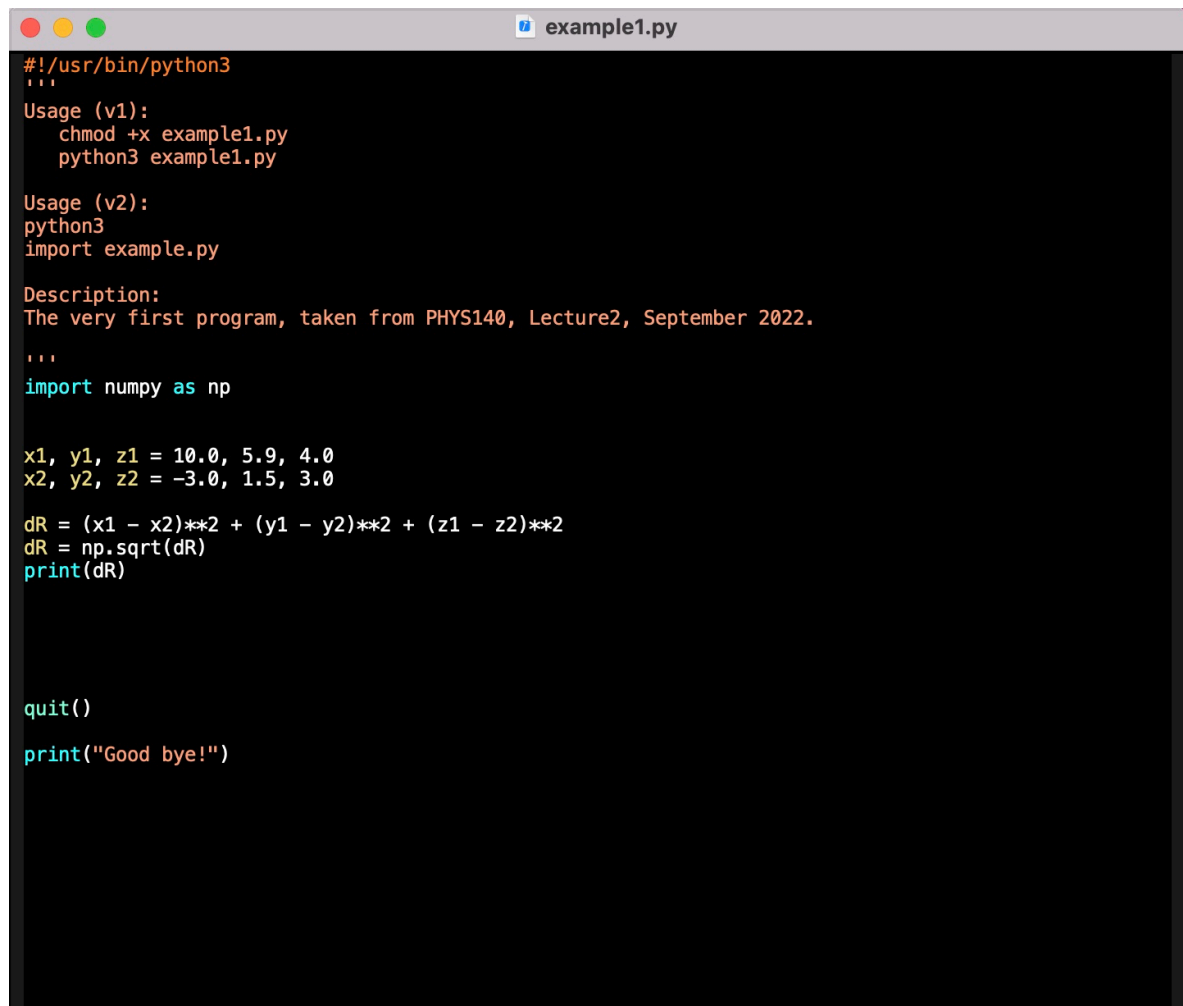# [ΦΥΣ140] Tutorial 2

Create working directory and file
- mkdir tutorial2
- cd tutorial2
- touch example1.py

```python
#!/usr/bin/python3
'''
Usage (v1):
    chmod +x example1.py
    python3 example1.py

Usage (v2):
python3
import example.py

Description:
The very first program, taken from PHYS140, Lecture2, September 2022.

'''
import numpy as np


x1, y1, z1 = 10.0, 5.9, 4.0
x2, y2, z2 = -3.0, 1.5, 3.0

dR = (x1 - x2)**2 + (y1 - y2)**2 + (z1 - z2)**2
dR = np.sqrt(dR)
print(dR)




quit()

print("Good bye!")
```

chmod [άδειες αρχείων]
- chmod u=,g=,o= phys140_tutorial2.py
  - ls -lt
- python3 phys140_tutorial2.py
  - ls -lt
  - chmod u=rwx,g=rx,o=r phys140_tutorial2.py
- phys140_tutorial2.py

Write as new filename "example2.py" (Ctrl+x , Ctrl+w)

example3.py

```python
#!/usr/bin/python3
'''
Usage (v1):
    chmod +x example2.py
    python3 example2.py

Usage (v2):
python3
import example2.py
'''
import numpy as np


# Variable declaration
x1, y1, z1 = 1, 1, 1
x2, y2, z2 = 0, 0, 0


# User input
x2 = float(input("Please provide integer value for x2: "))
if (x2.is_integer() == True):
    print("x2 = ", x2)
else:
    print("x2 is not an integer!")
    quit()

y2 = float(input("Please provide value for y2: "))
if (y2.is_integer() == True):
    print("y2 = ", y2)
else:
    print("y2 is not an integer!")
    quit()

z2 = float(input("Please provide value for z2: "))
if (z2.is_integer() == True):
    print("z2 = ", z2)
else:
    print("z2 is not an integer!")
    quit()


# Calculations
dRsq = (x1 - x2)**2 + (y1 - y2)**2 + (z1 - z2)**2
dR = np.sqrt(dRsq)
print("dR = sqrt(%s) = %s" % (dRsq, dR) )
print("dR = sqrt(%.3f) = %.3f" % (dRsq, dR) )
print("dR = sqrt(%d) = %d" % (dRsq, dR) )


msg = "Thank you for using python3. GOOD BYE!"
print(msg)
print(msg.swapcase())
quit()
```

```
-:---   example3.py    All (1,0)      (Python ElDoc) 1:59PM 3.20
ESC <escape> is undefined
```

Write as new filename "example3.py" (Ctrl+x , Ctrl+w)

```
#!/usr/bin/python3
'''
Usage (v1):
  chmod +x example3.py
  python3 example3.py

Usage (v2):
python3
import example3.py
'''
import numpy as np
```

```python
# Declare a list with integers
intList = [9, 2, 3, 4, 0, 4, 3, 1, 7]

print("intList = ", intList)
print("len(intList) = ", len(intList))
print("sorted(intList) = ", sorted(intList)) # does not modify the object
print("intList = ", intList)

# Permanently modify the intList (sort() instead of sorted())
intList.sort()
print("\n1) intList = ", intList)

intList.reverse()
print("\n2) intList = ", intList)


rm = intList.pop(0)
print("\n3a) intList = ", intList)
print("3b) len(intList) = ", len(intList))

intList.insert(0, rm)
print("\n4a) intList = ", intList)
print("4b) len(intList) = ", len(intList))

intList.insert( len(intList), -1)
print("\n5a) intList = ", intList)
print("5b) len(intList) = ", len(intList))

list1 = [1, 2, 3]
list2 = ["a", "b", "c"]
newList = list1 + list2
print("="*100)
print("6a) newList = ", newList)
print("6b) len(newList) = ", len(newList))
print("6c) newList[0] = ", newList[0])
print("6d) newList[-1] = ", newList[-1])
print("6e) newList[:1] = ", newList[:1])
print("6f) newList[:2] = ", newList[:2])
print("6g) newList[1:] = ", newList[1:])
print("6h) newList[2:] = ", newList[2:])
print("6i) newList[-1:] = ", newList[-1:])
print("6i) newList[-2:] = ", newList[-2:])
print("="*100)

print("*"*100)
newList.append("A")
newList.extend(["B", "C"])
print("7) newList = ", newList)
print("*"*100)
```

```
msg = "quit()"
print(msg)
quit()
```

Create tarball with today's work and email it to yourselves
- cd ../
- ls -lt
  - do you see tutorial2? if no cd so that you are one directory up relative to it
- tar -cvzf tut2.tgz tutorial2/

Confirm that tarball contains the desired files by examining its contents"
(detailed table of contents for this archive)
tar -tvf tut2.tgz
tar -tvf tut2.tgz '*.py'