

## Manual de Uso – Aplicación Runge-Kutta

### Integrantes:

- Juan David Buendía Loyo
- Carlos David Páez Ferreira
- María Fernanda Toloza Buitrago

### Enunciados

El programa que se desarrolló en Python para esta temática cumple con la solución adecuada para los ejercicios propuestos en cada uno de los ejecutables individuales.

- RK4:

#dy/dt = y - t<sup>2</sup> +1 Test 1

#dy/dt = 2 - e<sup>-4t</sup> -2y Test 2

- RK4-SegundaOpcion:

# dy/dx = x<sup>2</sup> - 3y Test1

#dy/dx = y - x<sup>2</sup> +1 F

- RK4\_Aplicaciones1:

Caso práctico de aplicación de Runge-Kutta de O(h)=4

Usando la ley de enfriamiento de Newton, se puede predecir la temperatura de un cuerpo al ser enfriado por convección para sistemas con Bi<0.1 (Número de Biot)

$$T(t) \quad pCpV[dT/dt] = -hA(T-T_{inf}) \quad T(0) = T_{inc}$$

Los datos son para una pieza metálica de acero del tamaño de una moneda enfriada con aire

- RK4\_Aplicaciones2:

Caso práctico de aplicación de Runge-Kutta de  $O(h)=4$  para un sistema de ecuaciones Oscilador Amortiguado:

Para una masa en un resorte con un amortiguador se tiene la siguiente ecuación utilizando las leyes de Newton

$$x(t) \quad v(t)=[dx/dt](t)$$

$$m[d^2x/dt^2] + c[dx/dt] + kx = 0$$

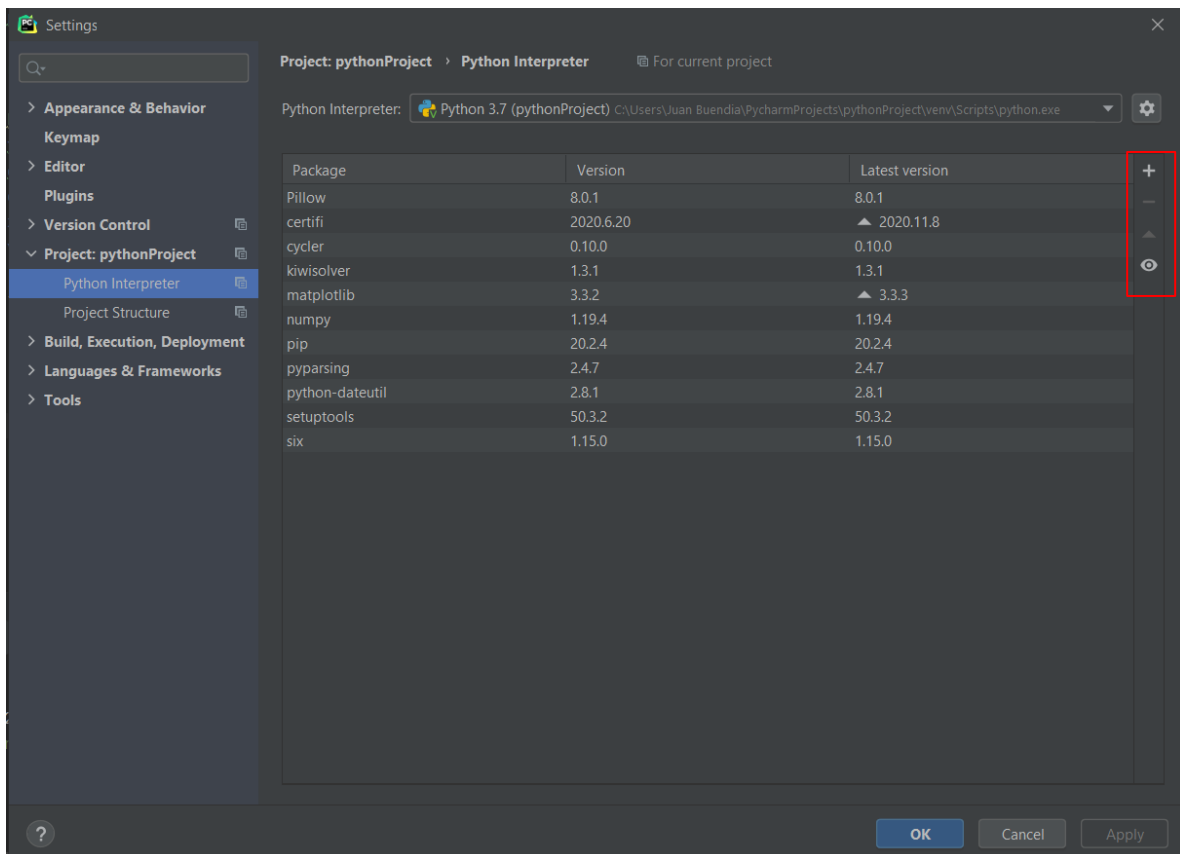
->  $ma+cv+kx = 0$  (masa\*aceleracion + fuerza de amortiguamiento + ley de Hooke)

$$x(0)=x_{inc} \quad v(0)=v_{inc}$$

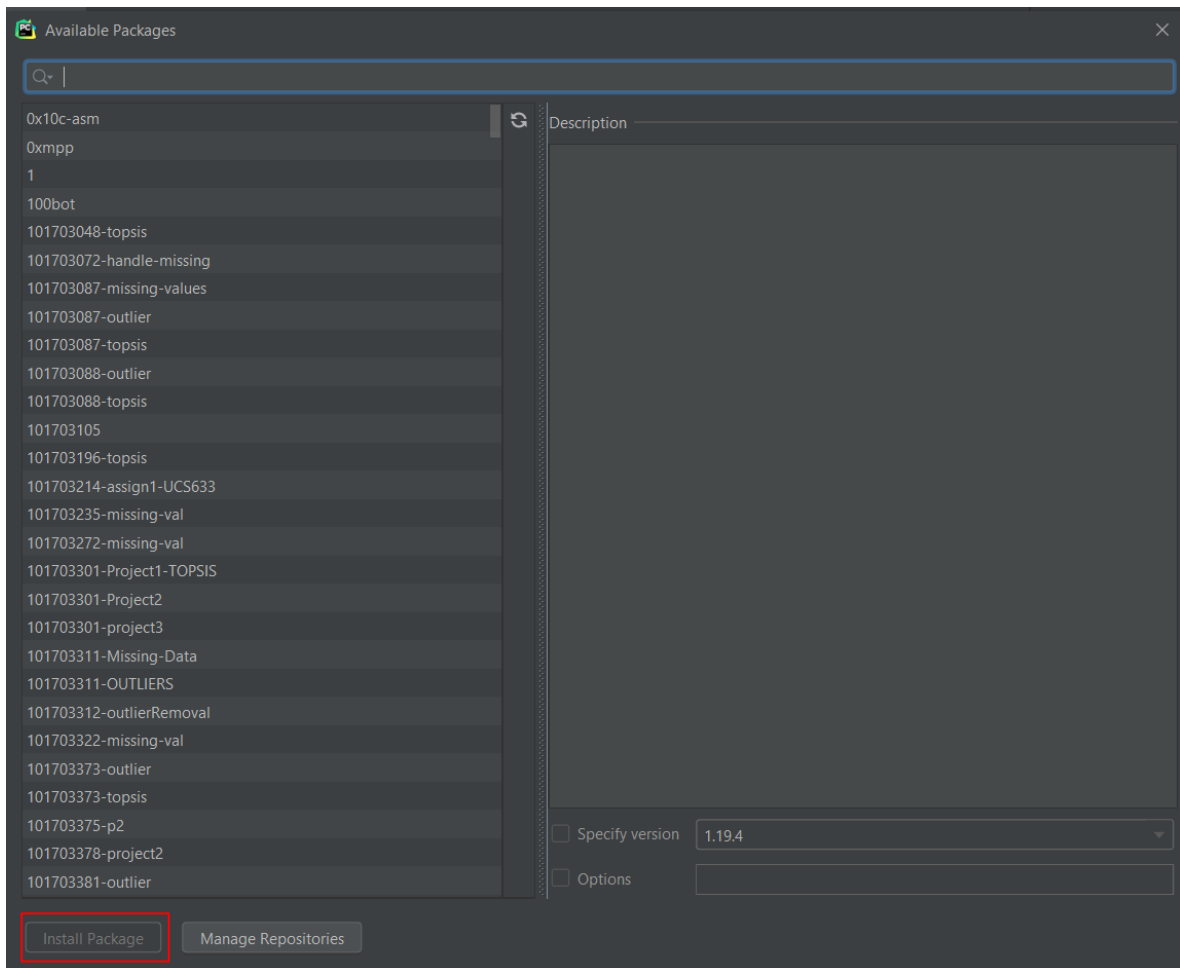
## Ejecución

Para correr los programas por medio de un IDE, depende del IDE en cuestión, a continuación, se va mostrar un ejemplo de cómo sería la ejecución con el IDE PyCharm:

Para instalar paquetes de Python se presiona Alt + Shift + F, donde se desplegará el siguiente menú y se da en el botón + para añadir.



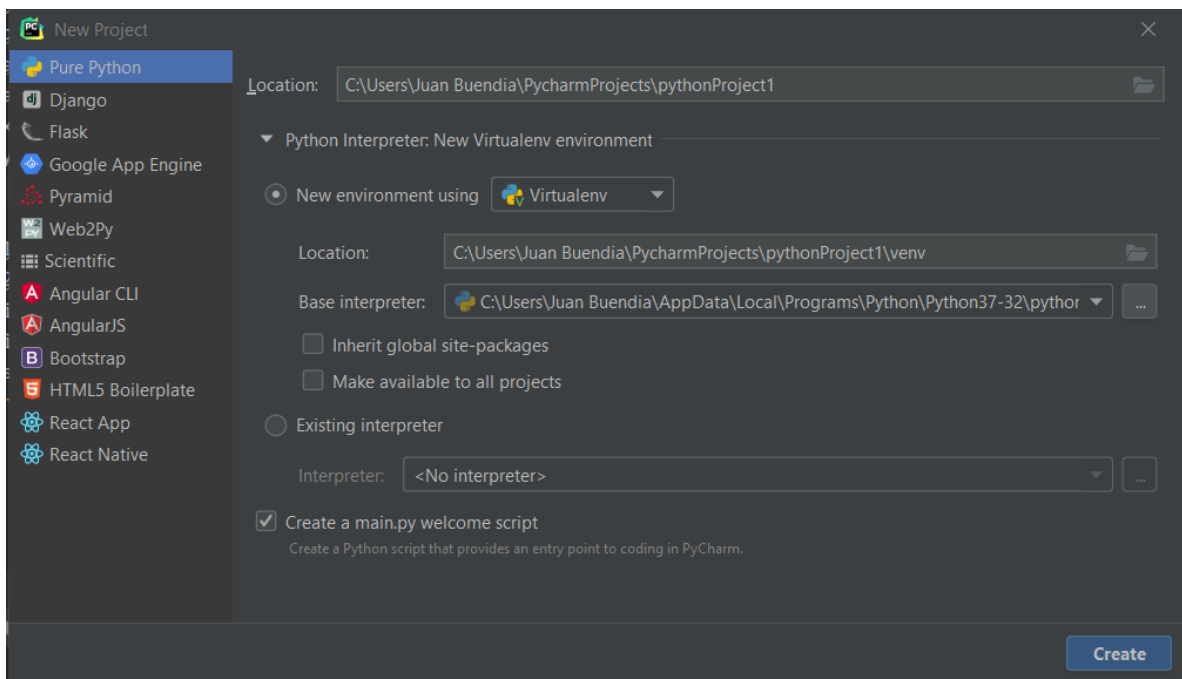
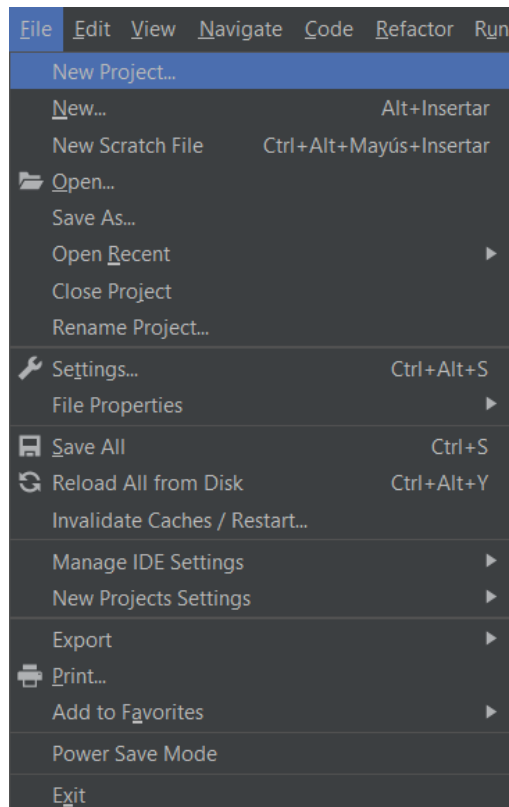
Donde se desplegará la siguiente pantalla.



Una vez desplegada se busca el nombre de los paquetes a instalar y se selecciona "Install Package"

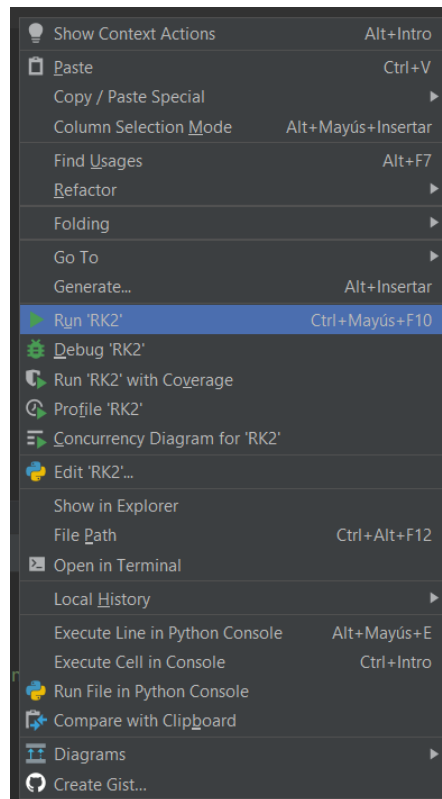
Una vez instalados los paquetes necesarios, se procede a hacer lo siguiente.

Se da clic en la pestaña de File o Archivo



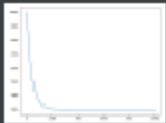
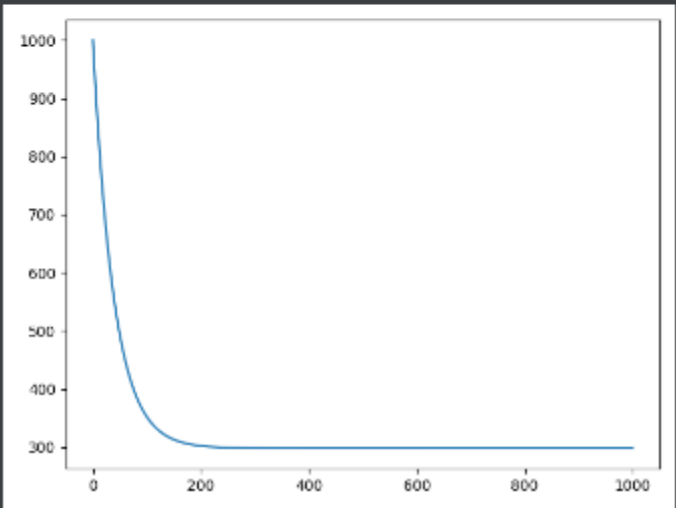
Aquí se escoge la carpeta del código “Código RK”

Después se abre el archivo que se desea ejecutar y se da clic derecho, donde se va a desplegar el siguiente menú



Seleccionada la opción Run el programa procederá a correr y dependiendo del programa mostrará los resultados, ya sea en la consola de Python o la pestaña de Plots

```
PyDev console: starting.  
  
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 21:26:53) [MSC v.1916 32 bit (Intel)] on win32  
>>> runfile('C:/Users/Juan Buendia/PycharmProjects/pythonProject/RK2.py', wdir='C:/Users/Juan Buendia/PycharmProjects/
```



Para correr los programas por CMD hay que asegurarse de lo siguiente:

1. Python está instalado.
2. Los módulos numpy y matplotlib se encuentran instalados.

Una vez comprobado esto, se navega a la carpeta donde se encuentren los archivos .py del proyecto.

Para ejecutar una aplicación de Python se escribe el siguiente comando:

```
python nombre_archivo.py
```

### **Composición**

El proyecto esta compuesto por los siguientes archivos:

- RK2.py
- RK4.py
- RK4-SegundaOpcion.py
- RK4\_Aplicaciones1.py
- RK4\_Aplicaciones2.py

Por tanto, los comandos para ejecutar cada uno de estos serían los siguientes:

- python RK2.py
- python RK4.py
- python RK4-SegundaOpcion.py
- python RK4\_Aplicaciones1.py
- python RK4\_Aplicaciones2.py



## Resultados

Programa      RK4.py      muestra      el      siguiente      resultado

Método RK4

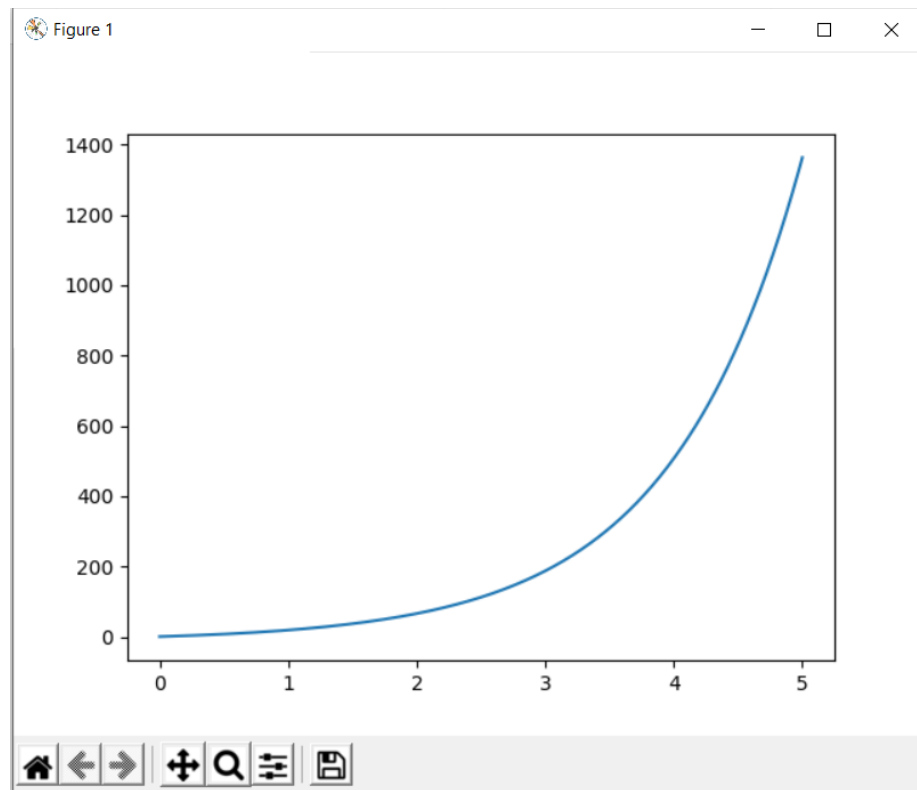
```
0 ---> 0.5
0.2 ---> 0.8292933333333334
0.4 ---> 1.2140762106666667
0.60000000000000001 ---> 1.6489220170416001
0.8 ---> 2.1272026849479437
1.0 ---> 2.6408226927287517
1.20000000000000002 ---> 3.1798941702322305
1.40000000000000001 ---> 3.7323400728549796
1.6 ---> 4.283409498318405
1.8 ---> 4.815085694579433
2.0 ---> 5.305363000692653
```

Método RK4

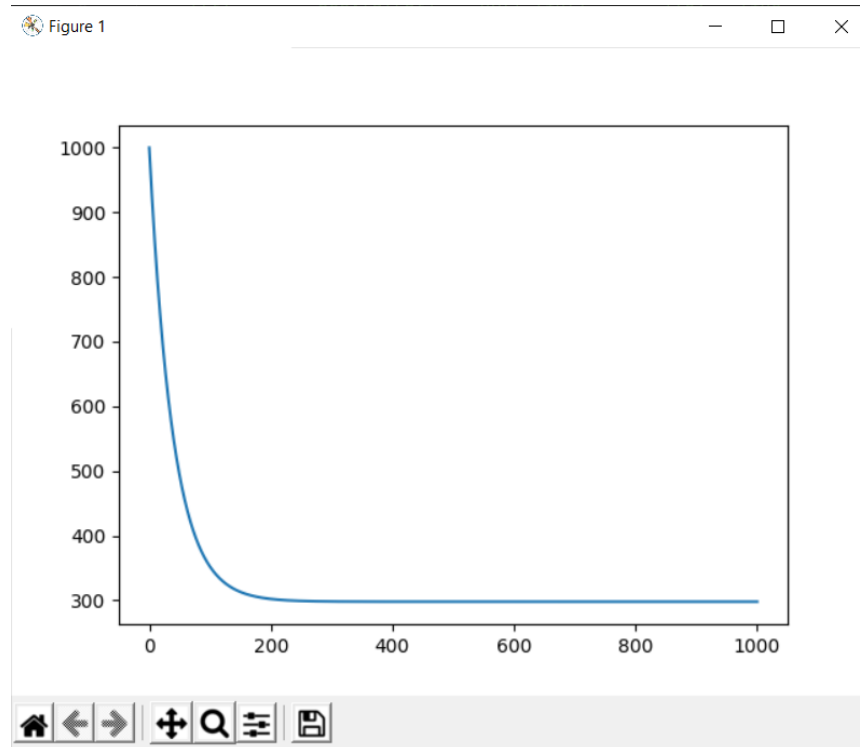
```
0 ---> 1
0.05 ---> 0.95694677392746
0.1 ---> 0.9257948263487807
0.15000000000000002 ---> 0.9039969357034371
0.2 ---> 0.889504715869562
0.25 ---> 0.8806746618725458
0.30000000000000004 ---> 0.8761915626139058
0.35000000000000003 ---> 0.8750061005394635
0.4 ---> 0.8762840376594968
0.45 ---> 0.8793648614932423
0.5 ---> 0.8837281524572481
0.55 ---> 0.8889662516042581
0.60000000000000001 ---> 0.8947420472504420
```

>>>

Programa RK4-SegundaOpcion.py muestra el siguiente resultado



Programa RK4\_Aplicaciones1.py muestra el siguiente resultado



Programa RK4\_Aplicaciones2.py muestra los siguientes resultados

