

显示自定义图像

学习目标：

上节课程学习了使用 Python 语言编程使 micro:bit 点阵显示内置图像的方法，这节课就来学一下如何使用 micro:bit 显示自定义图像，例如使用 micro:bit 在点阵上显示一辆小船，我们可以这样编写代码。

代码与解析：

```
from microbit import *
boat = Image("49494:"
"49494:"
"99999:"
"49994")
display.show(boat)
```

另一种表示方法：

```
from microbit import *
boat = Image("49494:49494:49494:99999:49994")
display.show(boat)
```

特别注意事项：

关于程序编写语法与大小写要求：

- 1 - 大小写必须正确！
- 2 - 拼写必须严格正确！
- 3 - 关键词如 **#** 与内容之间需要有个空格
- 4 - 程序以一行空白程序结尾
- 5 - 程序块体（如 **while** 所包含的程序体是以缩进为标志的），相对于 **C** 语言，**Python** 完全省去了大括号（连同后缀的分号），并使用缩进结构来表示所属关系，只能使用 **Tab** 键（制表键）进行缩进。

Micro:bit 有 5*5 个 LED 组成的点阵，在点阵上的每个 LED 亮度都可以设置为 0 至 9 中的一个值。如果一个 LED 的亮度被设置为 0，那么它就熄灭了。如果它的亮度被设置为 9，那么它就处于最亮的水平。利用这个特点我们就可以在 micro:bit 点阵上显示自定义的图像，我们这节课的代码实现的功能时在 micro:bit 点阵上显示一艘帆船，背景的亮度值是 5，船体部分的亮度要亮一些，亮度值是 9，同学们可以设置其他的亮度值，来显示不同的图案。

编程与下载:

1.打开 Mu 软件，如图 3-1 在编辑窗口输入代码，注意！所有的英文与符号都应该在英文状态下输入，使用 Tab 键（制表键）进行缩进，最后一行以空白程序结尾。

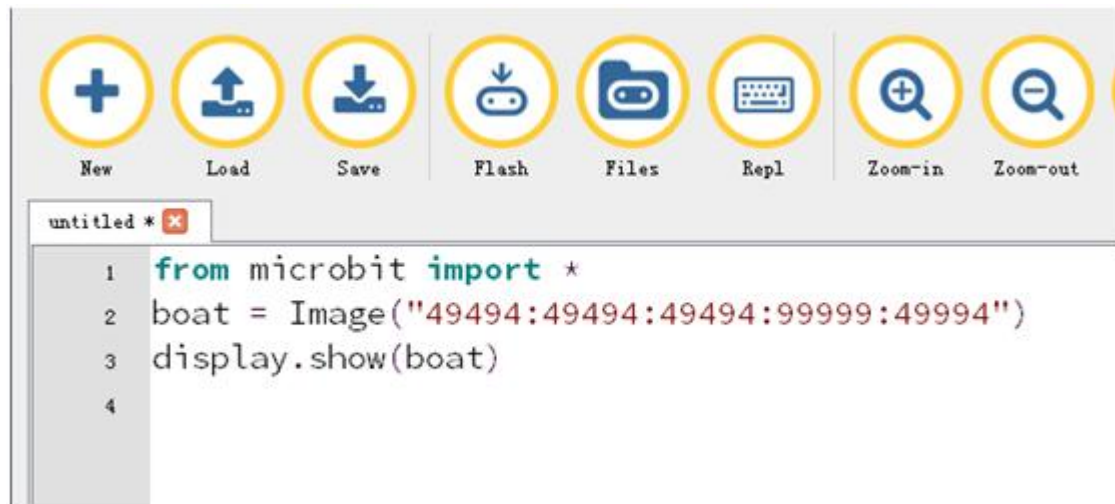


图 3-1

2.如图 3-2 所示，点击大拇指图标的 Check 键检查一下我们的代码是否有错误。如果某一行出现光标或下划线，表示某一行的程序出错，如果没有出现光标或下划线，则表示代码没有错误。

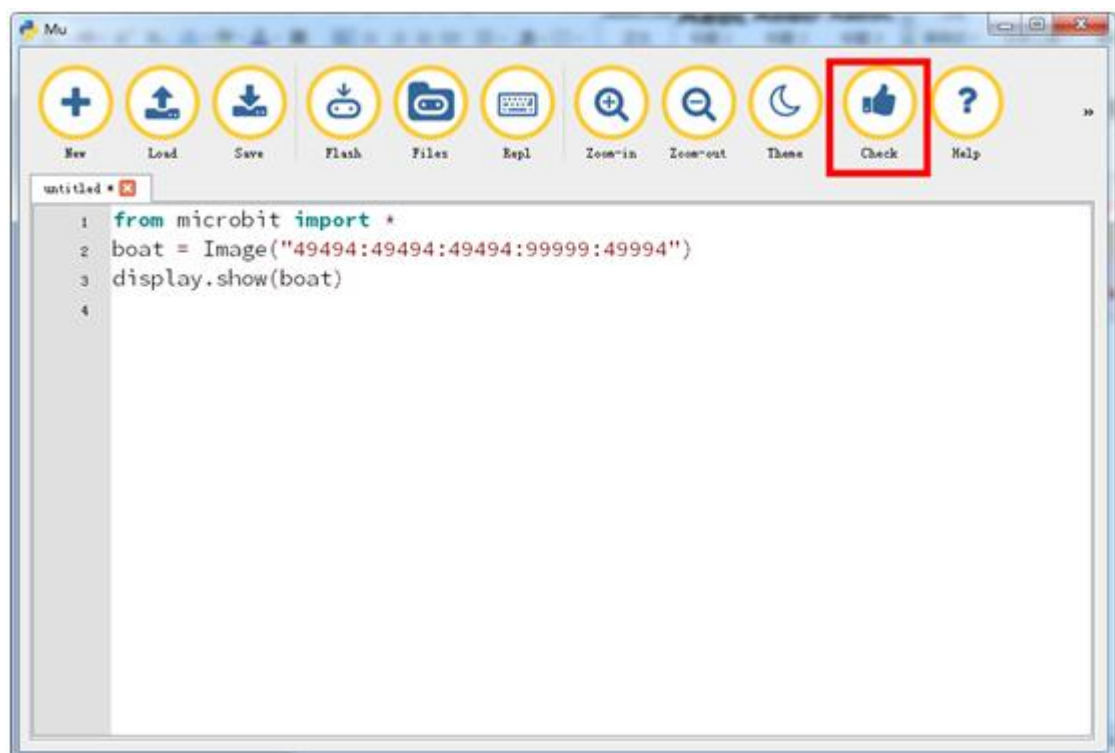


图 3-2

3.将 micro USB 数据线连接 micro:bit 与电脑，接着如图 3-3 点击 Flash 按键下载程序至 micro:bit。

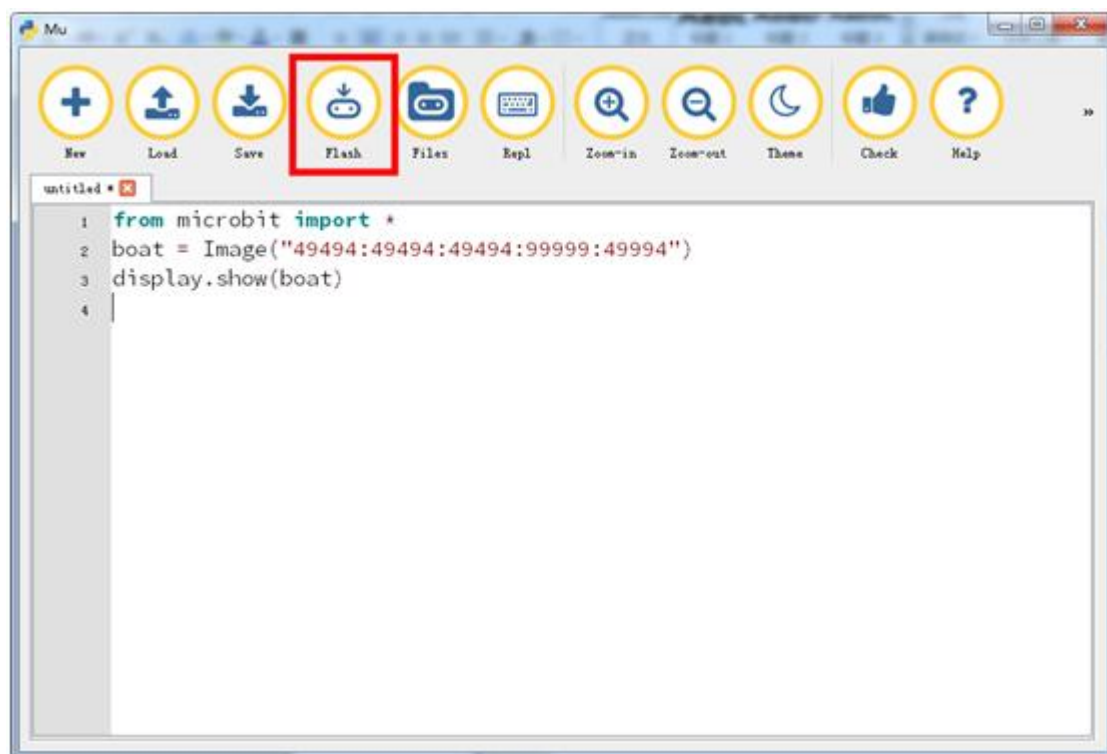


图 3-3

4. 下载成功后，可以观察到 micro:bit 点阵上出现帆船，其中背景的亮度要比船体的亮度弱，如图 3-4 所示。

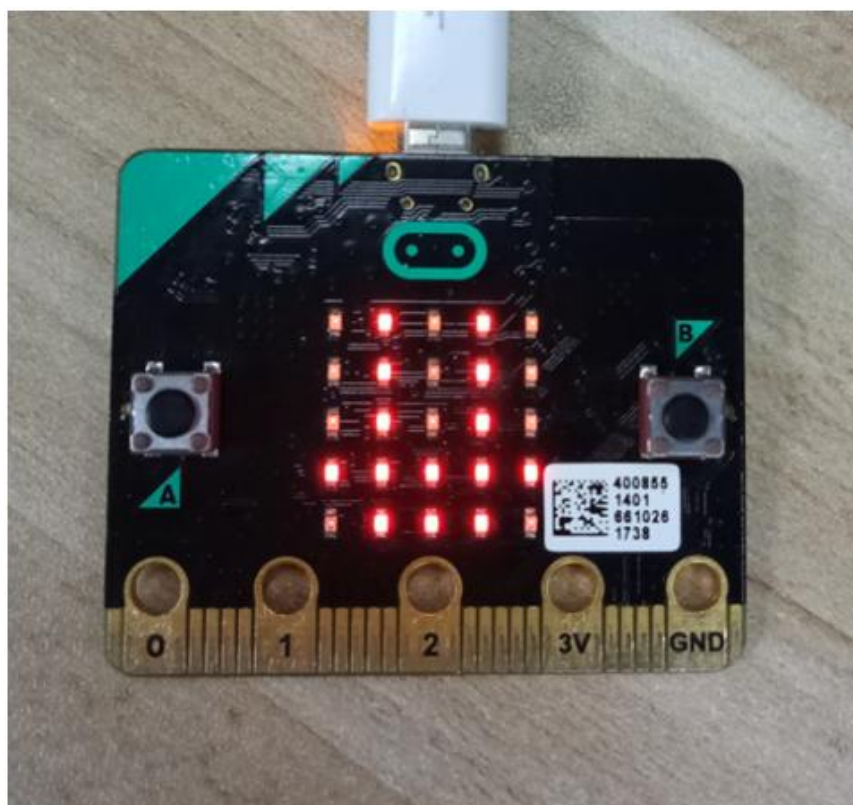


图 3-4