

```
# There are several operations more complicated than simple arithmetic,  
# such as logarithm, multiplying matrices, making graphs.  
# To do these things in Python, we need to call certain "packages".
```

```
print("-----")  
print("FUNCTIONS and PACKAGES")  
print("-----")  
print()
```

```
#####  
# NATURAL LOGARITHM  
#####  
# Tell the computer to import the logarithm function from the math package  
from math import log  
# Once the function has been imported,  
# it becomes available until the program ends.
```

```
# To compute the NATURAL logarithm (ln)  
x = log(3.5)  
print("ln(3.5) = ",x)
```

```
#####  
# LOG in the BASIS 10  
#####  
# To compute the logarithm in basis 10  
from math import log10  
x = log10(3.5)  
print("log10(3.5) = ",x)
```

```
#####  
# PI  
#####  
# You can import a list from a package  
from math import exp,sin,cos,sqrt,pi,e  
x = 17.4*pi  
print("17.4*Pi = ",x)
```

```
#####  
# TRIGO FUNCTIONS  
#####  
# CAREFUL! The trigonometric functions work with angles in RADIANS  
# Remember that 360 degrees = 2*Pi radians  
# So the angle "x" in degrees is equal to x*2*Pi/360 or x*Pi/180  
# Write 60 degrees in radians  
x = 60.*pi/180.  
x = cos(x)  
print("Cos[60 degrees] = ", x)
```

```
#####
# IMPORT ALL
#####
# You can also import ALL of the functions by using "*", as in
from math import *
# CAREFUL!! If you happen to give to a variable the same name
# of an existing function, unexpected things can happen!!
# To avoid it, either import only what you need
# or use long and not so familiar names for the variables.
```

```
#####
#####
# IMPORT MATH
#####
# Another option is to use the function only when called
import math
```

```
x = math.log(3.5)
print("ln(3.5) = ", x)
```

```
x_pi = math.pi
print("pi = ", x_pi)
```

```
angle = math.radians(60.)
x = math.cos(angle)
print("cos(60 degrees) = ", x)
```

```
#####
```

```
#####
# MODULES
#####
# Some large packages can be split into subpackages called "modules"
# They are referred to as "packagename.modulename"
# The package called "numpy" has many modules, such as
# numpy.linalg, which is a linear algebra module
```

```
# We can import a function from a particular module,
# such as "inv" from numpy.linalg,
# which we use to invert a matrix.
```

```
# This is done as
# from numpy.linalg import inv
# We will see more about this later.

# Small packages, such as the "math" package used above,
# do not have modules.

print()
print("-----")
print("EXAMPLE 2.2")
print("-----")

# Use r=2 and the angle theta=60 degrees
r = 2.
theta = 60. # Do not forget to convert it to RADIANS!
theta = theta*pi/180.

# Convert r and theta to Cartesian coordinates using
#  $x = r \cos(\theta)$  and  $y = r \sin(\theta)$ 
x = r*cos(theta)
y = r*sin(theta)
print("x =",x," y=",y)
```