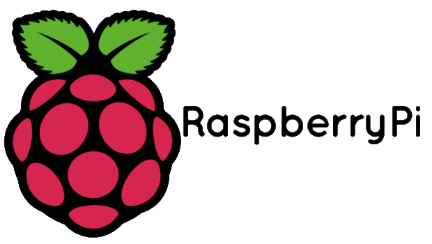
**Connexion de Raspberry Pi à la base de données FireBase à l'aide de Python3**

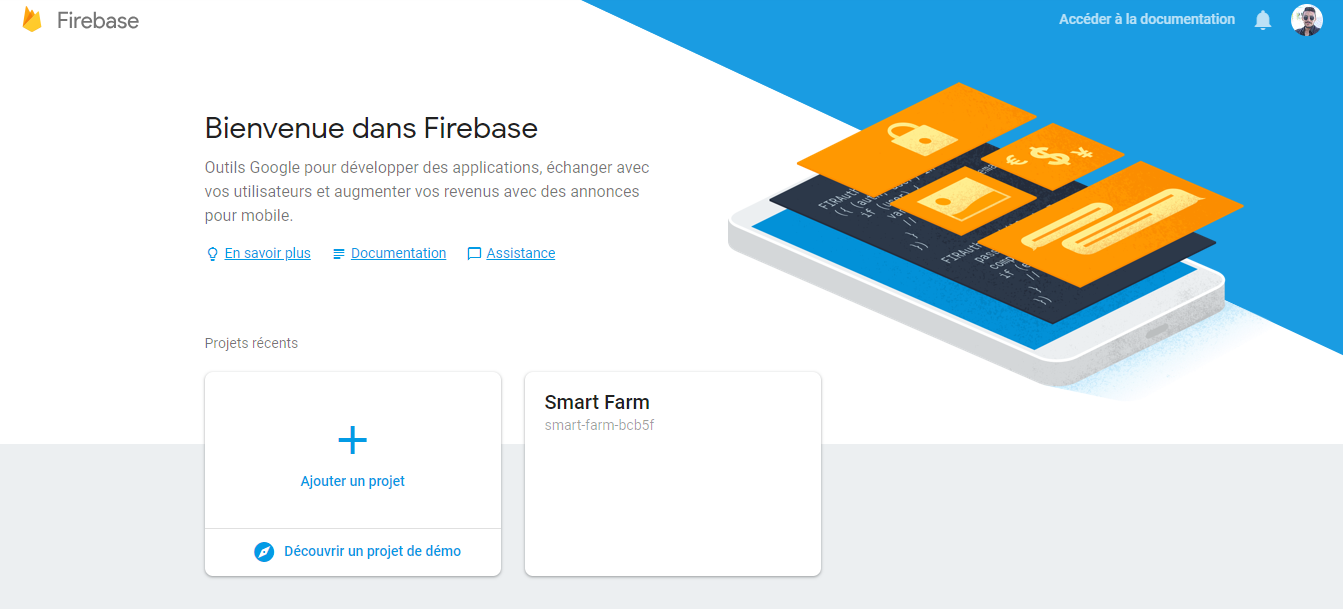
 

#### 1)Configuration de votre Raspberry Pi :

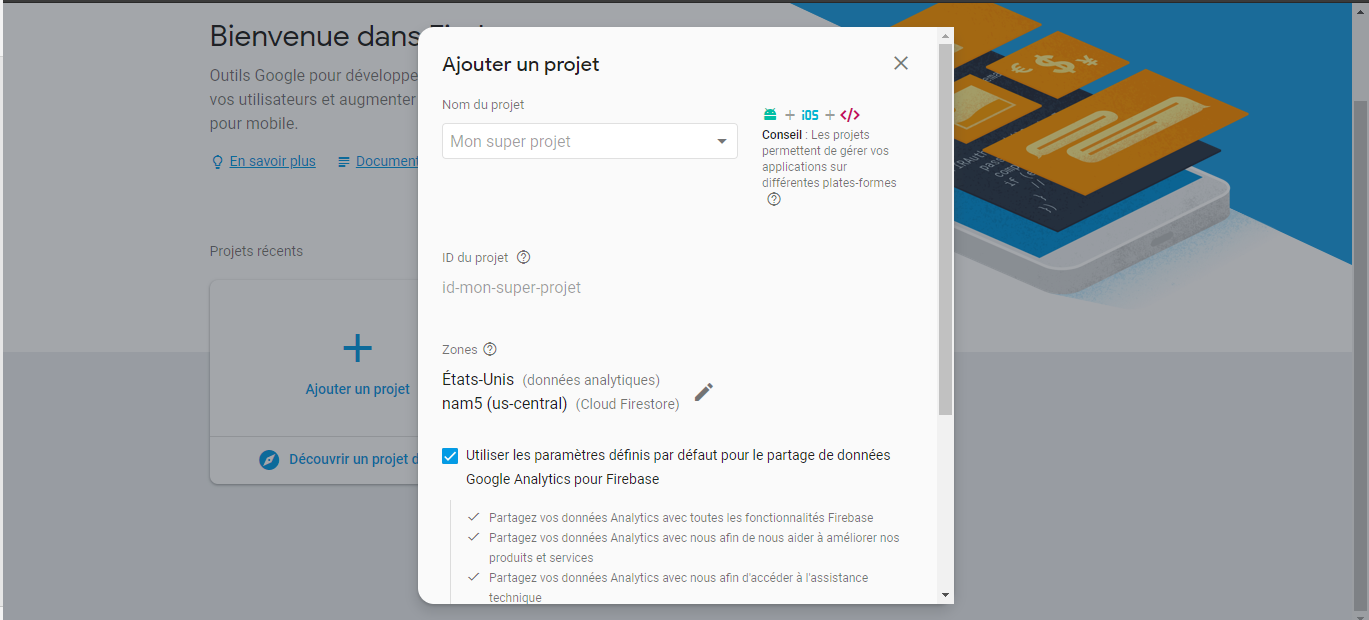
1. sudo apt-get update
2. sudo apt-get upgrade
3. sudo apt-get install python3-dev
4. sudo pip3 install requests==1.1.0
5. sudo pip3 install python-firebase

#### 2)Création d’un compte FireBase :

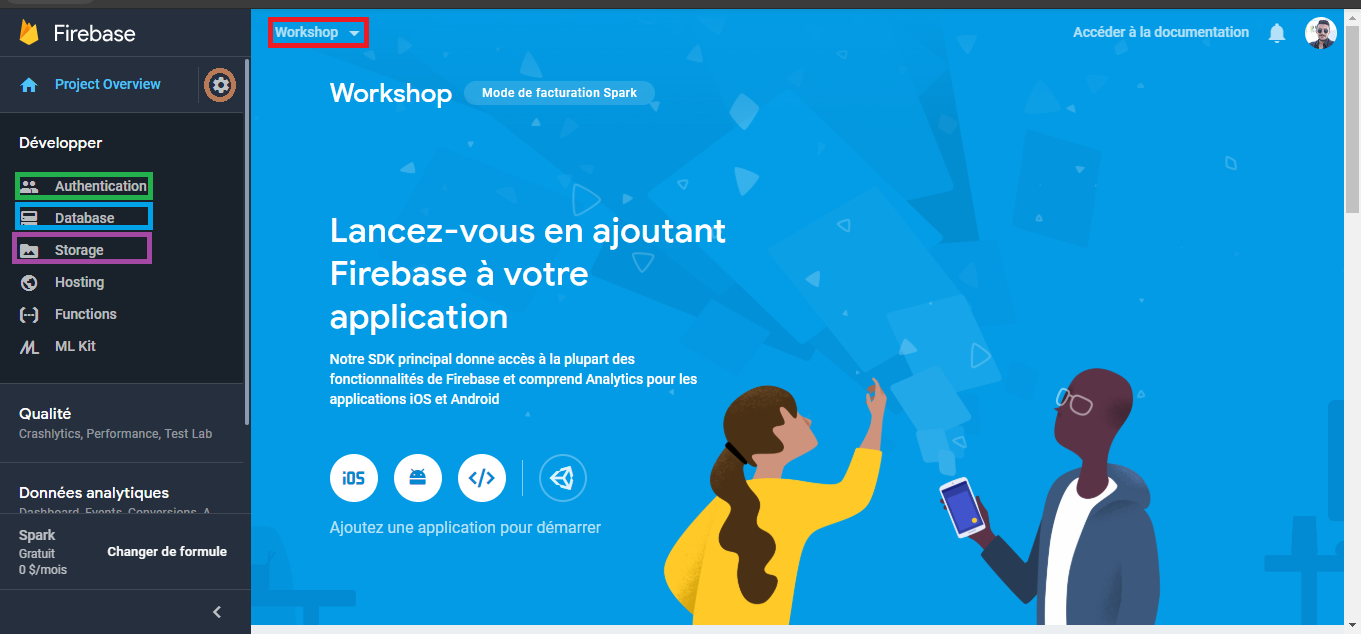
* Lien vers FireBase : <https://firebase.google.com/>
* Connecter-Vous avec un compte Gmail.
* Ajouter Un projet :

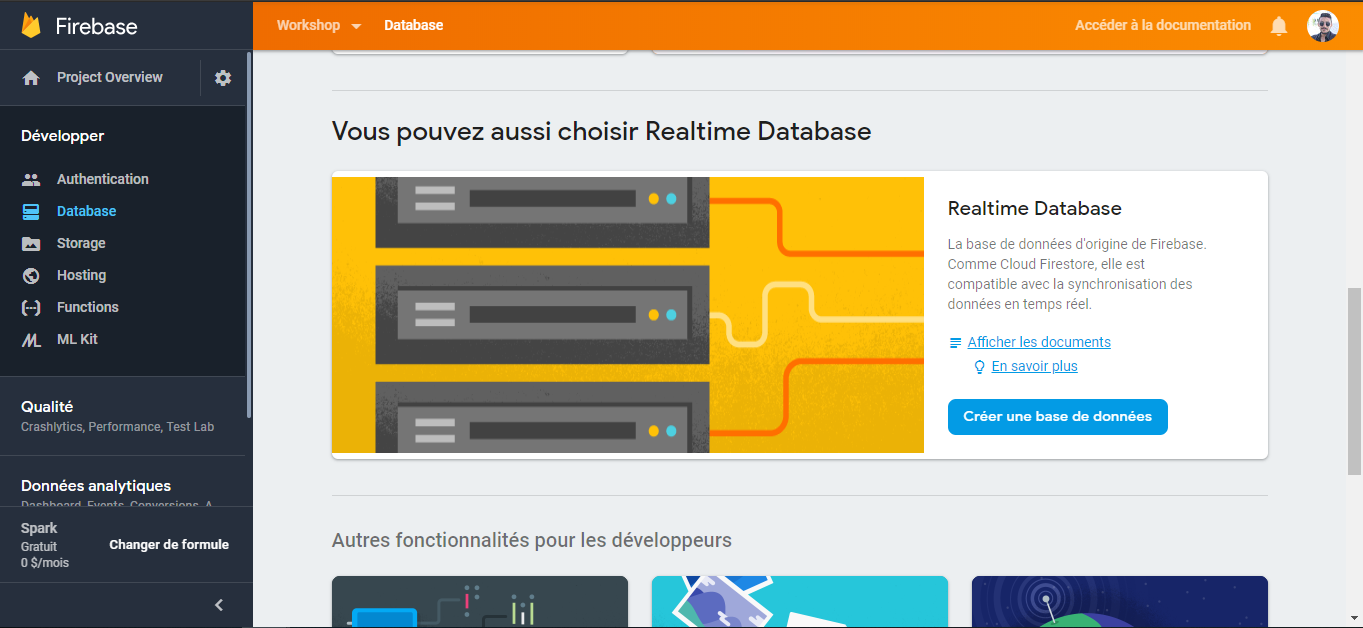


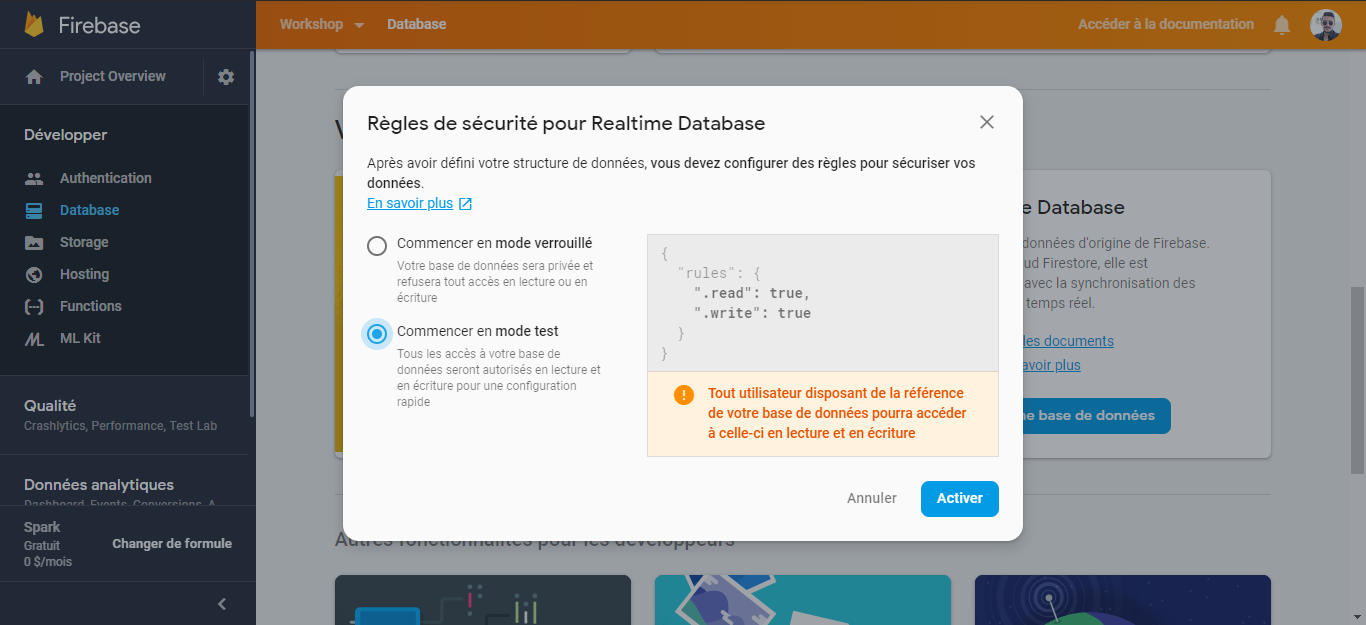
* Choisissez un nom pour votre projet :



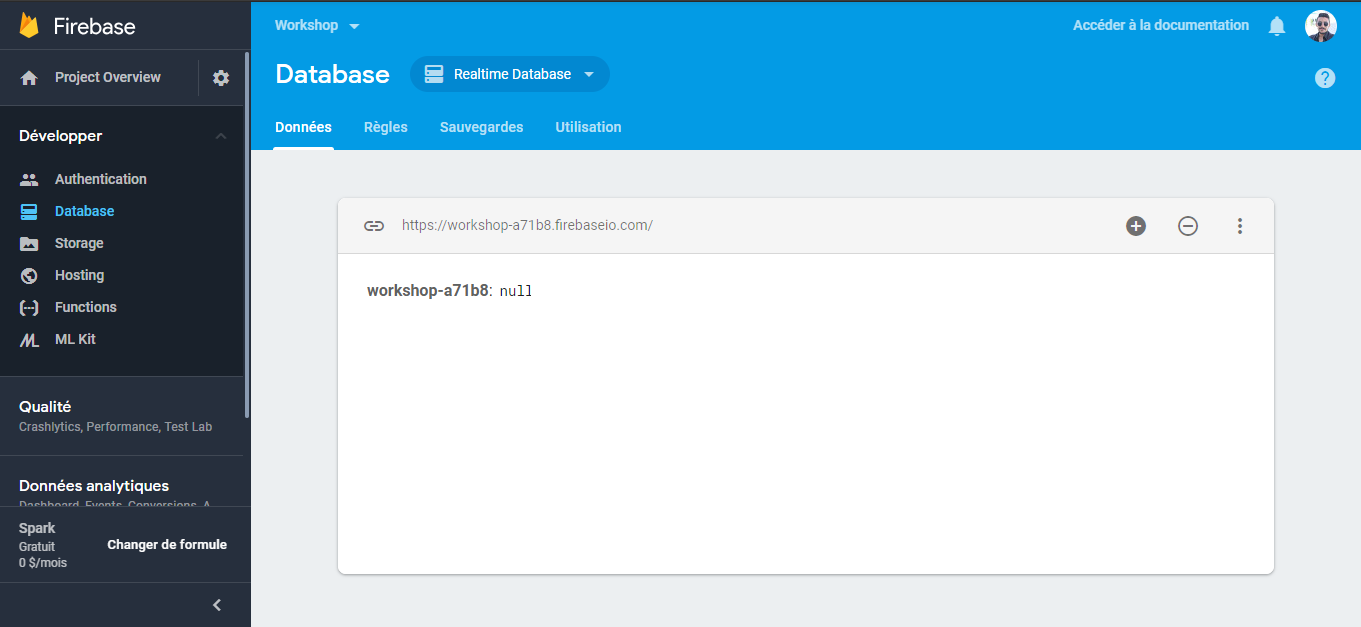
* Dashboard FireBase :



* Créez une « Real-Time Database » 
* Choisissez commencer en mode test :



* La base de données est maintenant vide :



#### 3)Premiers pas avec « Real-Time Database » :

1. Connexion à la carte Raspberry pi avec SSH (exemple : PuTTY).
2. Créez une nouvelle ficher python avec la commande « nano ».

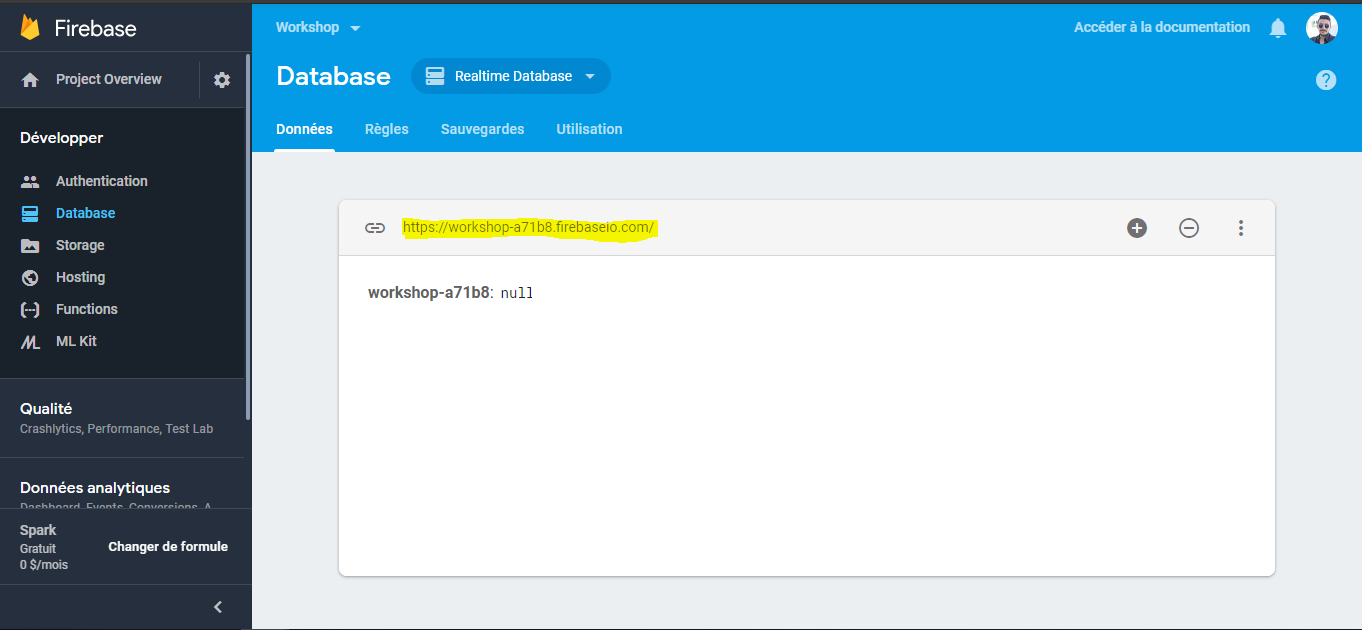
pi@raspberry: ~$ nano firetest.py

1. Dans cette ficher on va ajouter les bibliothèques suivantes :

* from firebase import firebase
* import requests
* import json
* import os
* import RPi.GPIO as GPIO
* from time import sleep

1. Ajouter le ligne ci-dessous pour assurer la liaison avec votre base de données :

firebase = firebase.FirebaseApplication('ajouter ici l’ID de votre base', None)



1. Maintenant on va créez une fonction nommée « update\_firebase » dans notre ficher firetest.py :

def update\_firebase():

data = {"variable1" : "hello world"}

firebase.patch('/', data)

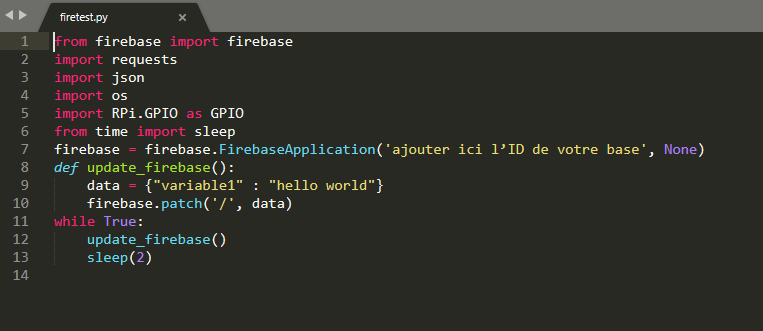
1. On va créez une boucle infinie pour l’envoi de de données vers notre base :

while True:

update\_firebase()

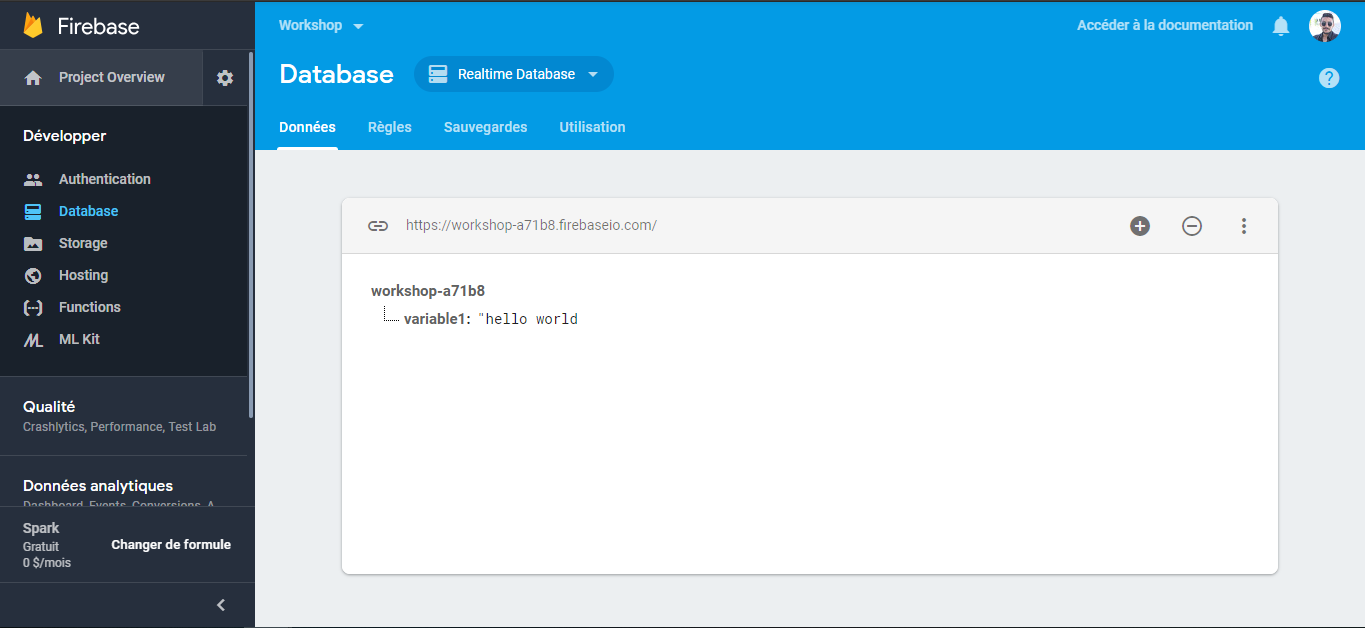
sleep(2)

* Notre fichier va être comme suit :



------> Tester votre code avec la commande : python3 firetest.py

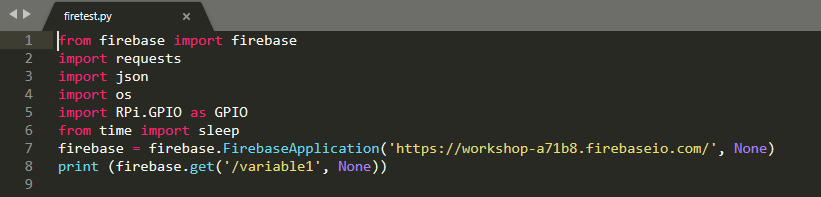
* La Résultat :



1. Récupération d’une valeur de Firebase :

C’est le même principe mais avec la commande :

* firebase.get('/variable1', None)



* La Résultat :

