

## Ayudantía 3

Actividad	Objetivo
Repaso	<ul style="list-style-type: none"><li>• Imprimiendo en Python.</li><li>• Variables, tipos de datos y expresiones.</li><li>• Listas.</li><li>• Funciones, métodos y paquetes.</li><li>• Procesos Iterativos.</li></ul>
Sublime	<ul style="list-style-type: none"><li>• Instalación de Software.</li></ul>
Programa sencillo	<ul style="list-style-type: none"><li>• Importar archivo .txt en python.</li><li>• Leer y editar archivo.</li></ul>
Pasar parámetros a Python	<ul style="list-style-type: none"><li>• Uso de sys.</li><li>• Input.</li></ul>
Python desde terminal	<ul style="list-style-type: none"><li>• Abrir terminal desde Python.</li><li>• Ejecutar archivos desde terminal.</li><li>• Uso de batch</li></ul>
Dudas	<ul style="list-style-type: none"><li>• Ayudantía / Stata / Tarea.</li></ul>

### Repaso

#### Imprimiendo en Python

In [9]:

```
print("Hola \n #2")
print("""Primera línea
Segunda línea
Tercera línea""")
print(2+3)
siete = 4+3
print(siete**2 - siete*2)
```

```
Hola
#2
Primera línea
Segunda línea
Tercera línea
5
35
```

#### Variables, tipos de datos y expresiones.

In [29]:

```
string = "Esto es una línea de texto"
verdadero = True
falso = False
uno = 1
numero = 987654.5
print(type(string) , type(verdadero), type(falso),type(uno), type(numero))
print(False + True + True) # Los booleanos son expresados como 1 y 0
string2 = str(numero)
numero2 = float(string2)
print(type(string2) , type(numero2))
```

```
#Qué pasa si no se declara una variable?
```

```
#Que pasaria en estos casos:
#print("Hola " * 2)
#print("Hola" * 2)
```

```
<class 'str'> <class 'bool'> <class 'bool'> <class 'int'> <class 'float'>
2
<class 'str'> <class 'float'>
Hola Hola
```

## Listas

In [7]:

```
sem1 = ["Microeconomía I", 45.40, "Conta II", 39.49, "Macro II", 50.0, "Economía y Ciencia de los d
atos", 70, "Ética", 60.5]
sem1[0] = "Micro I"
sem1[3] = 40
sem1[-1] = 10
print(sem1)
tav = sem1 + ["Calculo IV", 51]
del(tav[2:4])
del(sem1[:])
print(tav , sem1)
```

```
['Micro I', 45.4, 'Conta II', 40, 'Macro II', 50.0, 'Economía y Ciencia de los datos', 70, 'Ética'
, 10]
['Micro I', 45.4, 'Macro II', 50.0, 'Economía y Ciencia de los datos', 70, 'Ética', 10, 'Calculo I
V', 51] []
```

In [8]:

```
notas = [ tav,
          ["Micro II", 50, "Conta II", 45, "Macro Internacional", 45, "OFG", 68, "OPR", 60],
          ["Competencia", 46, "Cyt", 50, "Mercados", 50, "OFG", 60, "OPR", 65],
          ["Teoría Micro", 46, "Teoría Macro", 52, "Eco Mate", 48, "Análisis Real", 39.5]
        ]
print(notas[0] , "\n" , notas[3][-1])
print(notas[2][4:])
```

```
['Micro I', 45.4, 'Macro II', 50.0, 'Economía y Ciencia de los datos', 70, 'Ética', 10, 'Calculo I
V', 51]
39.5
['Mercados', 50, 'OFG', 60, 'OPR', 65]
```

## Funciones métodos y paquetes

In [3]:

```
#print(pi)
from math import pi as pi
r = float(input("Escriba el radio: "))
diametro = 2*pi*r
area = pi*r**2
print("Diametro es " + str(diametro) + " y area es " + str(area))
```

```
Escriba el radio: 45
Diametro es 282.7433388230814 y area es 6361.725123519331
```

In [5]:

```
import datetime
fecha = datetime.datetime.now().strftime ("%d/%m/%Y")
print(fecha)
```

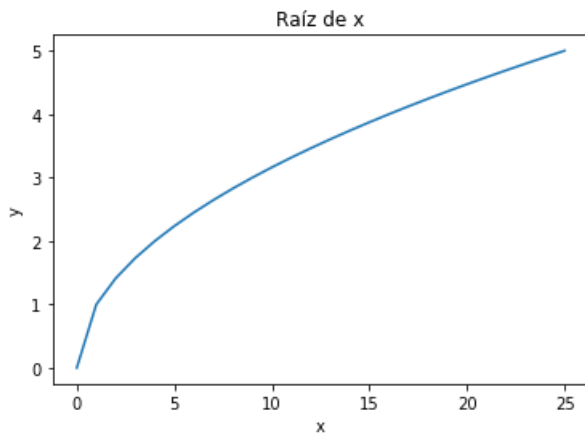
21/08/2018

## Procesos iterativos

In [7]:

```
import matplotlib.pyplot as plt
linea = []
cuadrado = []
for x in range(26):
    linea.append(x)
    cuadrado.append(x**0.5)

plt.plot(linea,cuadrado) #Mostrar con scatter también
plt.xlabel("x")
plt.ylabel("y")
plt.title("Raíz de x")
plt.show()
```

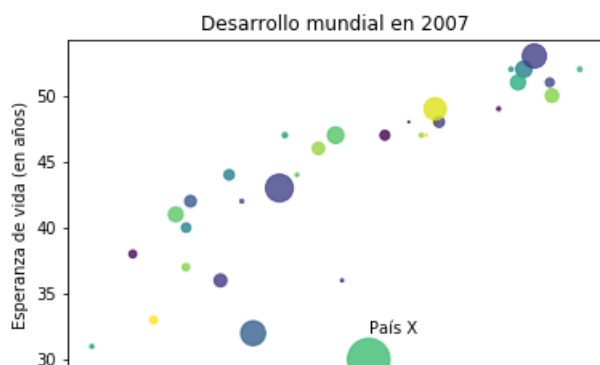


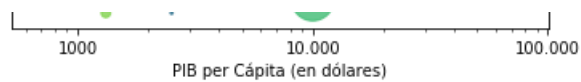
In [89]:

```
import matplotlib.pyplot as plt
import numpy as np

pib = [10000,45712.0, 11712.0, 50649.600000000006, 7230.400000000001, 2340.8, 2548.8, 43144.0, 6016
9.600000000006, 16737.600000000002, 2510.4, 664.0, 19291.2, 1672.0, 1214.4, 19923.2,
1668.8000000000002, 2884.8, 17531.2, 19164.8, 4953.6, 14806.400000000001, 6112.0, 1318.4, 1510.4, 7
697.6, 1745.6000000000001, 58876.8, 40296.0, 4398.400000000001, 992.0, 3222.4, 78971.20000000001, 3
5705.6, 4169.6]
esperanza = [30,52.0, 47.0, 53.0, 47.0, 36.0, 44.0, 51.0, 50.0, 47.0, 29.0, 31.0, 48.0, 40.0, 33.0,
48.0, 37.0, 42.0, 47.0, 49.0, 44.0, 48.0, 46.0, 29.0, 41.0, 36.0, 42.0, 51.0, 52.0, 47.0, 38.0, 32.
0, 52.0, 49.0, 43.0]
poblacion = [700,101.7625, 36.365, 223.0725, 105.08749999999999, 62.31749999999995, 40.775, 85.82,
69.825, 6.859999999999999, 3.5174999999999996, 5.5825, 10.57, 33.5475, 23.3275, 43.435, 21.0525, 5.
7225, 2.1875, 190.225, 5.0225, 1.1900000000000002, 59.08, 34.9125, 83.58, 3.605, 50.574999999999996
, 28.997500000000002, 7.21, 9.94, 22.5575, 236.3025, 8.1025, 5.6000000000000005, 296.2225]
colores = np.random.rand(len(poblacion))

plt.scatter(pib, esperanza, s = poblacion, c = colores,alpha=0.8)
plt.xscale('log')
plt.xlabel('PIB per Cápita (en dólares)')
plt.ylabel('Esperanza de vida (en años)')
plt.title('Desarrollo mundial en 2007')
plt.xticks([1000, 10000, 100000],['1000', '10.000', '100.000'])
plt.text(10000, 32, 'País X')
plt.show()
```





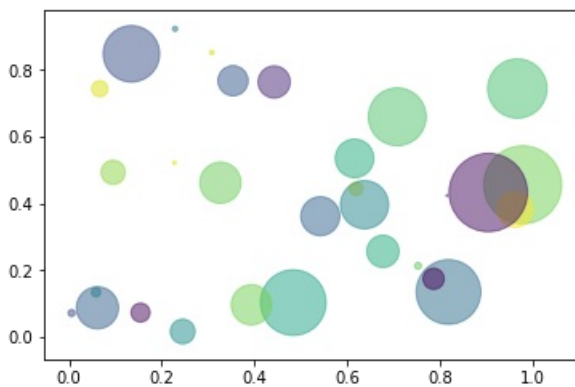
In [30]:

```
#API en https://matplotlib.org/api/pyplot_api.html
import numpy as np
import matplotlib.pyplot as plt

np.random.seed(1996) #Dejamos una semilla para que se pueda reproducir

N = 30
x = np.random.rand(N)
y = np.random.rand(N)
colors = np.random.rand(N)
area = (50 * np.random.rand(N))**2

plt.scatter(x, y, s=area, c=colors, alpha=0.5)
plt.show()
```



## Sublime

### [Descargar Software](#)

## Programa sencillo

Con el siguiente código nos moveremos a nuestro directorio de trabajo e importaremos el archivo abrir.txt para leerlo.

### Uso de open()

In [6]:

```
import os
os.getcwd()
os.system("C:\\Users\\pablo\\Desktop\\Ayudantía 3")
f=open('abrir.txt','r') #Abrimos el archivo
f.read() #Lo leemos
```

Out[6]:

```
'Código de Honor de la universidad\n"Como miembro de la comunidad de la Pontificia Universidad
Católica de Chile me comprometo a respetar los principios y normativas que la rigen. Asimismo, pro
meto actuar con rectitud y honestidad en las relaciones con los demás integrantes de la comunidad
y en la realización de todo trabajo, particularmente en aquellas actividades vinculadas a la
docencia, el aprendizaje y la creación, difusión y transferencia del conocimiento. Además, velaré
por la integridad de las personas y cuidaré los bienes de la Universidad".Acepto|Rechazo'
```

In [7]:

```
#Alternativamente: Lo leemos línea por línea.
f=open('abrir.txt','r') #Nuevamente, ya que el archivo se cerró.
for line in f:
```

```
print(repr(line))
f.read()
```

```
'Código de Honor de la universidad\n'
'"Como miembro de la comunidad de la Pontificia Universidad Católica de Chile me comprometo a
respetar los principios y normativas que la rigen. Asimismo, prometo actuar con rectitud y honesti
dad en las relaciones con los demás integrantes de la comunidad y en la realización de todo
trabajo, particularmente en aquellas actividades vinculadas a la docencia, el aprendizaje y la cre
ación, difusión y transferencia del conocimiento. Además, velaré por la integridad de las personas
y cuidaré los bienes de la Universidad".Acepto|Rechazo'
```

```
Out[7]:
```

```
''
```

### ¿Y si quisiéramos escribir algo en el archivo?

```
In [ ]:
```

```
with open('abrir.txt','r+') as f:
    texto = f.read()
    print(texto)
    f.write("\n Acepto|Rechazo \n") #Escribimos una nueva línea en el archivo
```

```
In [ ]:
```

```
f.read() #Error (Ya se cerró el archivo)
print(texto) #Se sigue viendo
```

### Operaciones con sys

```
In [8]:
```

```
import sys
count = 0
for line in sys.stdin:
    count += 1
print(count)
```

```
0
```

Guardemos ese código bajo el nombre contar lineas.py y ejecutémoslo de la siguiente forma.  
python hola.py < usuario.txt

### Uso de input

```
In [1]:
```

```
a = input("Escribe algo ")
print(" "+a)
```

```
Escribe algo Hola
Hola
```

## Python desde línea de comandos

```
In [ ]:
```

```
import os
os.system("start cmd") #Aparecerá la pantalla de comando.
```

Dentro de la terminal escribir:

```
cd [Dirección del archivo]
```

cd [Dirección del archivo]

python "Stuck in a time loop.py" #Prueba escribir un número en lo que aparezca...

python "Apaxiaaaaaaaaaaaaaans.py" #NOOOOOOOOOOOOOOOOOOOOOOO!!!!

python

print("hola")

print(2+3)

for x in range(10):

print(x,x\*\*2) #Como ven, podemos ejecutar código desde la termina

Ahora veamos como funcional los archivos .bat: [Apaxiaaaaaaaaaaaaaans / Acción](#)

---

¿Dudas?