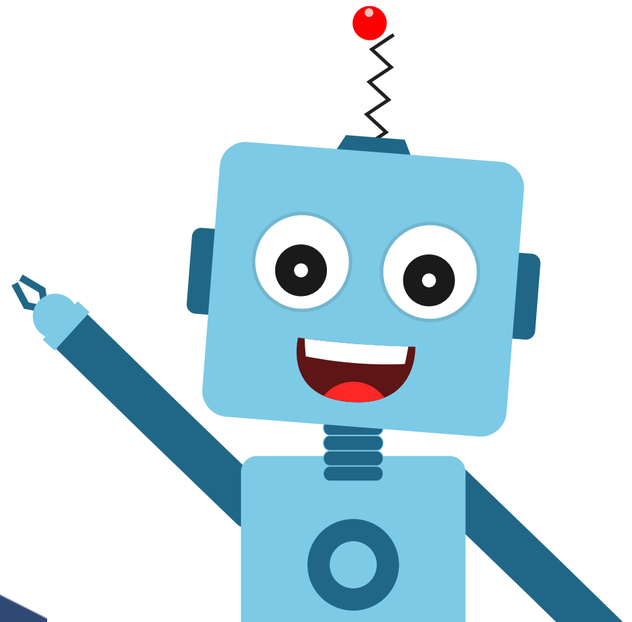
POWERED BY STEMpedia

Getting Started with Quarky

Session 4



www.ai.thestempedia.com

Topics covered

POWERED BY STEMpedia

- Iintroduction to Quarky
- **Activity 1**: Display emotions with Quarky
- **Activity 2**: Name badge with Quarky

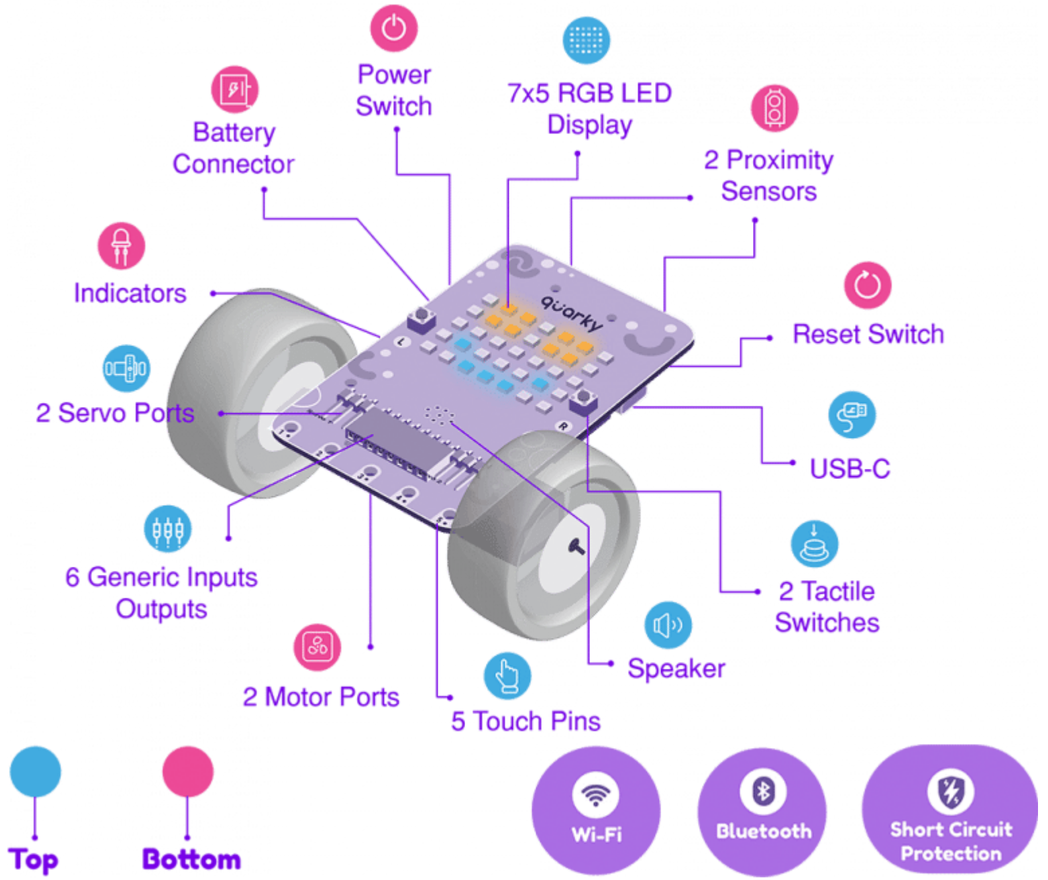
Introduction to Quarky

Quarky: Your AI Learning Buddy

POWERED BY STEMpedia

- Quarky is an electronic board with a lot of features using which you can make numerous exciting projects.

- You can program Quarky to make expressions, play sounds, detect touch, and much more.

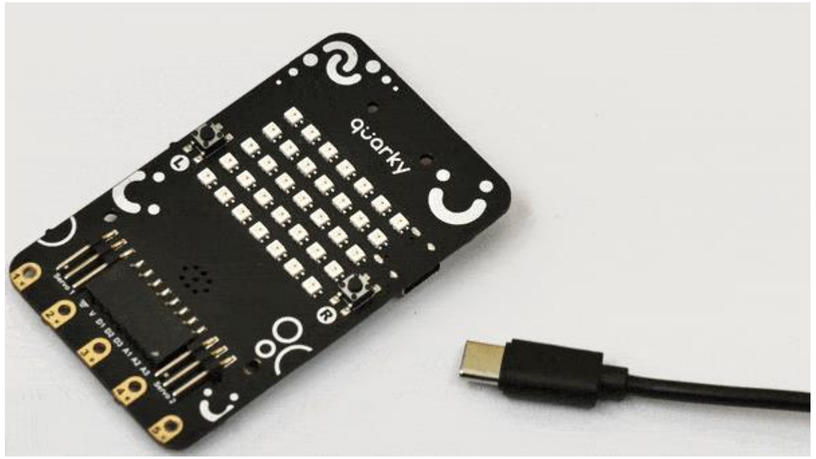


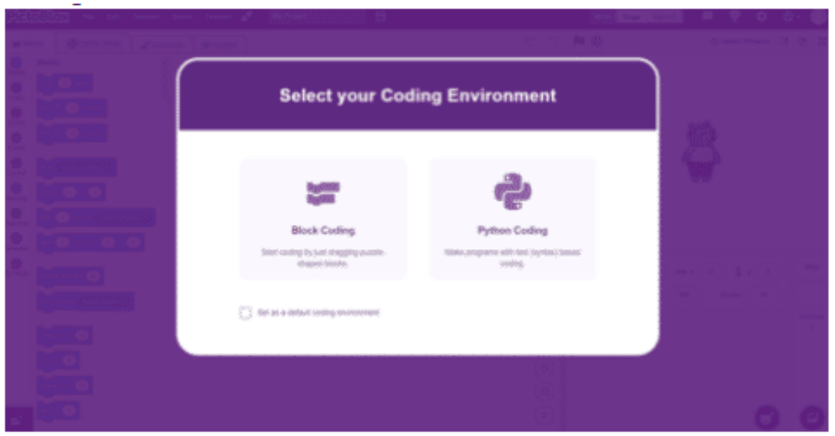
Quarky Meets PictoBlox

POWERED BY STEMpedia

- You can control Quarky using PictoBlox, by connecting both of them together.

- Let’s begin by first connecting Quarky to PictoBlox.
  - First, connect Quarky to your laptop using C Type cable.
  - OR, connect Quarky to your laptop using Bluetooth.
  - Next, open PictoBlox on your desktop.
  - After that, select Python Coding as your coding environment.





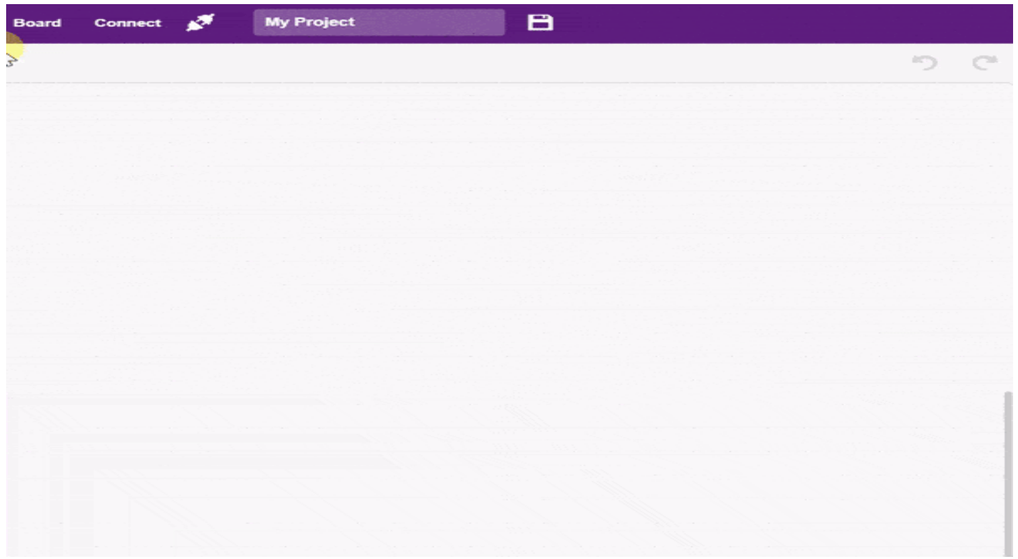
Quarky meets Pictoblox

POWERED BY STEMpedia

- Then, click the Board button from the toolbar, and select board: Quarky.

- Next, select the appropriate serial port from Serial Ports if you want to connect Quarky via C Type Cable.

- If you want to connect Quarky via Bluetooth, you must download an app called PictoBlox Link on your computer.

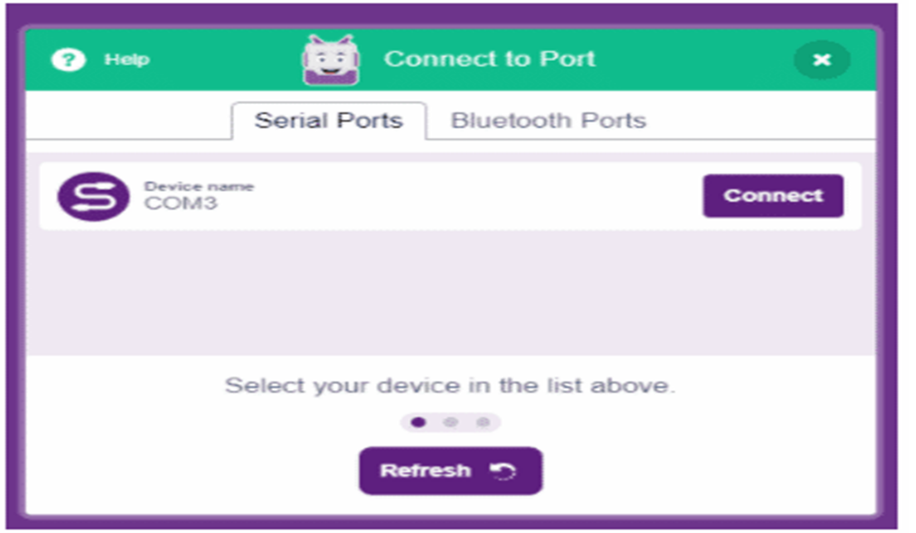


Quarky meets Pictoblox

POWERED BY STEMpedia

- You need to activate this app every time you want to connect with your quarky (or any other board) via Bluetooth.

- Now you can click on Bluetooth Ports and press Connect on the appropriate Device name.

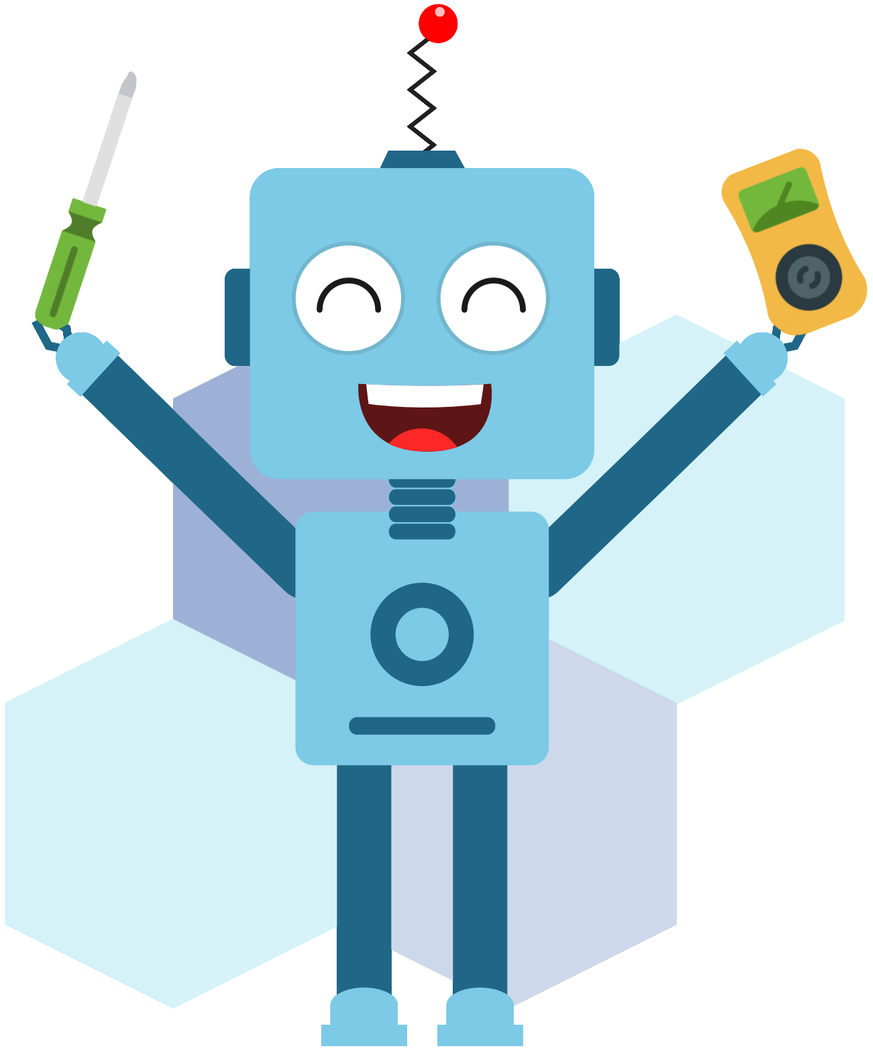


ACTIVITY

POWERED BY STEMpedia

Display emotions with Quarky

Did you know that our friend Quarky can be expressive just like us? We can tell what it's feeling just by looking at its eyes - which happen to be an RGB LED matrix! We're going to use this matrix to display different expressions, so that Quarky can show us how it's feeling.

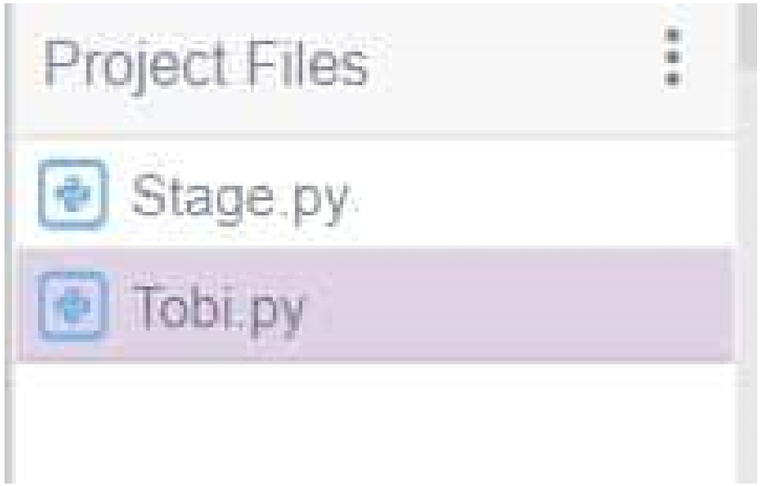


Lets Code

POWERED BY STEMpedia

- Let’s begin by first connecting Quarky to PictoBlox:

  - First, connect Quarky to your laptop using the Bluetooth.
  - Open PictoBlox and create a new file. Select the coding environment as Python Coding.
  - Select the Board as Quarky. Next, select the Serial port to connect Quarky and press Connect.



Let's Code

POWERED BY STEMpedia

- Now, select the Tobi.py file from the Project Files section and by default, the syntax will be written in sprite as an object.

```
sprite = Sprite('Tobi')
```

- We need to use time module in python:
  - As we want to have certain time gap between display of two emotions on Quarky.
  - Python time module allows to work with time in Python.
  - So before starting with this module we need to import it

```
import time
```

Let's Code

POWERED BY STEMpedia

- We would be using Quarky in this activity, so we would also be writing quarky related functions and for writing those functions we need to define an object for quarky in the same manner as we did for the sprite.

```
quarky = Quarky()
```

- Let us now see how to define a function for a particular emotion to be displayed on Quarky :

    - For Inside the while loop, make Quarky show an emotion by typing.

```
quarky.showemotion("emotion"):
```

Let's Code

POWERED BY STEMpedia

- For displaying happy emotion on quarky we will be using showemotion() function along with quarky's object which we have defined in the beginning.

```
quarky.showemotion('happy')
```

- Now, for ensuring that the emotion is displayed atleast for 2 seconds on quarky we will be using time.sleep() function.

```
time.sleep(2)
```

Let's Code

POWERED BY STEMpedia

- After a particular emotion is displayed on Quarky we need to make sure that the Quarky's display is clear so that another emotion can be displayed on it, for that we will be using cleardisplay() function along with quarky's object.

```
quarky.cleardisplay()
```

- Now, our complete, function for displaying happy emotion on quarky would look like this:

```
while 1:
  quarky.showemotion('happy')
  time.sleep(2)
```

Let's Code

POWERED BY STEMpedia

- Now, We will then repeat steps 6 and 7 for the remaining emotions, such as "angry", "crying", and "love".

```
quarky.showemotion('love')
 time.sleep(2)
```

- Similarly, we have to define function for angry emotion:

```
quarky.showemotion('angry')
 time.sleep(2)
```

Let's Code

POWERED BY STEMpedia

- So, our complete function for displaying happy emotion would look like this :

```
sprite = Sprite('Tobi')
import time
quarky = Quarky()

while 1:
  quarky.showemotion('happy')
  time.sleep(2)
```

Let’s Code

POWERED BY STEMpedia

- Function for sad emotion:

```
quarky.showemotion(‘sad')
 time.sleep(2)
```

Final Code

POWERED BY STEMpedia

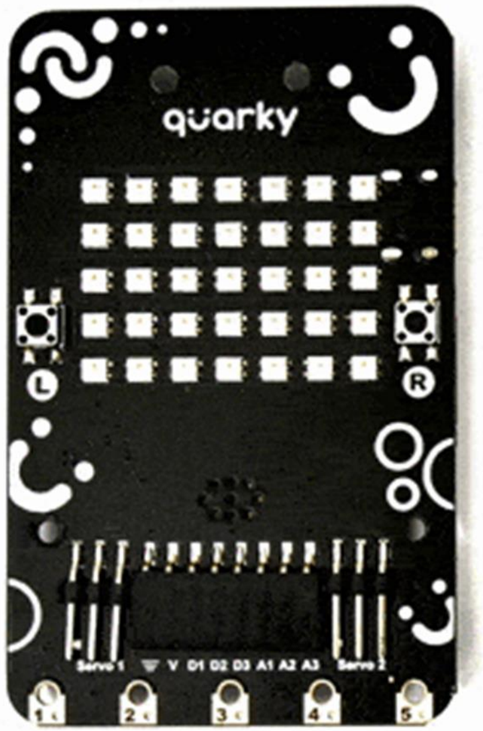
```
sprite = Sprite('Tobi')

import time

quarky = Quarky()


while 1:
  quarky.showemotion('happy')
  time.sleep(2)
  quarky.showemotion('angry')
  time.sleep(2)
  quarky.showemotion('crying')
  time.sleep(2)
  quarky.showemotion('love')
  time.sleep(2)
```

Final Output



LED Emotion

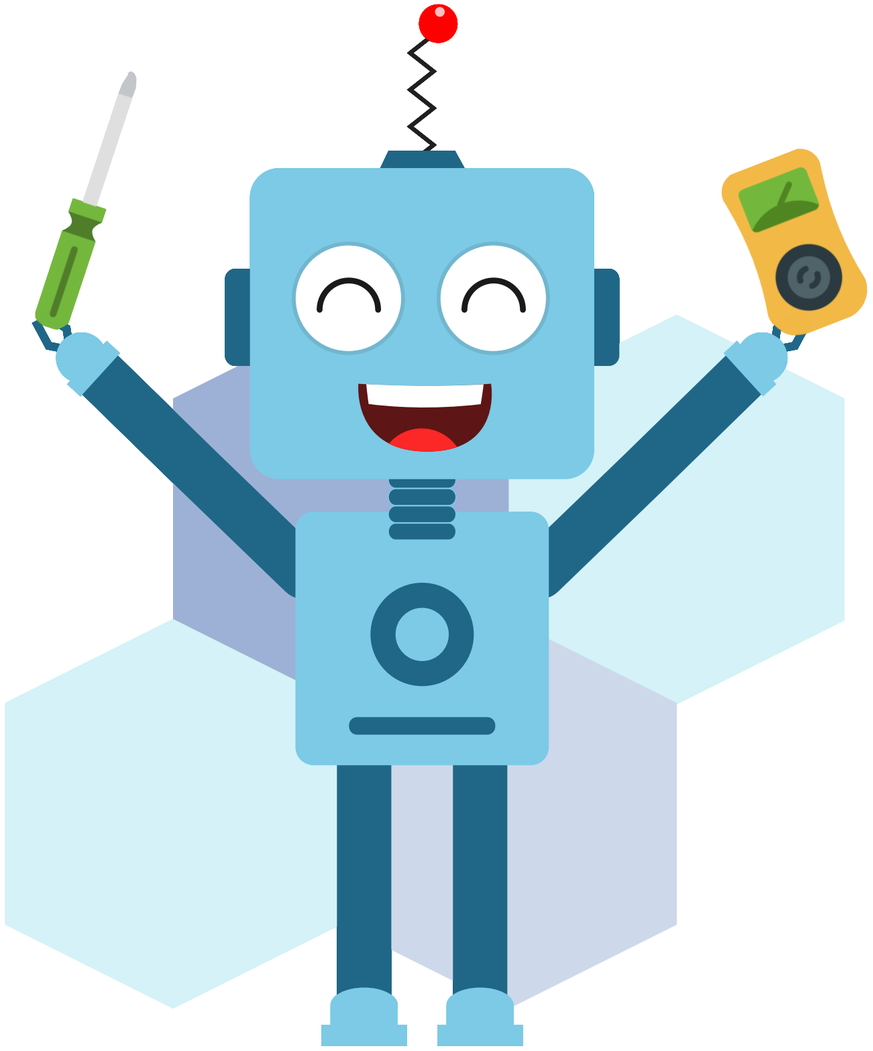
Happy

ACTIVITY

POWERED BY STEMpedia

Name badge with Quarky

Can you believe Quarky can display text too. Let’s make a script to display Quarky’s name on the matrix. We have to use
function to make this happen. This function displays the given text on Quarky as scrolling text. It takes 3 parameters as inputs: Text, Color and Speed of scrolling.



Name badge with Quarky

POWERED BY STEMpedia

- For setting the brightness of pixels on Quarky's display, we will be using setbrightness() function along with quarky's object which we have defined.

```
quarky.setbrightness(15)
```

- Now , we will be writing the while loop by providing the condition as while(1):
  - This implies that the loop should keep on executing to display the output according to the statements mentioned inside the loop.

```
while 1:
 quarky.showanimation("blink")
 quarky.showanimation("blink")
 quarky.showscrollingtext("Quarky", 2, [0, 255, 0])
 quarky.showscrollingtext("Robotics", 2, [0, 0, 255])
```

Name badge with Quarky

POWERED BY STEMpedia

- We will be using **showanimation()** function to display blinking animation on Quarky.
  - Using this function **twice** implies that the action of blinking should be performed twice on Quarky's display.
- Then, we will be using **showscrollingtext()** function to display "Quarky" .
  - The syntax of **showscrollingtext()** function is:

```
showscrollingtext([1],[2],[3])
[1]:String-TEXT="Quarky"([1]:String)
[2]:String-SPEED=2 ([2]:{1,2,3})
[3]:Num_Array-COLOR=[R,G,B] ([3]:R-(0-255),G-(0-255),B-(0-255))


quarky.showscrollingtext("Quarky", 2, [0, 255, 0])
```

Name badge with Quarky

POWERED BY STEMpedia

- Then, we will be using the same function to display “Robotics” as well.

```
quarky.showscrollingtext("Robotics", 2, [0, 0, 255])
```

- At the end, we will be using cleardisplay() function to clear Quarky’s display.

```
quarky.cleardisplay()
```

Name badge with Quarky

POWERED BY STEMpedia

```
sprite = Sprite('Tobi')
quarky = Quarky()
quarky.setbrightness(15)
while 1:
  quarky.showanimation("blink")
  quarky.showanimation("blink")
  quarky.showscrollingtext("Quarky", 2, [0, 255, 0])
  quarky.showscrollingtext("Robotics", 2, [0, 0, 255])
```

Name badge with Quarky

POWERED BY
STEMpedia

www.ai.thestempedia.com