**MAJOR PROJECT-2**

**IoT Chatbot**

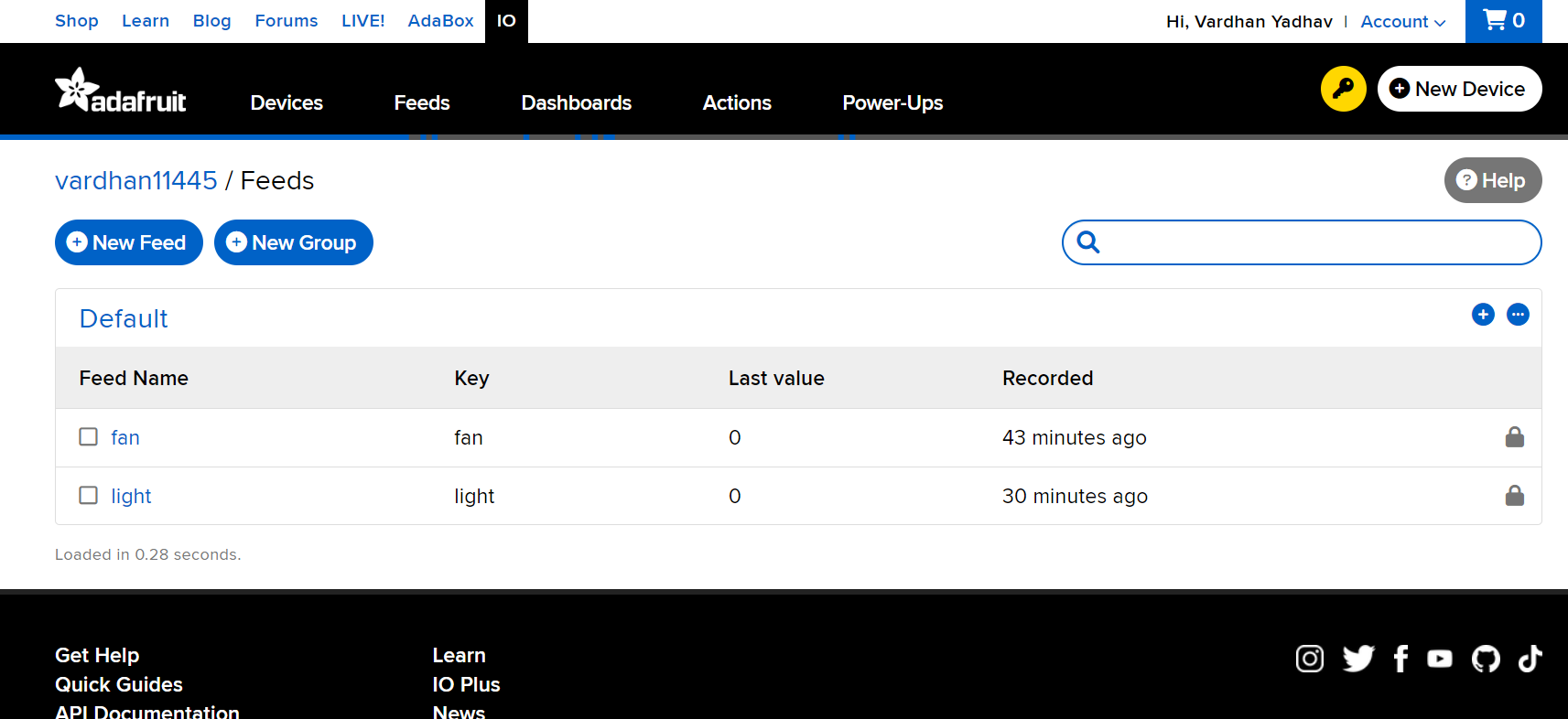
Smart Home Automation

**Step-1:**

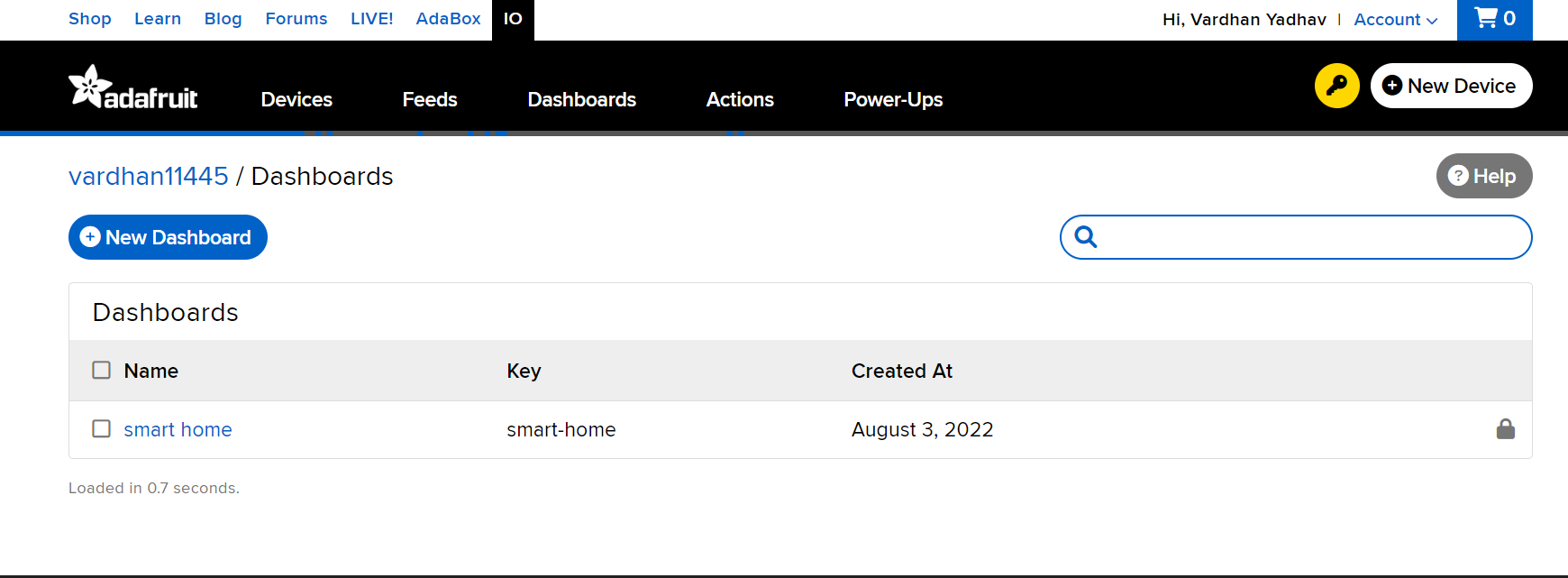
Create an account in adafruit website.

**Step-2:**

Create two feeds in adafruit and give names fan and lights.



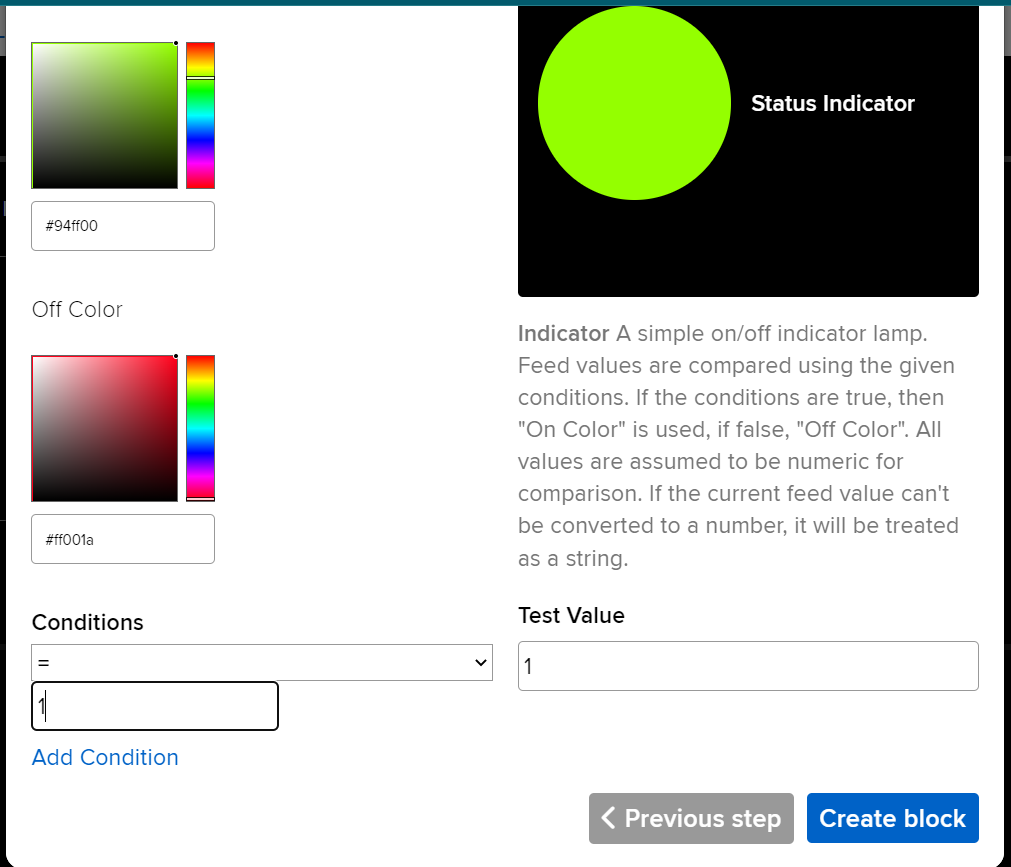
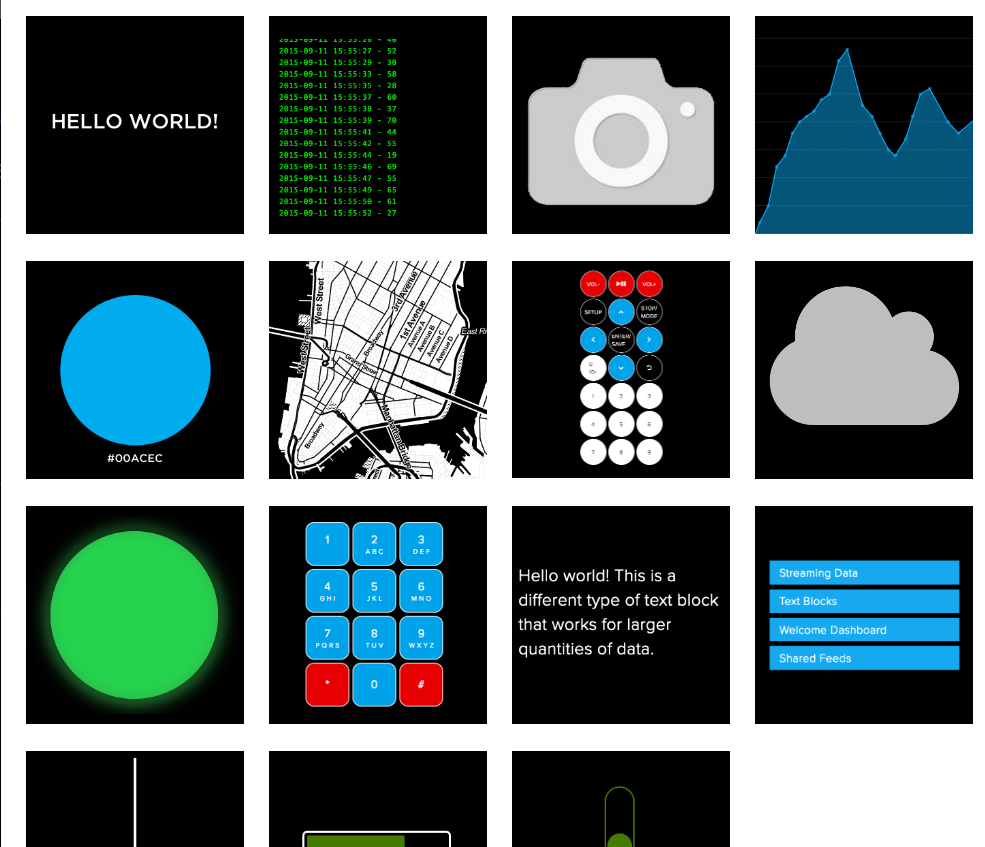
**Step-3:**

Create a dashboard in adafruit and give it a nam

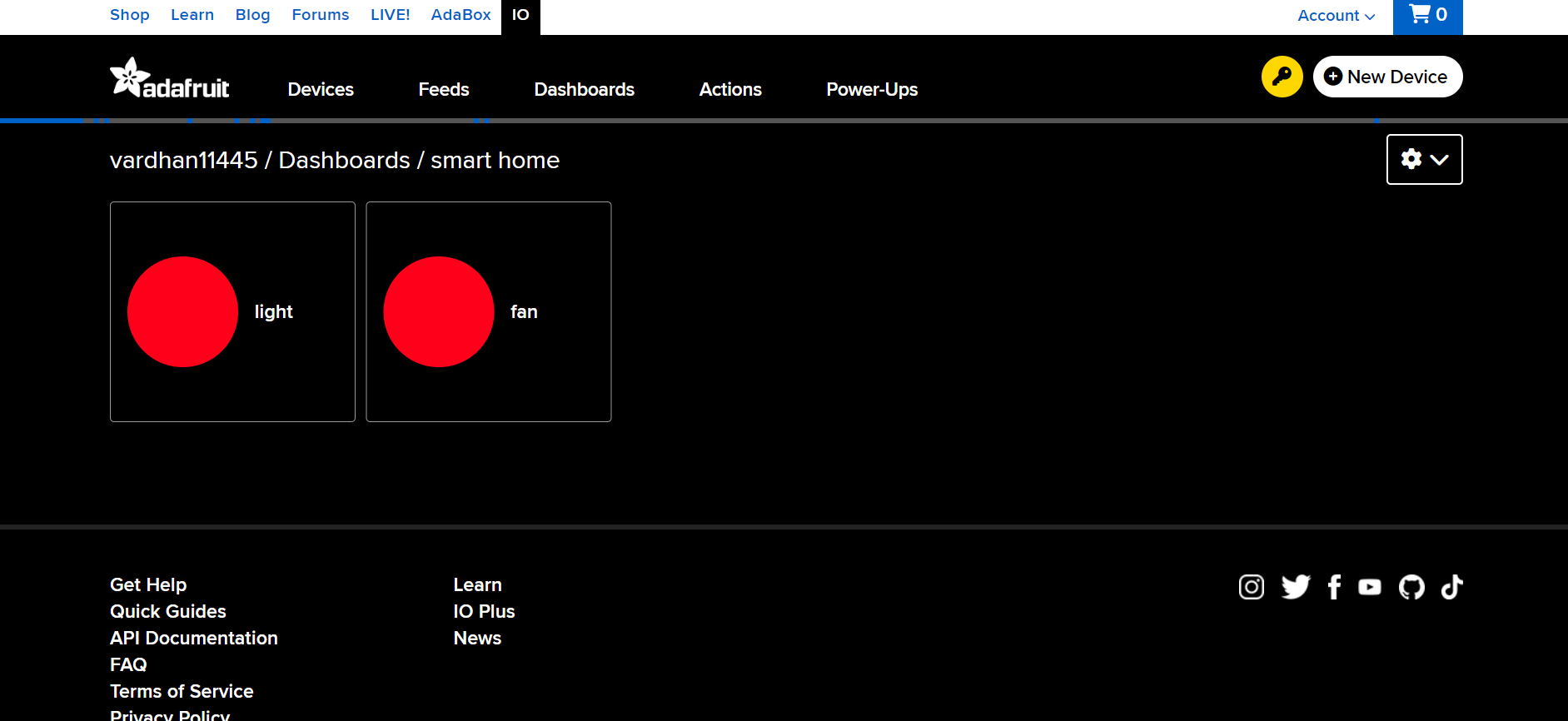
**Step-4:**

Create two new indicator blocks in dashboard one for fan feed and other for

lights feed

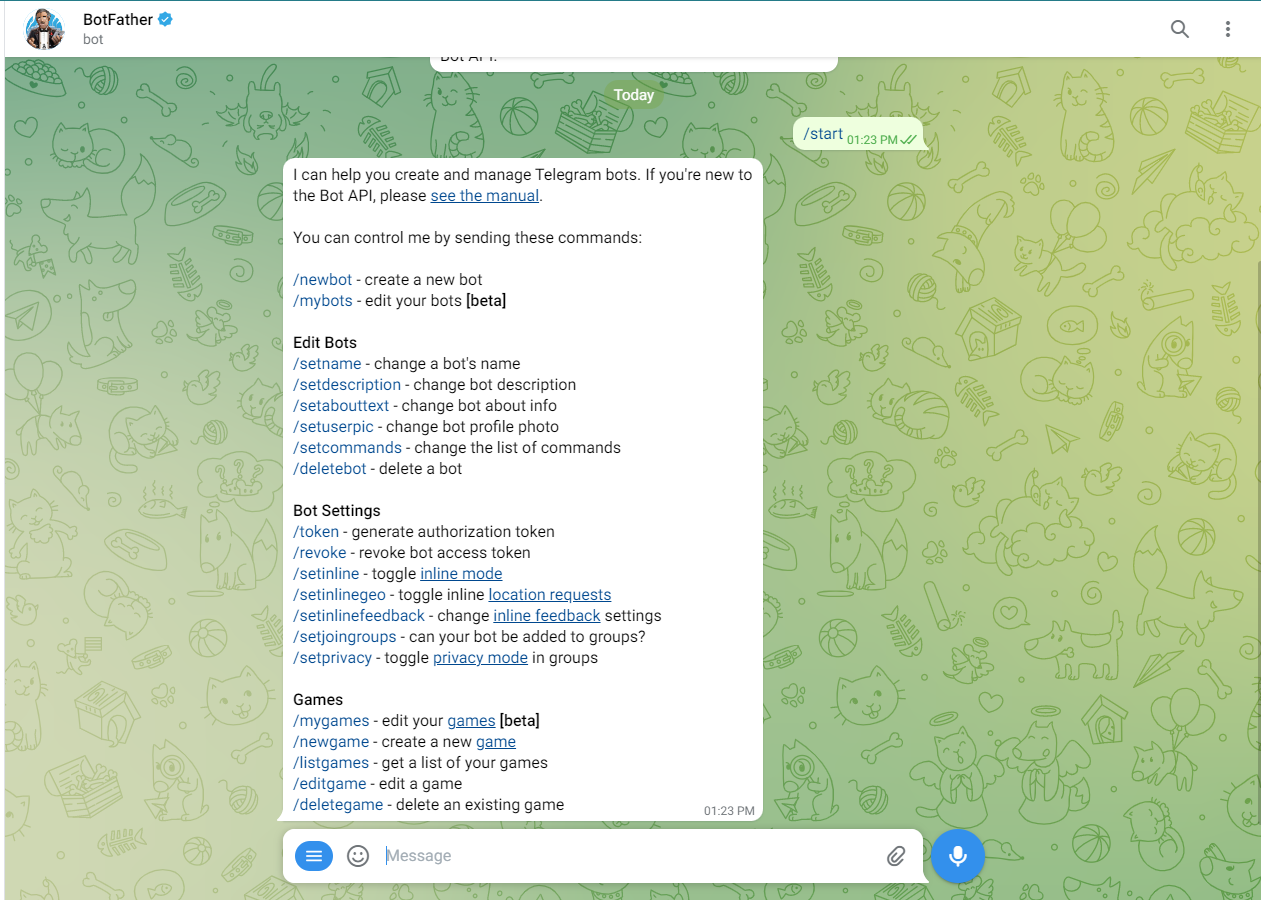


Final dashboard will appear

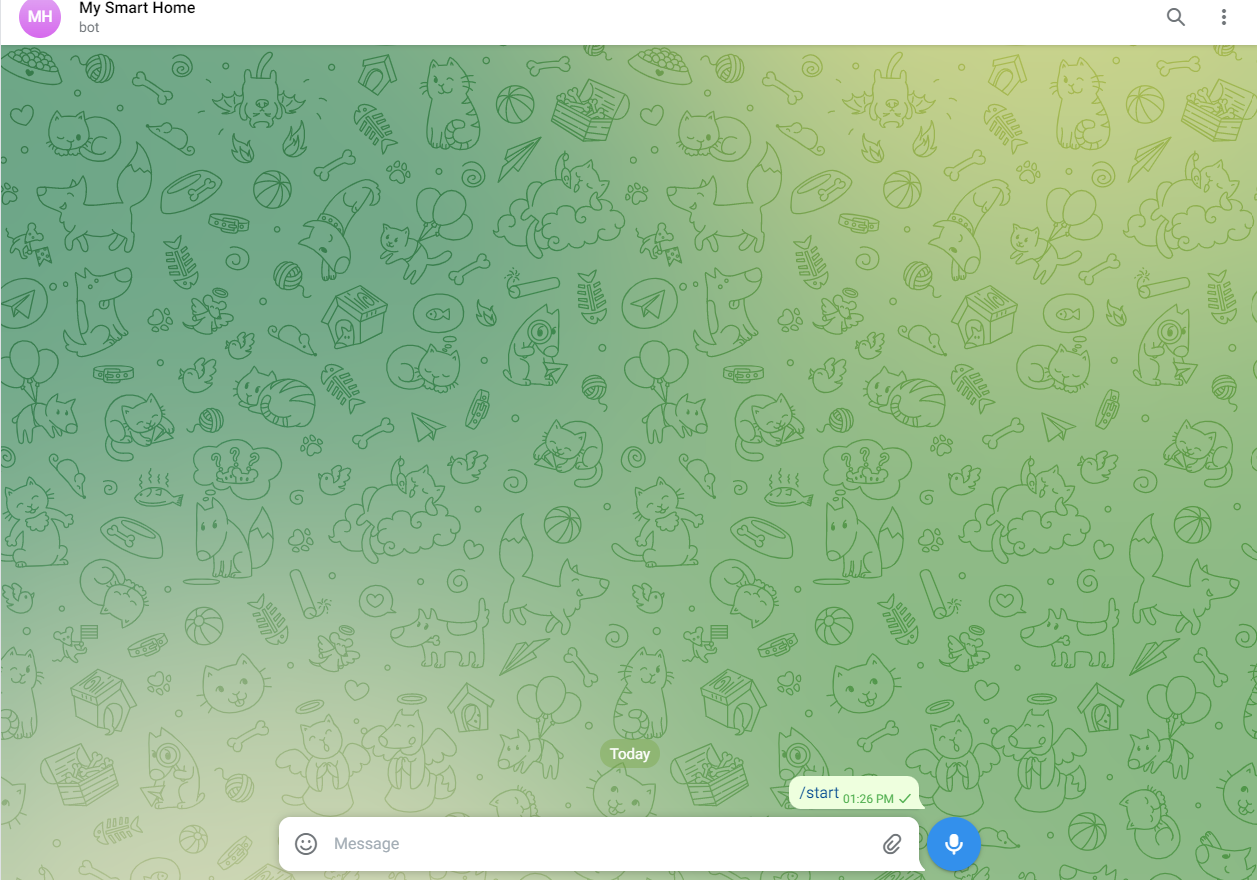


**Step-5:**

In telegram search for BotFather and click on start



Select the option newbot and give it a name and unique bot id.Your bot account was ready.Open your bot and click on start.

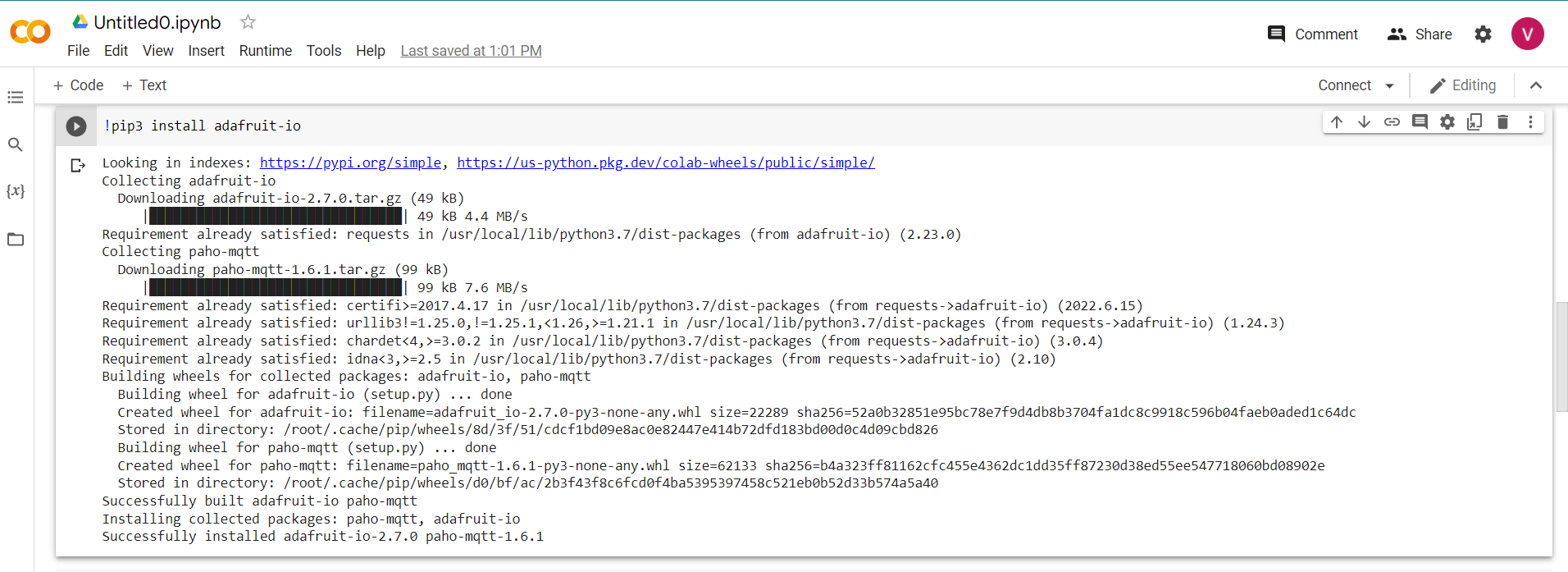


**Step-6:**

Now in any python IDLE or in google colaboratory run the following commands

one after the other.It install required libraries

!pip install adafruit-io



!pip install python-telegram-bot --pre

**Step-7:**

Then add the following code to the IDLE

#import required libraries for telegram

from telegram import Update

from telegram.ext import ApplicationBuilder, CommandHandler, ContextTypes

#import required libraries for adafruit

from Adafruit\_IO import Client, Feed ,Data

#make a telegram bot using botfather and enter the API key here

app = ApplicationBuilder().token("5454771209:AAEYDCjO\_qKbQ2sCMDcxfgKZsKAWDnE6I18").build()

#make commands using telegram bot and follow through adafruit API\_IO\_KEY and give specified value to feeds

#for lights to turn on

async def turn\_on\_lights(update: Update, context: ContextTypes.DEFAULT\_TYPE) -> None:

await update.message.reply\_text("Lights turned on")

aio = Client('vardhan11445', 'aio\_eswW77cGORK72xNm7or4b8sLHL33')

aio.send('light', 1)

data = aio.receive('light')

#for lights to turn off

async def turn\_off\_lights(update: Update, context: ContextTypes.DEFAULT\_TYPE) -> None:

await update.message.reply\_text("Lights turned off")

aio = Client('vardhan11445', 'aio\_eswW77cGORK72xNm7or4b8sLHL33')

aio.send('light', 0)

data = aio.receive('light')

#for lights to fan off

async def turn\_on\_fan(update: Update, context: ContextTypes.DEFAULT\_TYPE) -> None:

await update.message.reply\_text("fan turned on")

aio = Client('vardhan11445', 'aio\_eswW77cGORK72xNm7or4b8sLHL33')

aio.send('fan', 1)

data = aio.receive('fan')

#for lights to fan off

async def turn\_off\_fan(update: Update, context: ContextTypes.DEFAULT\_TYPE) -> None:

await update.message.reply\_text("fan turned off")

aio = Client('vardhan11445', 'aio\_eswW77cGORK72xNm7or4b8sLHL33')

aio.send('fan', 0)

data = aio.receive('fan')

async def turn\_on\_fan(update: Update, context: ContextTypes.DEFAULT\_TYPE) -> None:

await update.message.reply\_text("fan turned on")

aio = Client('vardhan11445', 'aio\_eswW77cGORK72xNm7or4b8sLHL33')

aio.send('fan', 1)

data = aio.receive('fan')

#to read commands from bot channel

app.add\_handler(CommandHandler("turnonlights",turn\_on\_lights))

app.add\_handler(CommandHandler("turnofflights",turn\_off\_lights))

app.add\_handler(CommandHandler("turnofffan",turn\_off\_fan))

app.add\_handler(CommandHandler("turnonfan",turn\_on\_fan))

app.run\_polling()

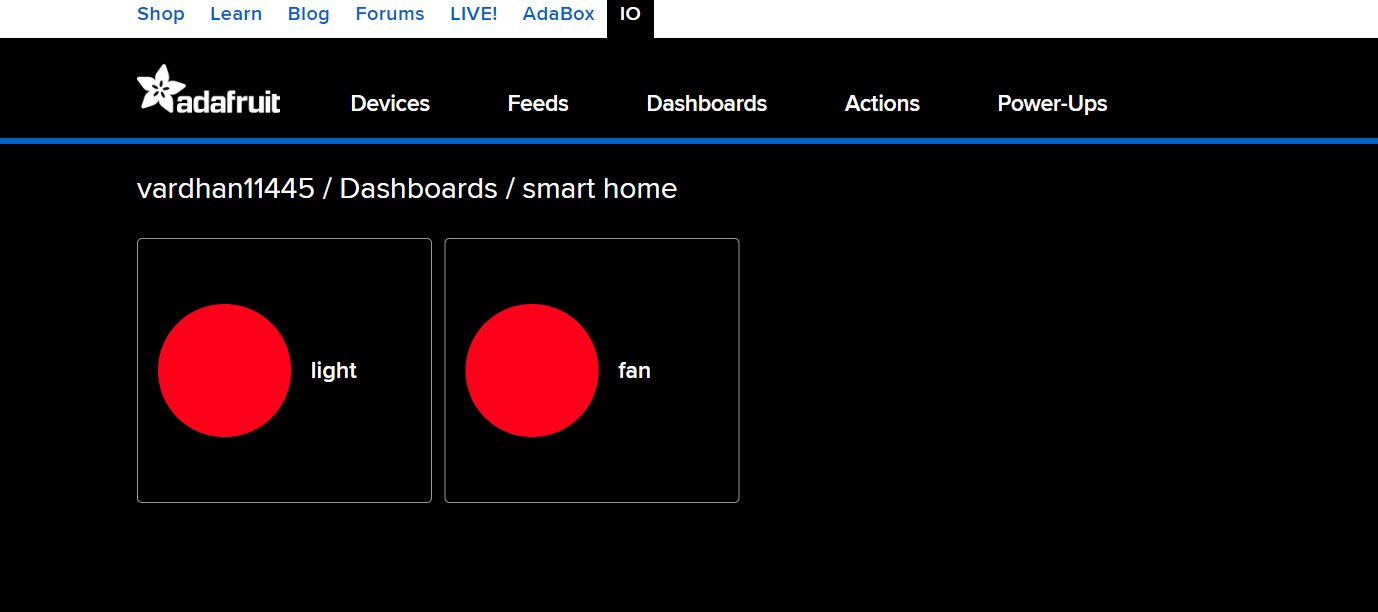
Here in client section add your user name and API key of bot channel created before.

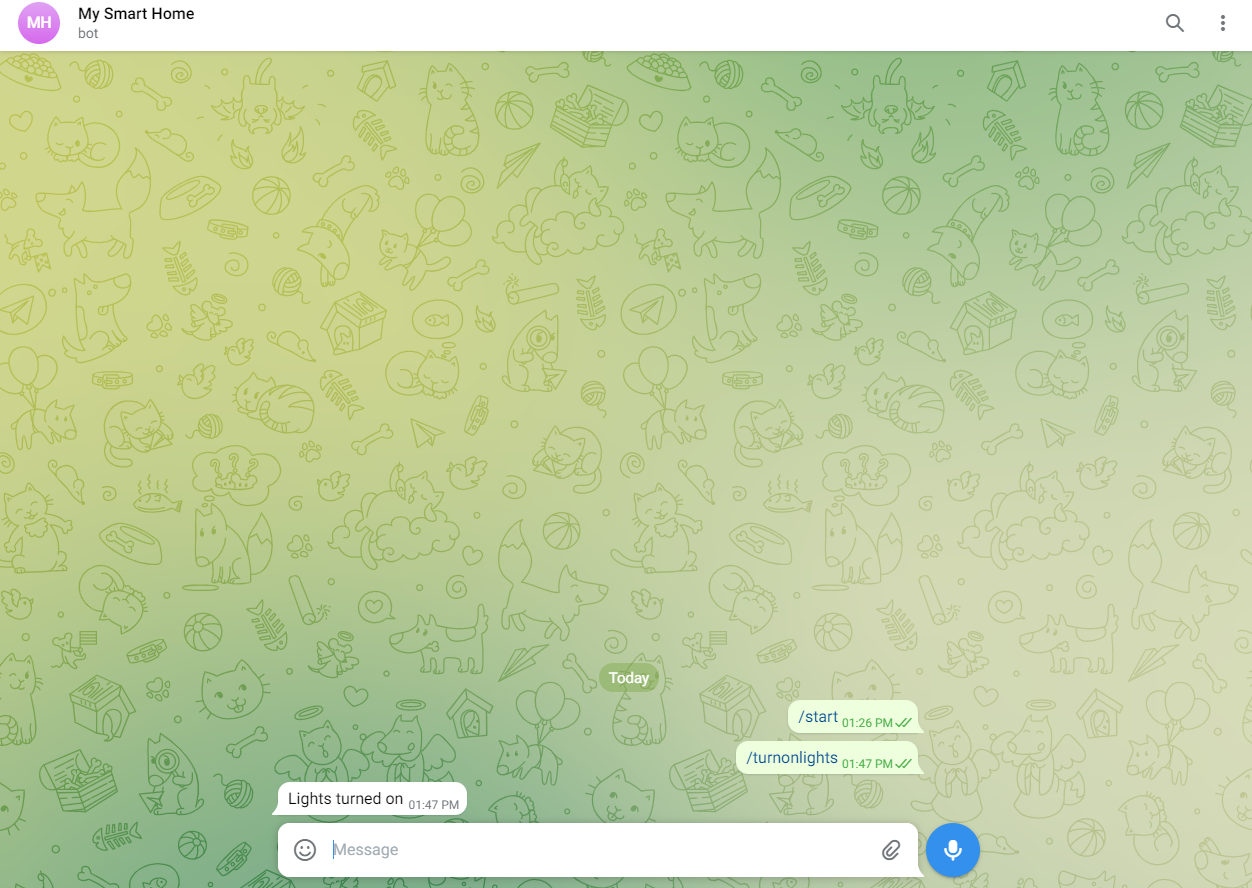
I added four commands and linked it with the adafruit feeds.

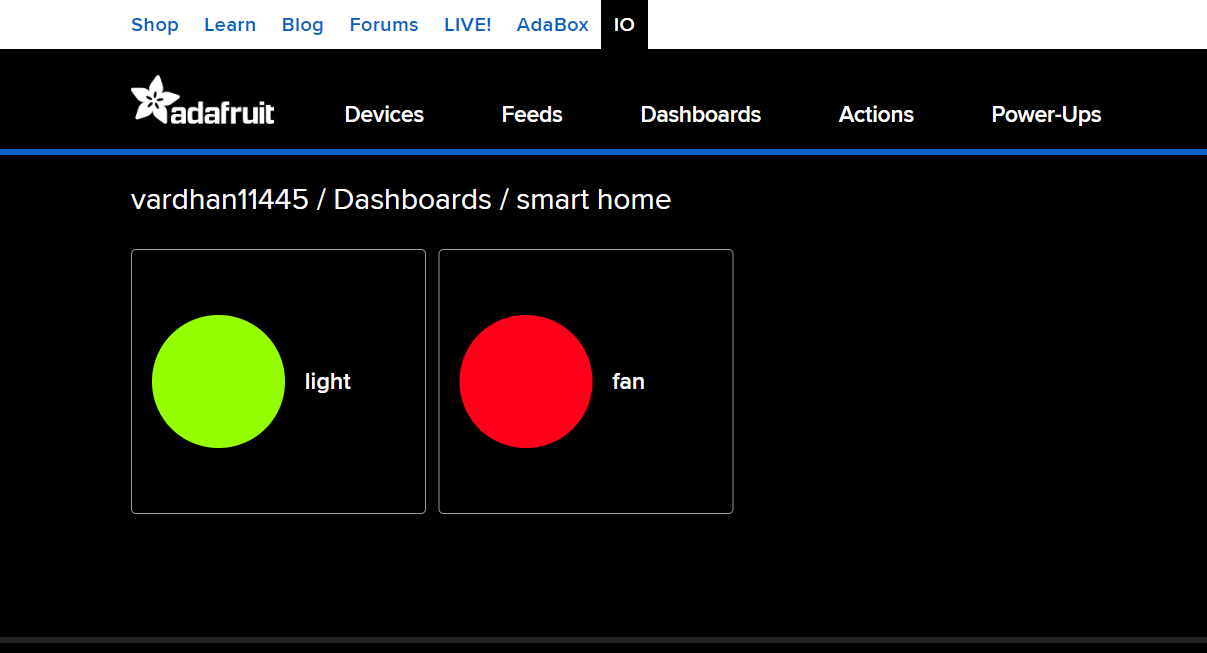
**Step-8:**

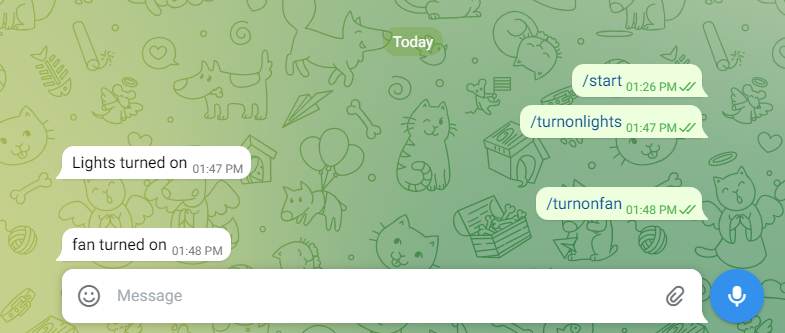
Run the code and open your telegram and adafruit dashboard to see the outputt

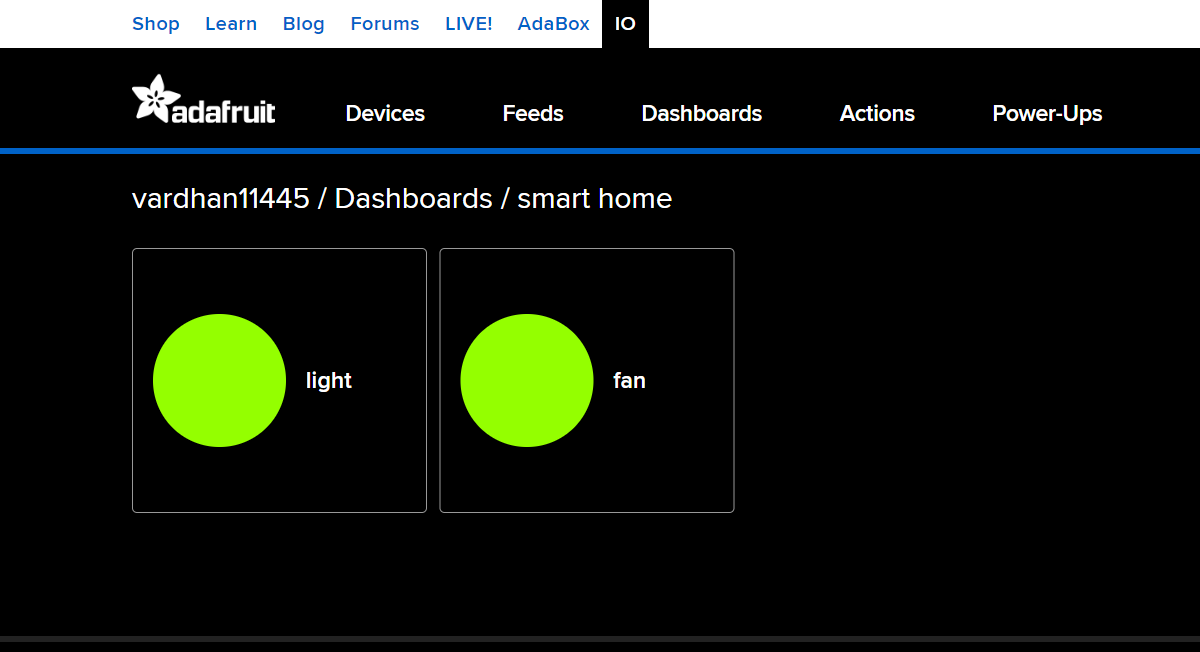
Initially the dashboard will look as shown

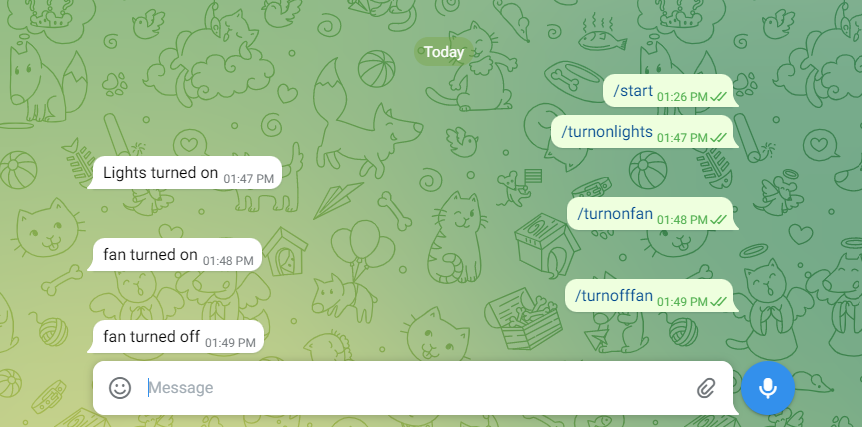


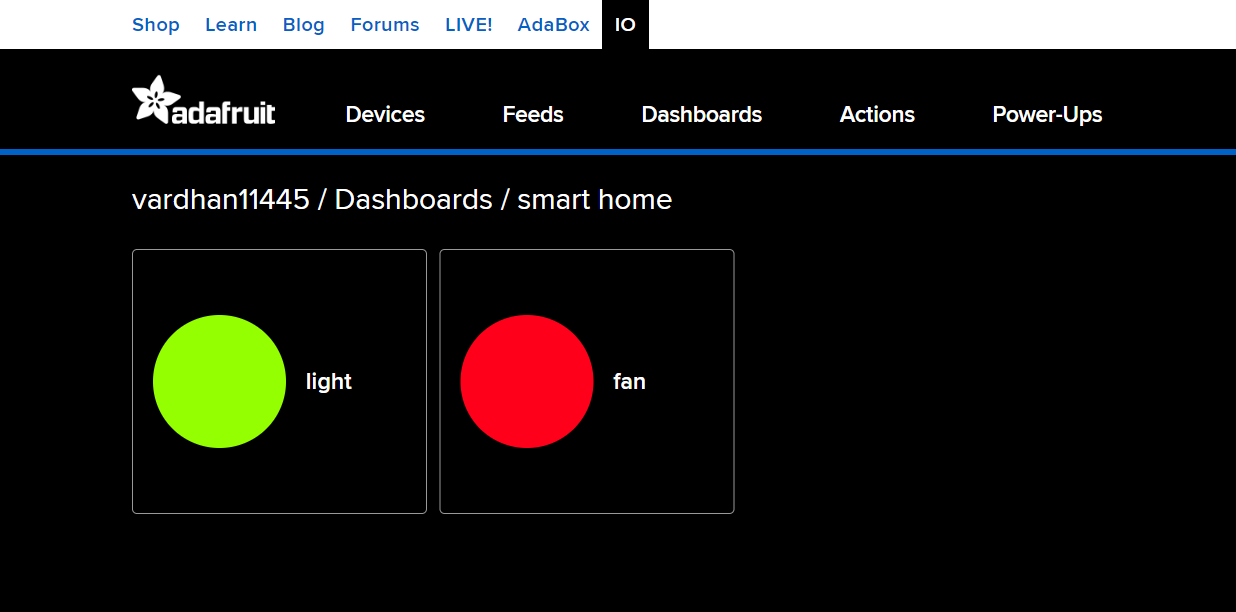
After entering your first command in telegram it looks as shown

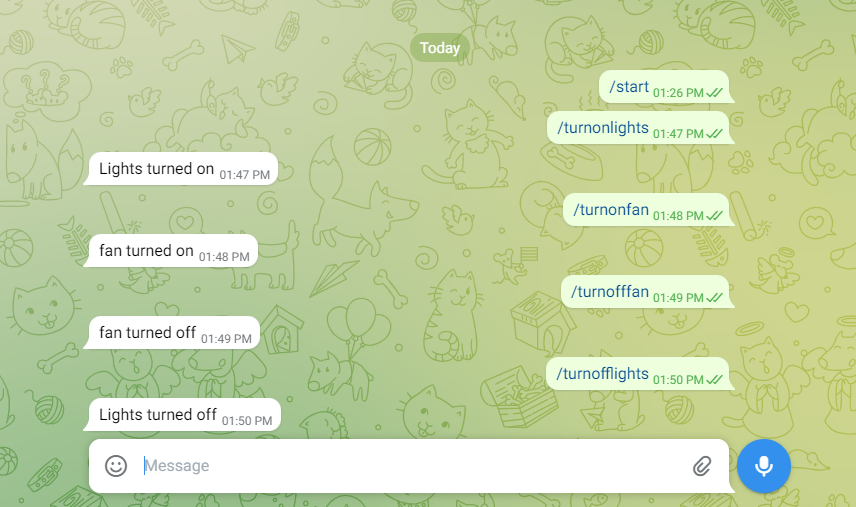


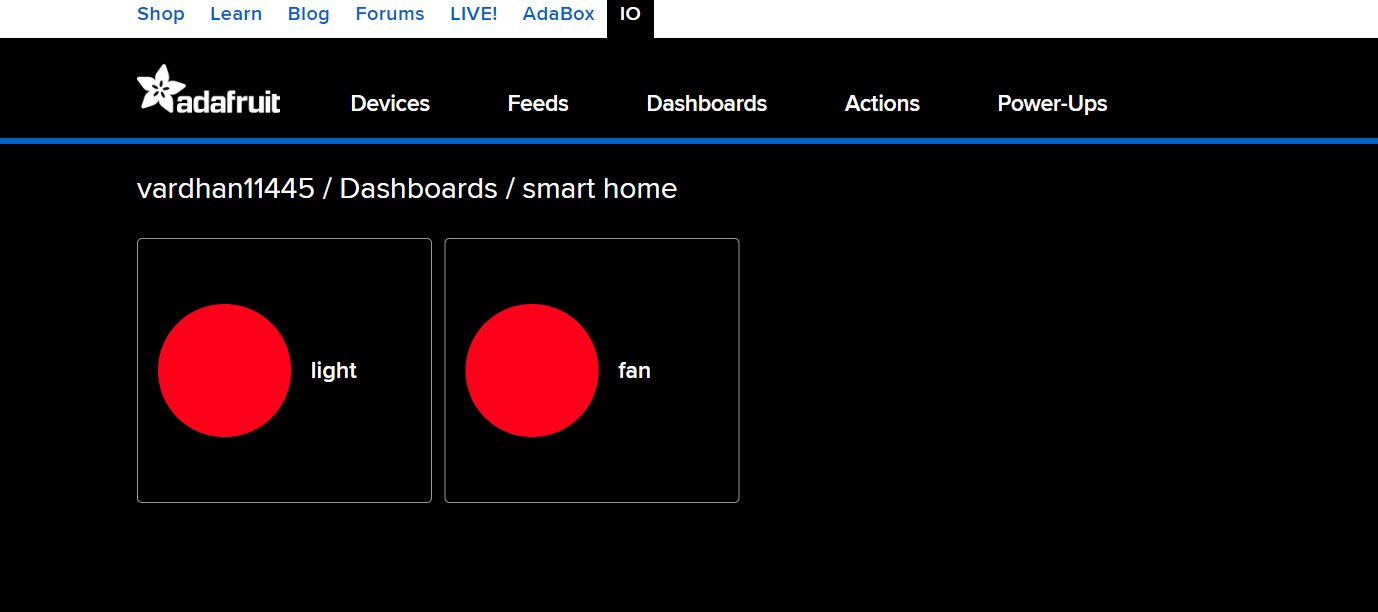
Enter turn on fan



Turn off your fan



Turn off your light



You can repeat this any number of times until your code was running in back.

**NOTE**: When you do the coding part in google colab for some users it shows runtime

error