```
#########
# LISTS
########
# Lists are lists of something, such as numbers.
# The elements of the list can be of different types.
# The length of the list can be changed.

##########
# ARRAYS
#########
# We will use arrays more than lists.
# Arrays also provide a list of things.
#
# BUT......
# Arrays are FASTER
# The elements can only be of the SAME type.
# The length CANNOT be changed
#
# Arrays are used as VECTORS and MATRICES
# Arrays can be two-dimensional = MATRICES
# Lists CANNOT.
#
# We can add arithmetic with lists, such as add them
# With List we CANNOT.

from math import *
from numpy import array, zeros, ones, loadtxt, sort, dot, copy

#########
# LISTS
########
print()
print("-----")
print("LISTS")
print("-----")


print()
print("This is a list")
r = [1, 1, 2, 3, 5, 8, 13, 21]
print(r)


# ---------------------------------------------
print()
print("Another list")
x = 1.0
y = 1.5
z = -2.2
r = [x, y, z]
print(r)
```

```
# ------------------------------------------------
print()
print("The elements can be calculated directly in the list")
r = [2*x, x+y, z/sqrt(x**2+y**2)]
print(r)


# ------------------------------------------------
print()
print("Check whether two lists are equal")
r = [1,2,3]
s = [1,2,3]
print(r==s)


# ------------------------------------------------
print()
print("The individual elements can be extracted and used in calculations")
print("NOTE that the first element starts from ZERO")
pyta=r[0]**2 + r[1]**2
print("r[0]^2 + r[1]^2 = ",pyta)
# NOTE that the first element starts from ZERO!!!!!!!


# ------------------------------------------------
print()
print("Negative indexes refer to the right edge of the list")
r = [1, 2, 3, 4, 5, 6, 7, 8]
print(r[0])
print(r[-1])
print(r[-2])


# ------------------------------------------------
print()
print("Replace an element")
# We can replace elements of the list
r = [1.0, 1.5, -2.2]
r[1] = 43.6
print(r)


# ------------------------------------------------
print()
print("Sum the elements")
# Python contains a built-in function called "sum"
r = [1.0, 1.5, -2.2]
print(r)
sumEle = sum(r)
```

```
print(sumEle)
print("Formatted value %4.2f" % sumEle)


# ------------------------------------------
# Other built-in functions
# max(r)     maximum element of the list
# min(r)     minimum element of the list
# len(r)     how many elements in the list
# abs(r[2])  absolute value of the 2nd element
# ------------------------------------------


# ------------------------------------------
print()
print("Find the mean value")
mean = sum(r)/len(r)
print(mean)
print("Formatted value %4.2f" % mean)



# ------------------------------------------
print()
print("You cannot apply a function to all elements")
# If you want to compute the log of each element
# you CANNOT simply type log(r)
# But this is what you can do

r = [10.0, 1.5, 32.2]
print(r)

print()
print("But you can do one by one, as in log(r[0])")
logR = [log(10.0), log(1.5), log(2.2)]
print("'ln' of each element done one by one")
print(logR)

print()
print("You can also use list(map(log,r))")
logR = list( map( log,r ))
print(logR)




# ------------------------------------------
print()
print("Append an element to a list")
r.append(6.1)
print(r)
```

```
print()
print("Start with an empty list and append")
r = []
r.append(2.)
r.append(4.)
r.append(6.)
r.append(8.)
print(r)


# -----------------------------------------------
print()
print("Remove an element from the list")
r.pop(2)
print(r)









# -----------------------------------------------
# -----------------------------------------------
# -----------------------------------------------
# -----------------------------------------------
##########
# ARRAYS
#########
print()
print("------")
print("ARRAYS")
print("------")


print()
print("Convert a list to an array")
# We can convert a list into an array
# BUT
# we need to call "array" from numpy
from numpy import array
r = [10.0, 1.5, 32.2]
a = array(r, float)
print(a)



########################################
# NOTE #
########
```

```
# Just as we did with "import math"
# Here, we can also write "import numpy"
# and then use array only when needed
import numpy
r = [10.0, 1.5, 32.2]
a = numpy.array(r, float)
print(a)
#######################################



print()
print("We can type an array using 'ARRAY' ")
b = array([4., 6., 9.],float)
print(b)


print()
print("Array of zeros: a vector")
from numpy import zeros, ones
vec0 = zeros(5, float)
print(vec0)
vec1 = ones(5, float)
print(vec1)
# NOTE: we could have defined zeros(5,int),
# if we wanted five integers equal to zero



# ---------------------------------------------
print()
print("Array of zeros: a matrix")
mat = zeros([3,5], float)
print(mat)
# It should also accept (3,5) instead of [3,5]


# A matrix is a list of lists
# So it can be created from lists as
print()
print("A matrix is a list of lists")
mat = array([ [2., 4, 5], [4, 4, 4], [6, 7, 8] ], float)
print(mat)



# ---------------------------------------------
print()
print("Extract an element of the matrix")
print("The element of the first row and first column is mat[0,0] =", mat[0,0])
```

```
# ------------------------------------------
# We can modify the elements of the array
print()
print("Changing the elements of a matrix")
mat = zeros( [3,3], float)
print(mat)
print()
mat[0,0] = 1.
mat[1,1] = 1.
mat[2,2] = 1.
print(mat)
```

```
# ------------------------------------------
# ------------------------------------------
# ------------------------------------------
############
# INPUT DATA
############
print()
print("----------")
print("INPUT DATA")
print("----------")
from numpy import loadtxt

print("Vector 'vec' ")
vec = loadtxt("Lec05_InputVector.txt",float)
print(vec)

print()
print("Matrix 'mat' ")
mat = loadtxt("Lec05_InputMatrix.txt",float)
print(mat)
```

```
# ------------------------------------------
# ------------------------------------------
# ------------------------------------------
#################
# USEFUL FUNCTIONS
#################
print()
print("----------")
print("Useful info")
```

```
print("-----------")

print()
print("Length of 'vec' ")
HowMany = len(vec)
print(HowMany)

print("Number of elements of 'vec' or 'mat' ")

HowMany = vec.size
print("Number of elements of 'vec' = ",HowMany)

HowMany = mat.size
print("Number of elements of 'mat' = ",HowMany)

print("Shape of 'vec' or 'mat' ")

shapeVec = vec.shape
print("Shape of 'vec' = ",shapeVec)

shapeMat = mat.shape
print("Shape of 'mat' = ",shapeMat)

print()
print("Sort a vector in increasing order ")
from numpy import sort
vecToSort = array( [6.,3.,9.,2.,1.,4.,5.], float)
print(vecToSort)
print(sort(vecToSort))




# ------------------------------------------
# ------------------------------------------
# ------------------------------------------
########################
# ARITHMETIC with ARRAYS
########################
print()
print("----------------------")
print("ARITHMETIC with ARRAYS")
print("----------------------")
from numpy import dot

print()
print("Sum the first two elements of 'vec' ")
tot = vec[0] +vec[1]
print(tot)

print()
```

```
print("Sum all elements of 'vec' ")
tot = sum(vec)
print(tot)

print()
print("BUT 'sum(mat)' sums all elements of each column")
tot = sum(mat)
print(tot)

print()
print("vec")
print(vec)
print("2*vec")
vec2 = 2*vec
print(vec2)

print()
print("vec + vec2")
print(vec+vec2)

print()
print("vec")
print(vec)
print("vec+1")
print(vec+1)

print()
print("vec*vec2")
print(vec*vec2)



# ------------------------------------------
# ------------------------------------------
# ------------------------------------------
print()
print("----------")
print("DOT PRODUCT")
print("----------")
print("vec.vec2")
print(dot(vec,vec2))
# DOT PRODUCT between two vectors results in a number
# DOT PRODUCT between a matrix and a vector results in a vector
# DOT PRODUCT between two matrices results in a matrix




# ------------------------------------------
# ------------------------------------------
# ------------------------------------------
```

```
# SLICING
print()
print("SLICING a vector")
vec3 = array([1.,2.,3.,4.,5.,7.,9.,11.],float)
print(vec3)
print(vec3[:])
print(vec3[2:])
print(vec3[:3])

# ------------------------------------------
print()
print("SLICING a matrix")
mat3 = array([ [1.,2.,3.,4.,5.],[5.,4.,3.,2.,1.],[11.,22.,33.,44.,55.] ],float)
print(mat3)

print()
print("A vector of the matrix")
print(mat3[2:])

print()
print("A block of the matrix")
print(mat3[0:2,3:5])




# ------------------------------------------
# ------------------------------------------
# ------------------------------------------
print()
print("-----------------------------------")
print("COPY: an annoying problem in Python!")
print("-----------------------------------")
print("vecB is identical to vecA and changes with it!")
vecA = array([1.,1.],float)
vecB = vecA

vecA[1]=2
print(vecA)
print(vecB)

print("To create vecB independently of vecA")
from numpy import copy
vecA = array([1.,1.],float)
vecB = copy(vecA)
vecA[1]=2
print(vecA)
print(vecB)
```