python

UNIVERSIDAD DE LAS FUERZAS ARMADAS

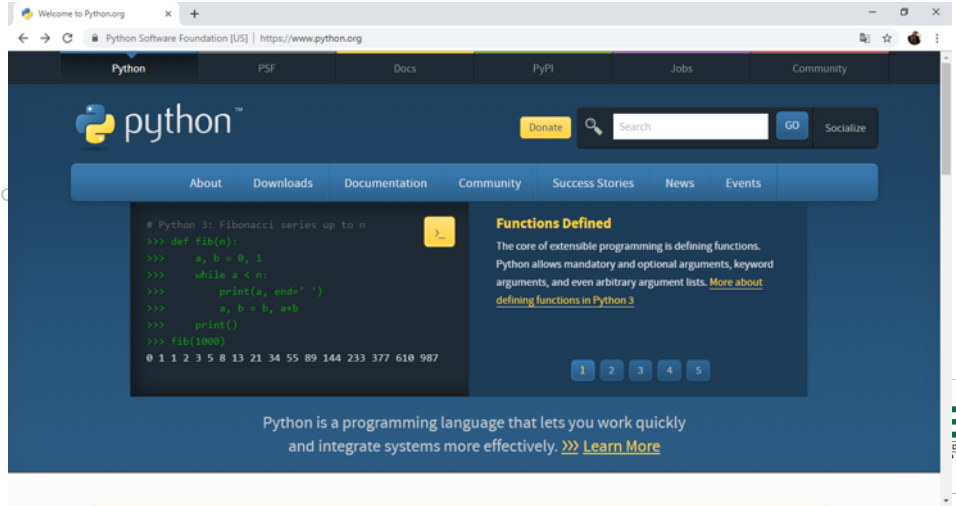
fig/tecnologia-informatica-placas-base-tecnologias_382884.jpg

May 28, 2019

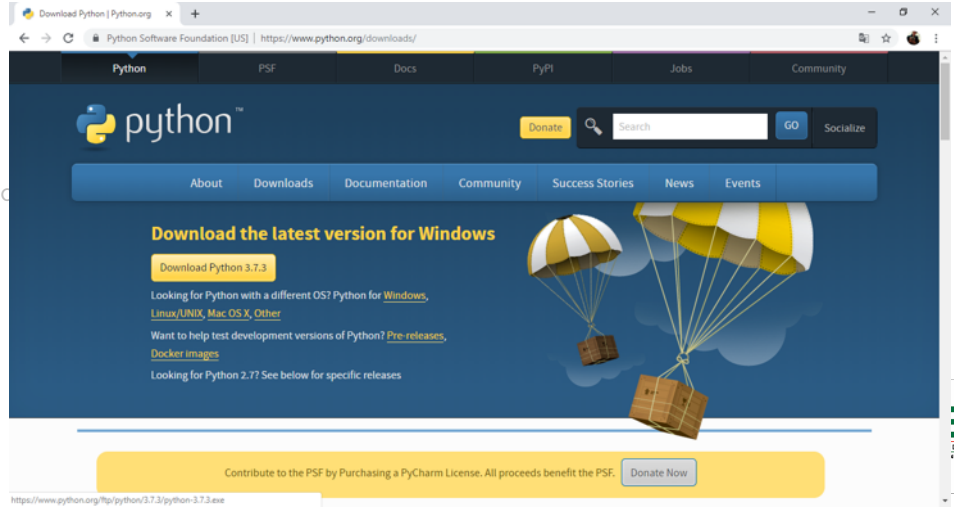
kivy

Kivy es una biblioteca gratuita y de código abierto de Python para desarrollar aplicaciones móviles y otro software de aplicación multitáctil con una interfaz de usuario natural.

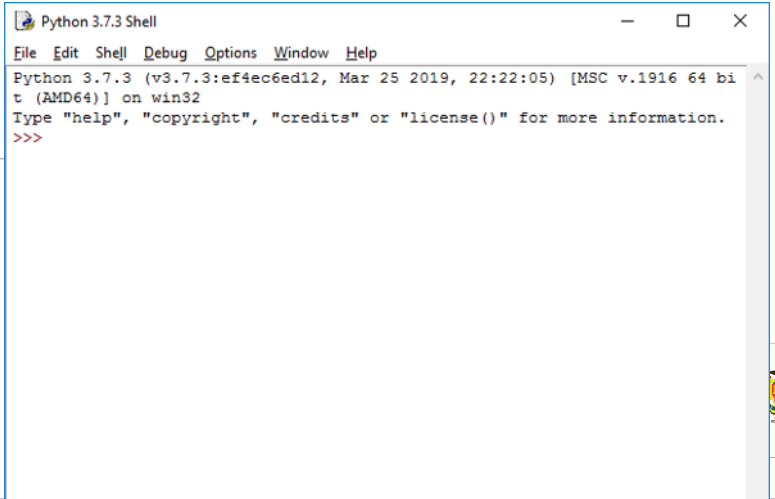
1. Descargar Phyton ingresando a la pagina (https://www.python.org/) .



1. Hacer clik en Download y entraremos a esta sección, descargamos la ultima versión.

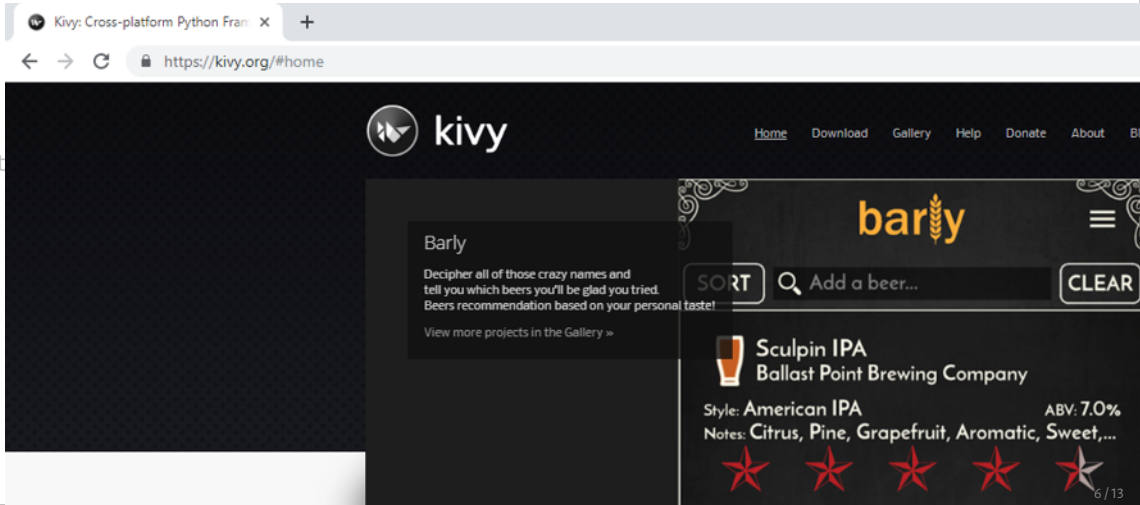


1. Ejecutamos y colocamos permitir después se comenzare a instalar, entramos a la ventana de Python.

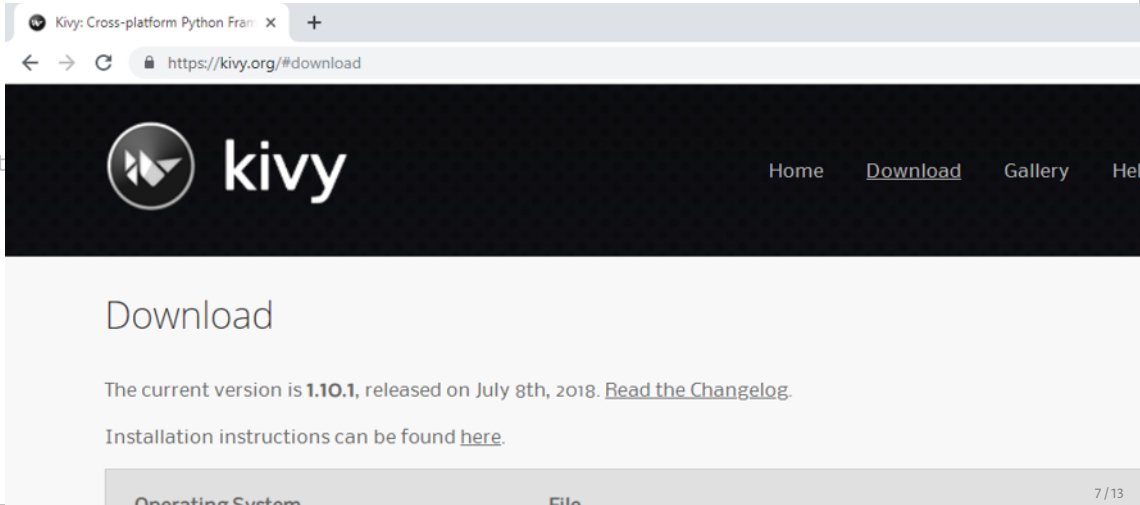


fig/tecnologia-

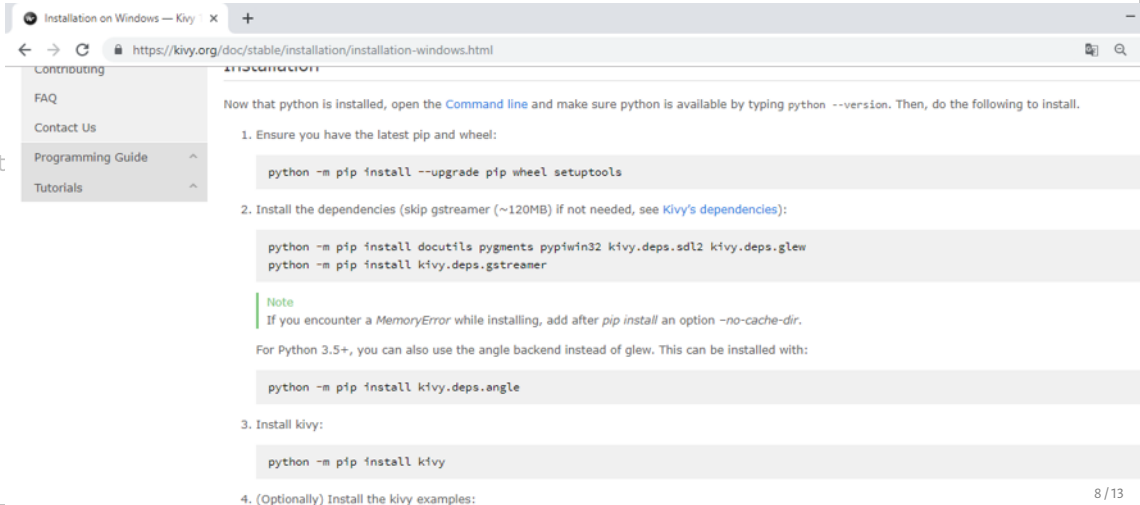
1. Descargar kivy (todos sus componentes) entramos a la pagina (https://kivy.org/home



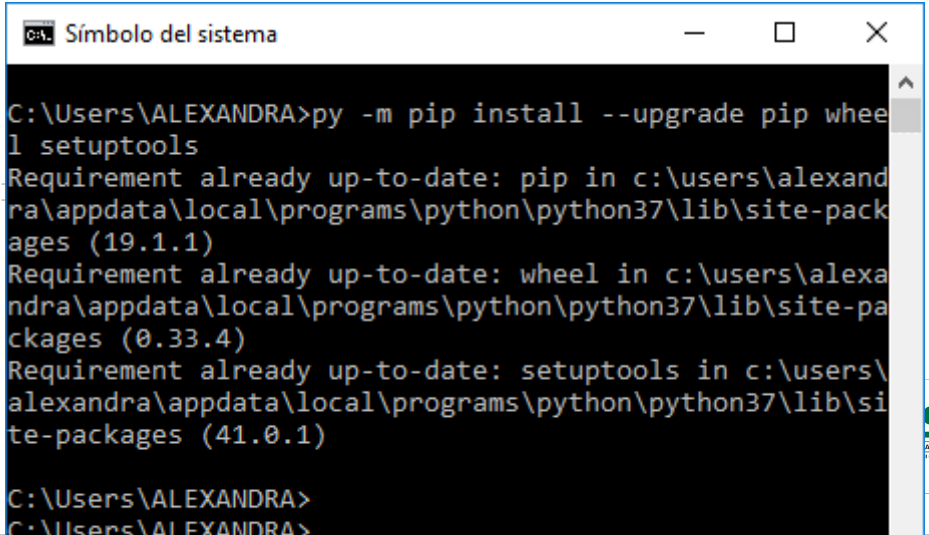
1. Nos vamos a Download y nos vamos donde dice descargar para Windows



1. Después copiamos todos los codigos en CDM para poder descargar kivy



1. Guiarse en los ejemplos que nos da en la parte de instalación de kivy



1. Aquí comenzamos a modificar para poder obtener lo que queremos lograr ver en la aplicación que estamos por crear.

33.py - C:\Users\USUARIO\Desktop\TRABAJO HERRAMIENTAS DE SOFTWARE\33.py (3.7.3)

File Edit Format Run Options Window Help

```
from kivy.app import App #SIRVE PARA IMPORTAR LA BIBLIOTECA DE KIVY
from kivy.uix.button import Button # GENERA UN BOTON A PARTIR DE LA DIRECCION UIX DE KIVI

class TestApp(App): #SIRVE PARA TENER METODOS DEFINIDOS POR SU CLASE
    def build(self): # SIRVE PARA CONSTRUIR
        return Button(text='Herramientas de Software para Ingenieria') #ES EL TEXTO QUE IRA DENTRO DE NUESTRO BOTON

TestApp().run() # PARA CORRER EL PROGRAMA
```

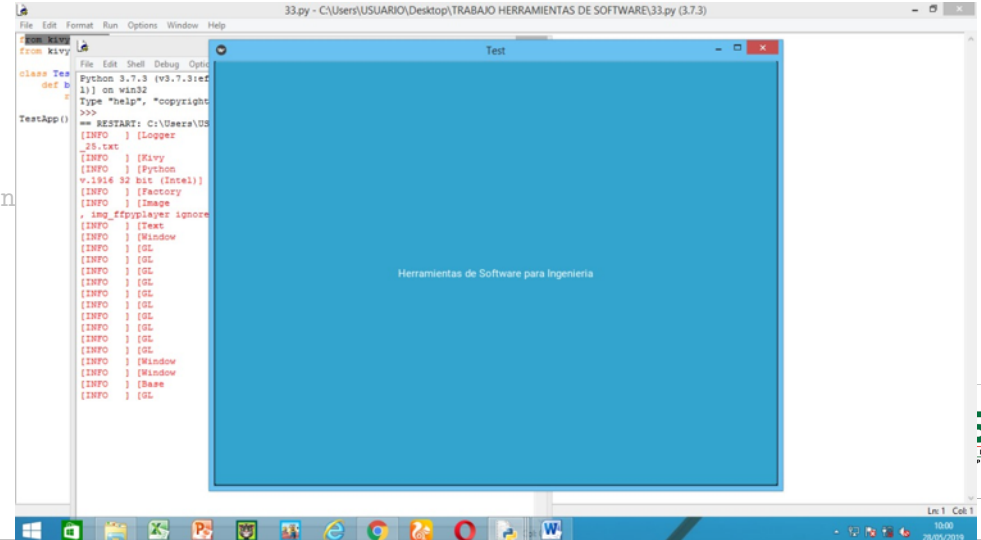
finalizar con la creación de la aplicación móvil vamos a poner correr obteniendo como resultado la aplicación con la frase que hemos ingresado

33.py - C:\Users\USUARIO\Desktop\TRABAJO HERRAMIENTAS DE SOFTWARE\33.py (3.7.3)

File Edit Format Run Options Window Help

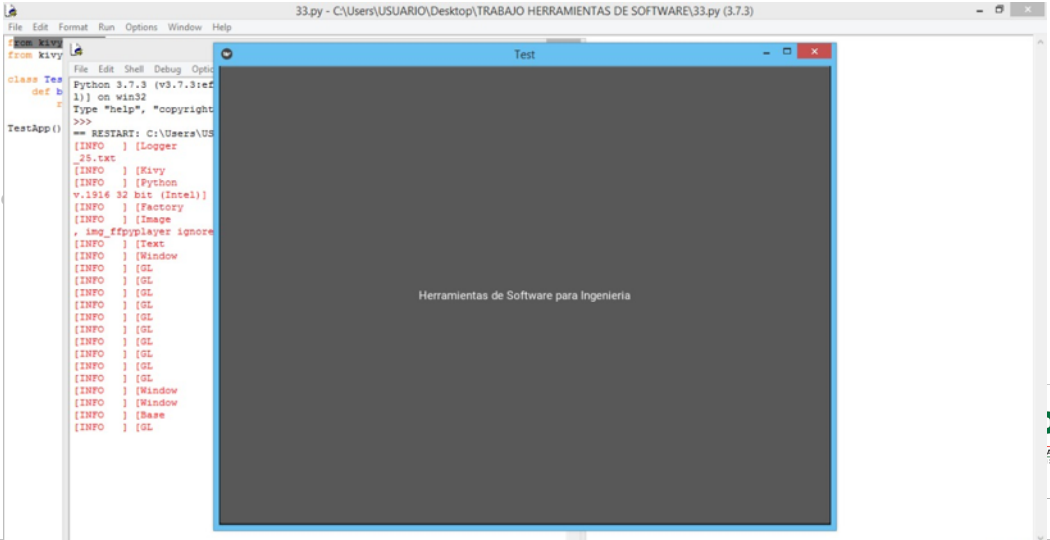
```
from kivy.app import App #SIRVE PARA IMPORTAR LA BIBLIOTECA DE KIVY
from kivy.uix.button import Button # GENERA UN BOTON A PARTIR DE LA DIRECCION UIX DE KIVI

class TestApp(App): #SIRVE PARA TENER METODOS DEFINIDOS POR SU CLASE
    def build(self): # SIRVE PARA CONSTRUIR
        return Button(text='Herramientas de Software para Ingenieria') #ES EL TEXTO QUE IRA DENTRO DE NUESTRO BOTON

TestApp().run() # PARA CORRER EL PROGRAMA
```

1.


33.py - C:\Users\USUARIO\Desktop\TRABAJO HERRAMIENTAS DE SOFTWARE\33.py (3.7.3)
File Edit Format Run Options Window Help
Test
Herramientas de Software para Ingenieria
Python 3.7.3 (v3.7.3:ef
1)] on win32
Type "help", "copyright
>>>
== RESTART: C:\Users\US
[INFO] [Logger
_25.txt
[INFO] [Kivy
[INFO] [Python
v.1916 32 bit (Intel)]
[INFO] [Factory
[INFO] [Image
, img_ffpyplayer ignore
[INFO] [Text
[INFO] [Window
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [Window
[INFO] [Window
[INFO] [Base
[INFO] [GL
Ln: 1 Col: 1

fig/tecn

1.


33.py - C:\Users\USUARIO\Desktop\TRABAJO HERRAMIENTAS DE SOFTWARE\33.py (3.7.3)
File Edit Format Run Options Window Help
from kivy
from kivy
class Tes
def b
r
TestApp()
File Edit Shell Debug Optic
Python 3.7.3 (v3.7.3:ef
1)] on win32
Type "help", "copyright
>>>
== RESTART: C:\Users\US
[INFO] [Logger
_25.txt
[INFO] [Kivy
[INFO] [Python
v.1916 32 bit (Intel)]
[INFO] [Factory
[INFO] [Image
, img_ffpyplayer ignore
[INFO] [Text
[INFO] [Window
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [GL
[INFO] [Window
[INFO] [Window
[INFO] [Base
[INFO] [GL
Test
Herramientas de Software para Ingenieria