



# Universidad Ricardo Palma

RECTORADO

*Formamos seres humanos para una cultura de paz*

## Primer Programa de Especialización INTRODUCCIÓN AL DATA SCIENCE

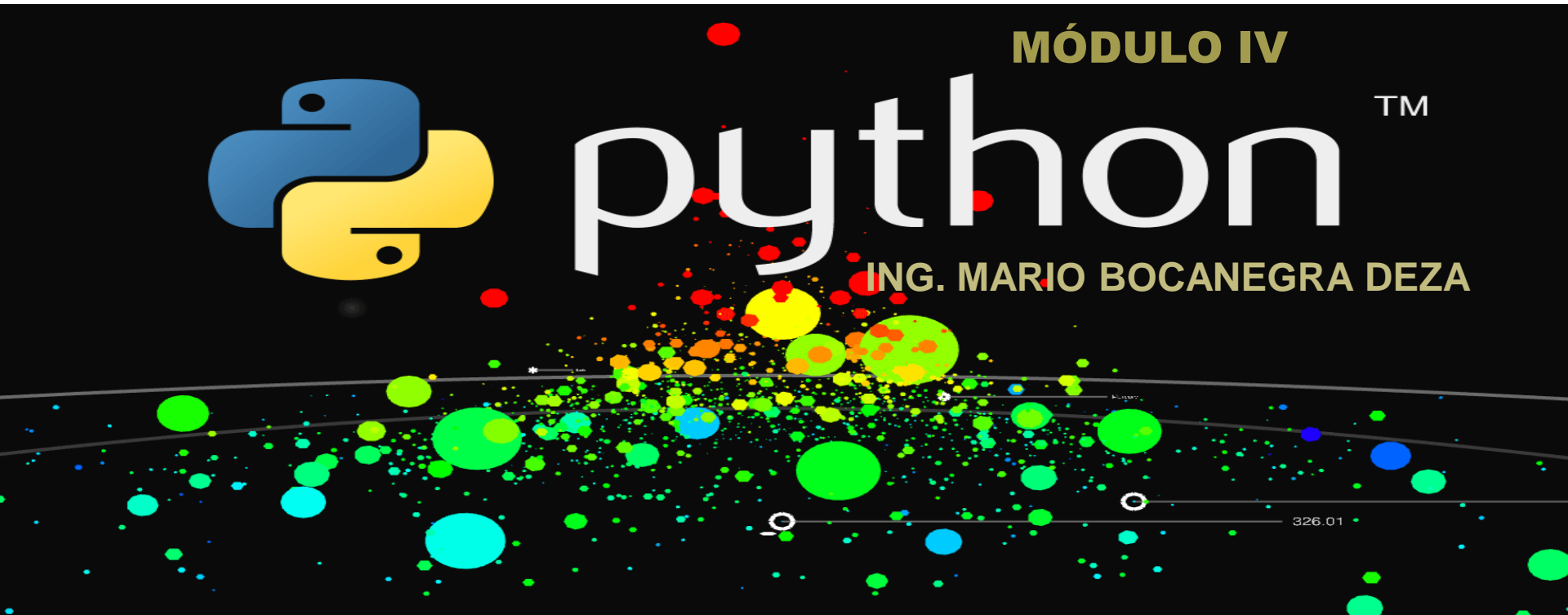
FUNCIONES



MÓDULO IV

python<sup>TM</sup>

ING. MARIO BOCANEGRA DEZA



# Contenido

---

- ✓ Definición de Funciones
- ✓ Ejemplo de función aplicando PM y POO
- ✓ Ejemplo de Módulo



# Definición de Funciones

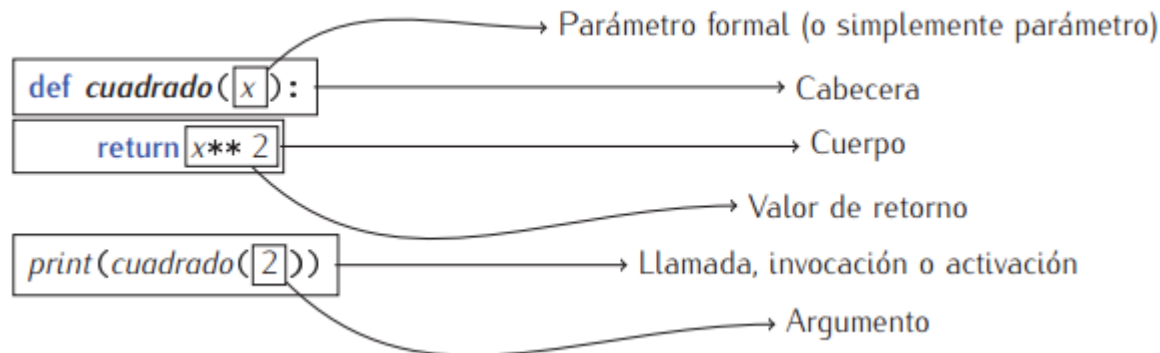
## Funciones con un solo parámetro :

```
def cuadrado(base):  
    return base**2
```

```
resultado=cuadrado(2)  
print("El resultado es", resultado)
```

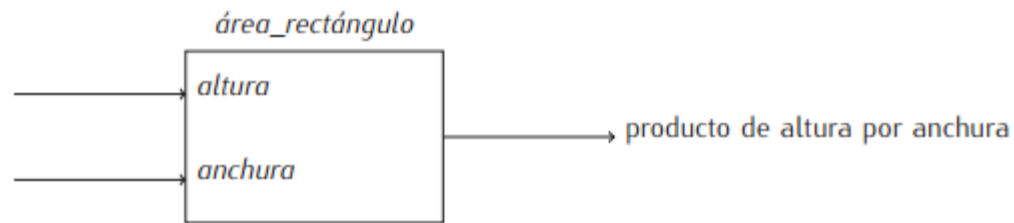
```
def mensaje(cadena):  
    return cadena
```

```
imprimir=mensaje('Utilizando funciones en Python')  
print("El mensaje es : ",imprimir)
```



## Funciones con varios parámetros:

---



```
def are_rectangulo(altura,anchura):  
    return altura*anchura  
  
resultado=are_rectangulo(2)  
print("El resultado es", resultado)
```

```
def devuelve_nro_mayor(a,b,c):  
    if a>b and a>c:  
        mayor=a  
    elif b>a and b>c:  
        mayor=b  
    else:  
        mayor=c  
    return mayor
```

```
resultado=devuelve_nro_mayor(22,15,6)  
print("El numero mayor es", resultado)
```



## Funciones sin parámetros:

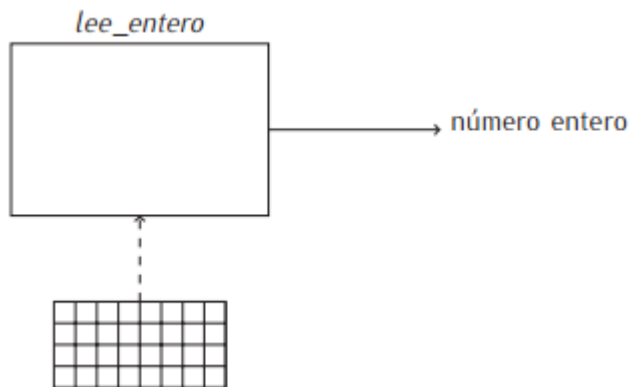
---

```
def limpia_pytonshell():  
    return print('\n'*100)  
  
limpia_pytonshell()
```



```
def leer_numero():  
    return int(input('Ingrese una numero:'))
```

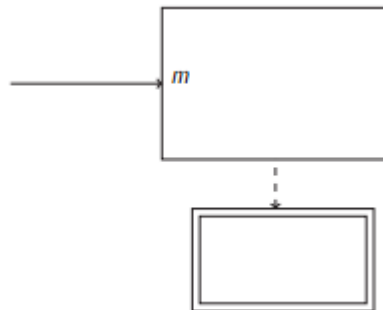
```
resultado=leer_numero()  
print("El numero ingresado es:", resultado)
```



# Funciones sin devolución valor:

---

## Procedimiento



```
def es_par(numero):  
    if numero%2==0:  
        print('El número ingreso es par')  
    else:  
        print('El número ingreso no es par')  
  
es_par(8)
```

```
def listar_nro_desc(inicio,final):  
    for n in range(inicio,final,-1):  
        print('Numero {}'.format(n))  
  
listar_nro_desc(20,0)
```



## Funciones que retornan valores mediante una lista:

---

```
def listar_nro_par(numero):  
  
    listapares=[]  
    listaimpares=[]  
  
    for n in range(2,numero):  
        if n%2==0:  
            listapares.append(n)  
        else:  
            listaimpares.append(n)  
  
    return [listapares,listaimpares]
```



```
n=10  
lista =listar_nro_par(n)  
print('Numeros pares {0}, Numeros impares {1}'.format(lista[0],lista[1]))
```



# Función aplicando P. Modular

```
#import math
from math import *

def media(lista,n):
    suma=0
    for i in lista:
        suma=suma+i
    return round(suma/n,2)

def varianza(lista,media,n):
    suma=0
    for i in lista:
        calculo=(i-media)**2
        suma=suma+calculo
    return round(suma/n,2)

def desviacion(lista,media,n):
    suma=0
    for i in lista:
        calculo=(i-media)**2
        suma=suma+calculo
    return round(sqrt(suma/n),2)

lista=[12,15,16,17,13,11]
n=len(lista)
mediaR=media(lista,n)
print('La media es :',mediaR)
print('La varianza es:',varianza(lista,mediaR,n))
print('La desviación estandar :',desviacion(lista,mediaR,n))
```





# Función aplicando POO

```
import math

class Estadistica:
    def __init__(self, lista, n):
        self.lista = lista
        self.n = n
        print('objeto inicializado')

    def media(self):
        suma=0
        for i in self.lista:
            suma=suma+i
        return round(suma/self.n,2)

    def varianza(self, media):
        suma=0
        for i in self.lista:
            calculo=(i-media)**2
            suma=suma+calculo
        return round(suma/self.n,2)

    def desviacion(self, media):
        suma=0
        for i in self.lista:
            calculo=(i-media)**2
            suma=suma+calculo
        return round(math.sqrt(suma/self.n),2)

lista=[12,15,16,17,13,18]
n=len(lista)
objeto=Estadistica(lista,n)
media=objeto.media()
print('La media es :',media)
print('La varianza es:',objeto.varianza(media))
print('La desviación estandar :',objeto.desviacion(media))
```



# Módulos

```
milibreria/  
    __init__.py  
    estadistica/  
        __init__.py  
        descriptiva.py  
    mate/  
        __init__.py  
    fisica/  
        __init__.py
```

```
from milibreria.estadistica import descriptiva  
  
lista=[12,15,16,17,13,18]  
n=len(lista)  
objeto=descriptiva.Estadistica(lista,n)  
media=objeto.media()  
print('La media es :',round(media,3))  
print('La varianza es:',round(objeto.varianza(media),3))  
print('La desviación estandar :',round(objeto.desviacion(media),3))
```

```
#descriptiva.py  
import math  
  
class Estadistica:  
    def __init__(self,lista,n):  
        self.lista = lista  
        self.n = n  
        print('objeto inicializado')  
  
    def media(self):  
        suma=0  
        for i in self.lista:  
            suma=suma+i  
        return suma/self.n  
  
    def varianza(self,media):  
        suma=0  
        for i in self.lista:  
            calculo=(i-media)**2  
            suma=suma+calculo  
        return suma/self.n  
  
    def desviacion(self,media):  
        suma=0  
        for i in self.lista:  
            calculo=(i-media)**2  
            suma=suma+calculo  
        return math.sqrt(suma/self.n)
```



[Features](#) [Business](#) [Explore](#) [Pricing](#)

Search GitHub

[Sign in](#) or [Sign up](#)

# Built for developers

GitHub is a development platform inspired by the way you work. From [open source](#) to [business](#), you can host and review code, manage projects, and build software alongside millions of other developers.

Use at least one letter, one numeral, and seven characters.

[Sign up for GitHub](#)

By clicking "Sign up for GitHub", you agree to our [terms of service](#) and [privacy policy](#). We'll occasionally send you account related emails.

## GitHub Inc.

Company

[github.com](https://github.com)

# GitHub

GitHub is a web-based Git or version control repository and Internet hosting service. It offers all of the distributed version control and source code management functionality of Git as well as adding its own features.





**Roger D. Peng**  
rdpeng

Professor of Biostatistics at Johns Hopkins University

Unfollow

Block or report user

Johns Hopkins University  
 Baltimore, MD  
 [rdpeng@gmail.com](mailto:rdpeng@gmail.com)  
 <http://www.biostat.jhsph.edu/~r...>

Overview

Repositories 46

Stars 16

Followers 4k

Following 3

### Popular repositories

#### [courses](#)

Forked from [DataScienceSpecialization/courses](#)

Course materials for the Data Science Specialization:  
<https://www.coursera.org/specialization/jhudatascience/1>

HTML ★ 1.1k 🍴 6.6k

#### [ProgrammingAssignment2](#)

Repository for Programming Assignment 2 for R Programming on Coursera

R ★ 462 🍴 100k

#### [practice\\_assignment](#)

Forked from [derekfranks/practice\\_assignment](#)

Practice assignment for the R programming class on Coursera

★ 247 🍴 1.5k

#### [ExData\\_Plotting1](#)

Plotting Assignment 1 for Exploratory Data Analysis

★ 128 🍴 42.8k

#### [RepData\\_PeerAssessment1](#)

Peer Assessment 1 for Reproducible Research

★ 56 🍴 26.8k

#### [rprogdatascience](#)

Makefile ★ 34 🍴 17



# Websites for you and your projects.

Hosted directly from your GitHub repository. Just edit, push, and your changes are live.

