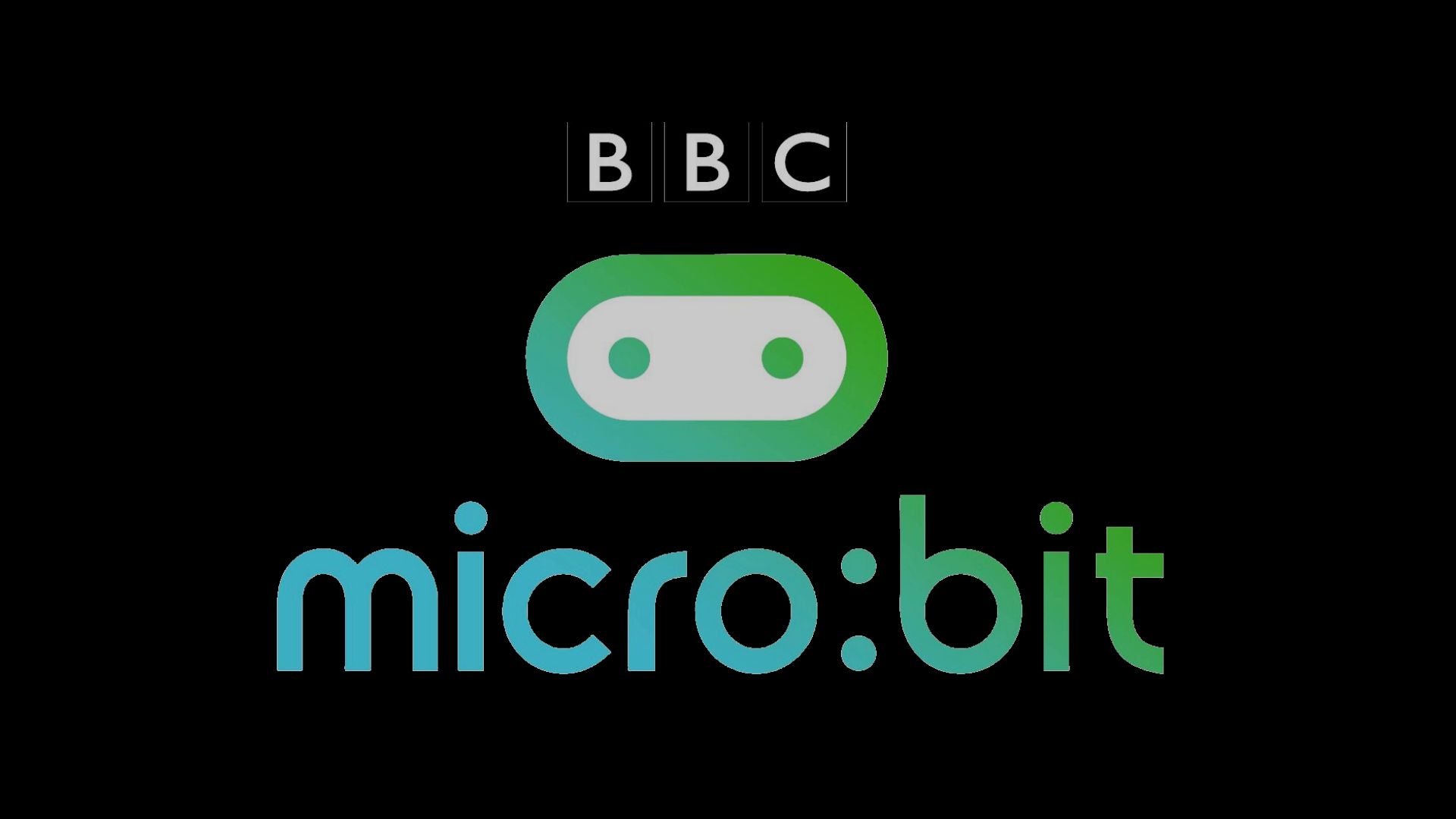
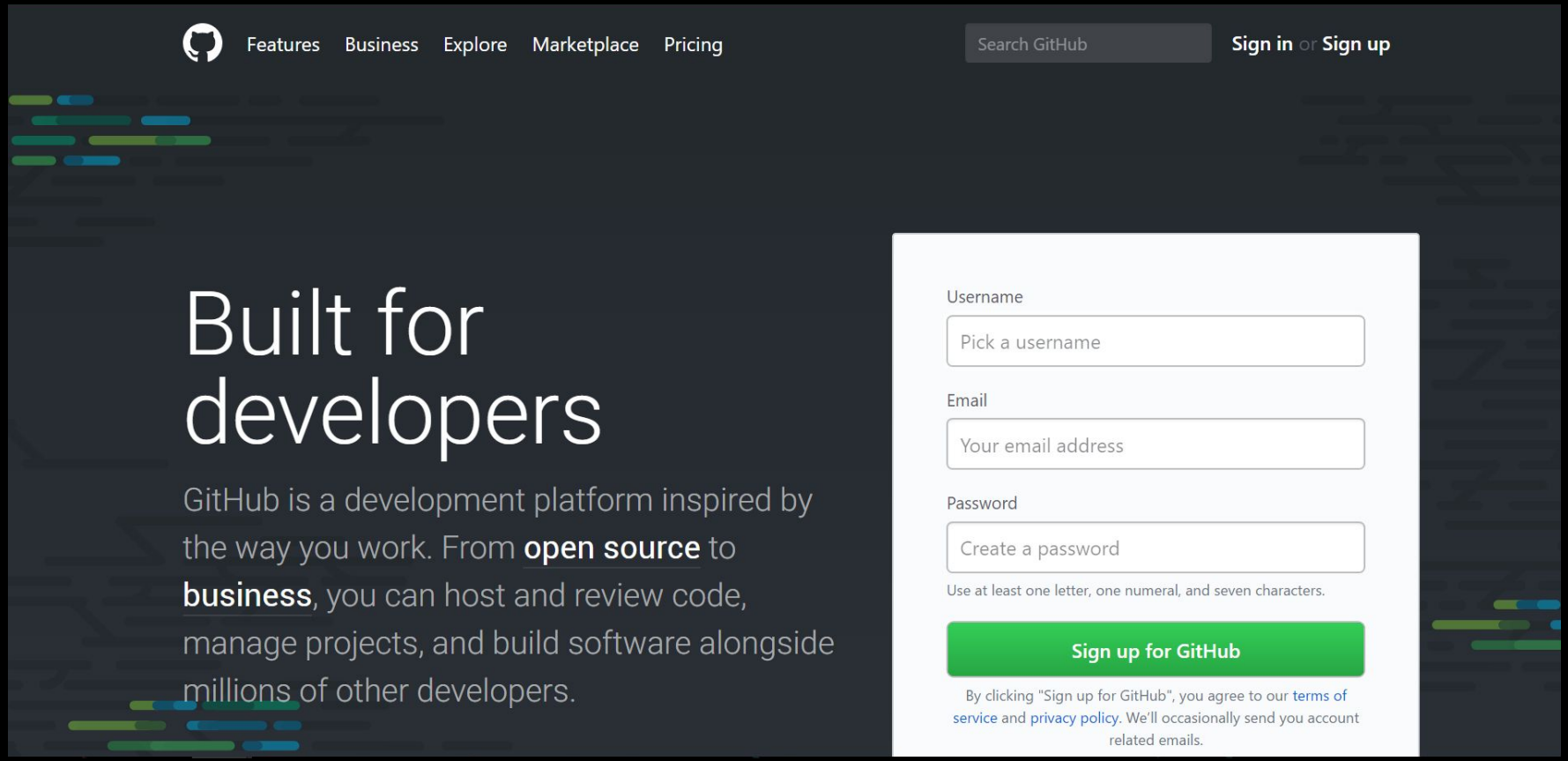
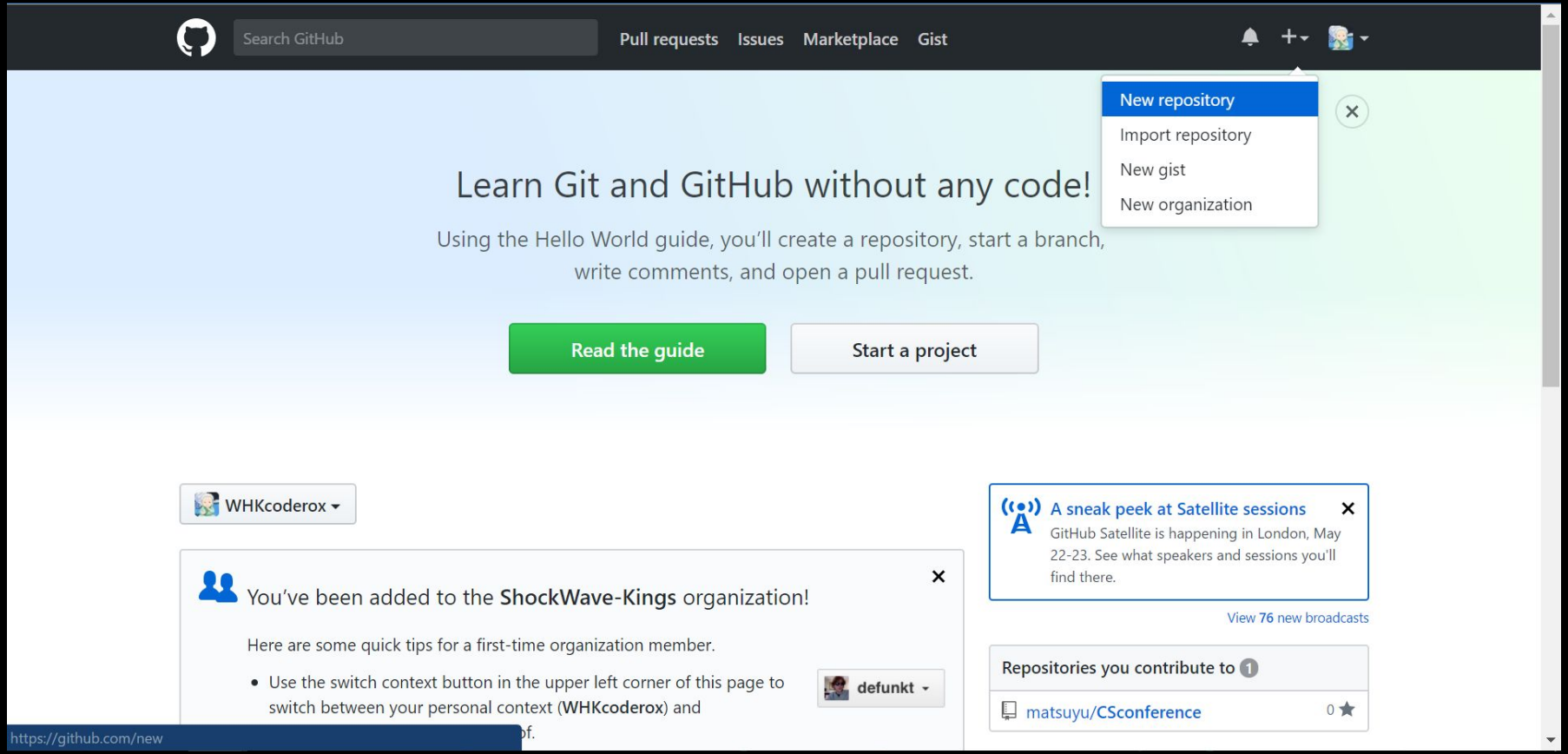

BBC
micro:bit

But Before We Get Into That...

Github


Features
Business
Explore
Marketplace
Pricing
Search GitHub
Sign in or Sign up
Built for developers
GitHub is a development platform inspired by the way you work. From open source to business, you can host and review code, manage projects, and build software alongside millions of other developers.
Username
Pick a username
Email
Your email address
Password
Create a password
Use at least one letter, one numeral, and seven characters.
Sign up for GitHub
By clicking "Sign up for GitHub", you agree to our terms of service and privacy policy. We'll occasionally send you account related emails.


Search GitHub
Pull requests
Issues
Marketplace
Gist
New repository
Import repository
New gist
New organization
Learn Git and GitHub without any code!
Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.
Read the guide
Start a project
WHKcoderox
You've been added to the ShockWave-Kings organization!
Here are some quick tips for a first-time organization member.
Use the switch context button in the upper left corner of this page to switch between your personal context (WHKcoderox) and
defunkt
A sneak peek at Satellite sessions
GitHub Satellite is happening in London, May 22-23. See what speakers and sessions you'll find there.
View 76 new broadcasts
Repositories you contribute to 1
matsuyu/CSconference
0
https://github.com/new

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner **Repository name**

WHKcoderox / Test repo

Great repository names are Your new repository will be created as **Test-repo** bout **expert-fiesta**.

Description (optional)

Just a social experiment

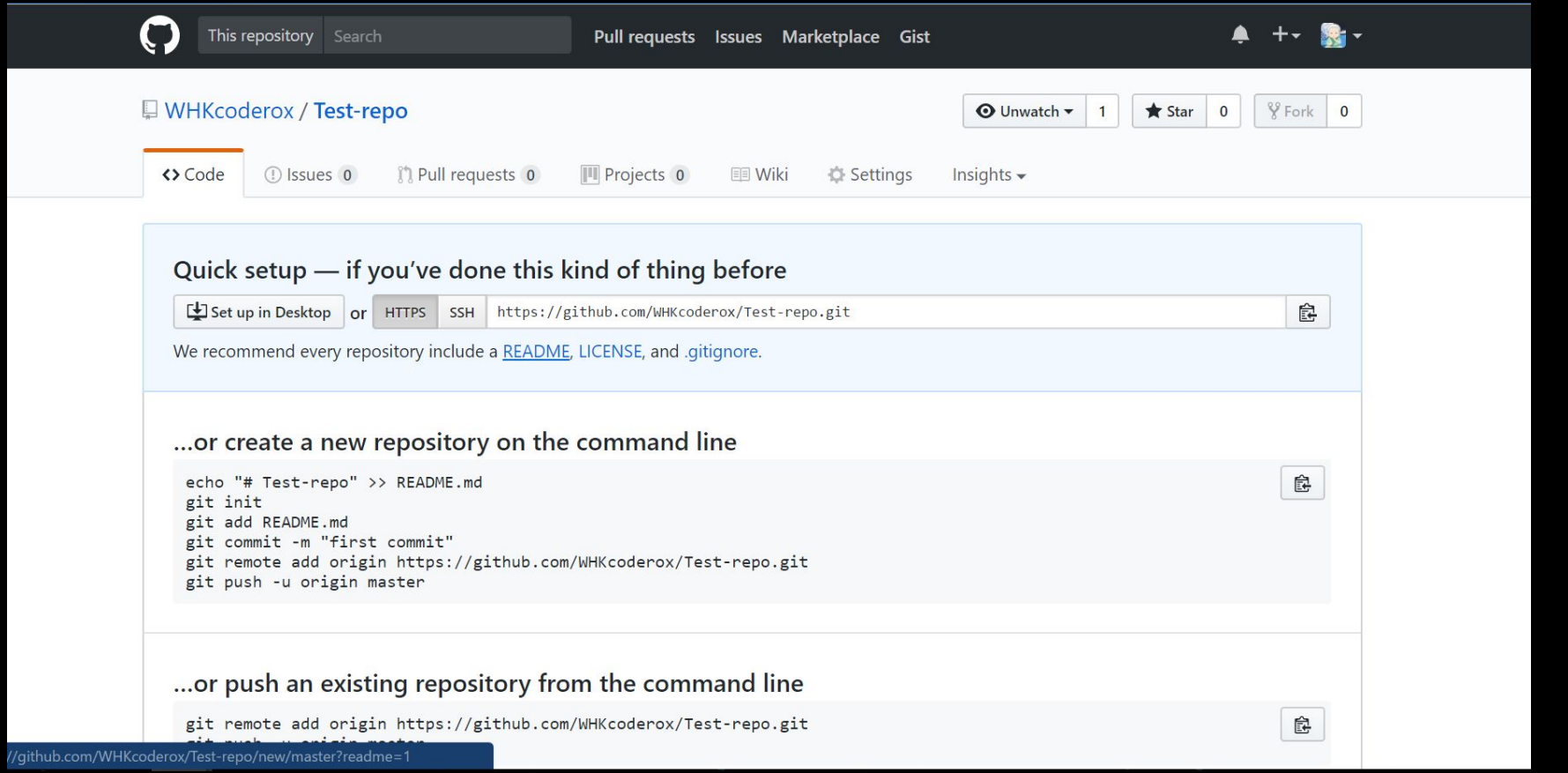
Public
Anyone can see this repository. You choose who can commit.

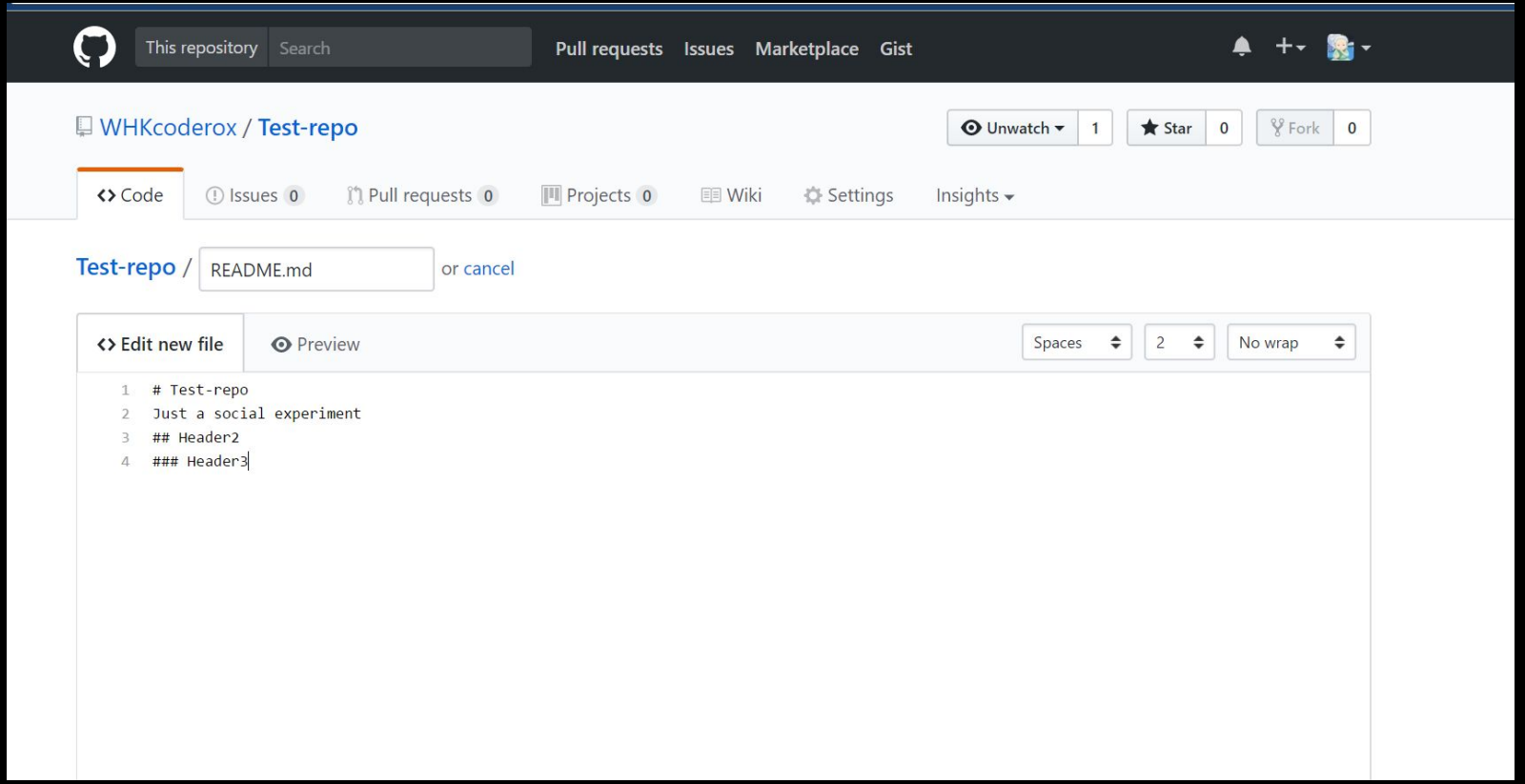
Private
You choose who can see and commit to this repository.

Initialize this repository with a README
This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: None | Add a license: None

Create repository


This repository
Search
Pull requests
Issues
Marketplace
Gist
WHKcoderox / Test-repo
Unwatch 1
Star 0
Fork 0
Code
Issues 0
Pull requests 0
Projects 0
Wiki
Settings
Insights
Quick setup — if you've done this kind of thing before
Set up in Desktop
or
HTTPS
SSH
https://github.com/WHKcoderox/Test-repo.git
We recommend every repository include a README, LICENSE, and .gitignore.
...or create a new repository on the command line
echo "# Test-repo" >> README.md
git init
git add README.md
git commit -m "first commit"
git remote add origin https://github.com/WHKcoderox/Test-repo.git
git push -u origin master
...or push an existing repository from the command line
git remote add origin https://github.com/WHKcoderox/Test-repo.git
//github.com/WHKcoderox/Test-repo/new/master?readme=1


This repository
Search
Pull requests
Issues
Marketplace
Gist
WHKcoderox / Test-repo
Unwatch 1
Star 0
Fork 0
Code
Issues 0
Pull requests 0
Projects 0
Wiki
Settings
Insights
Test-repo / README.md
or cancel
Edit new file
Preview
Spaces
2
No wrap
1 # Test-repo
2 Just a social experiment
3 ## Header2
4 ### Header3

Code | Issues 0 | Pull requests 0 | Projects 0 | Wiki | Settings | Insights

Test-repo / README.md or cancel

Edit new file | Preview | Spaces | 2 | No wrap

Test-repo

Just a social experiment

Header2

Header3

Commit new file

Create README.md

Commit new file

My first readme in my first repository

Gonna be that one IT dude in my teams from now on...

Commit new file | Cancel


BBC
micro:bit
We're back!

About the micro:bit

A microcontroller, a tiny computer to be incorporated in projects.

A *teensy* bit of storage

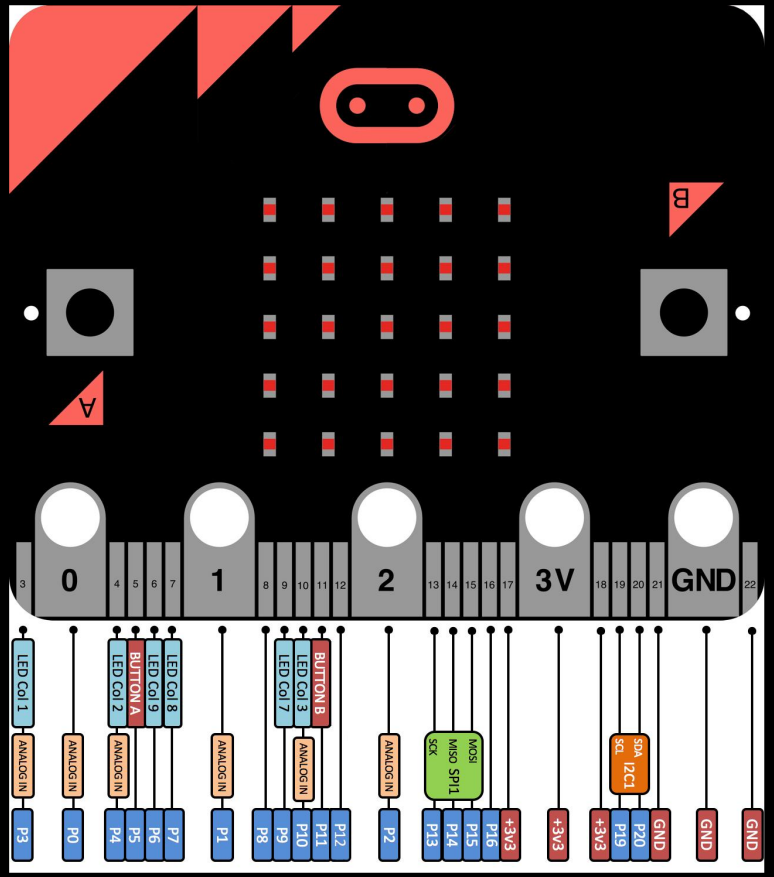
5x5 display of LED's (as if 144p was bad enough)

Three buttons - a, b and the reset button

Accelerometer - Gives a reading of tilt, or vertical movement of the micro:bit

Compass - reads your orientation

Pins - circuitry connections



Documentation

https://microbit-micropython.readthedocs.io/en/latest/

Github repo

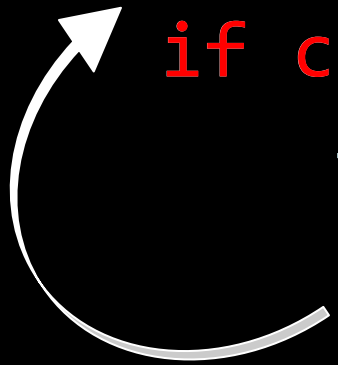
tinyurl.com/YCW-microbit-templates

Code explained!

```
from microbit import *
```

Code explained!

```
while True:
    if condition:
        # do something...
```



Animation

What if you have a list of images?

```
boat1 = Image("05050:05050:05050:99999:09990")

boat2 = Image("00000:05050:05050:05050:99999")

boat3 = Image("00000:00000:05050:05050:05050")

boat4 = Image("00000:00000:00000:05050:05050")

boat5 = Image("00000:00000:00000:00000:05050")

boat6 = Image("00000:00000:00000:00000:00000")

all_boats = [boat1, boat2, boat3, boat4, boat5, boat6]
display.show(all_boats, delay=200)
```

Button

```
from microbit import *

while True:
    if button_a.is_pressed():
        display.show(Image.HAPPY)
    elif button_b.is_pressed():
        break
    else:
        display.show(Image.SAD)
```

Accelerometer

```
from microbit import *


while True:
    reading = accelerometer.get_x()

    if reading > 20:
        display.show("L")
    elif reading < -20:
        display.show("R")
    else:
        display.show("-")
```

Compass

```
from microbit import *

# Start calibrating
compass.calibrate()

# Try to keep the needle pointed in (roughly) the correct direction
while True:
    sleep(100)
    needle = ((15 - compass.heading()) // 30) % 12
    display.show(Image.ALL_CLOCKS[needle])
```

So, how is micro:bit useful?

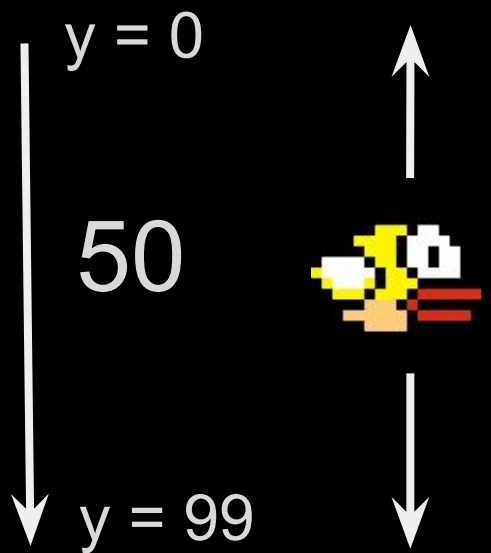
With all these basic functions of micro:bit, we can get...

And also a lot more of your other daily appliances and applications!

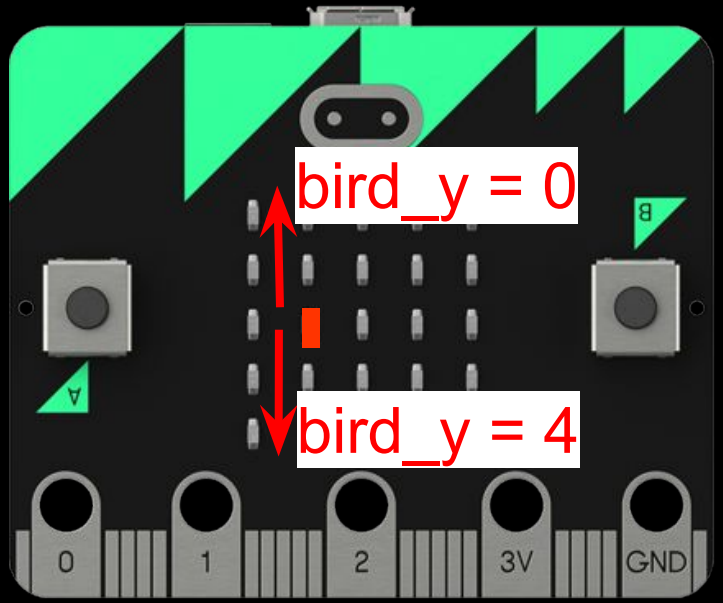
Flappy Bird: 1. Welcome message

```
# Welcome message
display.scroll("Get ready...")
```

Flappy Bird: 2. Draw the bird

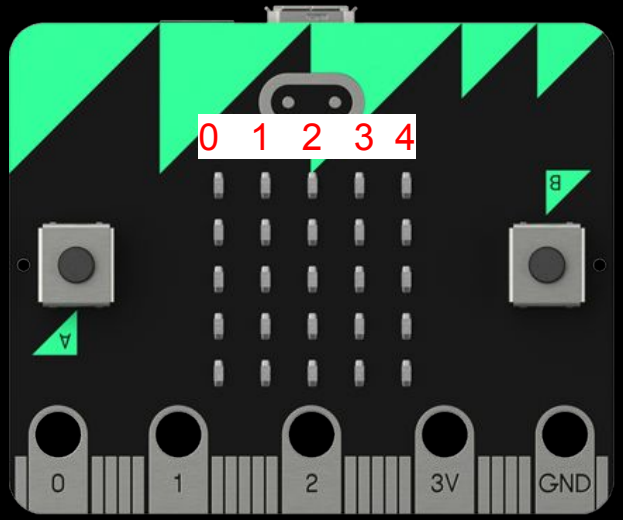


Scale down



Flappy Bird: 2. Draw the bird

```
# Global variables
y = 50

# Game loop
while True:
    # draw bird
    bird_y = int(y / 20)
    display.set_pixel(1, bird_y, 9)

    # wait 20ms
    sleep(DELAY)
```



x-value

Scaled down y-value

brightness

Flappy Bird 3. Simulate gravity

```
# Global variables
speed = 0
```

```
# accelerate down to terminal velocity
speed += 1
if speed > 2:
    speed = 2

# move bird, but not off the edge
y += speed
if y > 99:
    y = 99
if y < 0:
    y = 0
```

Flappy Bird 4. Click button to flap

```
# flap if button a was pressed
if button_a.was_pressed():
    speed = -8
```

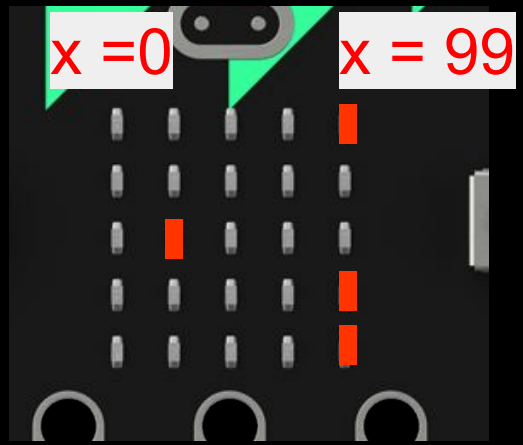
Flappy Bird: 5. Draw Pipes

```
def make_pipe():
    pipe = Image("00003:00003:00003:00003:00003")
    gap = random.randint(0,3)        # random position on the wall
    pipe.set_pixel(4, gap, 0)        # create hole in pipe
    pipe.set_pixel(4, gap+1, 0)      # hole is two dots tall
    return pipe

# create first pipe
pipe = make_pipe()

while True:
    # show pipe
    display.show(pipe)
```

Flappy Bird: 6. Move Pipes

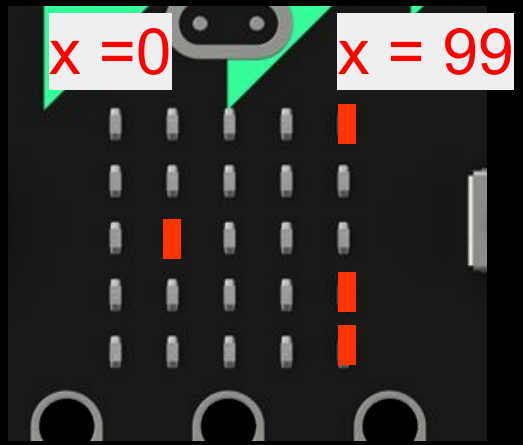
```
# Game constants - T
DELAY = 20                          # ms between each frame
FRAMES_PER_WALL_SHIFT = 20          # num of frames between each wall shift
FRAMES_PER_NEW_WALL = 100           # num of frames between each new wall
FRAMES_PER_SCORE = 50               # num of frames between score rising by 1
```

```
# Global variables
frame = 0
score = 0
```



Flappy Bird: 6. Move Pipes

```
 # move wall left
if(frame % FRAMES_PER_WALL_SHIFT == 0):
    pipe = pipe.shift_left(1)

# create new wall
if(frame % FRAMES_PER_NEW_WALL == 0):
    pipe = make_pipe()

# increase score
if(frame % FRAMES_PER_SCORE == 0):
    score += 1

# increment frame
frame += 1
```



Flappy Bird: 7. Collision

```
# check for collision
if pipe.get_pixel(1, bird_y) != 0:
    display.show(Image.SAD)
    sleep(500)
    display.scroll("Score: " + str(score))
    break
```

Radio

A bonus component of your microbit: Brings signal capabilities to the microbit!

(Can be found in the official documentation)

```
radio.on()   #of course, there's a radio.off() to save power & memory
radio.send("Hello Next World")
incoming_message = radio.receive()
radio.config(length=251,queue=3,channel=5,power=6,,,,)
#some extra settings you can go check out yourself ^
```

Mission: Pass the message!

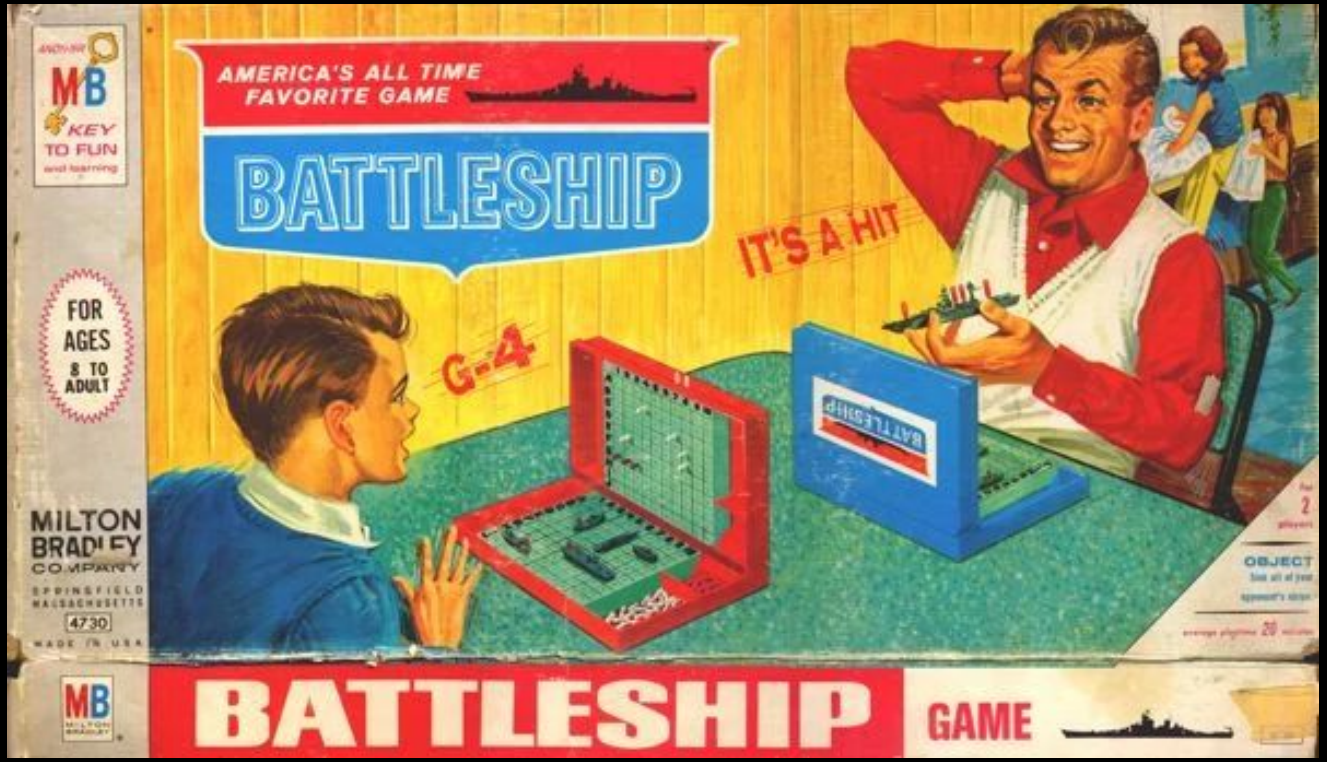
Objective: in groups of 5, plan how to get a secret message passed down (5 minutes). After everyone has planned, talking will be banned.

Remember, as much as you can transmit messages, you can also receive random messages from other teams. Try to play around that!

Each member will have a piece of paper with x columns, to guess other groups messages including theirs.

Once the game commences, one group member will get a secret text. This text must be passed down the group.

Battleships!



Touch Arpeggiator (Banana)

http://www.itpro.co.uk/desktop-hardware/26289/13-top-bbc-micro-bit-projects

Resources

Github link: https://tinyurl.com/buildingbloCSGH

Micro:bit official website: microbit.org

Documentation: https://microbit-micropython.readthedocs.io/en/latest/