)
plt.show()
print("velocity varying with time ")
```



velocity varying with time

In [260]:
```
import matplotlib.pyplot as plt
import numpy as np
x= np.arange(0,np.pi/6,0.0001 )
y=-10*45*10*(np.cos((45*x)+30))
plt.plot(x,y)
plt.show()
print("acceleration varying with time ")
```




```
acceleration varying with time
```

In [264]:
```
##q4
import numpy as np
sampleArray = np.arange(100, 200, 10)
sampleArray = sampleArray.reshape(5,2)
print (sampleArray)
```

```
[[100 110]
 [120 130]
 [140 150]
 [160 170]
 [180 190]]
```

In [270]:
```
##q5
import numpy as np
sarray=numpy.array([[34,43,73],[82,22,12],[53,94,66]])
print("here is a sample array",sarray)
b=np.delete(sarray,1,1)
print("here is the second column erased",b)
newcol=np.array([[10]])
newcol
c=np.insert(b,1,newcol,axis=1)
print("here we have 10 added into 1st column in all rows"
      ,c)
```

```
here is a sample array [[34 43 73]
 [82 22 12]
 [53 94 66]]
here is the second column erased [[34 73]
 [82 12]
 [53 66]]
here we have 10 added into 1st column in all rows [[34 10 73]
 [82 10 12]
 [53 10 66]]
```

In [139]:
```
##Q8a
import numpy as np
d=30
d=np.radians(d)
xcomp=7.3*np.cos(d)
ycomp=7.3*np.sin(d)
print("answer for a part a is " ,xcomp,"answer for a part b is ",ycomp)
```

```
answer for a part a is  6.321985447626402 answer for a part b is  3.649999999
9999995
```

In [159]:
```
##q8 b
import numpy as np
import math
aa=np.radians(30)
ab=np.radians(75)
ax=10*np.cos(aa)
ay=10*np.sin(aa)
bx=-10*np.cos(ab)
by=10*np.sin(ab)
rx=ax+bx
ry=ay+by
print("A)x component of r is",rx,"and y component is",ry)
##q8b b
r=np.sqrt(rx**2+ry**2)
print("B)The magnitude of r is ",r)
##q8 b c
angle=math.atan(ry/rx)
print("C)Angle r makes with postive x axis is ",angle,"in radians")
```

```
A)x component of r is 6.07206358681918 and y component is 14.659258262890681
B)The magnitude of r is  15.867066805824702
C)Angle r makes with postive x axis is  1.1780972450961724 in radians
```

In [207]:
```
##q8c
import numpy as np
import math
aa=math.radians(30)
ab=math.radians(195)
ac=math.radians(315)

ax=50*np.cos(aa)
ay=50*np.sin(aa)
bx=50*np.cos(ab)
by=50*np.sin(ab)
cx=50*np.cos(ac)
cy=50*np.sin(ac)
rx=ax+bx+cx
ry=ay+by+cy
r=np.sqrt((rx**2)+(ry**2))
ar=math.atan(ry/rx)
print(" ci)the magnitude of a+b+c=",r,"and the angle made with +ve x axis is",
ar,"radians")
rx2=ax-bx+cx
ry2=ay-by+cy
rr=np.sqrt((rx2**2)+(ry2**2))
ar2=math.atan(ry2/rx2)
print(" cii)the magnitude of a-b+c=",rr,"and the angle made with +ve x axis i
s",ar2,"radians")
rx3=ax+bx-cx
ry3=ay+by-cy
rrr=np.sqrt((rx3**2)+(ry3**2))
ar3=math.atan(ry3/rx3)
print(" ciii)the magnitude of a+b-c=d",rrr,"and the angle made with +ve x axis
is",ar3,"radians")
```

```
 ci)the magnitude of a+b+c= 38.26834323650899 and the angle made with +ve x a
xis is -0.6544984694978738 radians
 cii)the magnitude of a-b+c= 126.97922805308725 and the angle made with +ve x
axis is 0.020363896898836234 radians
 ciii)the magnitude of a+b-c=d 62.25974335830541 and the angle made with +ve
x axis is -0.8657132268937776 radians
```

In [219]:
```
##q8d
import numpy as np
import math
mag=np.sqrt((2**2)+(3**2)+(5**2))
theta=math.acos(2/(mag))
print("angle in radians with x axis is",theta)
theta1=math.acos(-3/mag)
print("angle in radians with y axis is",theta1)
theta2=math.acos(5/mag)
print("angle in radians with z axis is",theta2)
```

```
angle in radians with x axis is 1.2403736788834132
angle in radians with y axis is 2.079063572966181
angle in radians with z axis is 0.6247538687650431
```

In [272]:
```
##q8e
import numpy as np
import math
adotb=(5*-2)+(4*2)+(-6*3)
maga=np.sqrt(5**2+4**2+6**2)
magb=np.sqrt(2**2+2**2+3**2)
theta=math.acos((adotb)/(maga*magb))
print("angle between a and b is ",theta,"radians")
rx=5+-2+4
ry=4+2+3
rz=-6+3+2
magr=np.sqrt(rx**2+ry**2+rz**2)
angle=math.acos(rz/magr)
print('angle r makes with z axis is ',angle,"radians")
```

```
angle between a and b is  2.1565049037442687 radians
angle r makes with z axis is  1.658278274323415 radians
```

In [ ]: