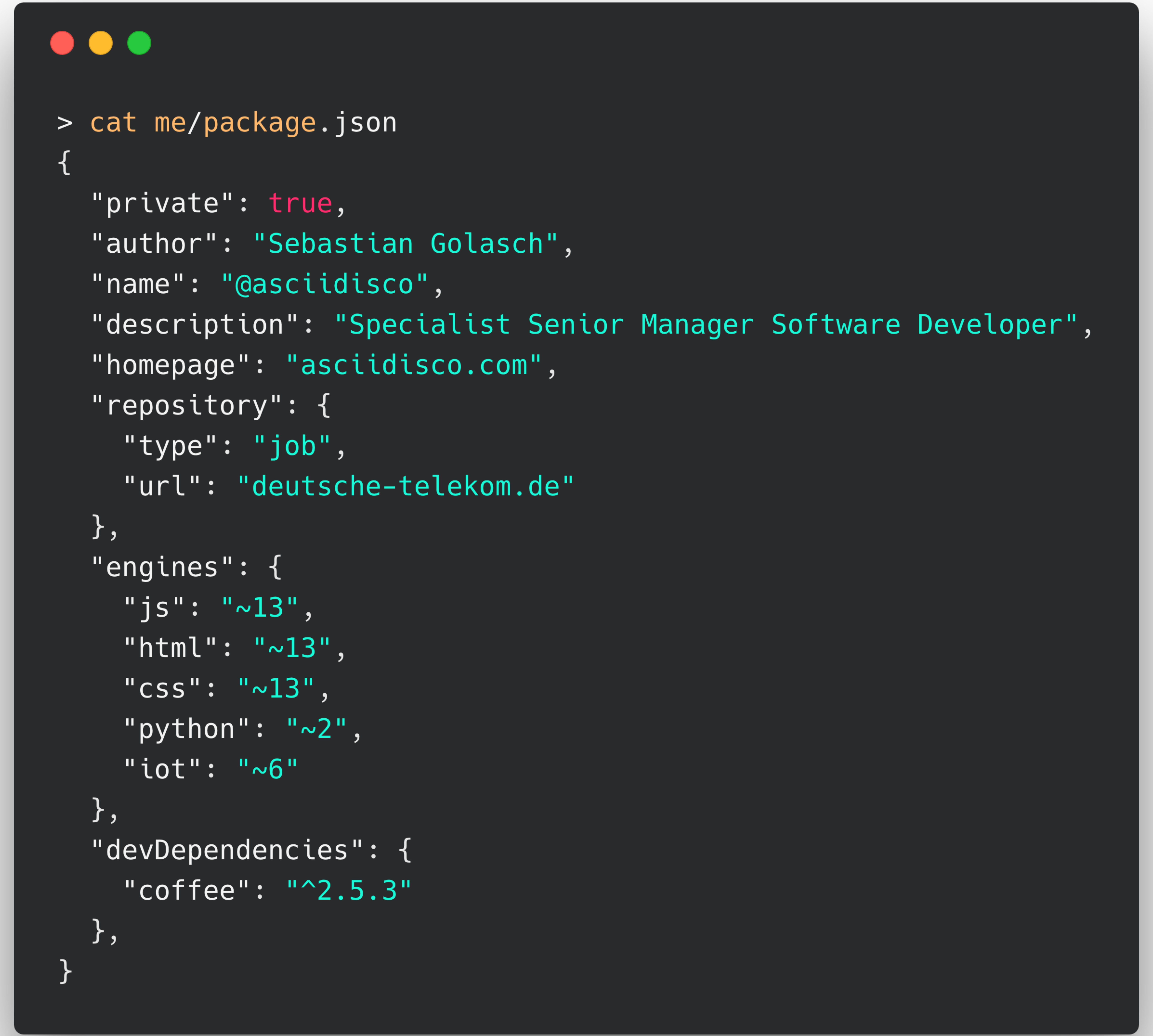
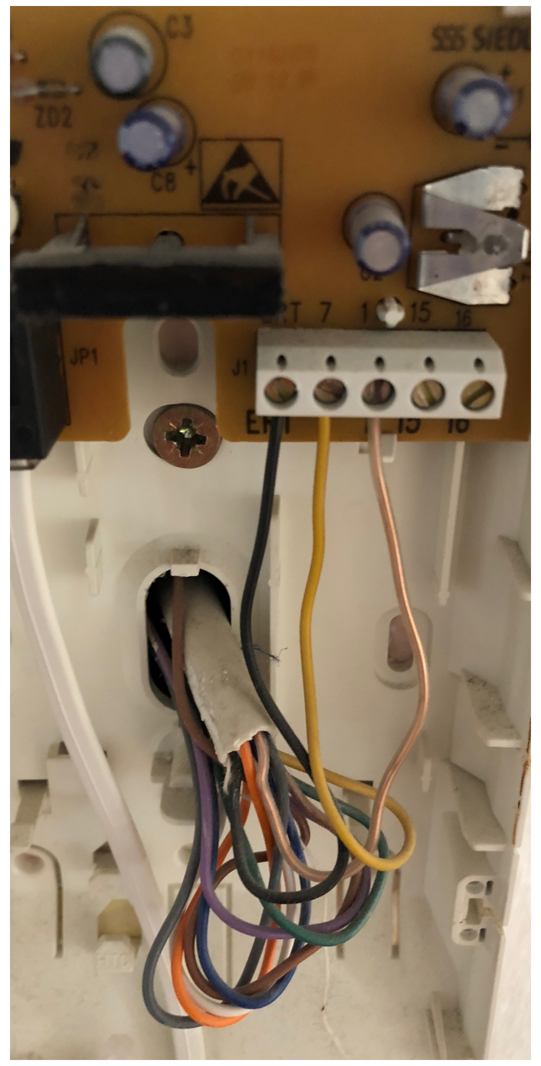
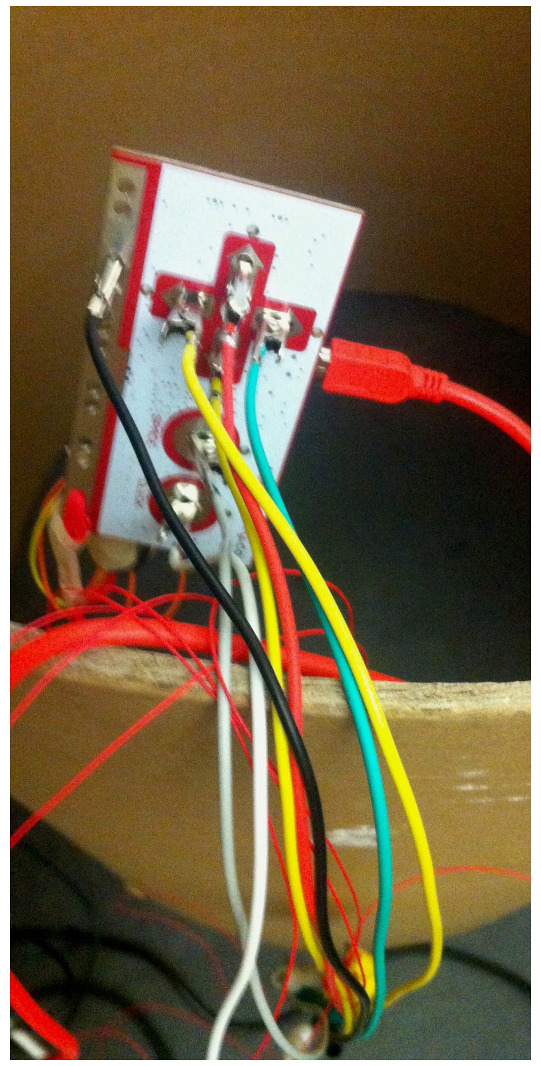
The Universal Serial Web

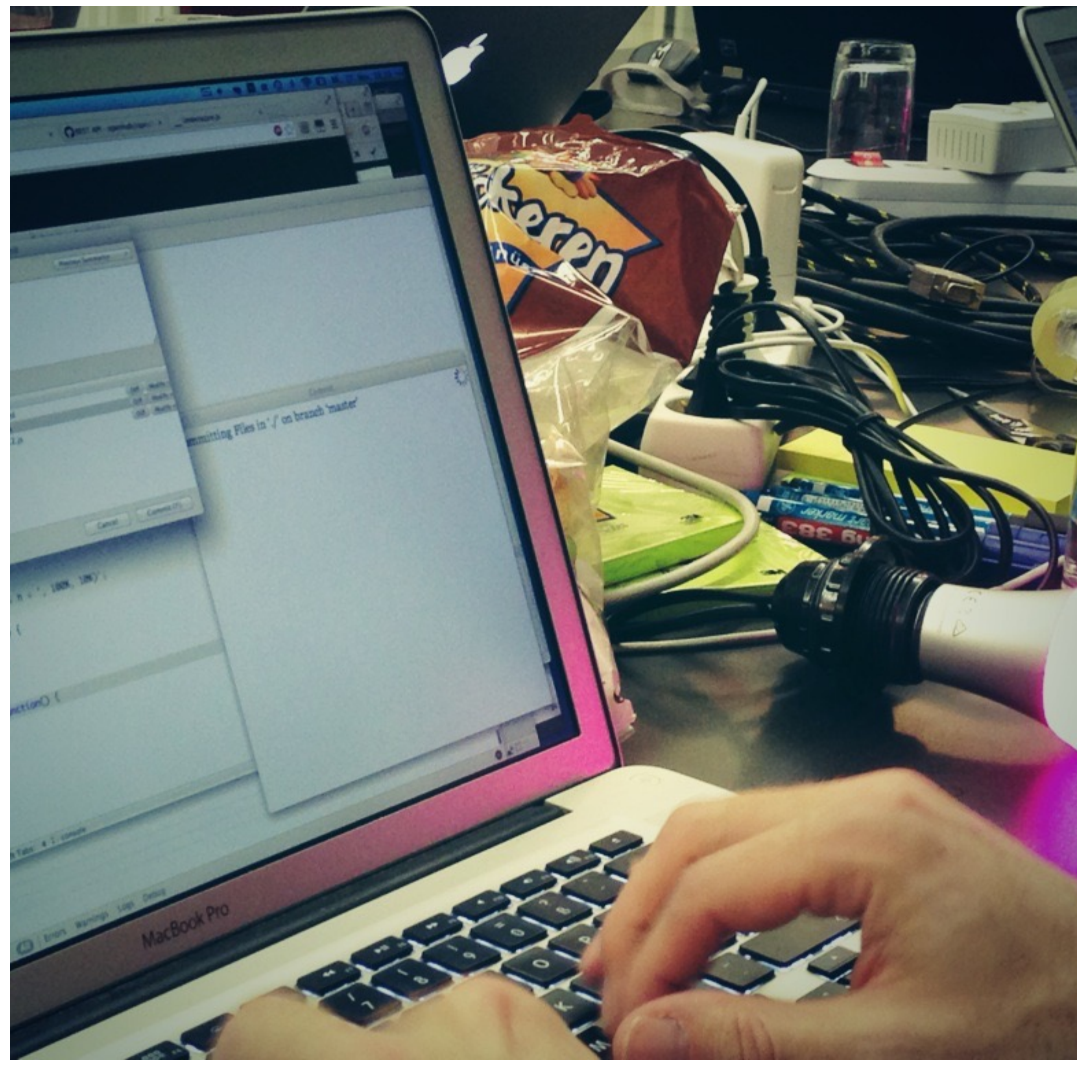
<ME/>

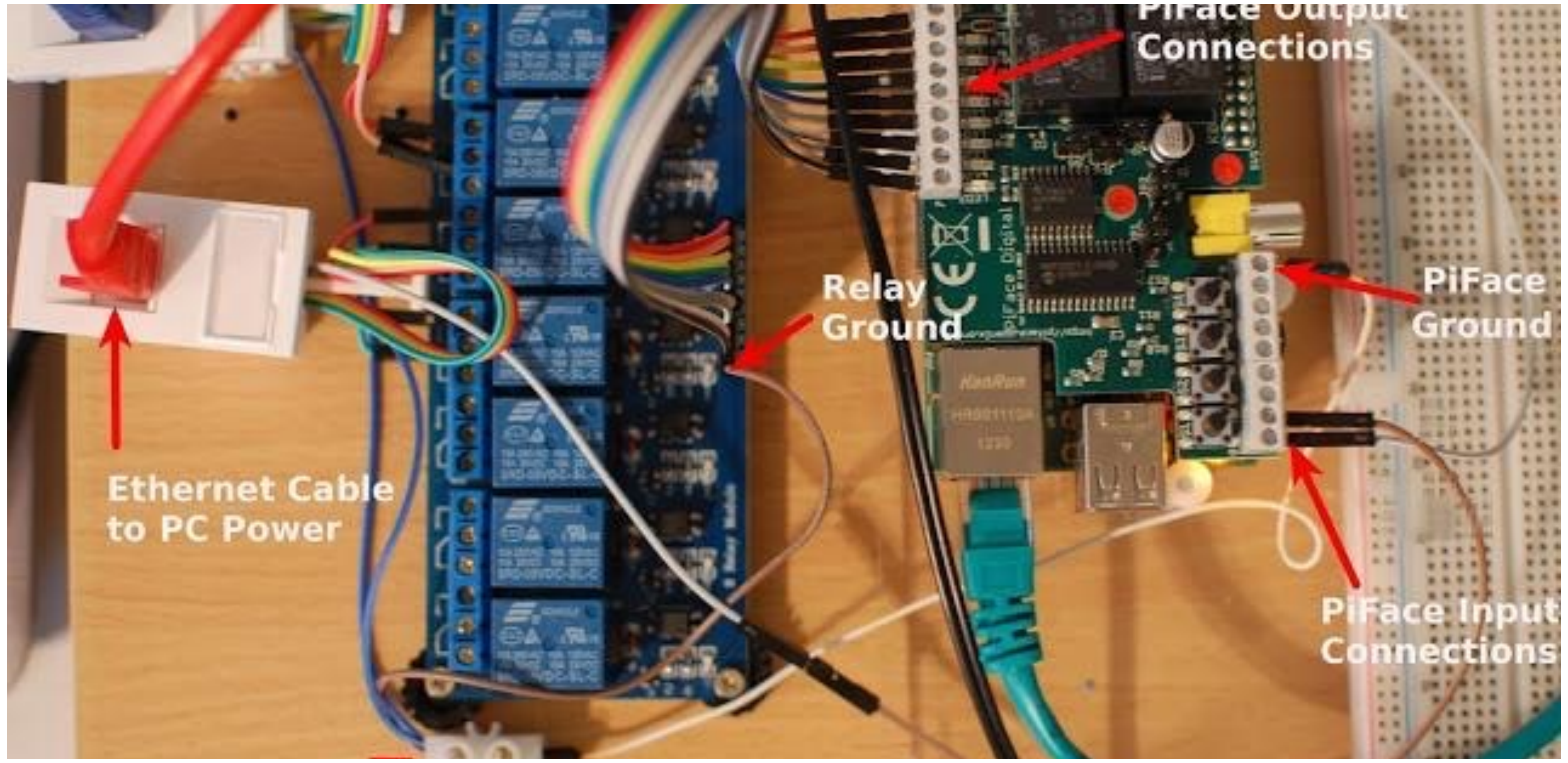


```
> cat me/package.json
{
  "private": true,
  "author": "Sebastian Golasch",
  "name": "@asciidisco",
  "description": "Specialist Senior Manager Software Developer",
  "homepage": "asciidisco.com",
  "repository": {
    "type": "job",
    "url": "deutsche-telekom.de"
  },
  "engines": {
    "js": "~13",
    "html": "~13",
    "css": "~13",
    "python": "~2",
    "iot": "~6"
  },
  "devDependencies": {
    "coffee": "^2.5.3"
  },
}
```

I´ve Been Tinkering With Hardware Since 2011…





Universal Hardware Interface: Usb

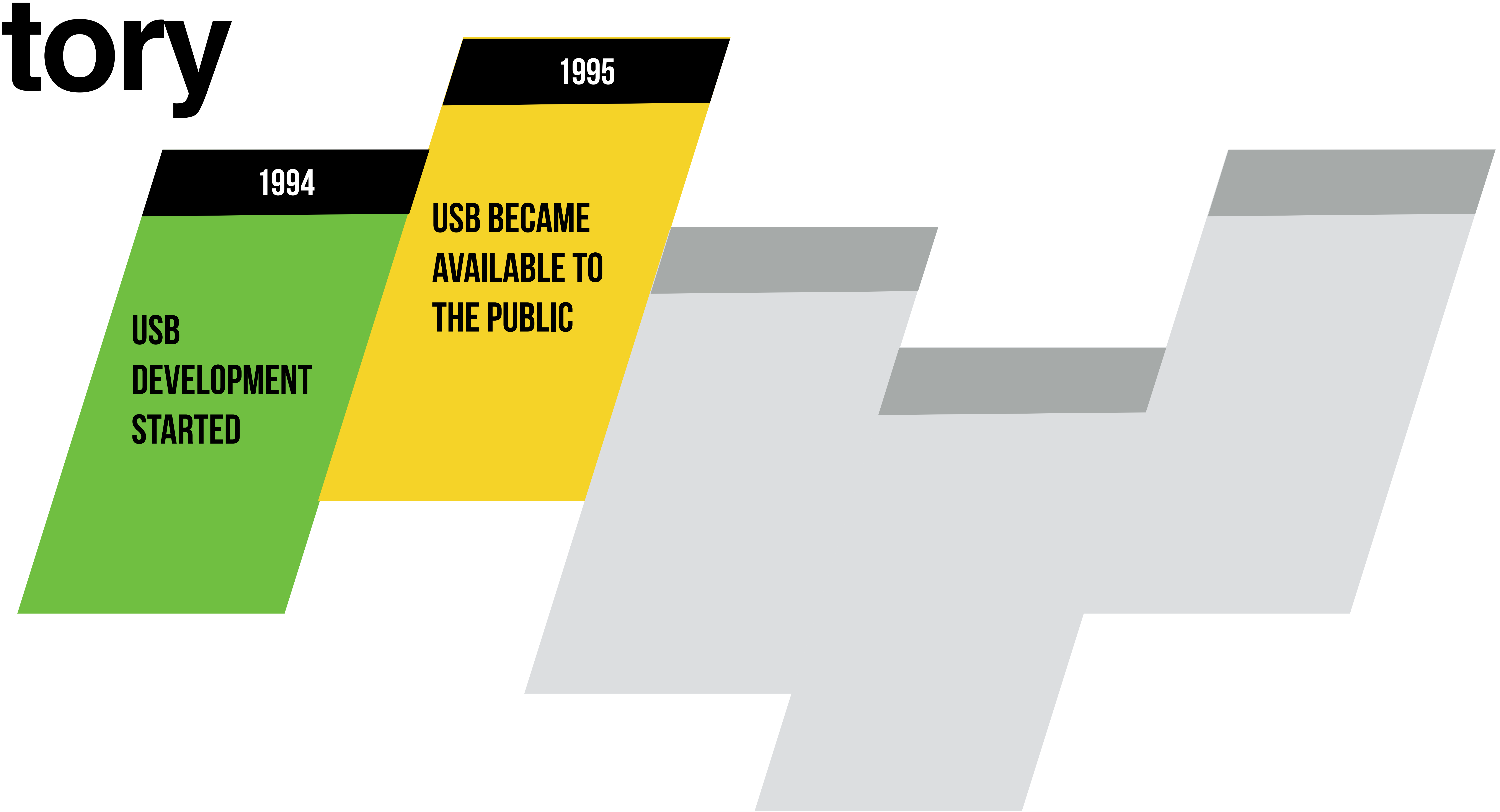
Usb Basics 💻

History

1994

USB DEVELOPMENT STARTED

History



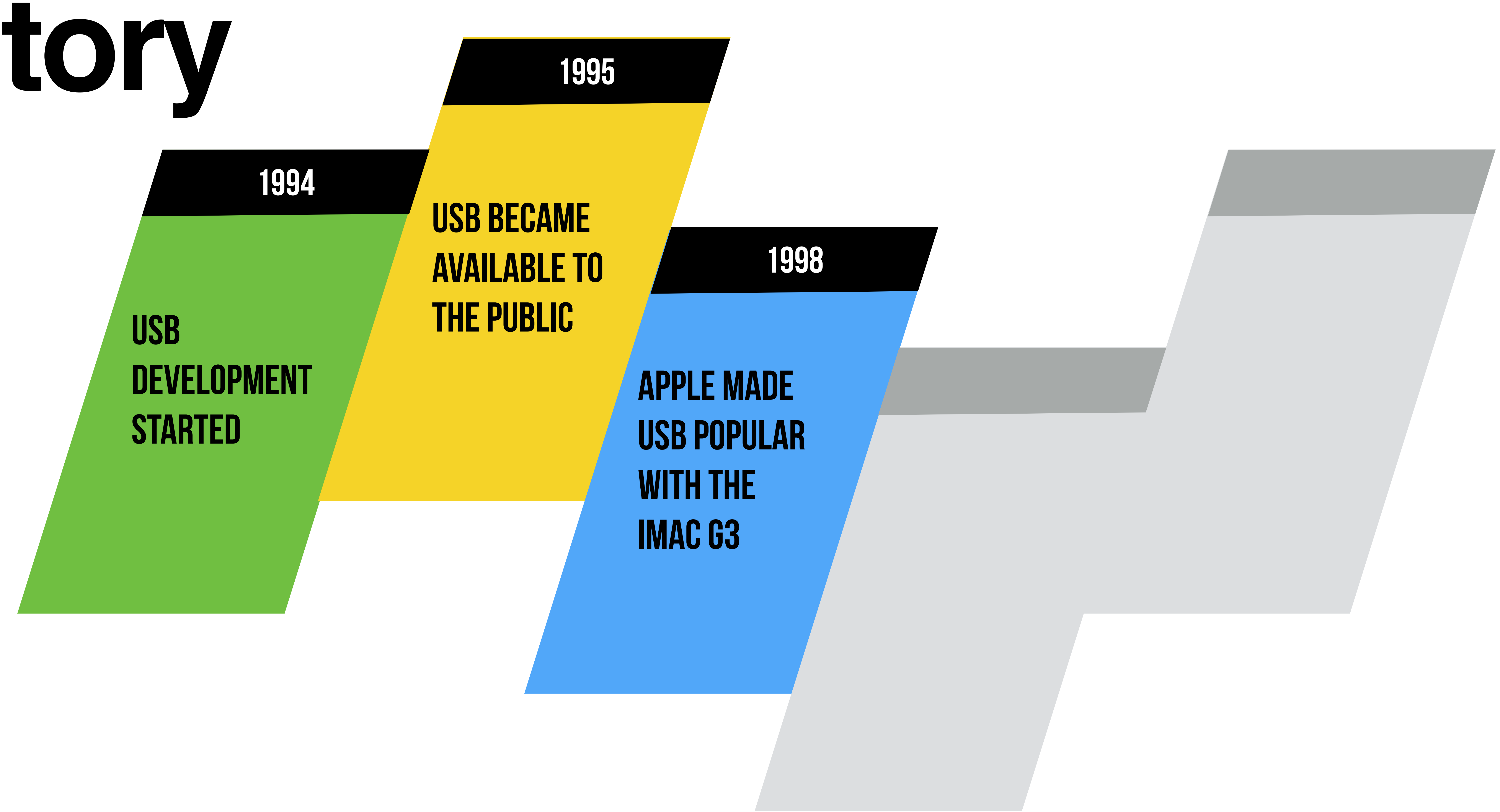
1994

USB DEVELOPMENT STARTED

1995

USB BECAME AVAILABLE TO THE PUBLIC

History



1994

USB DEVELOPMENT STARTED

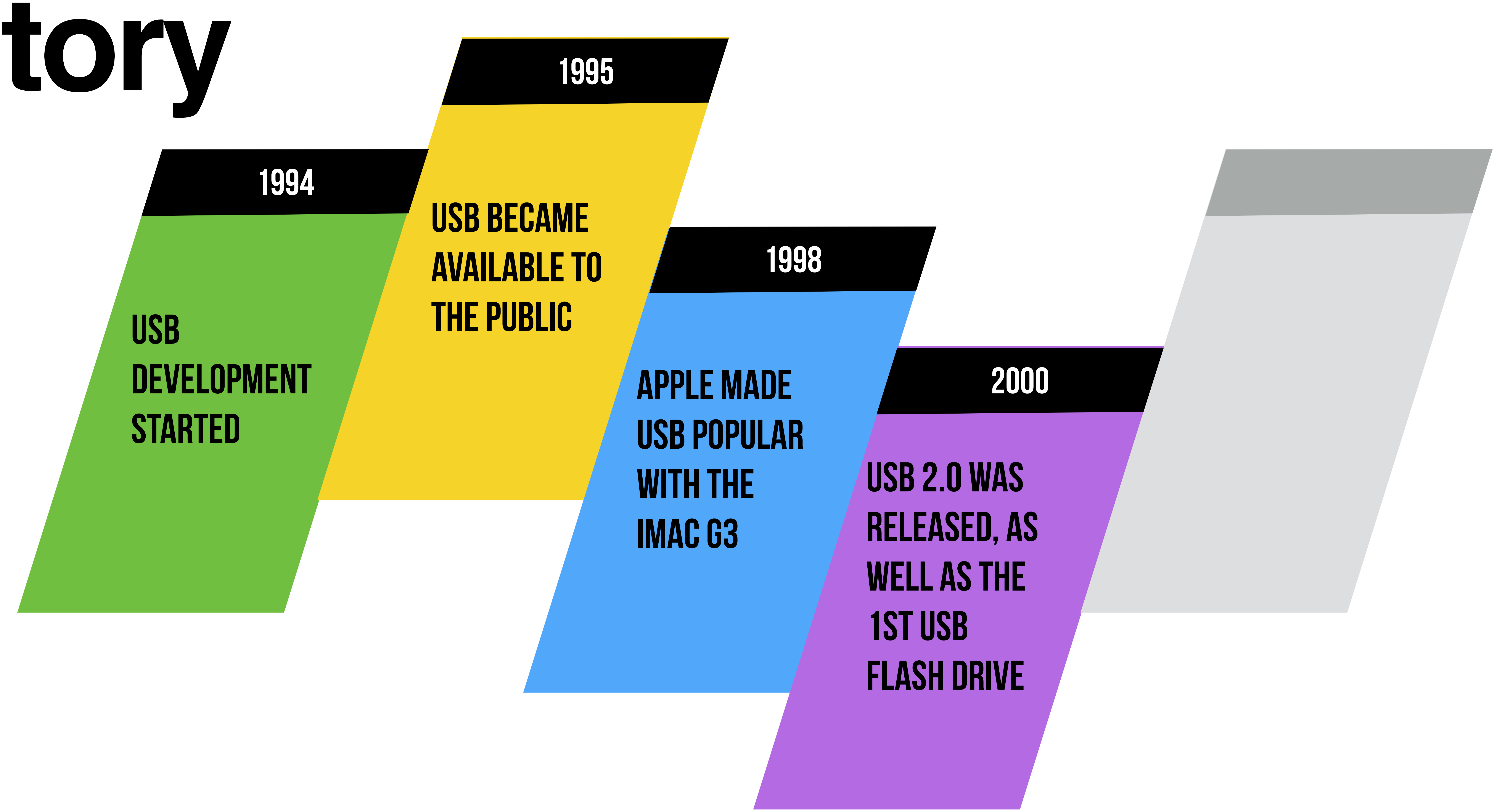
1995

USB BECAME AVAILABLE TO THE PUBLIC

1998

APPLE MADE USB POPULAR WITH THE IMAC G3

History



1994

USB DEVELOPMENT STARTED

1995

USB BECAME AVAILABLE TO THE PUBLIC

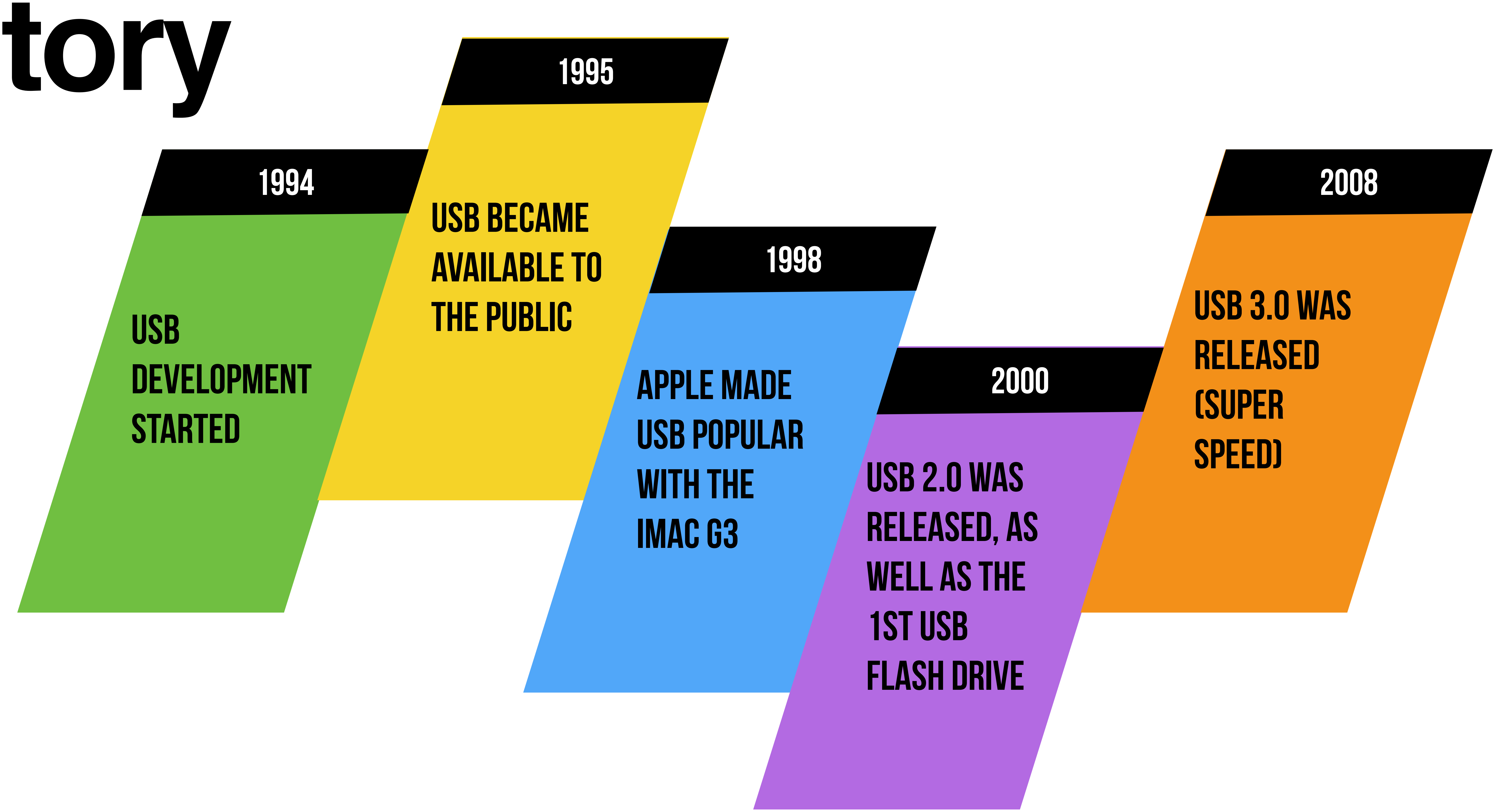
1998

APPLE MADE USB POPULAR WITH THE IMAC G3

2000

USB 2.0 WAS RELEASED, AS WELL AS THE 1ST USB FLASH DRIVE

History



1994

USB DEVELOPMENT STARTED

1995

USB BECAME AVAILABLE TO THE PUBLIC

1998

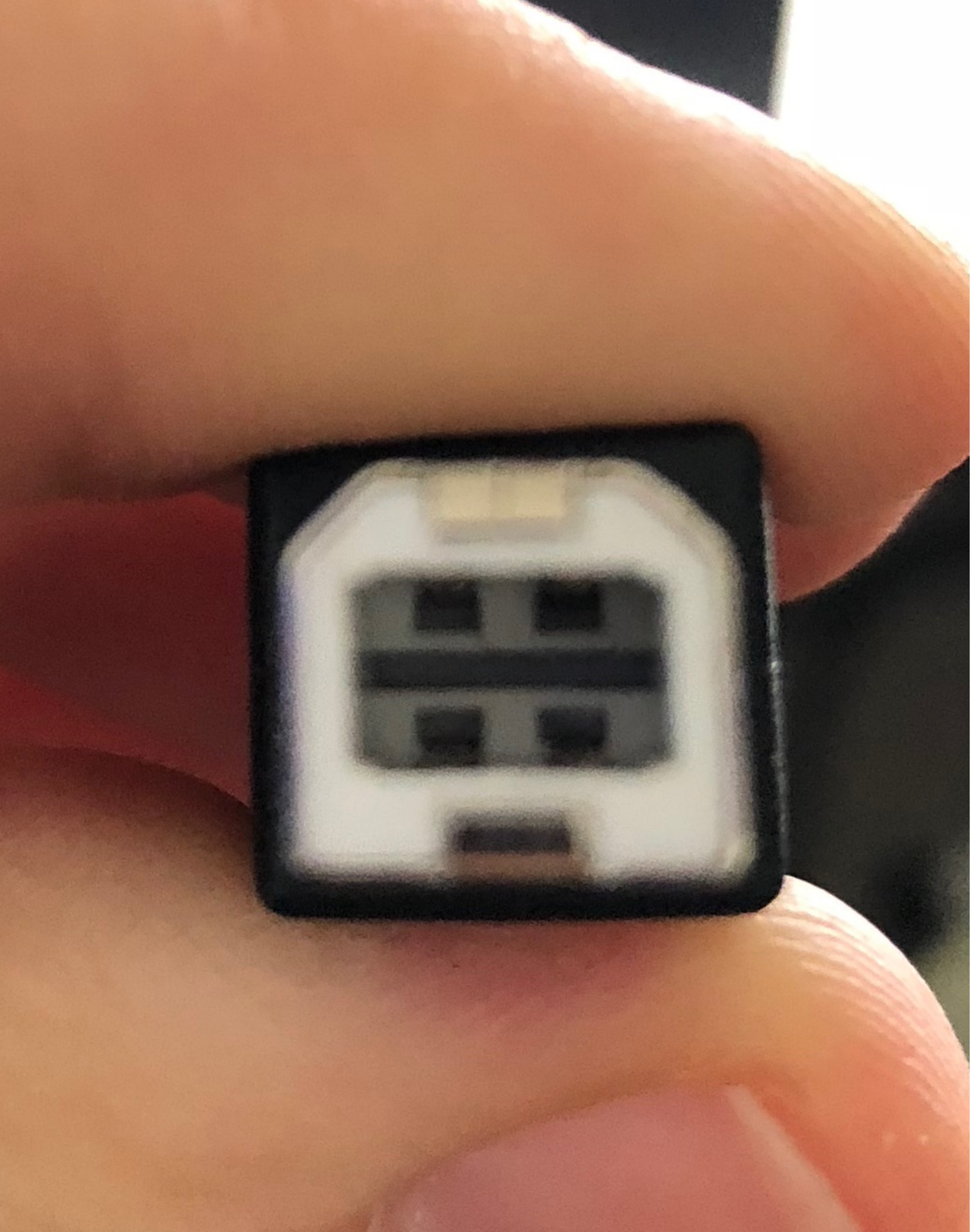
APPLE MADE USB POPULAR WITH THE IMAC G3

2000

USB 2.0 WAS RELEASED, AS WELL AS THE 1ST USB FLASH DRIVE

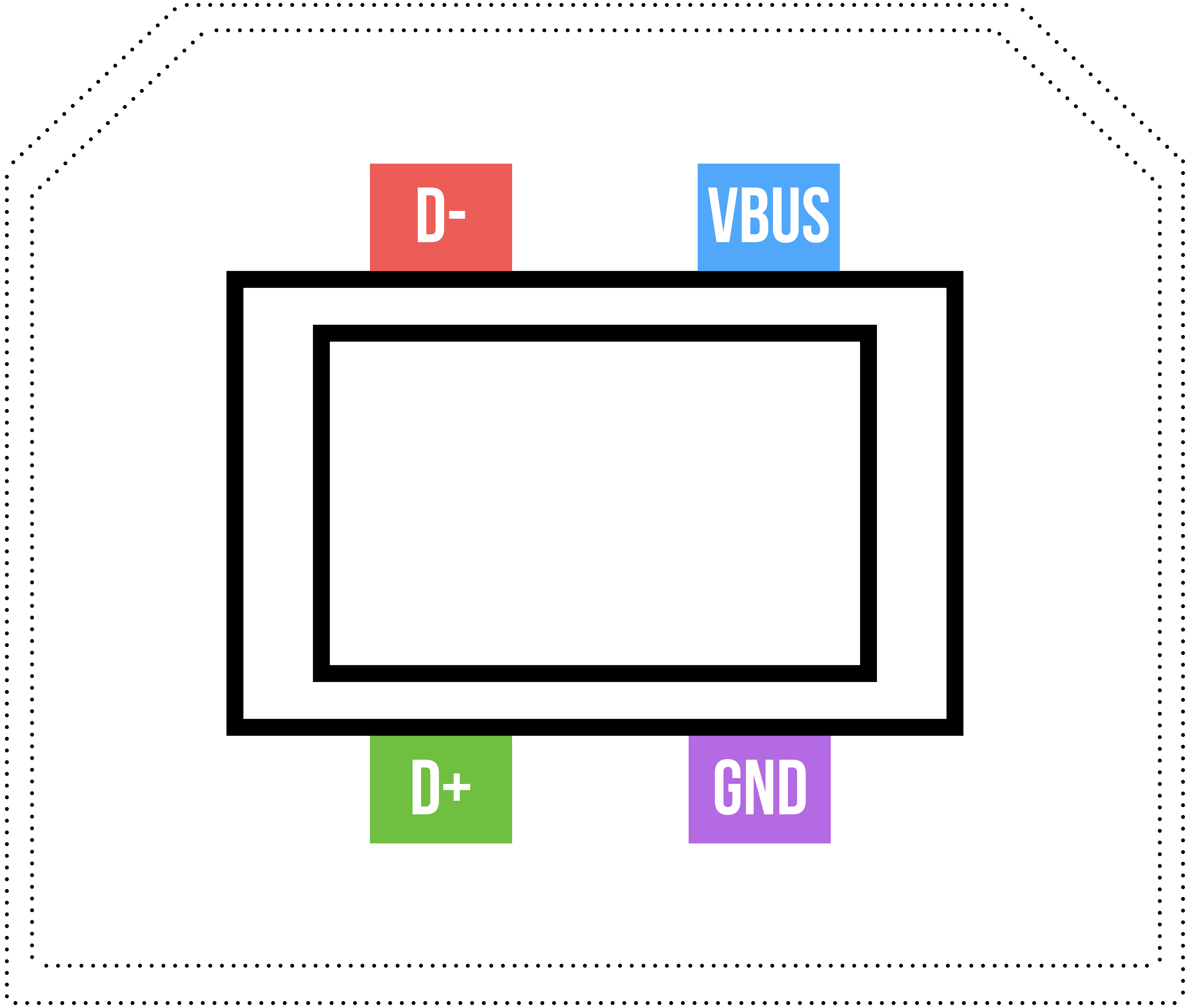
2008

USB 3.0 WAS RELEASED (SUPER SPEED)



Connector Layout

Connector Layout



VBUS - POWER 5V

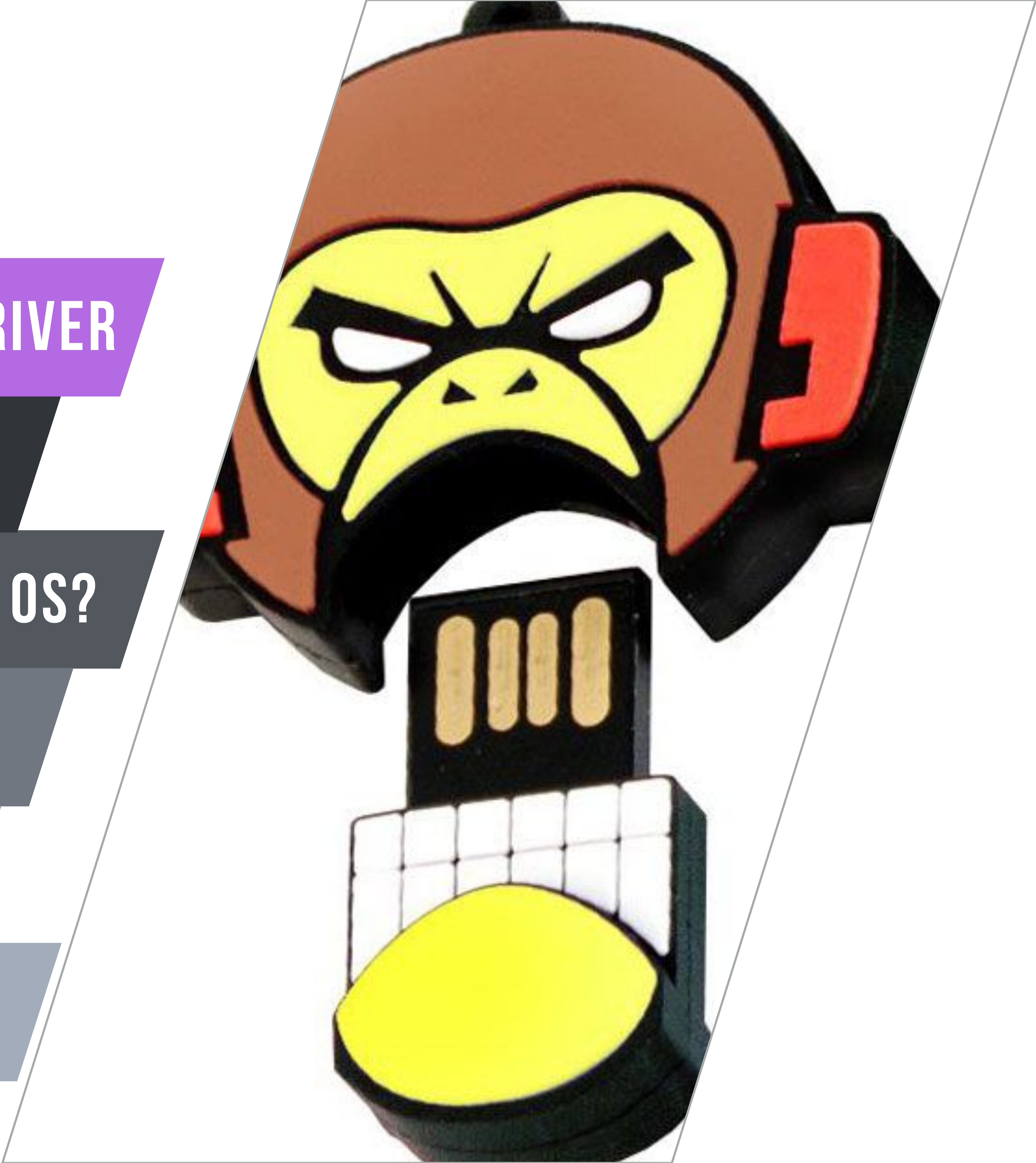
GND - GROUND

D+ - DATA

D- - DATA

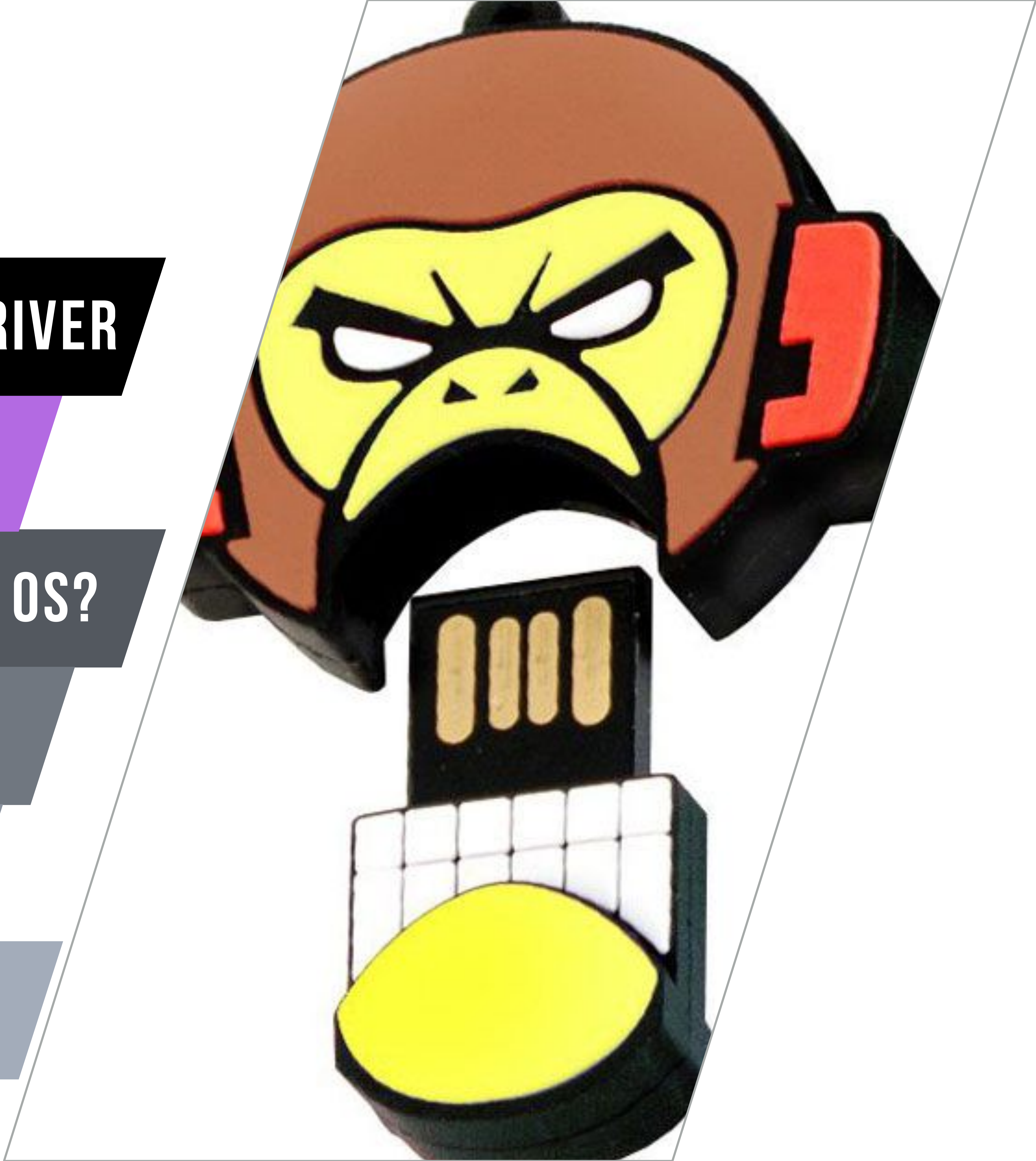
The Usb Way™

- SEARCH THE INTERNET FOR A DRIVER
- INSTALL A NATIVE APPLICATION
- OH, IS THAT SUPPORTED ON MY OS?
- SCARY OS POPUP TURNS UP
- MALFUNCTION KILLS EXISTING SETUP
- CODE STICKS AROUND FOREVER



The Usb Way™

- SEARCH THE INTERNET FOR A DRIVER
- INSTALL A NATIVE APPLICATION
- OH, IS THAT SUPPORTED ON MY OS?
- SCARY OS POPUP TURNS UP
- MALFUNCTION KILLS EXISTING SETUP
- CODE STICKS AROUND FOREVER

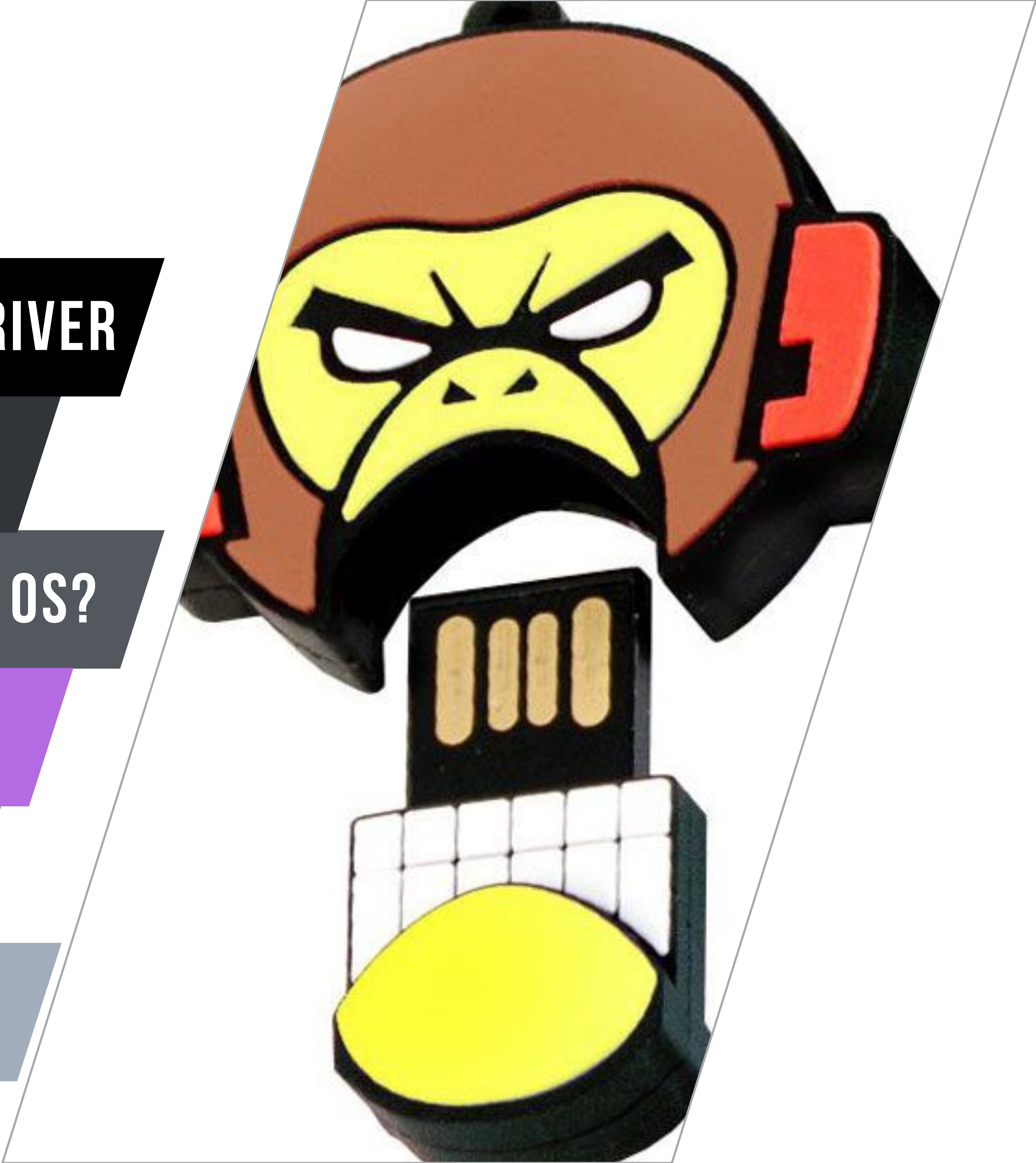


The Usb Way™

- SEARCH THE INTERNET FOR A DRIVER
- INSTALL A NATIVE APPLICATION
- OH, IS THAT SUPPORTED ON MY OS?
- SCARY OS POPUP TURNS UP
- MALFUNCTION KILLS EXISTING SETUP
- CODE STICKS AROUND FOREVER

The Usb Way™

- SEARCH THE INTERNET FOR A DRIVER
- INSTALL A NATIVE APPLICATION
- OH, IS THAT SUPPORTED ON MY OS?
- SCARY OS POPUP TURNS UP
- MALFUNCTION KILLS EXISTING SETUP
- CODE STICKS AROUND FOREVER



The Usb Way™

- SEARCH THE INTERNET FOR A DRIVER
- INSTALL A NATIVE APPLICATION
- OH, IS THAT SUPPORTED ON MY OS?
- SCARY OS POPUP TURNS UP
- MALFUNCTION KILLS EXISTING SETUP
- CODE STICKS AROUND FOREVER

The Usb Way™

SEARCH THE INTERNET FOR A DRIVER

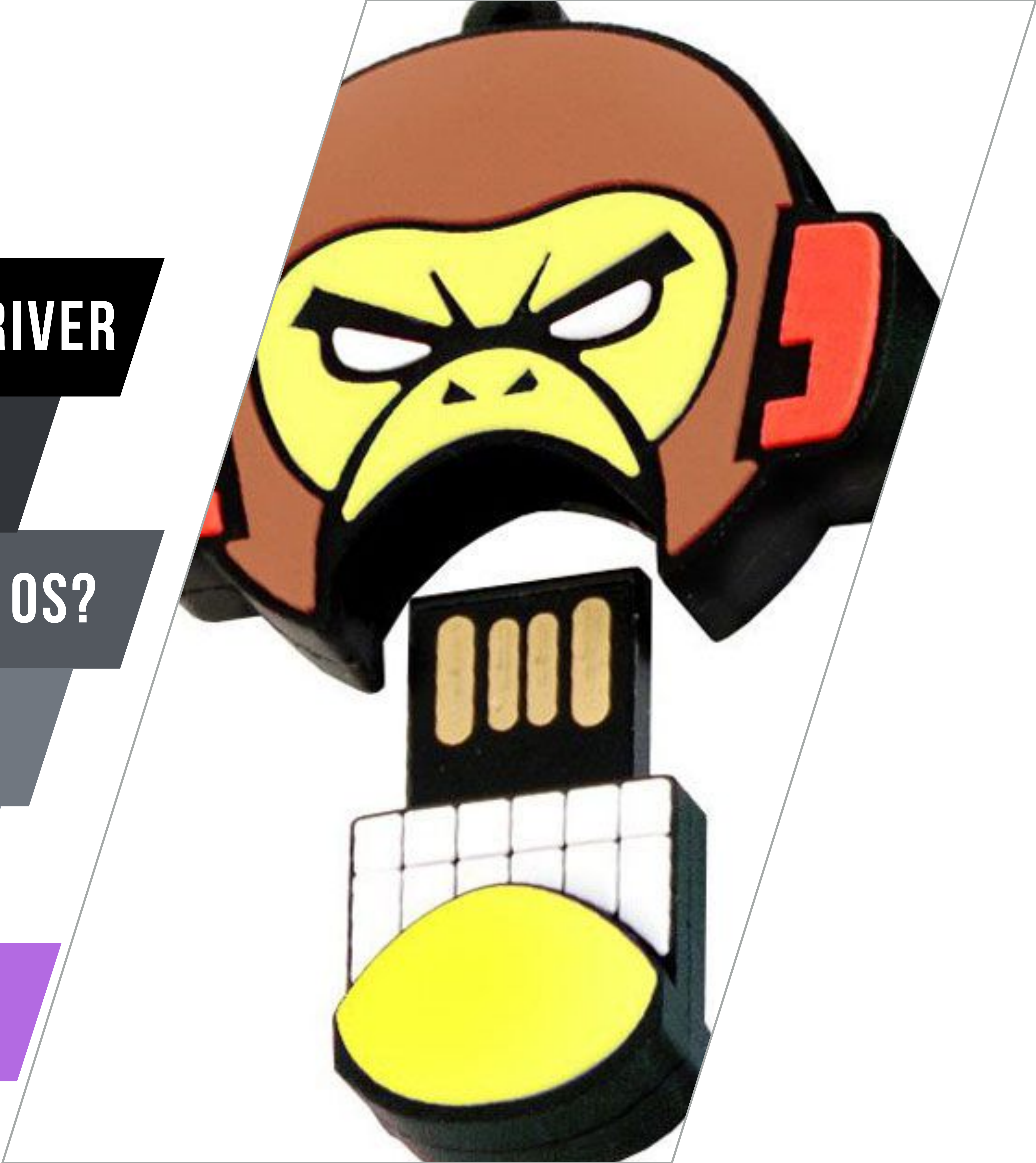
INSTALL A NATIVE APPLICATION

OH, IS THAT SUPPORTED ON MY OS?

SCARY OS POPUP TURNS UP

MALFUNCTION KILLS EXISTING SETUP

CODE STICKS AROUND FOREVER



A Nicer Usb Way™

- BUY A DEVICE
- PLUG IN
- NOTIFICATION APPEARS
- CLICK ON IT
- OPENS WEBSITE
- FUN!

A Nicer Usb Way™

- BUY A DEVICE
- PLUG IN
- NOTIFICATION APPEARS
- CLICK ON IT
- OPENS WEBSITE
- FUN!

A Nicer Usb Way™

BUY A DEVICE

PLUG IN

NOTIFICATION APPEARS

CLICK ON IT

OPENS WEBSITE

FUN!

A Nicer Usb Way™

- BUY A DEVICE
- PLUG IN
- NOTIFICATION APPEARS
- CLICK ON IT
- OPENS WEBSITE
- FUN!

A Nicer Usb Way™

- BUY A DEVICE
- PLUG IN
- NOTIFICATION APPEARS
- CLICK ON IT
- OPENS WEBSITE
- FUN!

A Nicer Usb Way™

BUY A DEVICE

PLUG IN

NOTIFICATION APPEARS

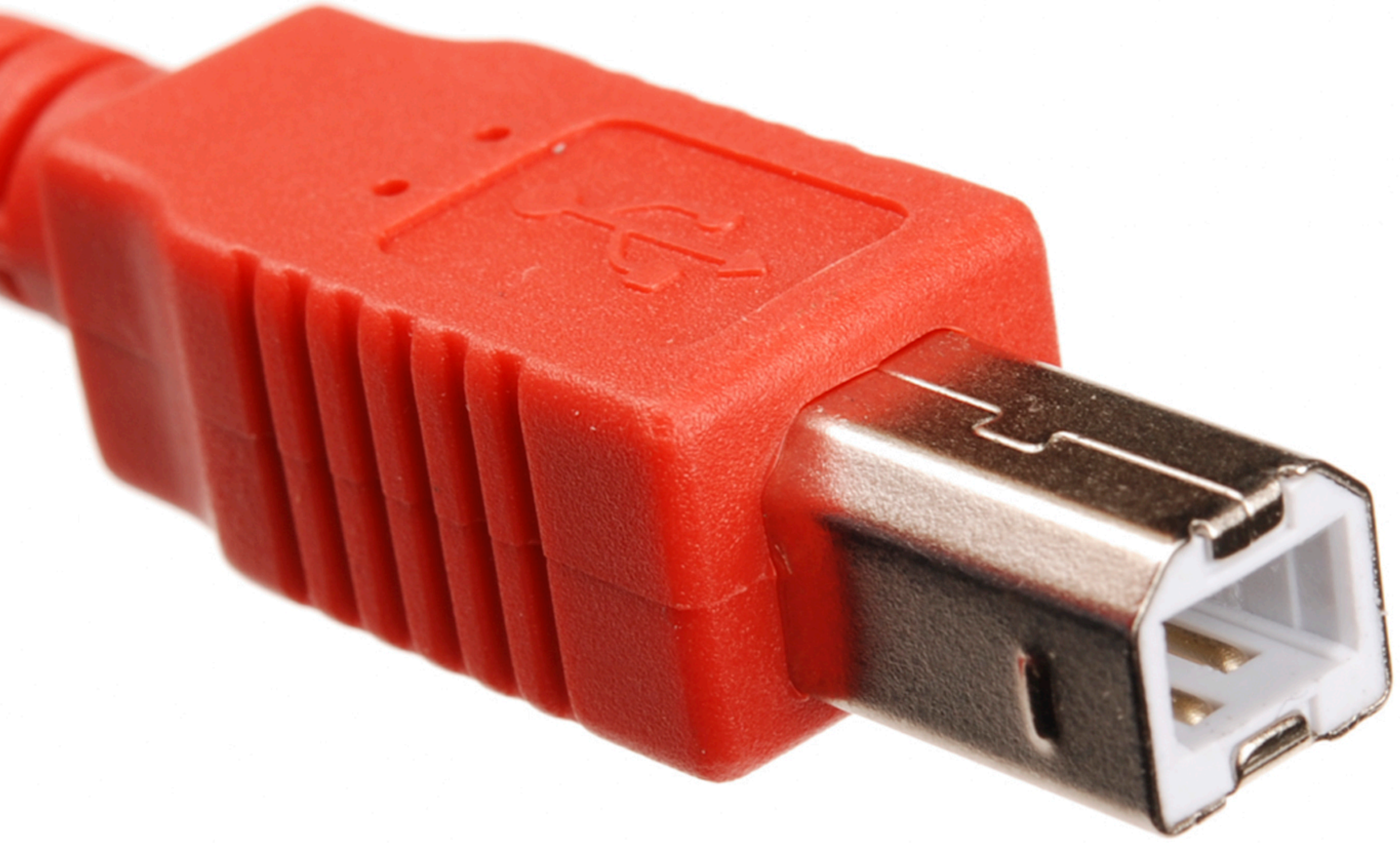
CLICK ON IT

OPENS WEBSITE

FUN!

Web Usb 💖

Webusb Facts



Only Available Via Https

* FOR SECURITY REASONS, LIKE DONE WITH ALL NEW PLATFORM FEATURES *

No „Native Code“ Needed

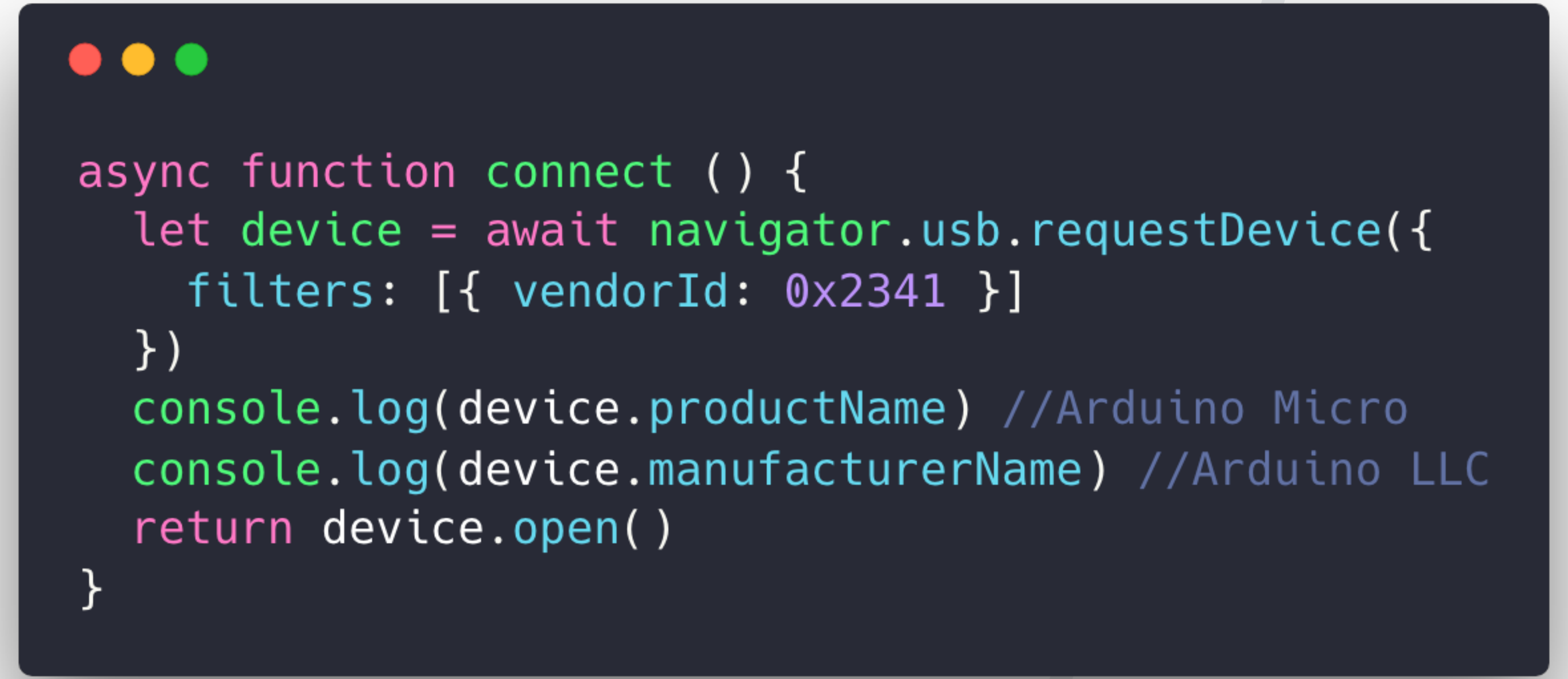
* NO NEED TO INSTALL ANY DRIVERS OR OTHER 3RD PARTY SOFTWARE*

Real Plug & Play

* PLUG IN DEVICE, CLICK LINK, USE IT*

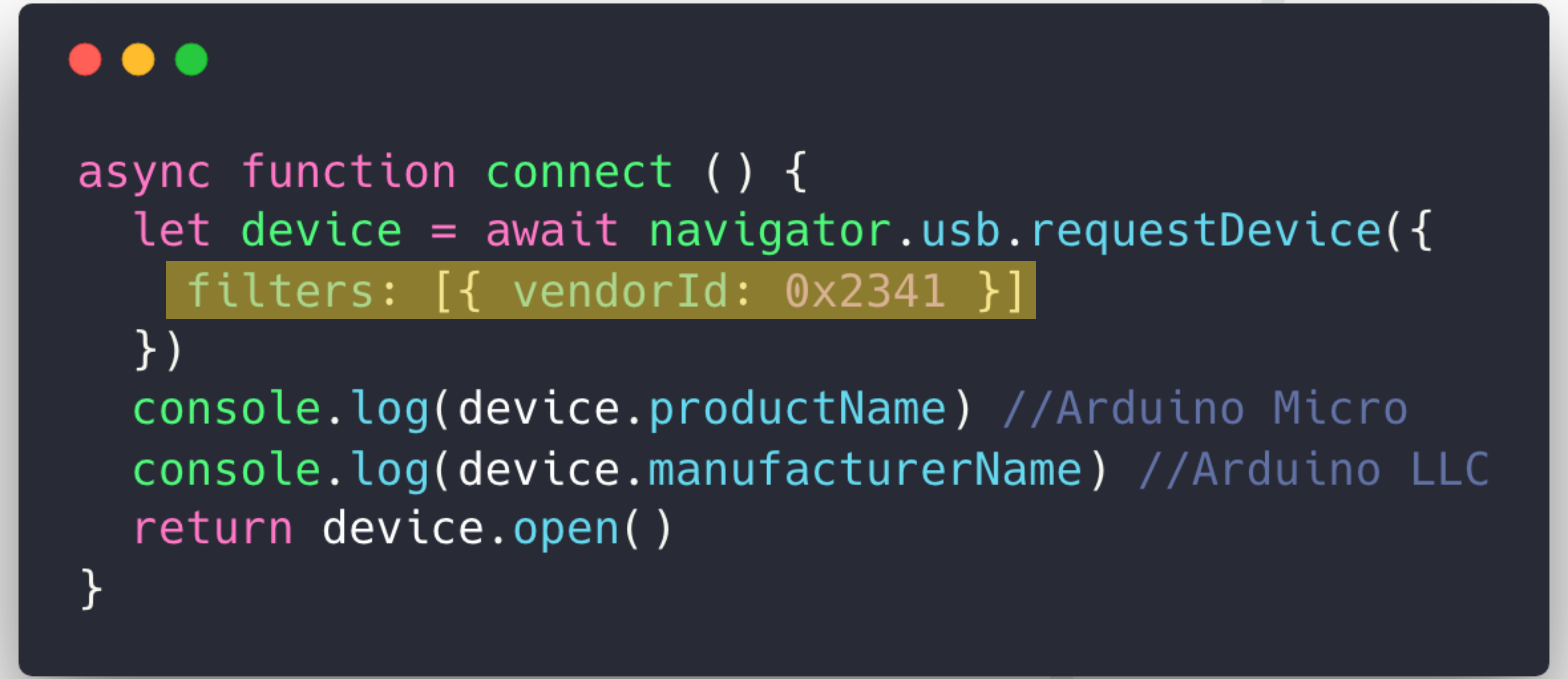
Code, Now, Please… 👾

DEVICE DESCRIPTOR



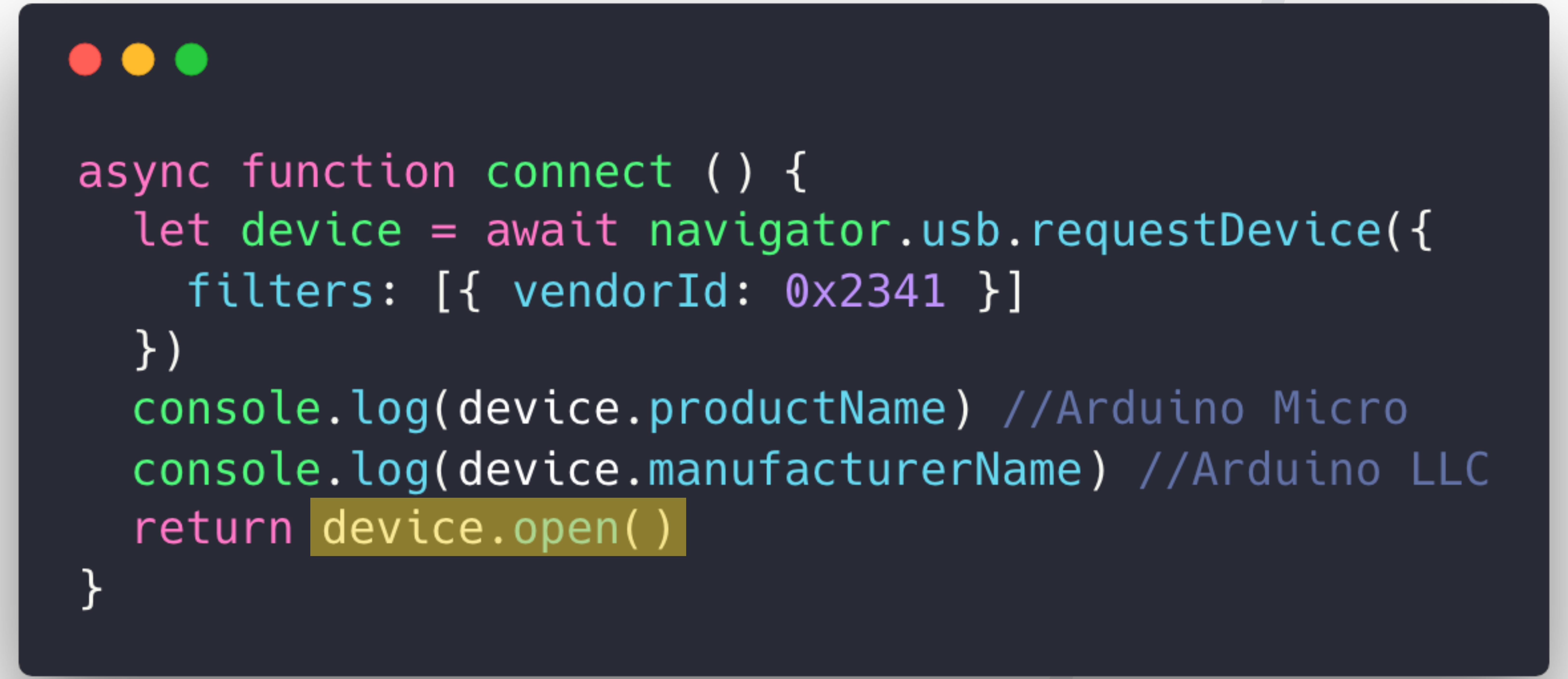
```
async function connect () {
  let device = await navigator.usb.requestDevice({
    filters: [{ vendorId: 0x2341 }]
  })
  console.log(device.productName) //Arduino Micro
  console.log(device.manufacturerName) //Arduino LLC
  return device.open()
}
```

DEVICE DESCRIPTOR

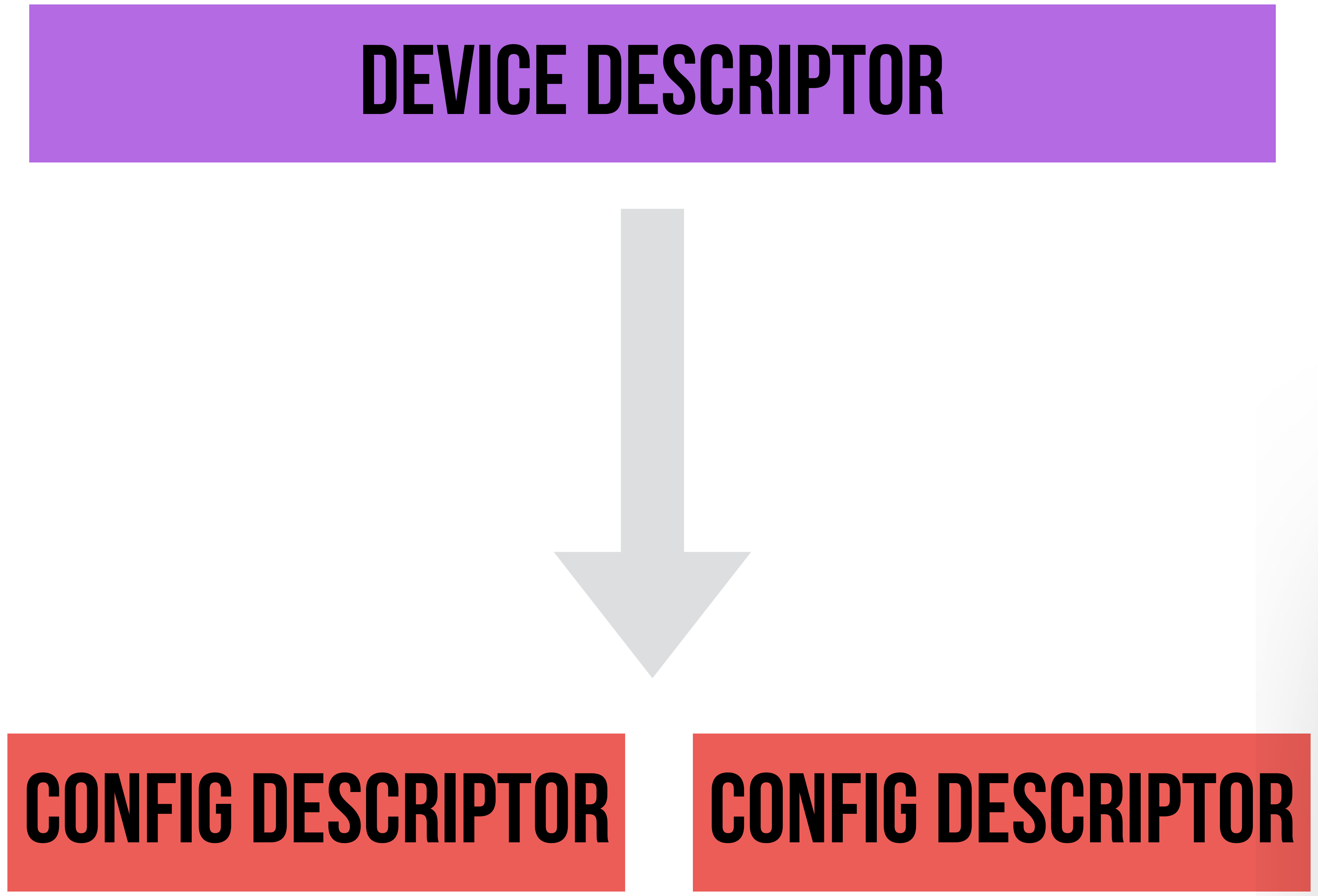


```
async function connect () {
  let device = await navigator.usb.requestDevice({
    filters: [{ vendorId: 0x2341 }]
  })
  console.log(device.productName) //Arduino Micro
  console.log(device.manufacturerName) //Arduino LLC
  return device.open()
}
```

DEVICE DESCRIPTOR



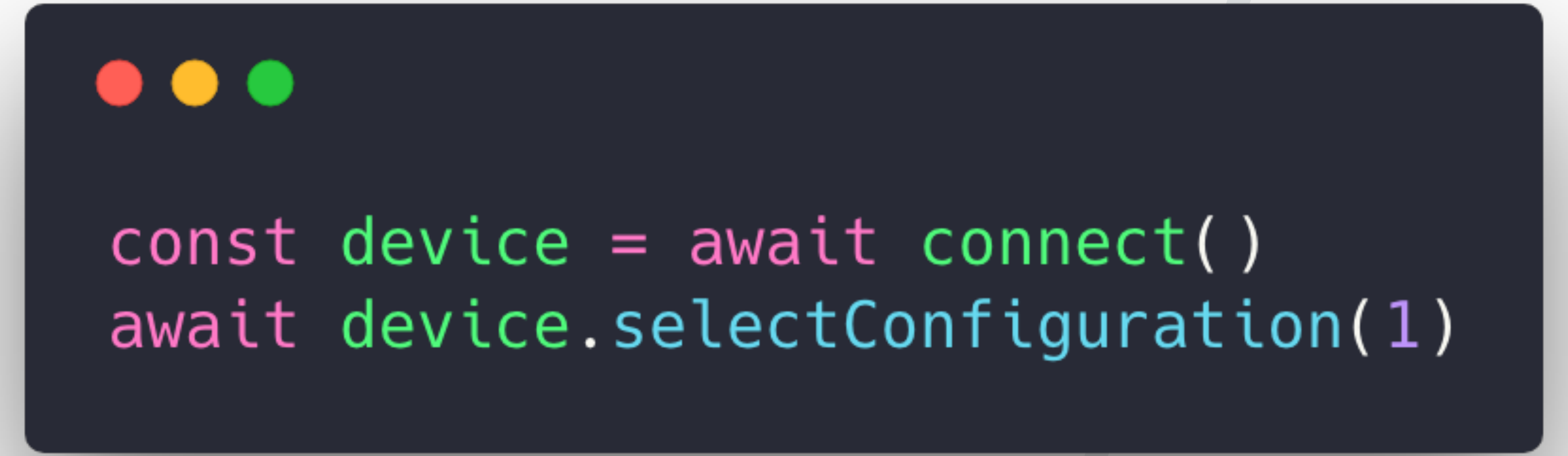
```
async function connect () {
  let device = await navigator.usb.requestDevice({
    filters: [{ vendorId: 0x2341 }]
  })
  console.log(device.productName) //Arduino Micro
  console.log(device.manufacturerName) //Arduino LLC
  return device.open()
}
```



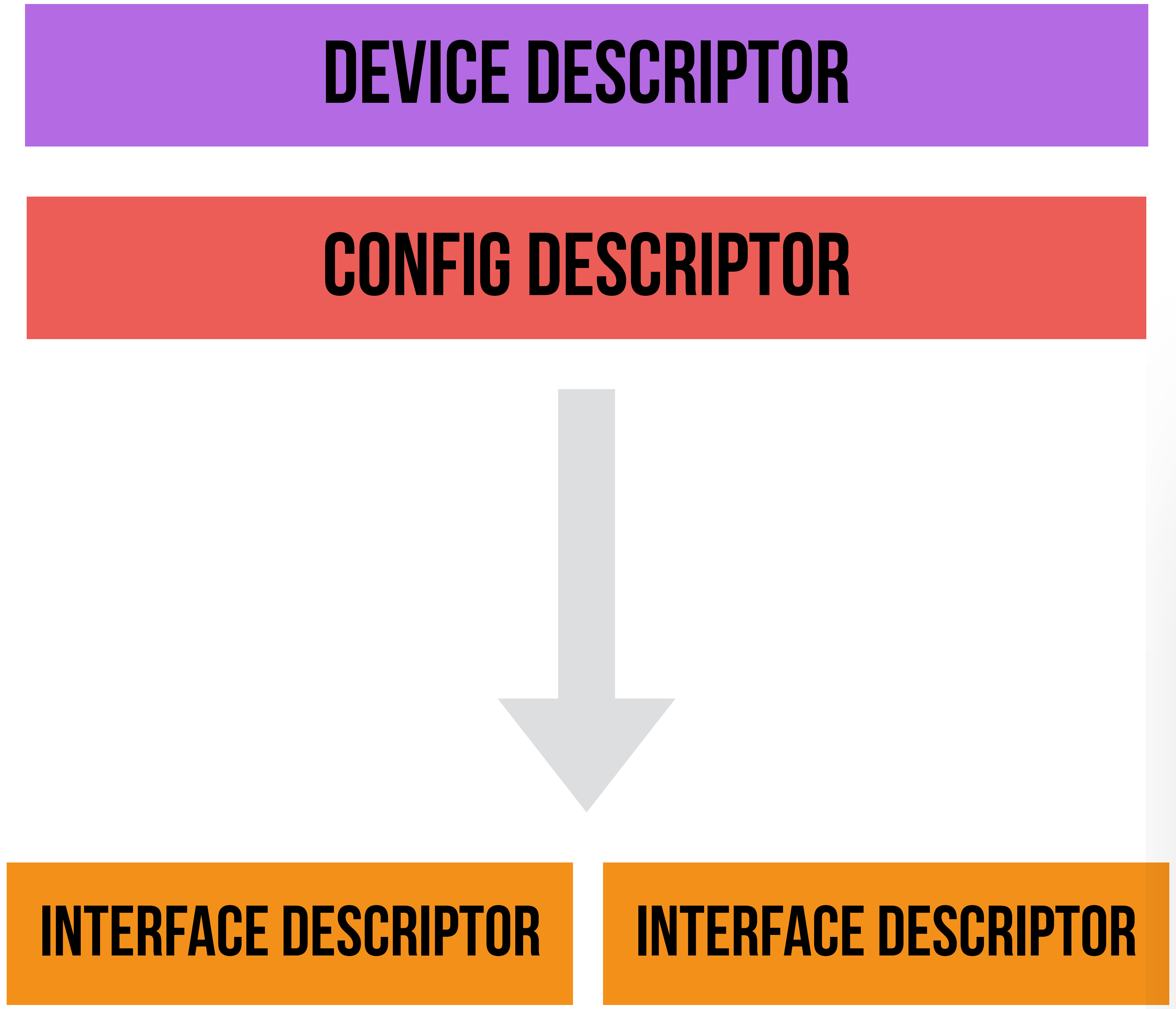
DEVICE DESCRIPTOR

CONFIG DESCRIPTOR

CONFIG DESCRIPTOR



```
const device = await connect()
await device.selectConfiguration(1)
```

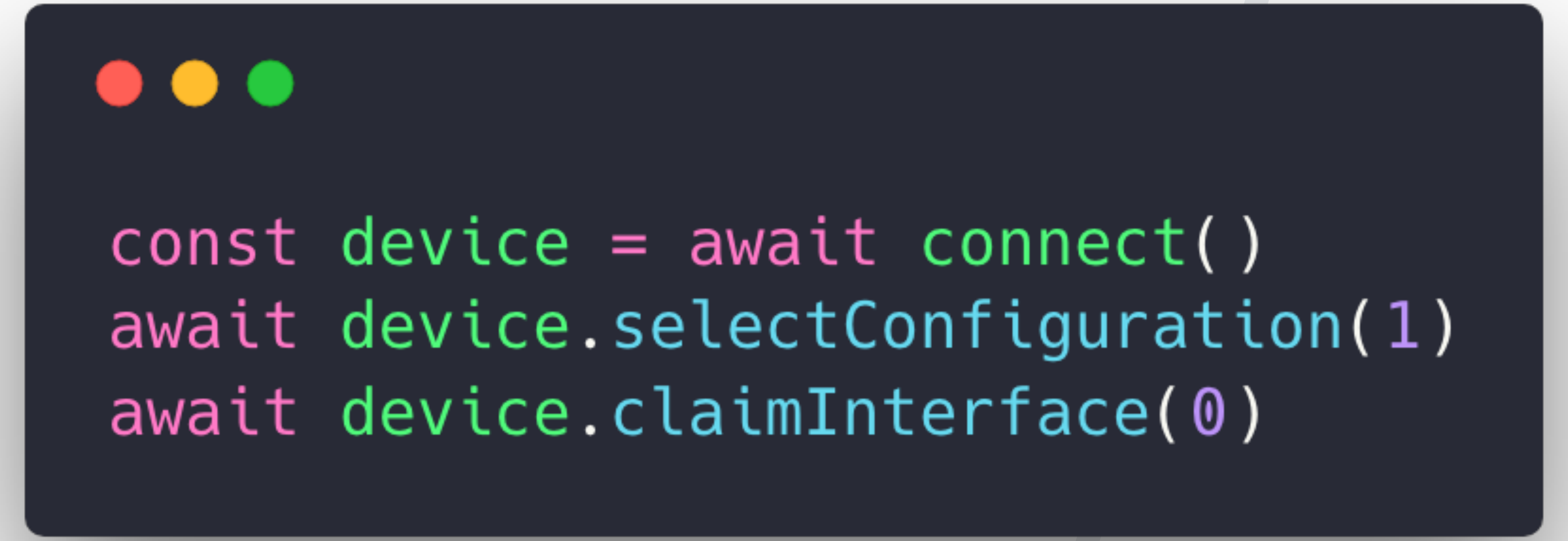


DEVICE DESCRIPTOR

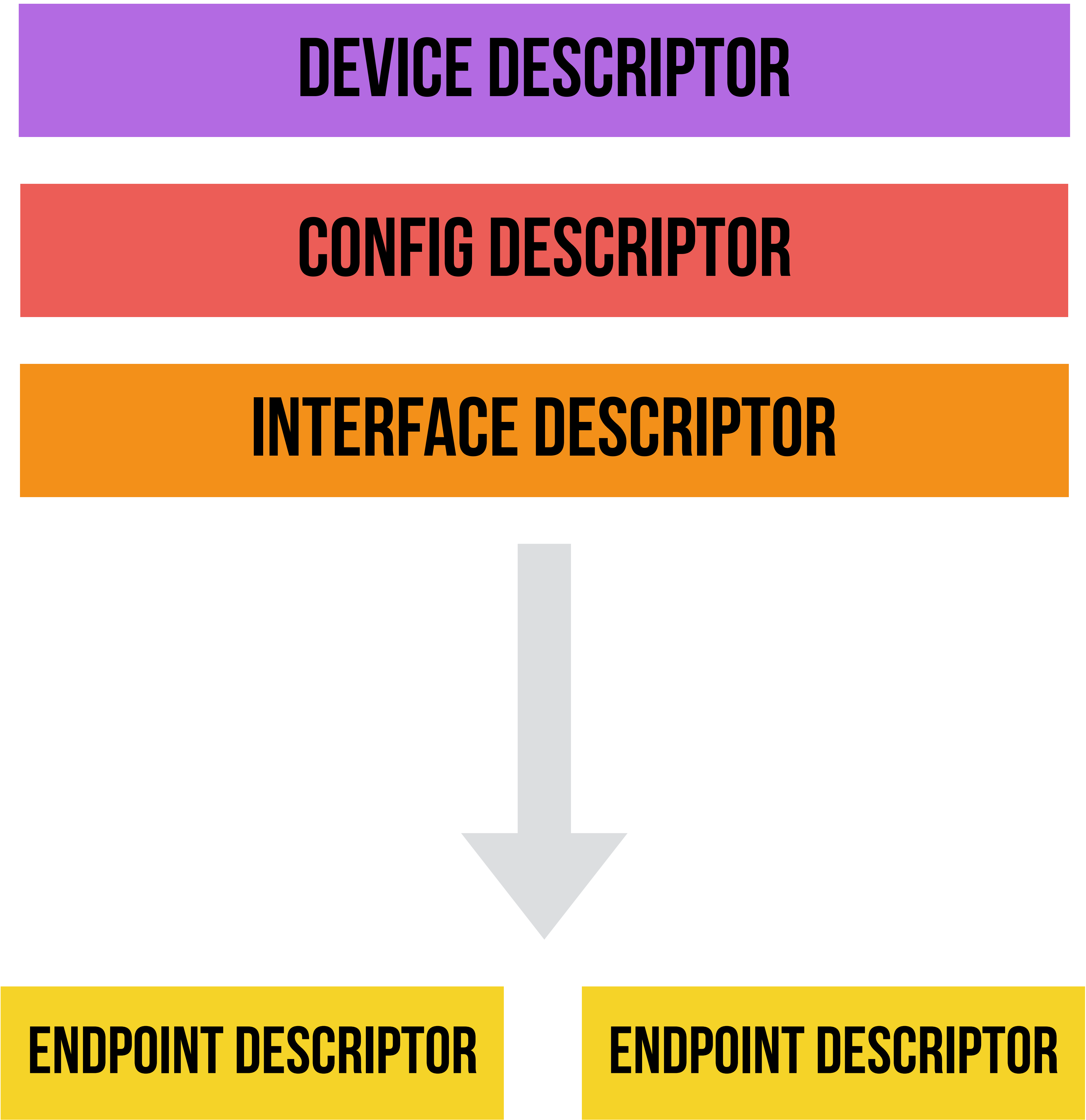
CONFIG DESCRIPTOR

INTERFACE DESCRIPTOR

INTERFACE DESCRIPTOR



```
const device = await connect()
await device.selectConfiguration(1)
await device.claimInterface(0)
```



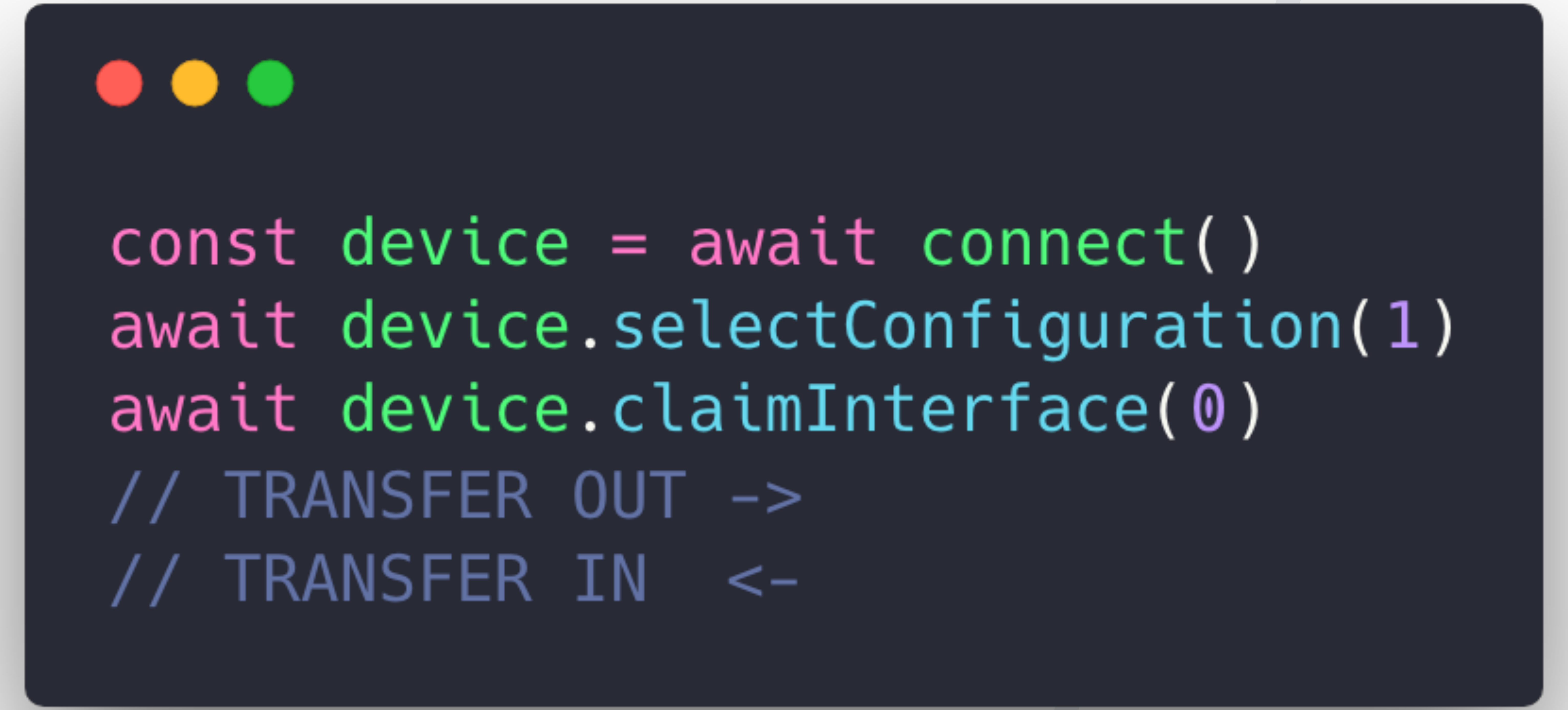
DEVICE DESCRIPTOR

CONFIG DESCRIPTOR

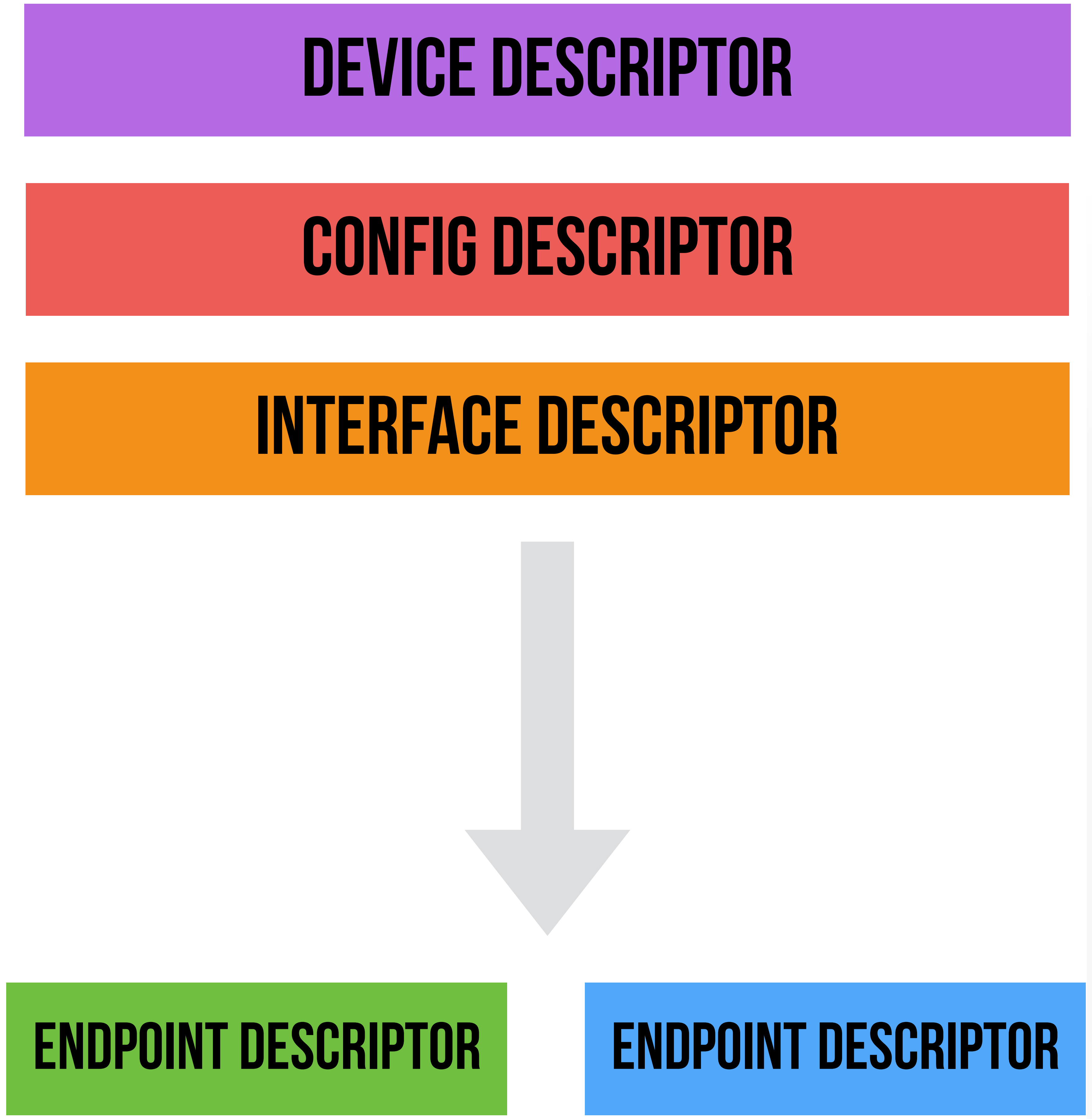
INTERFACE DESCRIPTOR

ENDPOINT DESCRIPTOR

ENDPOINT DESCRIPTOR



```
const device = await connect()
await device.selectConfiguration(1)
await device.claimInterface(0)
// TRANSFER OUT ->
// TRANSFER IN  <-
```



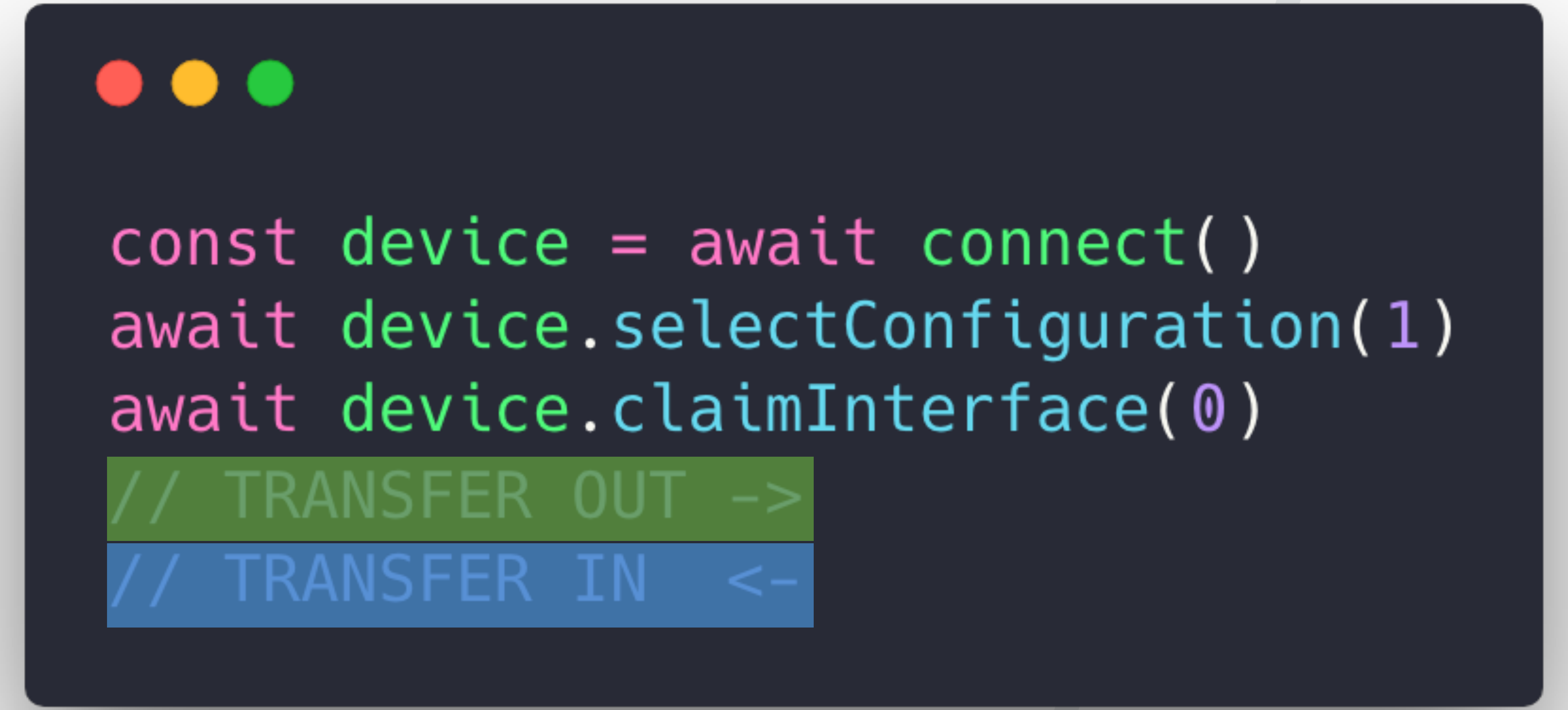
DEVICE DESCRIPTOR

CONFIG DESCRIPTOR

INTERFACE DESCRIPTOR

ENDPOINT DESCRIPTOR

ENDPOINT DESCRIPTOR



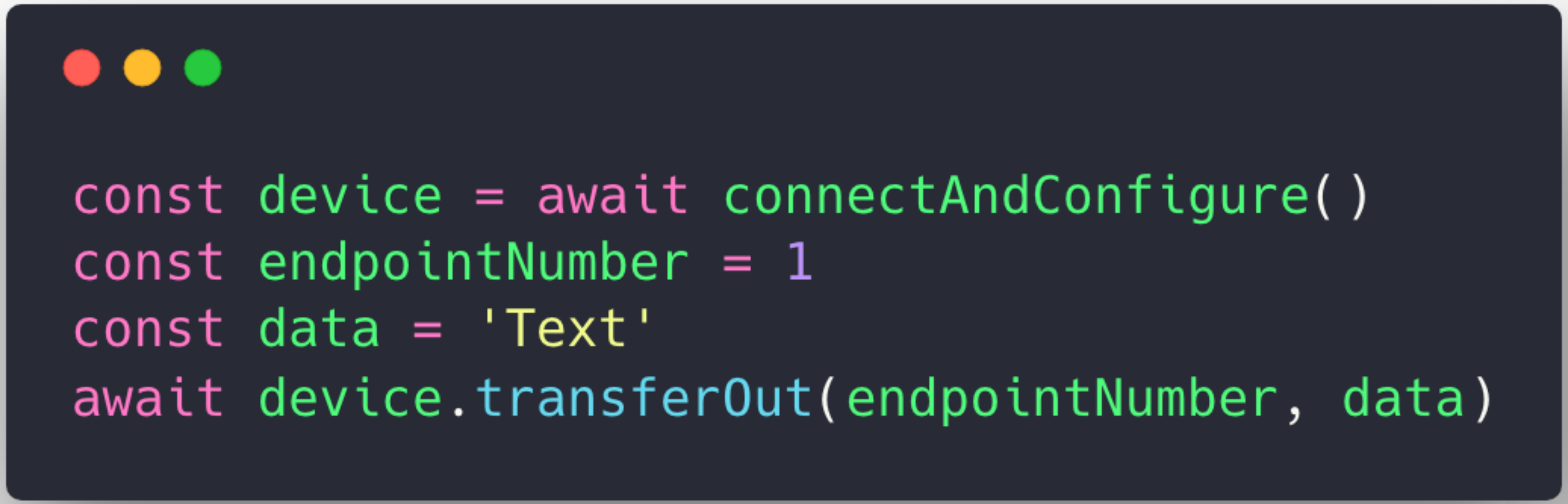
```
const device = await connect()
await device.selectConfiguration(1)
await device.claimInterface(0)
// TRANSFER OUT ->
// TRANSFER IN  <-
```

INTERRUPT TRANSFER

CONTROL

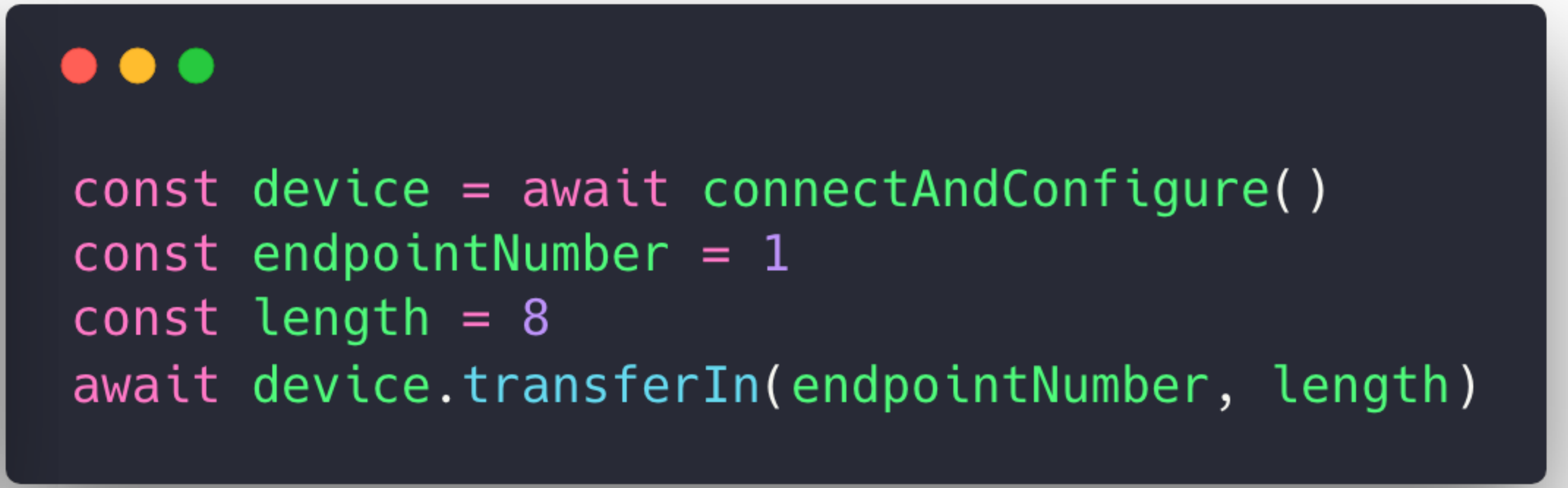
ISOCHRONOUS

OUT



```
const device = await connectAndConfigure()
const endpointNumber = 1
const data = 'Text'
await device.transferOut(endpointNumber, data)
```

IN

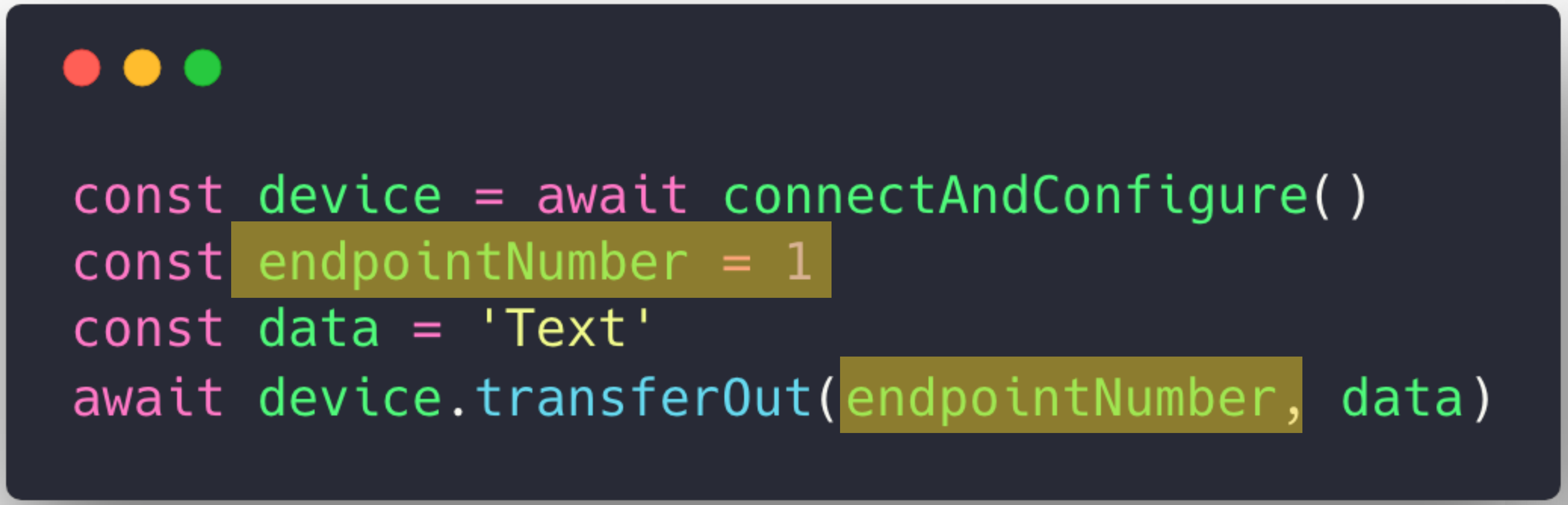


```
const device = await connectAndConfigure()
const endpointNumber = 1
const length = 8
await device.transferIn(endpointNumber, length)
```

* INTERRUPT TRANSFERS ARE TYPICALLY NON-PERIODIC, SMALL, DEVICE "INITIATED" COMMUNICATION THAT REQUIRE A SPECIFIC LATENCY *

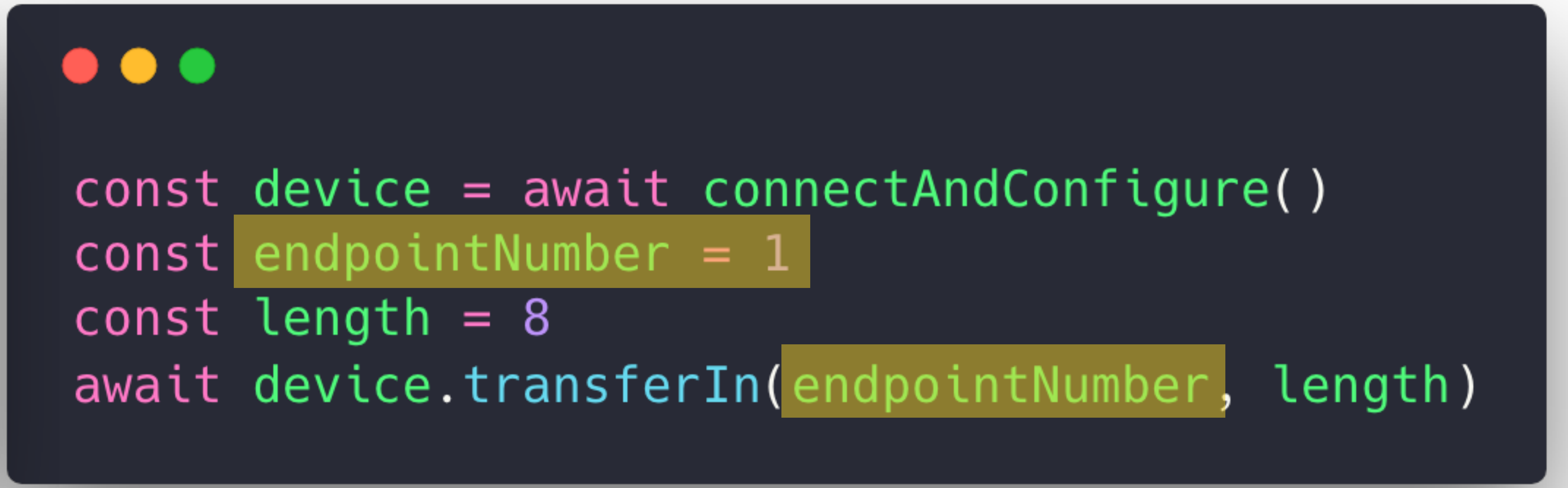
INTERRUPT TRANSFER | CONTROL | ISOCHRONOUS

OUT



```
const device = await connectAndConfigure()
const endpointNumber = 1
const data = 'Text'
await device.transferOut(endpointNumber, data)
```

IN

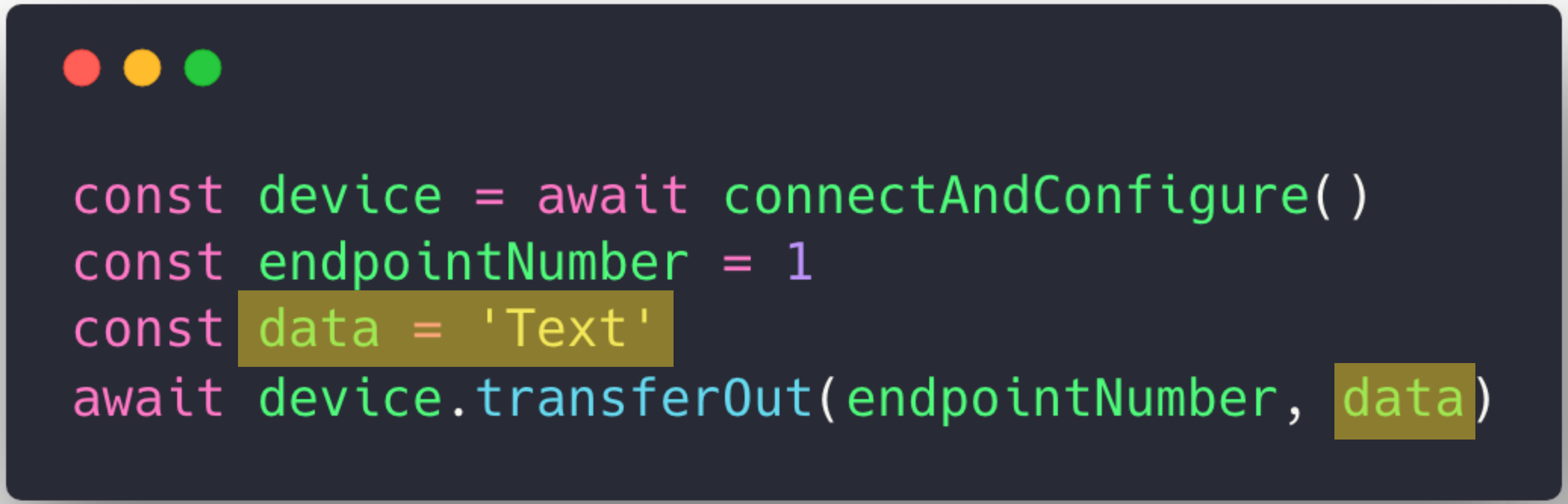


```
const device = await connectAndConfigure()
const endpointNumber = 1
const length = 8
await device.transferIn(endpointNumber, length)
```

* INTERRUPT TRANSFERS ARE TYPICALLY NON-PERIODIC, SMALL, DEVICE "INITIATED" COMMUNICATION THAT REQUIRE A SPECIFIC LATENCY *

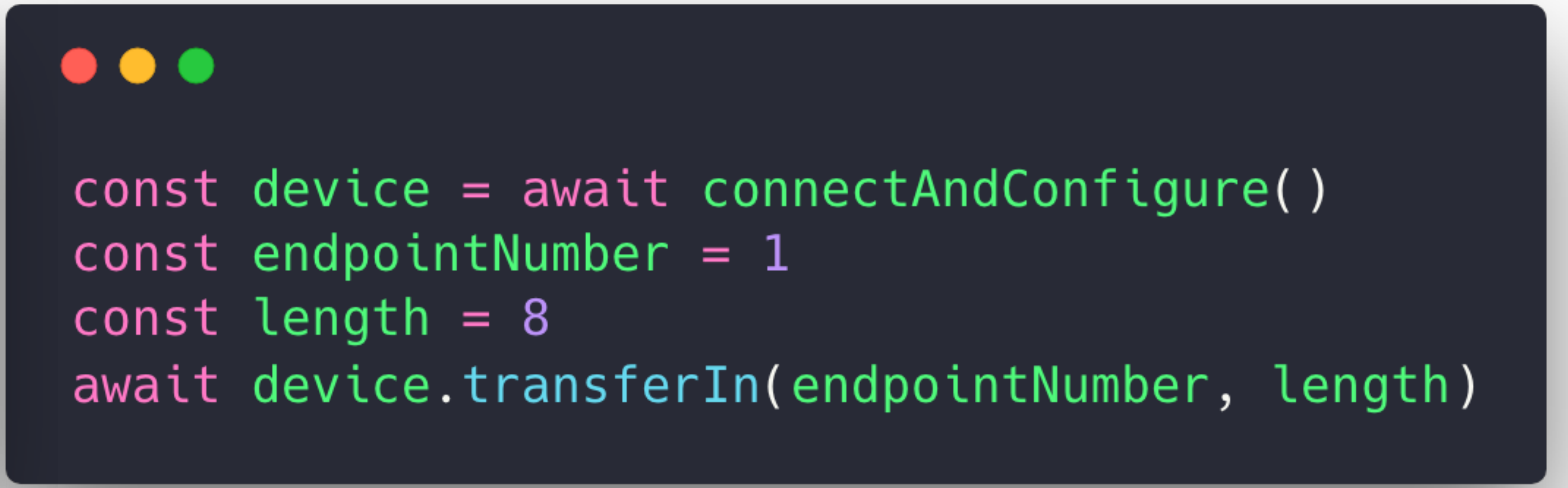
INTERRUPT TRANSFER | CONTROL | ISOCHRONOUS

OUT



```
const device = await connectAndConfigure()
const endpointNumber = 1
const data = 'Text'
await device.transferOut(endpointNumber, data)
```

IN



```
const device = await connectAndConfigure()
const endpointNumber = 1
const length = 8
await device.transferIn(endpointNumber, length)
```

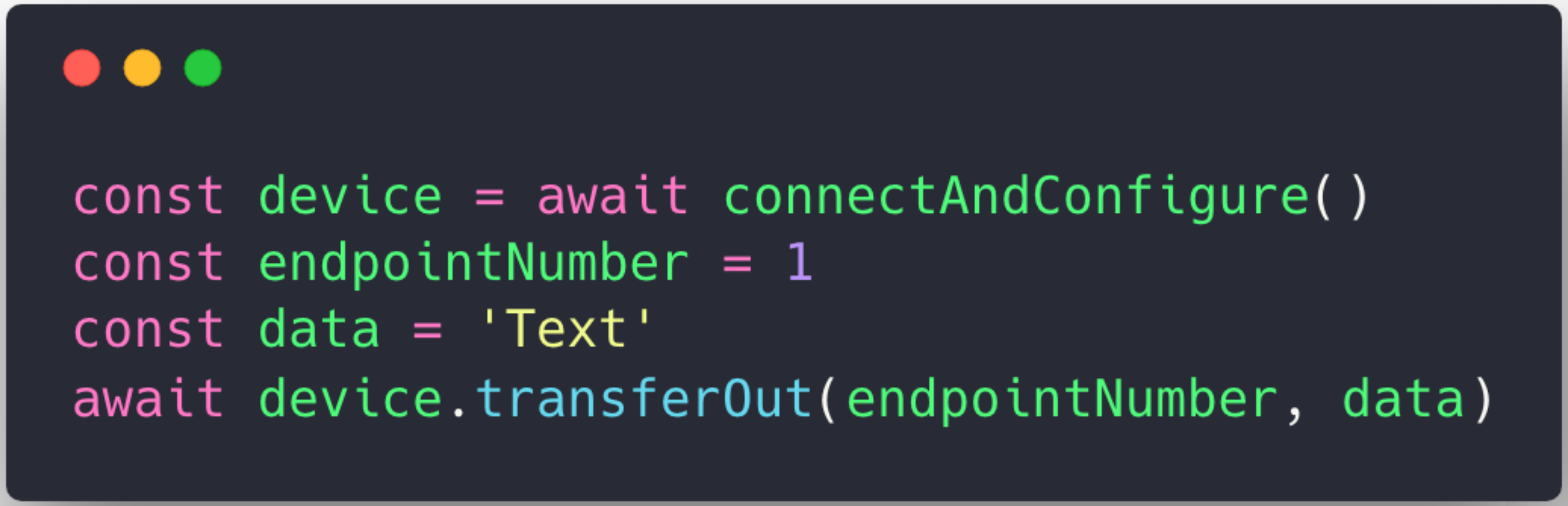
* INTERRUPT TRANSFERS ARE TYPICALLY NON-PERIODIC, SMALL, DEVICE "INITIATED" COMMUNICATION THAT REQUIRE A SPECIFIC LATENCY *

INTERRUPT TRANSFER

CONTROL

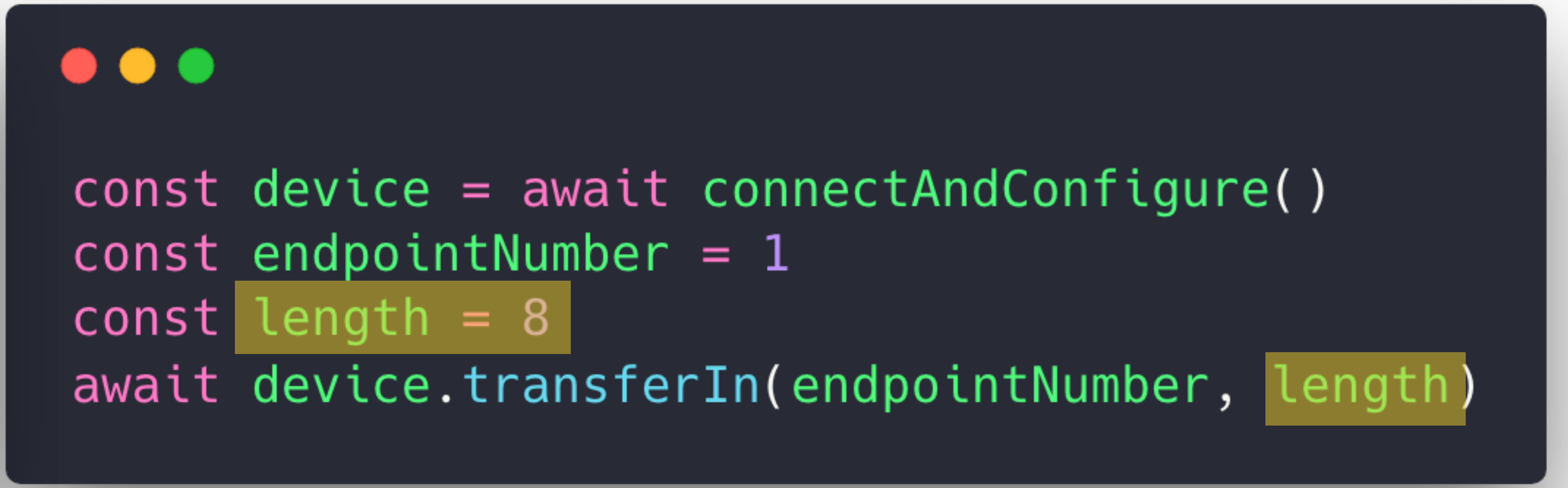
ISOCHRONOUS

OUT



```
const device = await connectAndConfigure()
const endpointNumber = 1
const data = 'Text'
await device.transferOut(endpointNumber, data)
```

IN

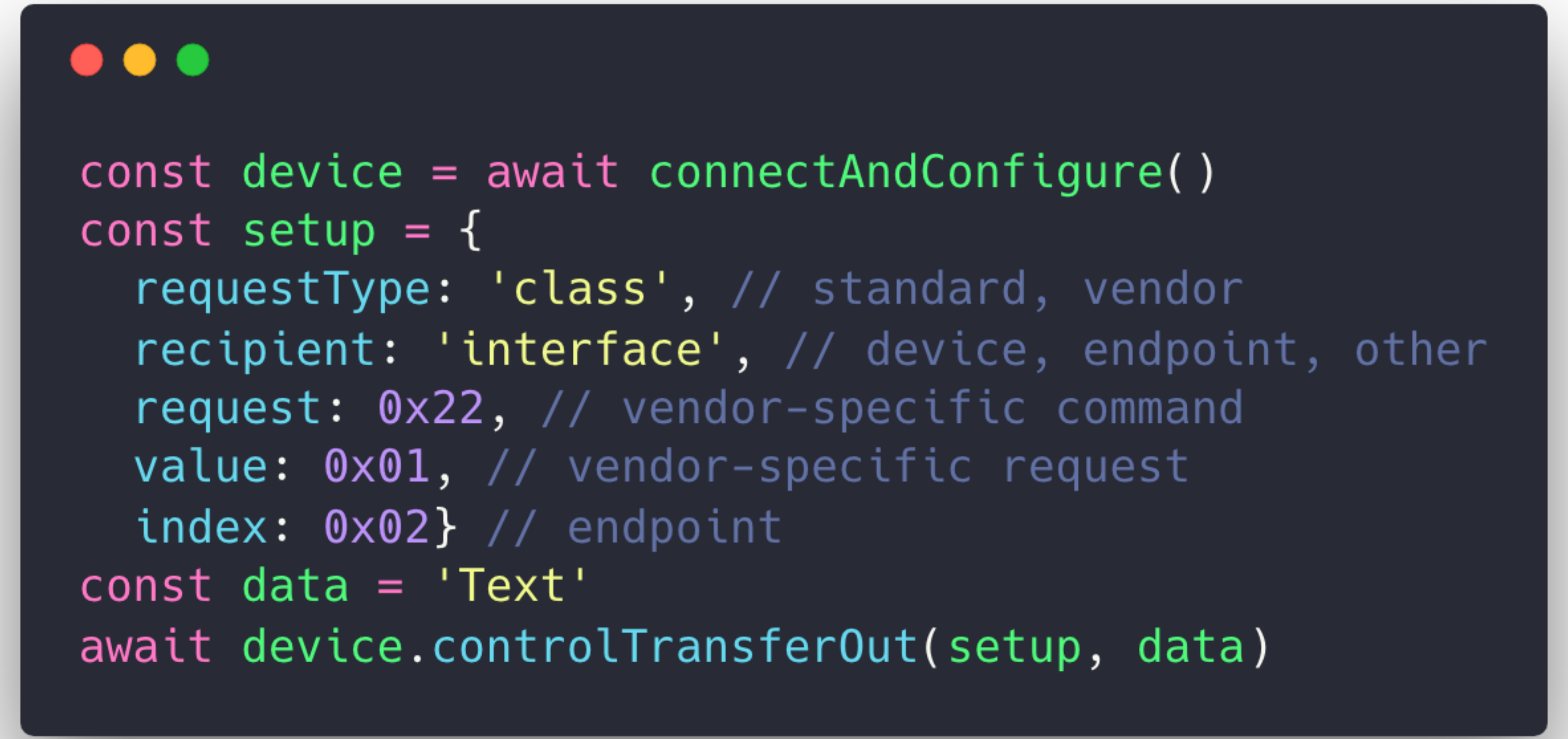


```
const device = await connectAndConfigure()
const endpointNumber = 1
const length = 8
await device.transferIn(endpointNumber, length)
```

* INTERRUPT TRANSFERS ARE TYPICALLY NON-PERIODIC, SMALL, DEVICE "INITIATED" COMMUNICATION THAT REQUIRE A SPECIFIC LATENCY *

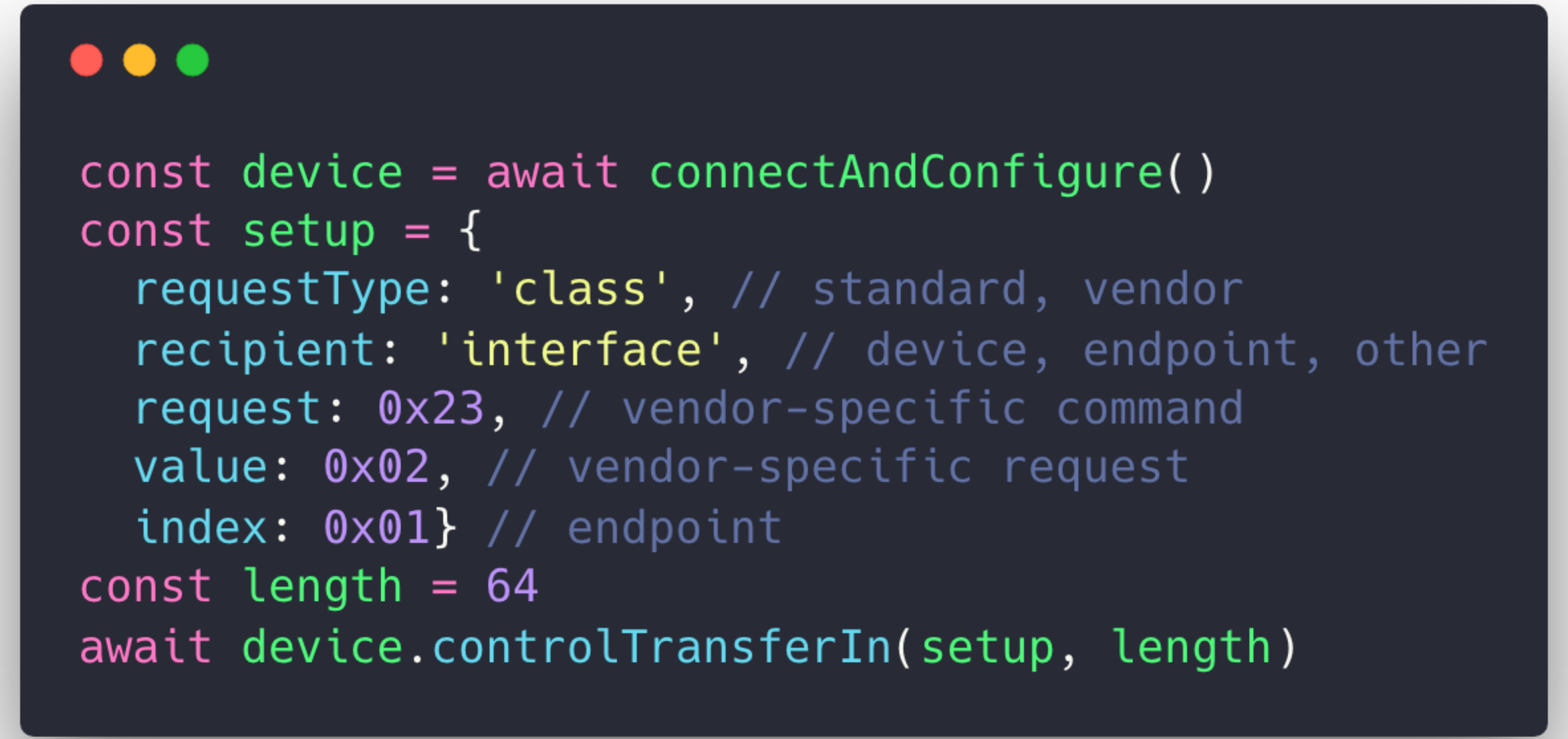
INTERRUPT | **CONTROL TRANSFER** | ISOCHRONOUS

OUT



```
const device = await connectAndConfigure()
const setup = {
  requestType: 'class', // standard, vendor
  recipient: 'interface', // device, endpoint, other
  request: 0x22, // vendor-specific command
  value: 0x01, // vendor-specific request
  index: 0x02} // endpoint
const data = 'Text'
await device.controlTransferOut(setup, data)
```

IN



```
const device = await connectAndConfigure()
const setup = {
  requestType: 'class', // standard, vendor
  recipient: 'interface', // device, endpoint, other
  request: 0x23, // vendor-specific command
  value: 0x02, // vendor-specific request
  index: 0x01} // endpoint
const length = 64
await device.controlTransferIn(setup, length)
```

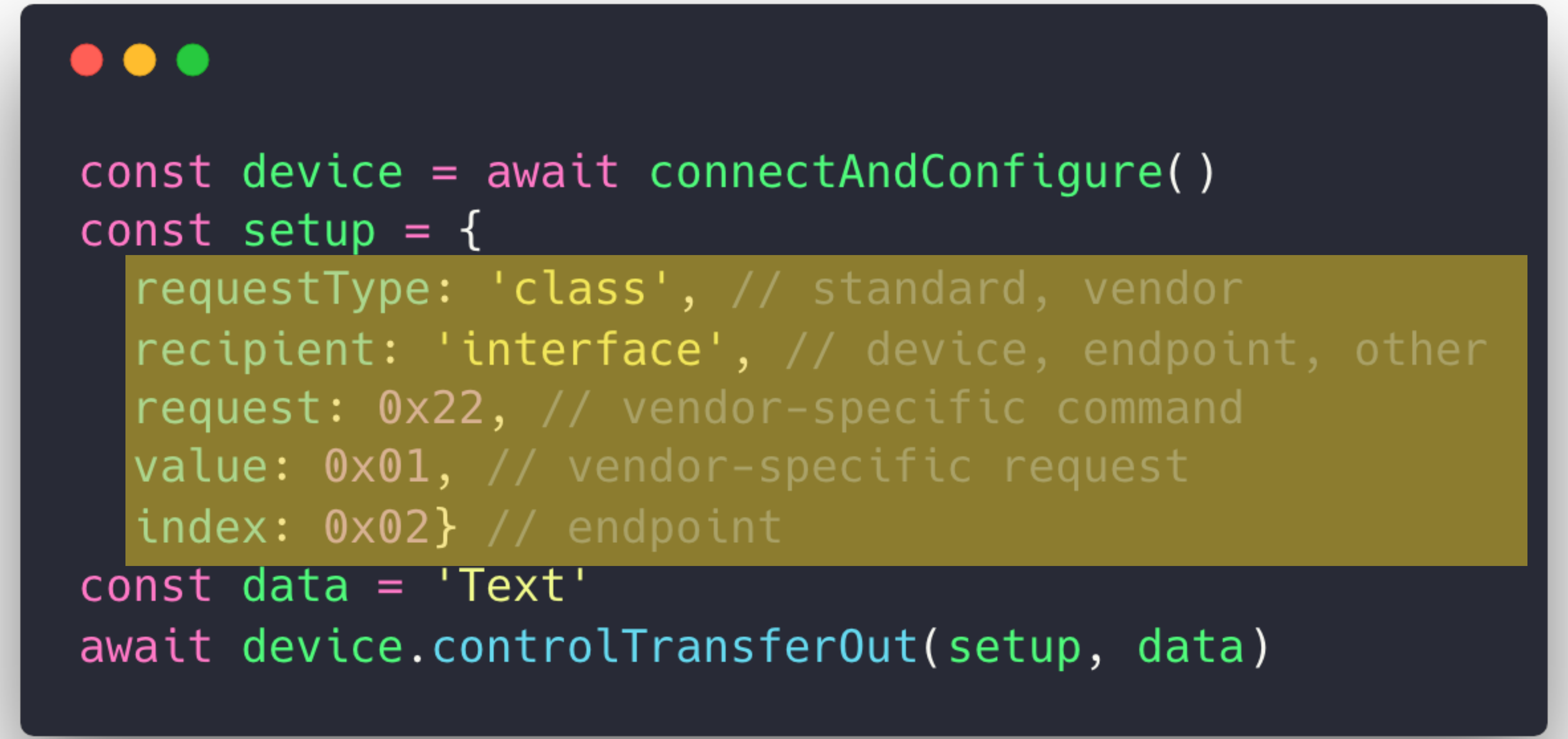
* CONTROL TRANSFERS ARE ESPECIALLY NICE FOR SMALL CONFIGURATION COMMANDS AS THEY GET BUS PRIORITY AND HAVE A WELL DEFINED STRUCTURE *

INTERRUPT

CONTROL TRANSFER

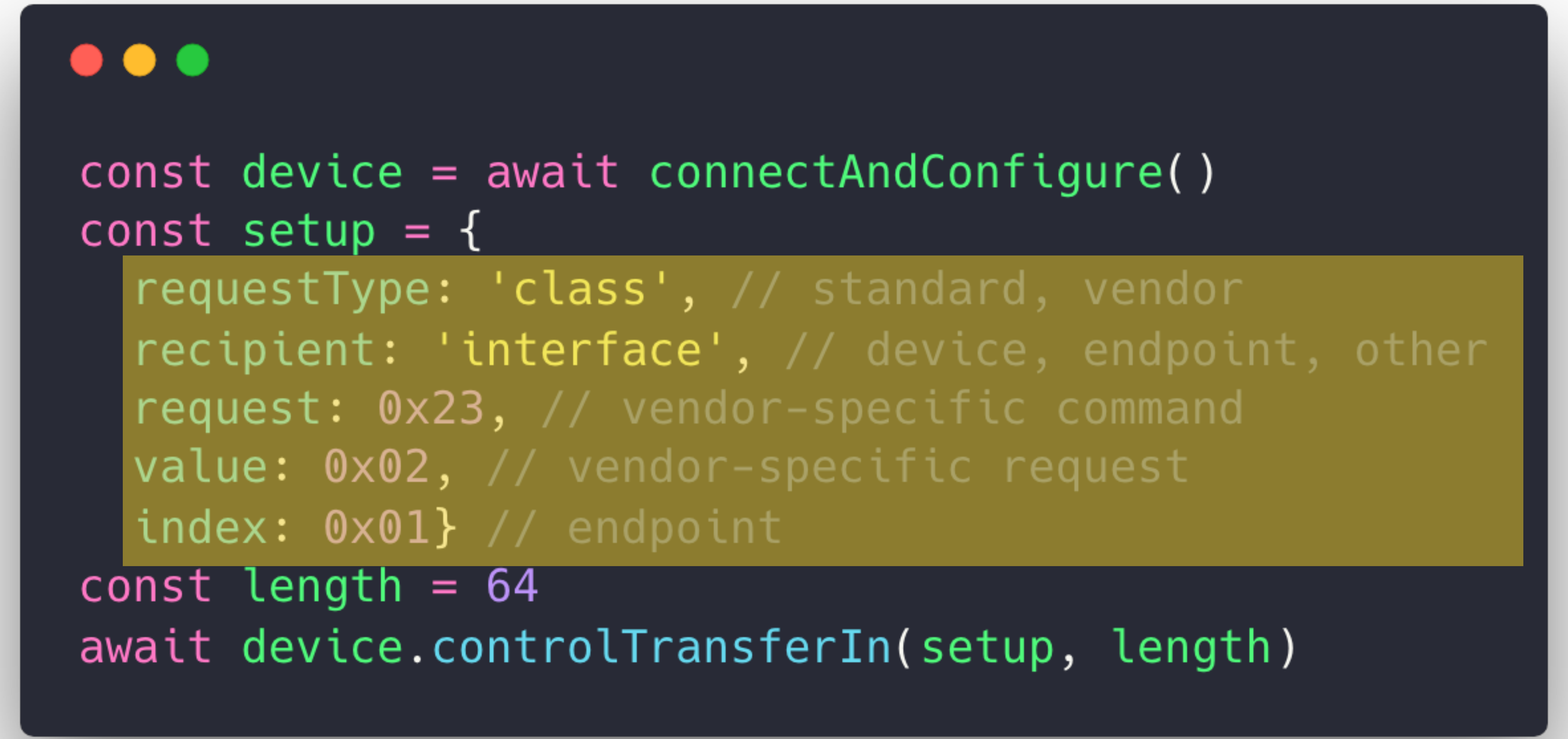
ISOCHRONOUS

OUT



```
const device = await connectAndConfigure()
const setup = {
  requestType: 'class', // standard, vendor
  recipient: 'interface', // device, endpoint, other
  request: 0x22, // vendor-specific command
  value: 0x01, // vendor-specific request
  index: 0x02} // endpoint
const data = 'Text'
await device.controlTransferOut(setup, data)
```

IN

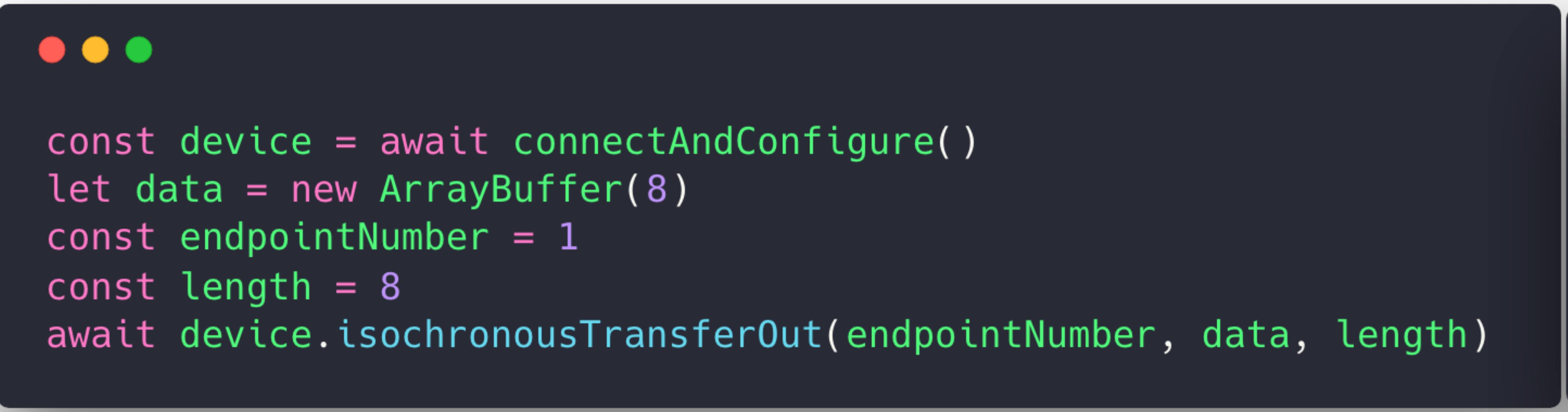


```
const device = await connectAndConfigure()
const setup = {
  requestType: 'class', // standard, vendor
  recipient: 'interface', // device, endpoint, other
  request: 0x23, // vendor-specific command
  value: 0x02, // vendor-specific request
  index: 0x01} // endpoint
const length = 64
await device.controlTransferIn(setup, length)
```

* CONTROL TRANSFERS ARE ESPECIALLY NICE FOR SMALL CONFIGURATION COMMANDS AS THEY GET BUS PRIORITY AND HAVE A WELL DEFINED STRUCTURE *

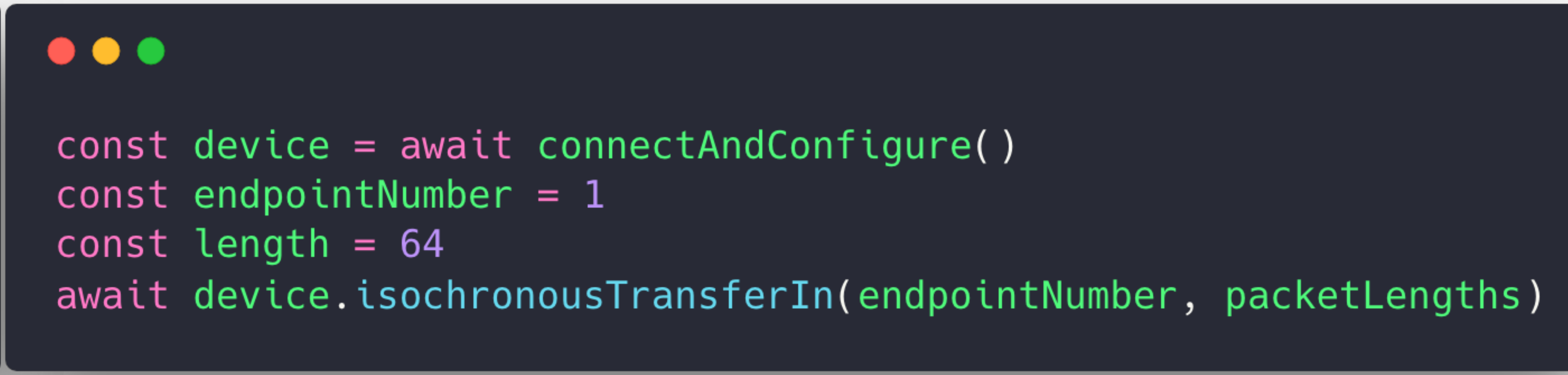
INTERRUPT | CONTROL | ISOCHRONOUS TRANSFER

OUT



```
const device = await connectAndConfigure()
let data = new ArrayBuffer(8)
const endpointNumber = 1
const length = 8
await device.isochronousTransferOut(endpointNumber, data, length)
```

IN



```
const device = await connectAndConfigure()
const endpointNumber = 1
const length = 64
await device.isochronousTransferIn(endpointNumber, packetLengths)
```

* ISOCHRONOUS TRANSFERS ARE RARELY USED, MOSTLY FOR AUDIO AND VIDEO DEVICES THAT SENT A PERSISTENT STREAM OF DATA *

Now Pray To The Demo Gods 🤞

Demos

- CREATE AN INTERACTIVE ARDUINO PLAYGROUND

J5
News
API
Examples
Articles
Platform Support
Fork me on GitHub

Johnny-Five

The JavaScript
BlendMicro
Robotics & IoT Platform

Johnny-Five is the JavaScript Robotics & IoT Platform. Released by Bocoup in 2012, Johnny-Five is maintained by a community of passionate software developers and hardware engineers. Over 75 developers have made contributions towards building a robust, extensible and composable ecosystem.

★ Star
Fork
Follow @nodebots
Tweet

The Johnny-Five Inventor's Kit:

Board

Board - Basic Initialization

Board - Specify port

Board - Cleanup in 'exit' event

Board - Multiple in one program

REPL

Board - Specify Sampling Interval

Pin

Custom Data Properties

LED

LED

LED - PCA9685

LED - Tessel Servo Module

LED - Blink

LED - Pulse

LED - Pulse with animation

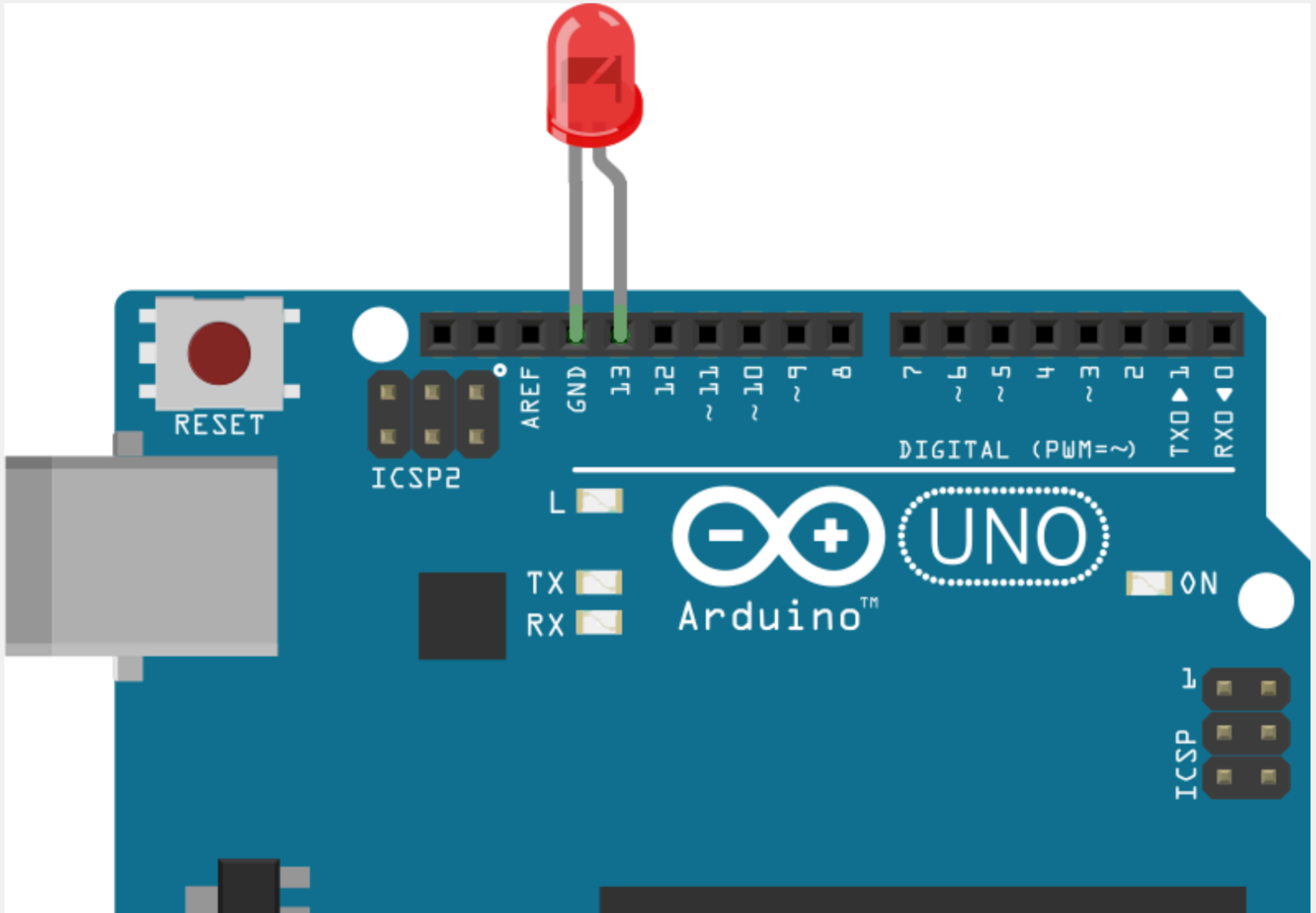
LED - Fade

LED - Fade callback

LED - Fade with animation

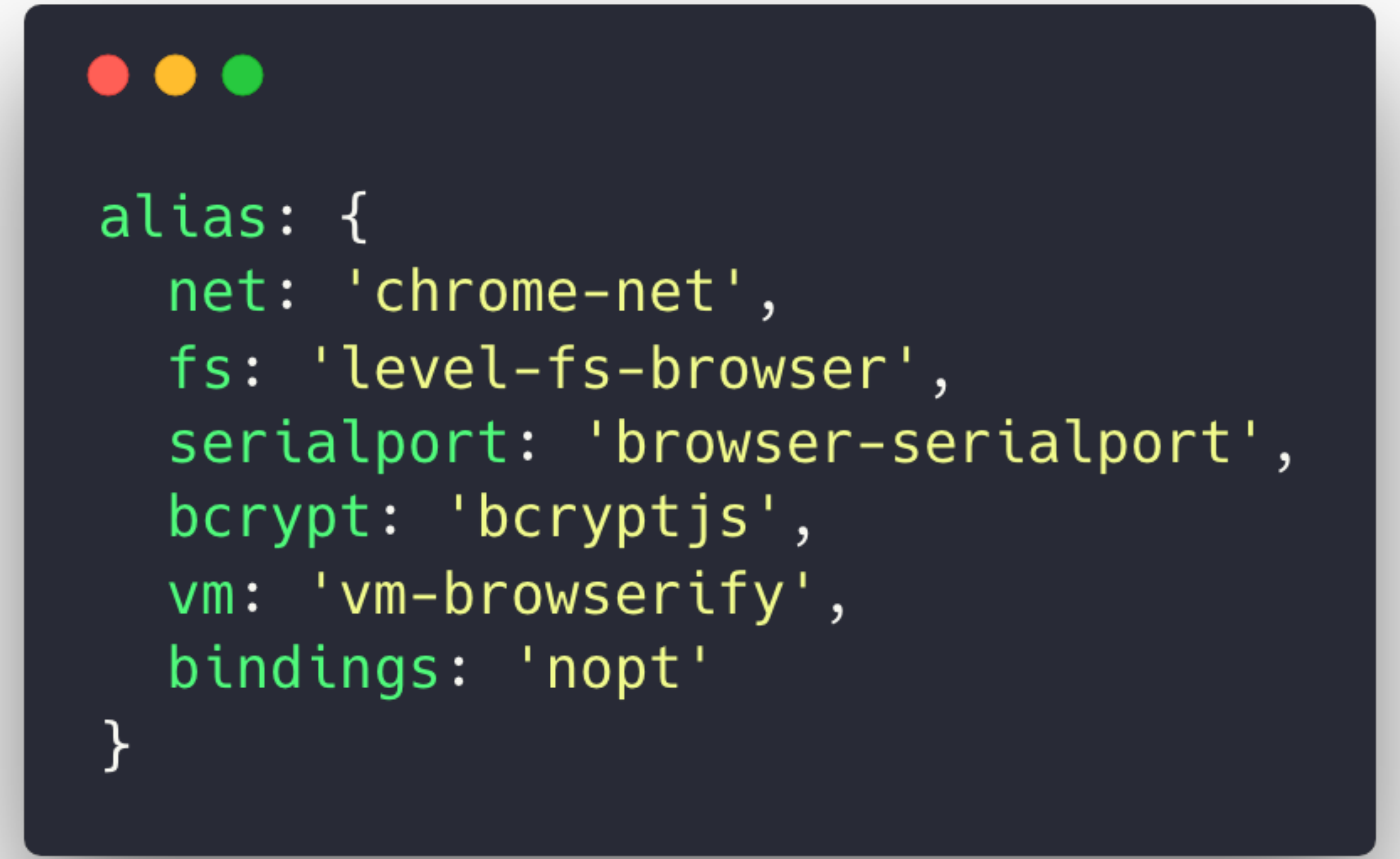
Hello World!

```
var five = require("johnny-five");
var board = new five.Board();

board.on("ready", function() {

  // Create a standard `led` component instance
  var led = new five.Led(13);

  // "blink" the led in 500ms
  // on-off phase periods
  led.blink(500);
});
```



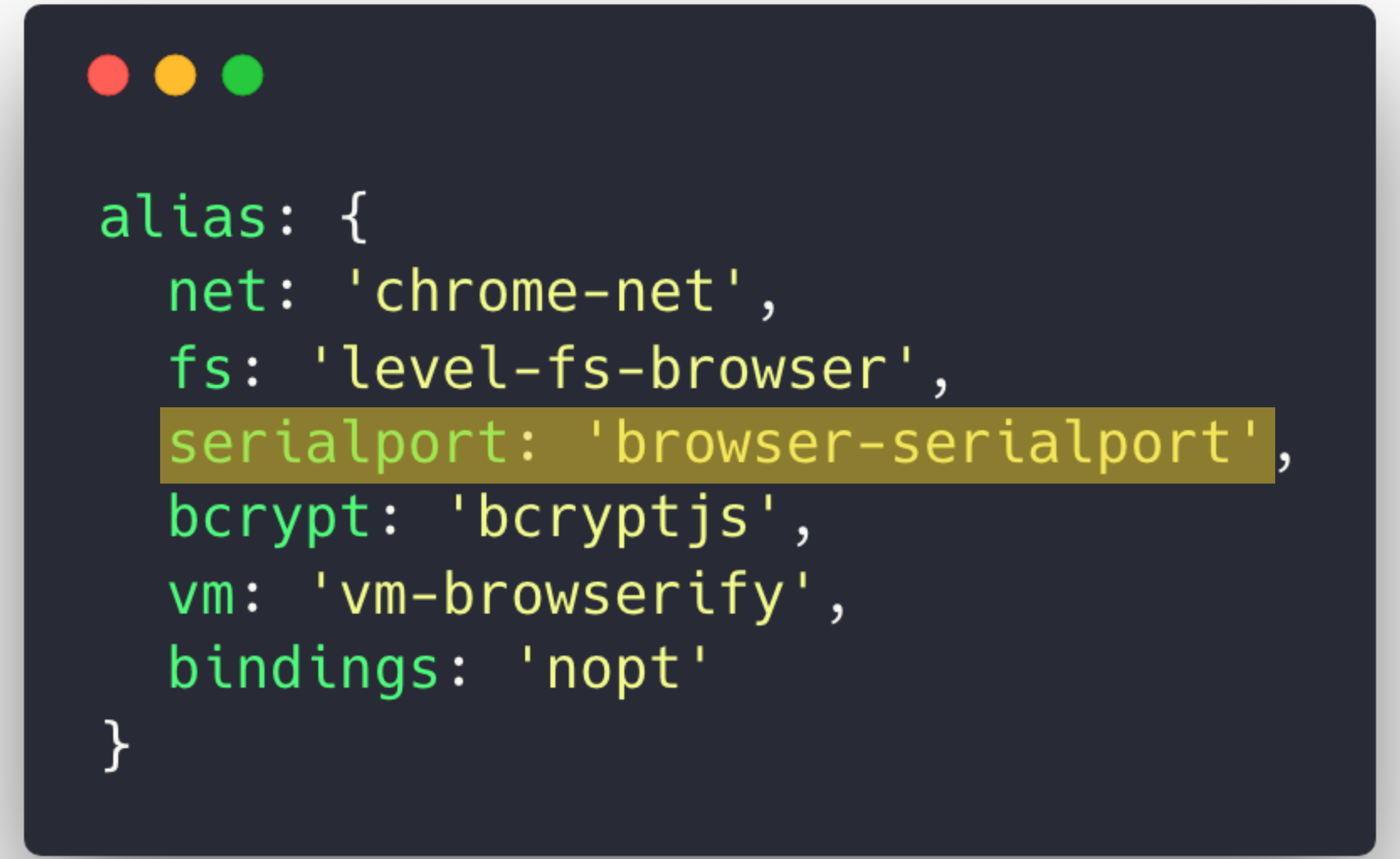
Demo

Node
-> Browser



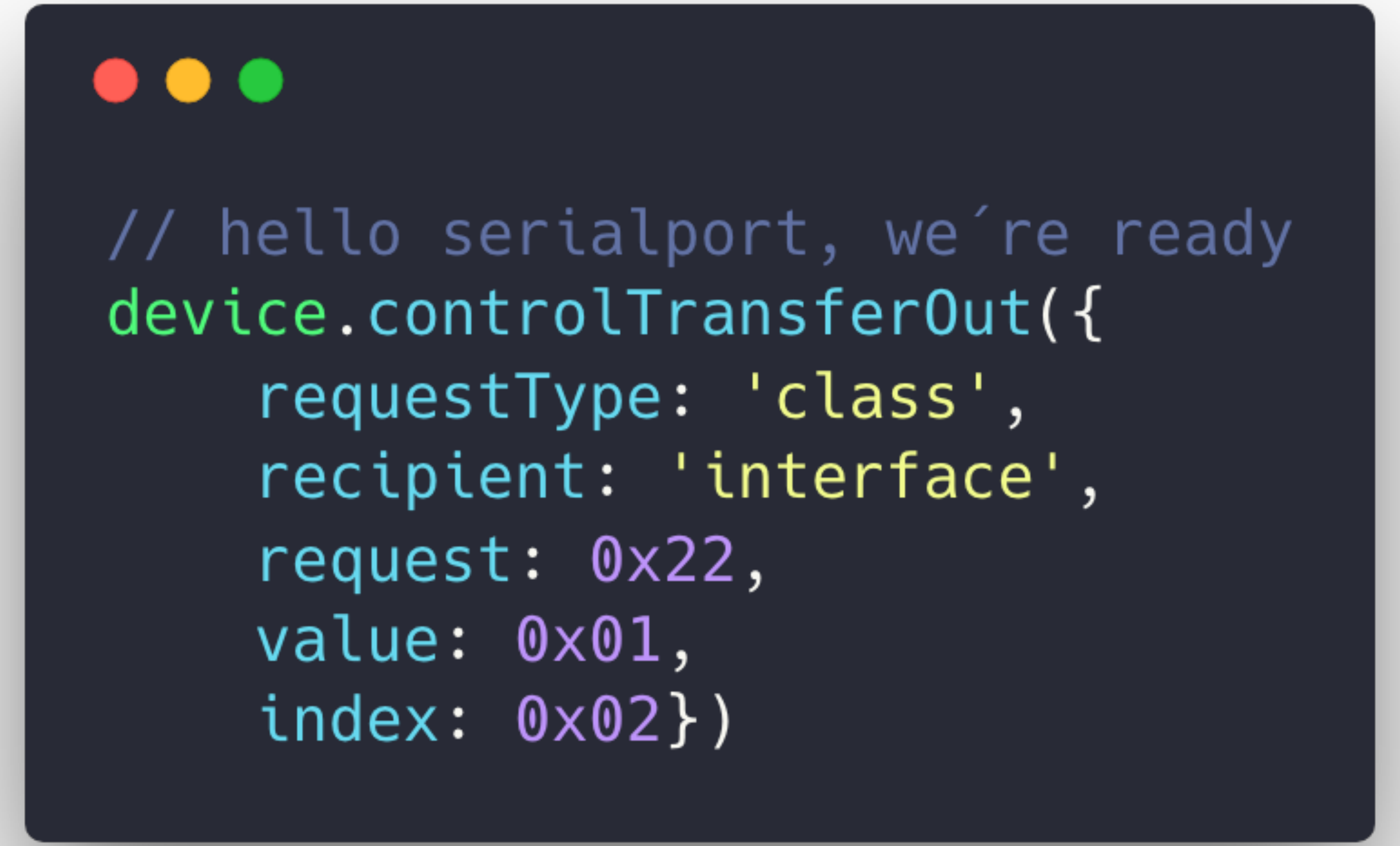
```
alias: {
  net: 'chrome-net',
  fs: 'level-fs-browser',
  serialport: 'browser-serialport',
  bcrypt: 'bcryptjs',
  vm: 'vm-browserify',
  bindings: 'nopt'
}
```

Node
-> Browser



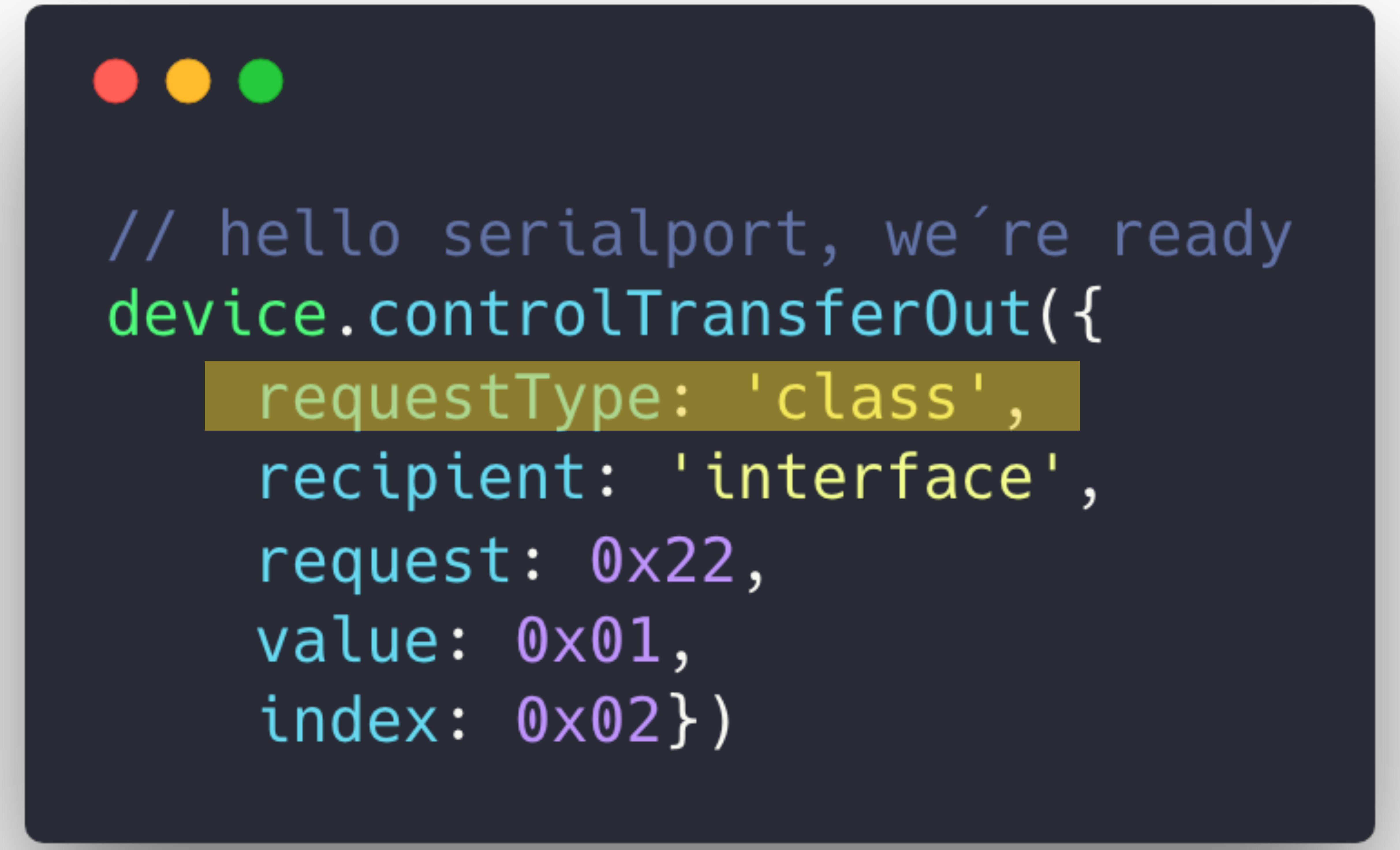
```
alias: {
  net: 'chrome-net',
  fs: 'level-fs-browser',
  serialport: 'browser-serialport',
  bcrypt: 'bcryptjs',
  vm: 'vm-browserify',
  bindings: 'nopt'
}
```

Serialport Communication



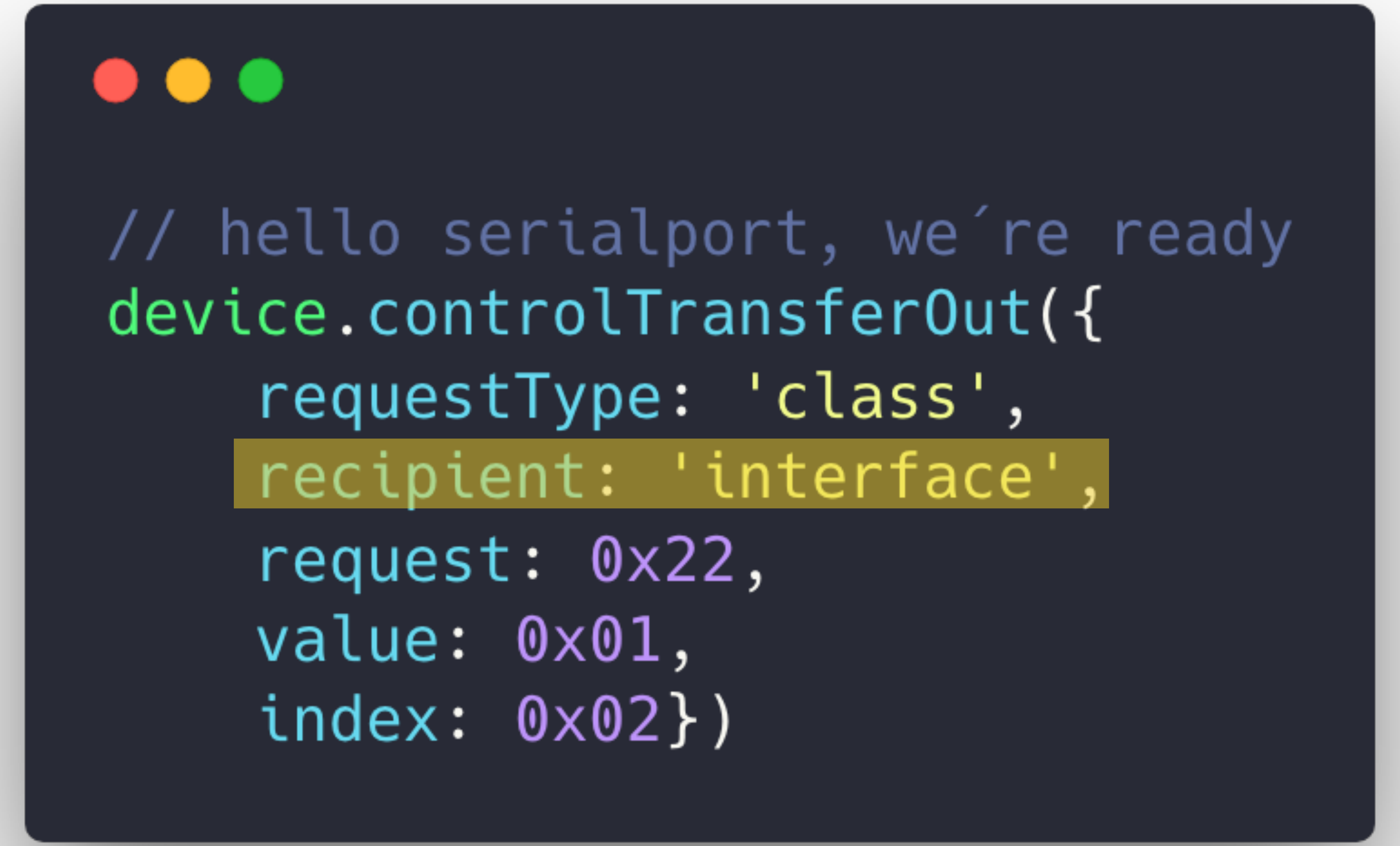
```
// hello serialport, we´re ready
device.controlTransferOut({
    requestType: 'class',
    recipient: 'interface',
    request: 0x22,
    value: 0x01,
    index: 0x02})
```

Which?



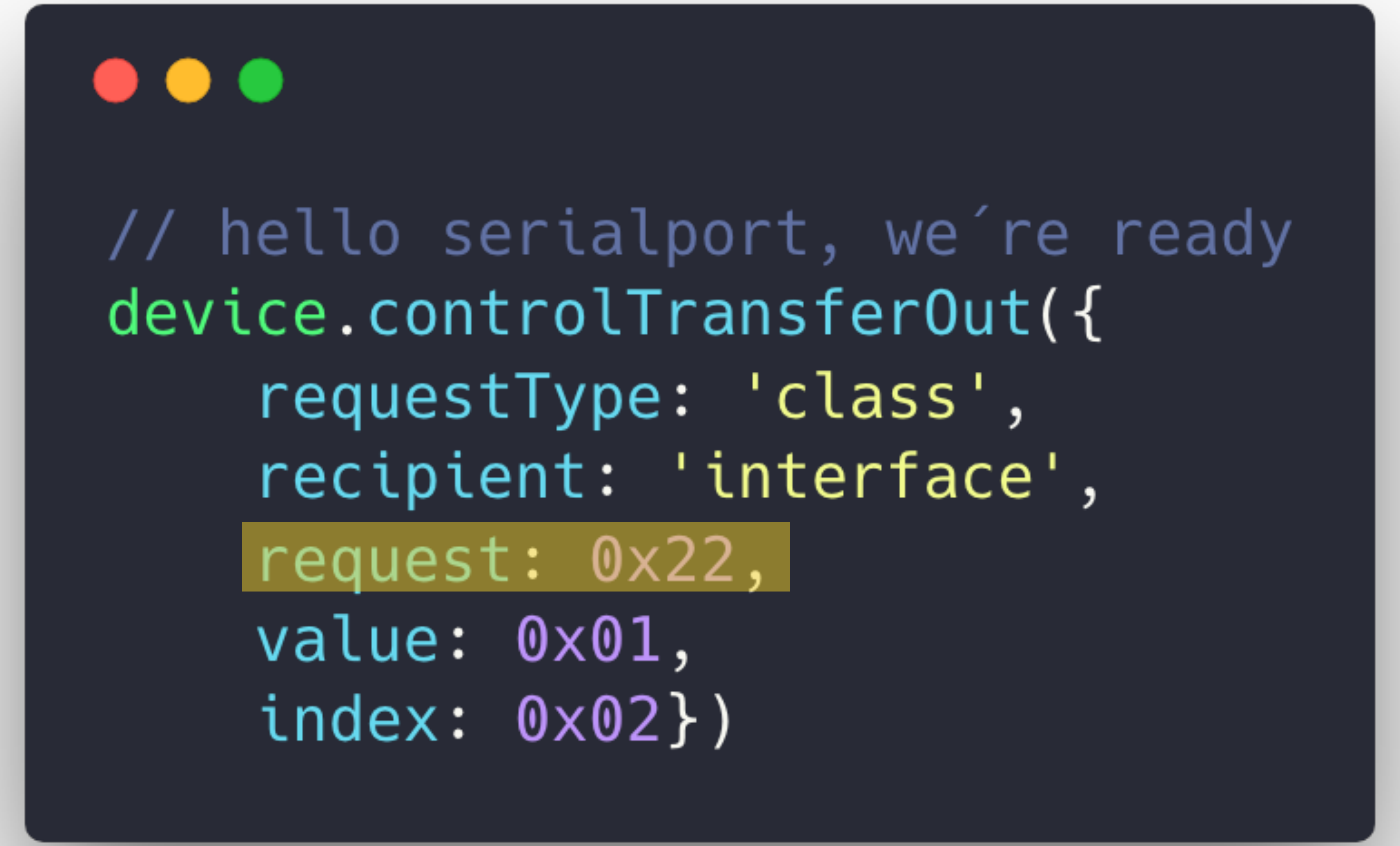
```
// hello serialport, we´re ready
device.controlTransferOut({
    requestType: 'class',
    recipient: 'interface',
    request: 0x22,
    value: 0x01,
    index: 0x02})
```

Who?



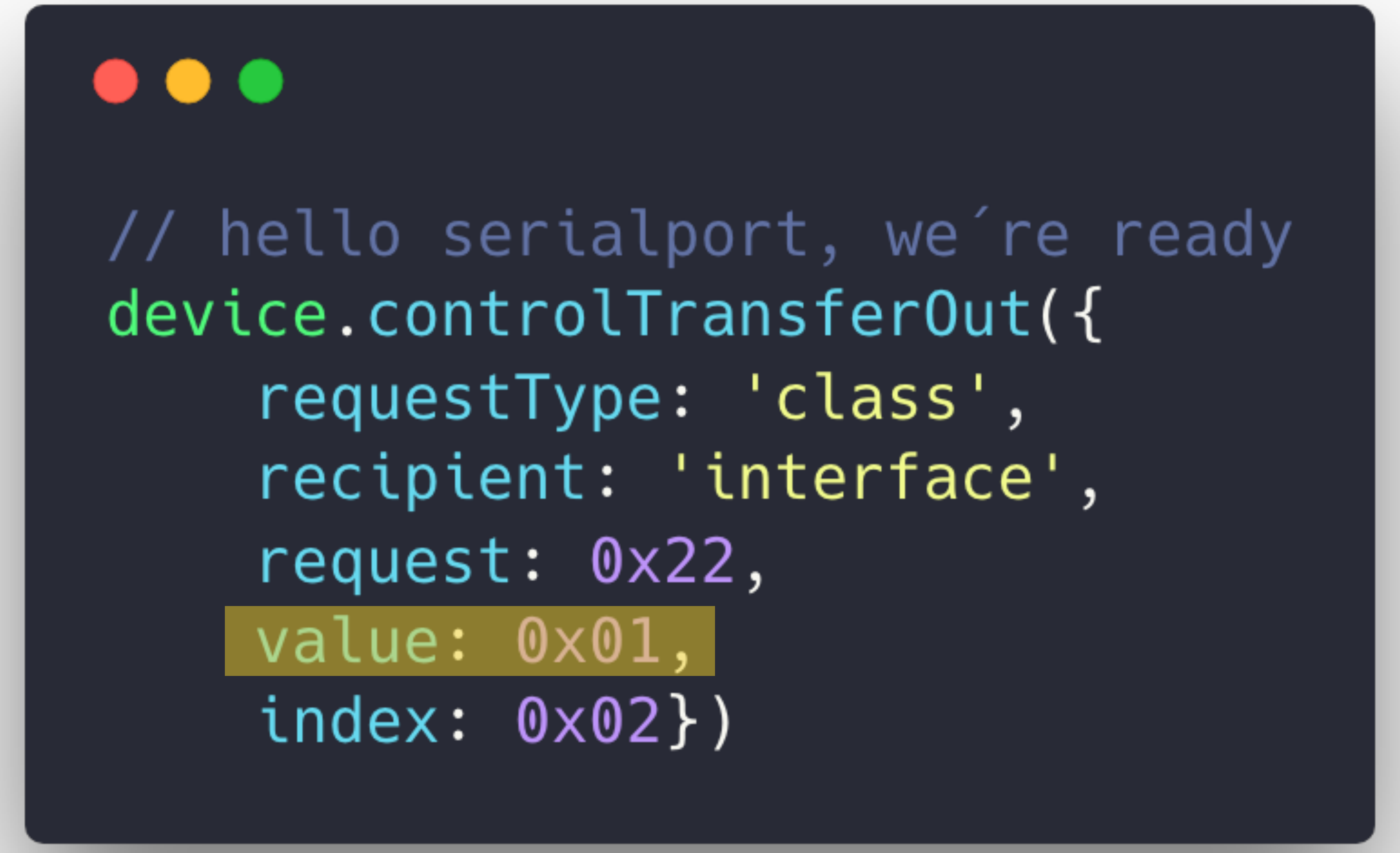
```
// hello serialport, we´re ready
device.controlTransferOut({
    requestType: 'class',
    recipient: 'interface',
    request: 0x22,
    value: 0x01,
    index: 0x02})
```

What?



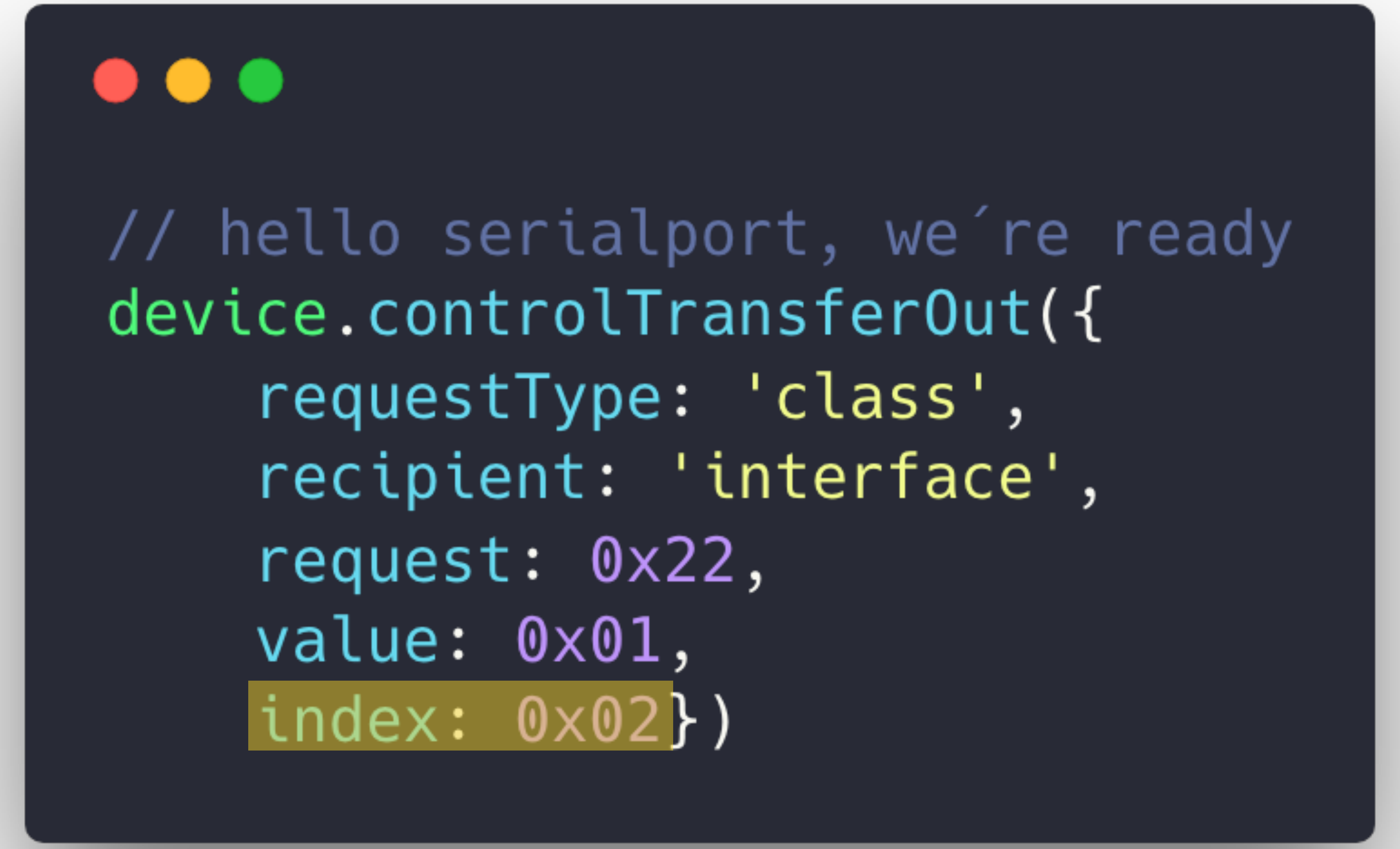
```
// hello serialport, we´re ready
device.controlTransferOut({
    requestType: 'class',
    recipient: 'interface',
    request: 0x22,
    value: 0x01,
    index: 0x02})
```

What? Pt.2



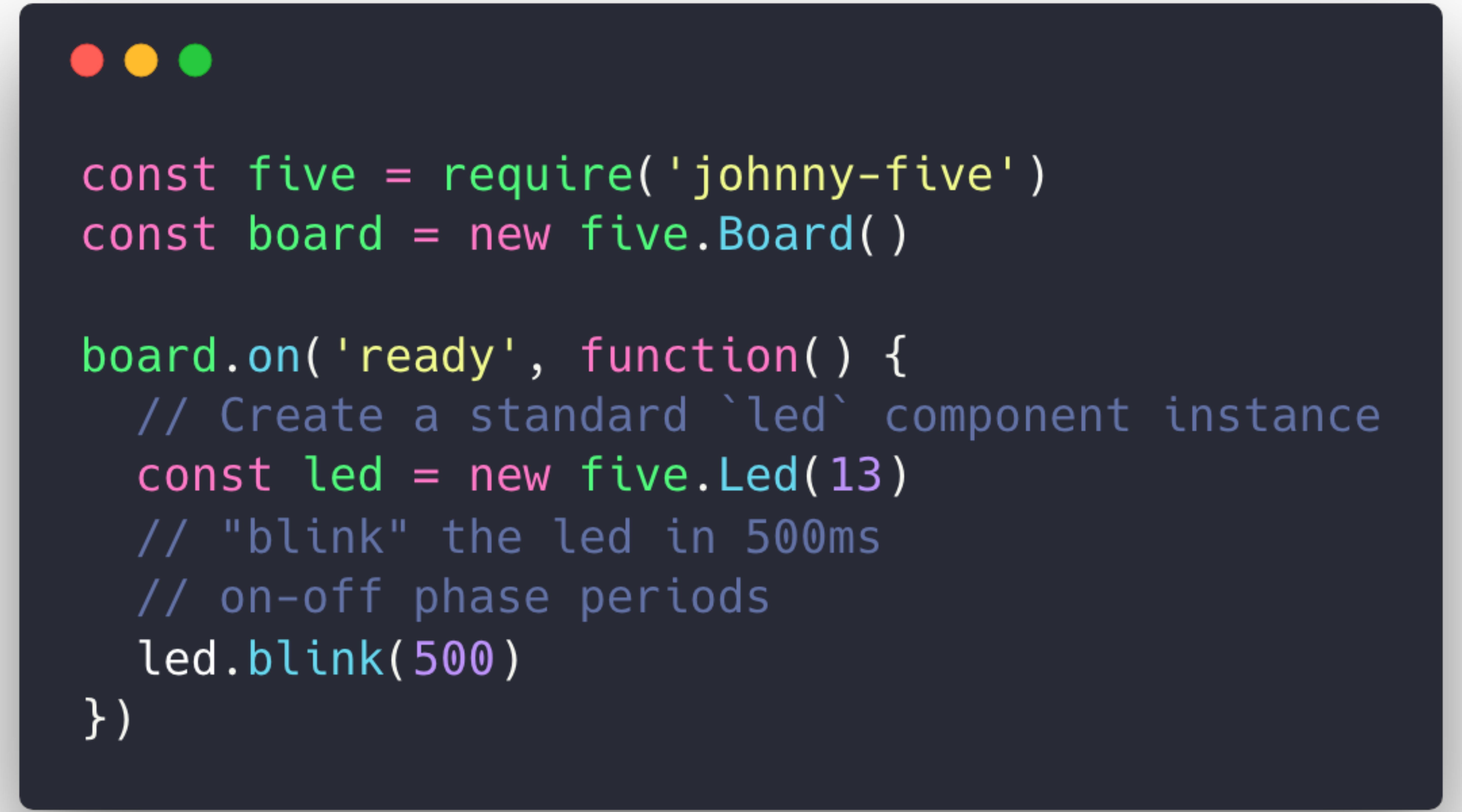
```
// hello serialport, we´re ready
device.controlTransferOut({
    requestType: 'class',
    recipient: 'interface',
    request: 0x22,
    value: 0x01,
    index: 0x02})
```

Whom Again?



```
// hello serialport, we´re ready
device.controlTransferOut({
    requestType: 'class',
    recipient: 'interface',
    request: 0x22,
    value: 0x01,
    index: 0x02})
```

Node
-> Browser



```
const five = require('johnny-five')
const board = new five.Board()

board.on('ready', function() {
  // Create a standard `led` component instance
  const led = new five.Led(13)
  // "blink" the led in 500ms
  // on-off phase periods
  led.blink(500)
})
```

Demos

- CREATE AN INTERACTIVE ARDUINO PLAYGROUND
- CONNECT TO AN ANDROID DEVICE & DOWNLOAD SOME DATA

Demo

Meet Android Studio
Workflow basics
Manage your project
Write your app
Build and run your app
Configure your build
Debug your app
Test your app
Profile your app
Publish your app
Command line tools
Übersicht
aapt2
adb
apkanalyzer
apksigner
avdmanager
bmgr
bundletool
d8
dmtracedump
dumpsys
etc1tool
jobb
logcat

Android Debug Bridge (adb)

☆☆☆☆☆

Android Debug Bridge (adb) is a versatile command-line tool that lets you communicate with a device. The adb command facilitates a variety of device actions, such as installing and debugging apps, and it provides access to a Unix shell that you can use to run a variety of commands on a device. It is a client-server program that includes three components:

- **A client**, which sends commands. The client runs on your development machine. You can invoke a client from a command-line terminal by issuing an adb command.
- **A daemon (adbd)**, which runs commands on a device. The daemon runs as a background process on each device.
- **A server**, which manages communication between the client and the daemon. The server runs as a background process on your development machine.

`adb` is included in the Android SDK Platform-Tools package. You can download this package with the SDK Manager, which installs it at `android_sdk/platform-tools/`. Or if you want the standalone Android SDK Platform-Tools package, you can download it here.

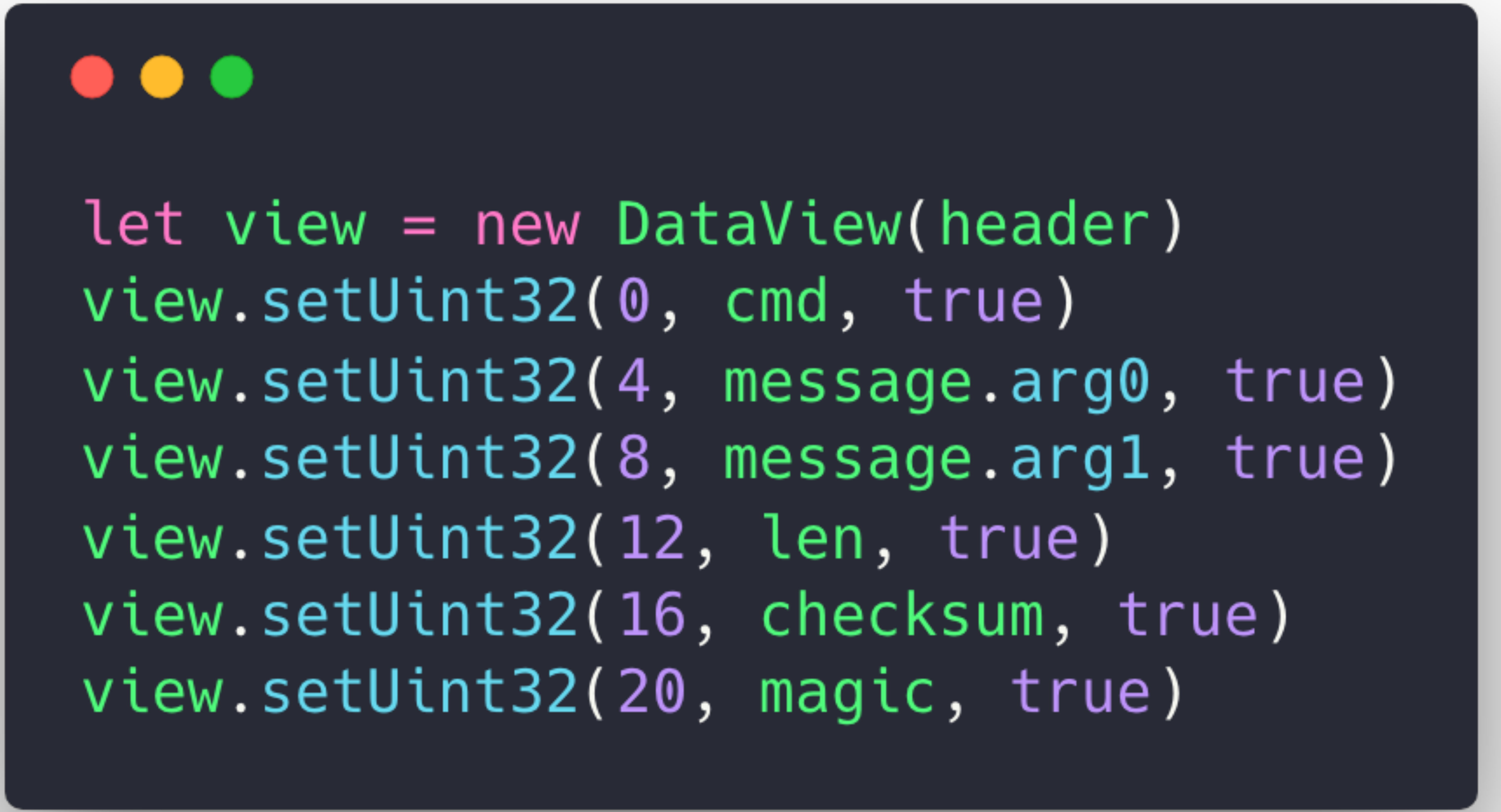
For information on connecting a device for use over ADB, including how to use the Connection Assistant to troubleshoot common problems, see Run apps on a hardware device.

How adb works

When you start an adb client, the client first checks whether there is an adb server process already running. If there isn't, it starts the server process. When the server starts, it binds to local TCP port 5037 and listens for commands sent from

Inhalt
How adb works
Enable adb debugging on your device
Connect to a device over Wi-Fi
Query for devices
Emulator not listed
Send commands to a specific device
Install an app
Set up port forwarding
Copy files to/from a device
Stop the adb server
adb commands reference
Issue shell commands
Call activity manager (am)
Call package manager (pm)
Call device policy manager (dpm)
Take a screenshot
Record a video
Read ART profiles for apps
Other shell commands

Send Adb Commands



```
let view = new DataView(header)
view.setUint32(0, cmd, true)
view.setUint32(4, message.arg0, true)
view.setUint32(8, message.arg1, true)
view.setUint32(12, len, true)
view.setUint32(16, checksum, true)
view.setUint32(20, magic, true)
```

Search or jump to...
Pull requests
Issues
Trending
Explore
webadb / webadb.js
Watch 4
Star 43
Fork 16
Code
Issues 0
Pull requests 0
Wiki
More
23 commits
1 branch
0 releases
3 contributors
MIT
Branch: master
Create new file
Find file
Clone or download

ADB host implementation based on WebUSB webadb.github.io

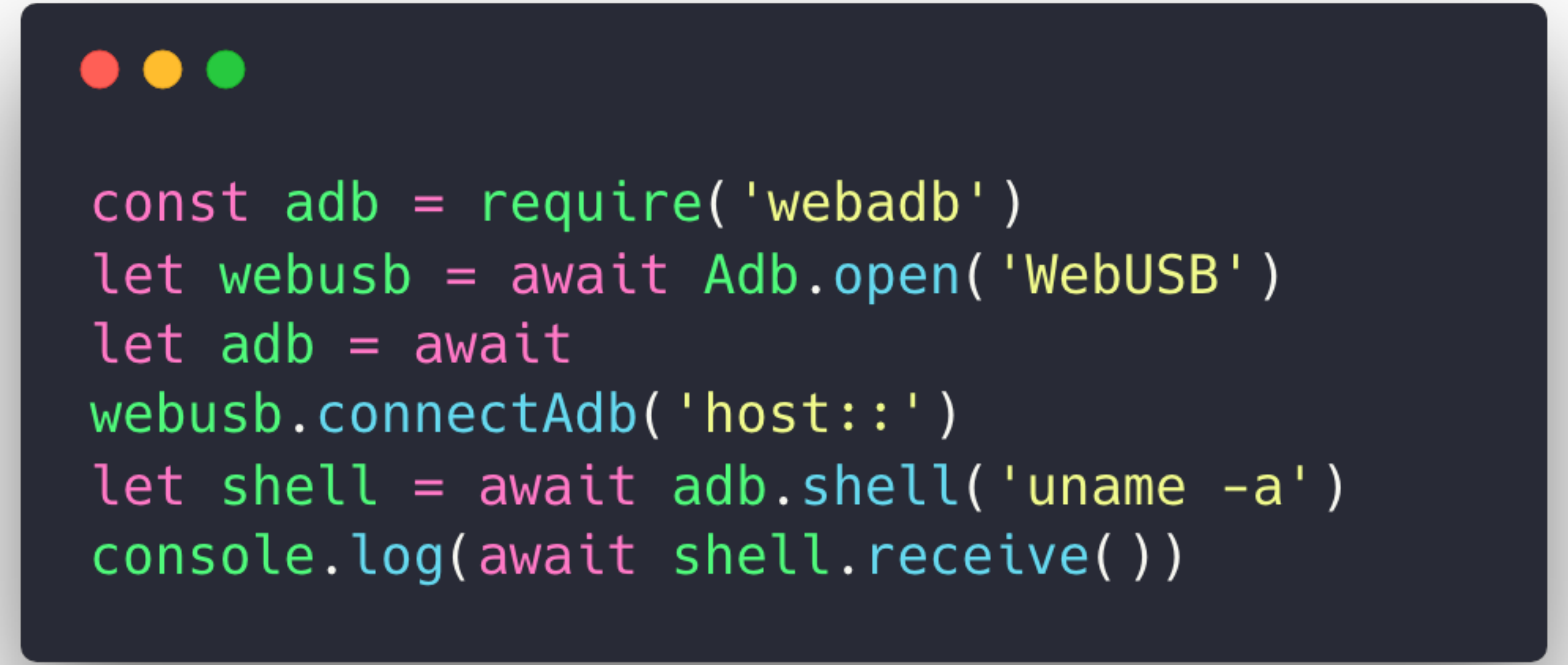
cavokz Merge pull request #6 from klausw/master ··· Latest commit 0c03cf1 on 17 Jun

.gitignore	Initial commit	a year ago
LICENSE	Let There Be MIT	9 months ago
README.md	(GH) add README.md	a year ago
tcpip.html	Add GUI and tcpip command, add no-checksum protocol	5 months ago
test.html	Parsing out the full file data from the pull	10 months ago
webadb.js	Add GUI and tcpip command, add no-checksum protocol	5 months ago

One-stop .js for Android Debug Bridge (adb) over WebUSB

```
let webusb = await Adb.open("WebUSB");
let adb = await webusb.connectAdb("host::");
let shell = await adb.shell("uname -a");
console.log(await shell.receive());
```

Easy Adb



```
const adb = require('webadb')
let webusb = await Adb.open('WebUSB')
let adb = await
webusb.connectAdb('host::')
let shell = await adb.shell('uname -a')
console.log(await shell.receive())
```

Security ⚠️

Security

01. Only Via User Gesture

🖱️

02.User Has To Grant Permission

🖖

03. No Cam/Microphone/Storage

📹

04. Kill Switch

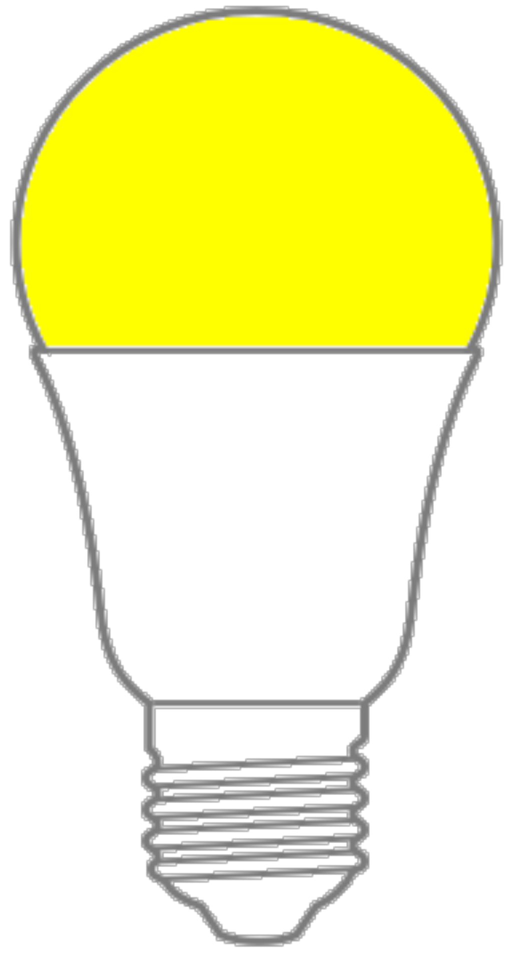
🆘

Demos

- CREATE AN INTERACTIVE ARDUINO PLAYGROUND
- CONNECT TO AN ANDROID DEVICE & DOWNLOAD SOME DATA
- CONTROL SMART HOME DEVICES USING THE BROWSER AS THE RUNTIME

enocean®

Bluetooth™

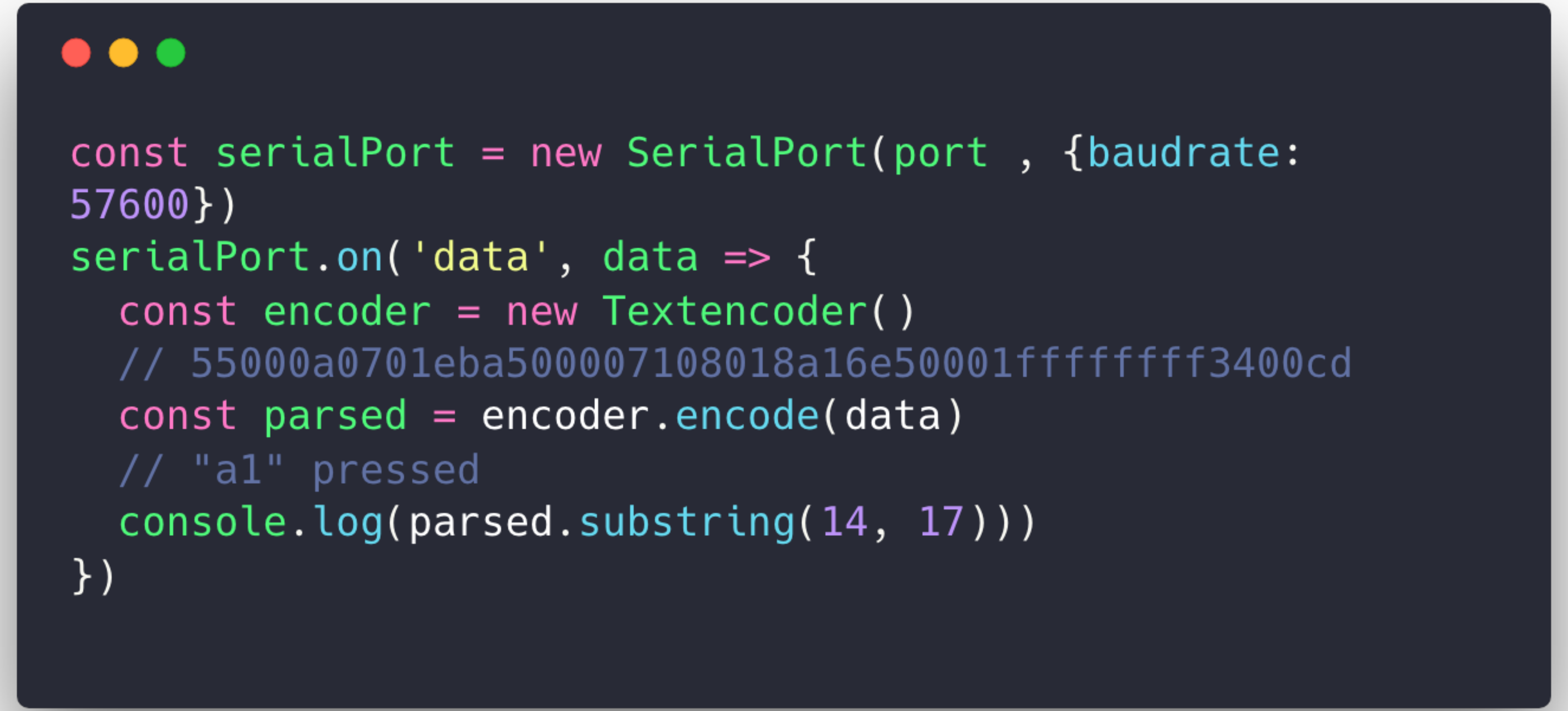


enocean®

Bluetooth™

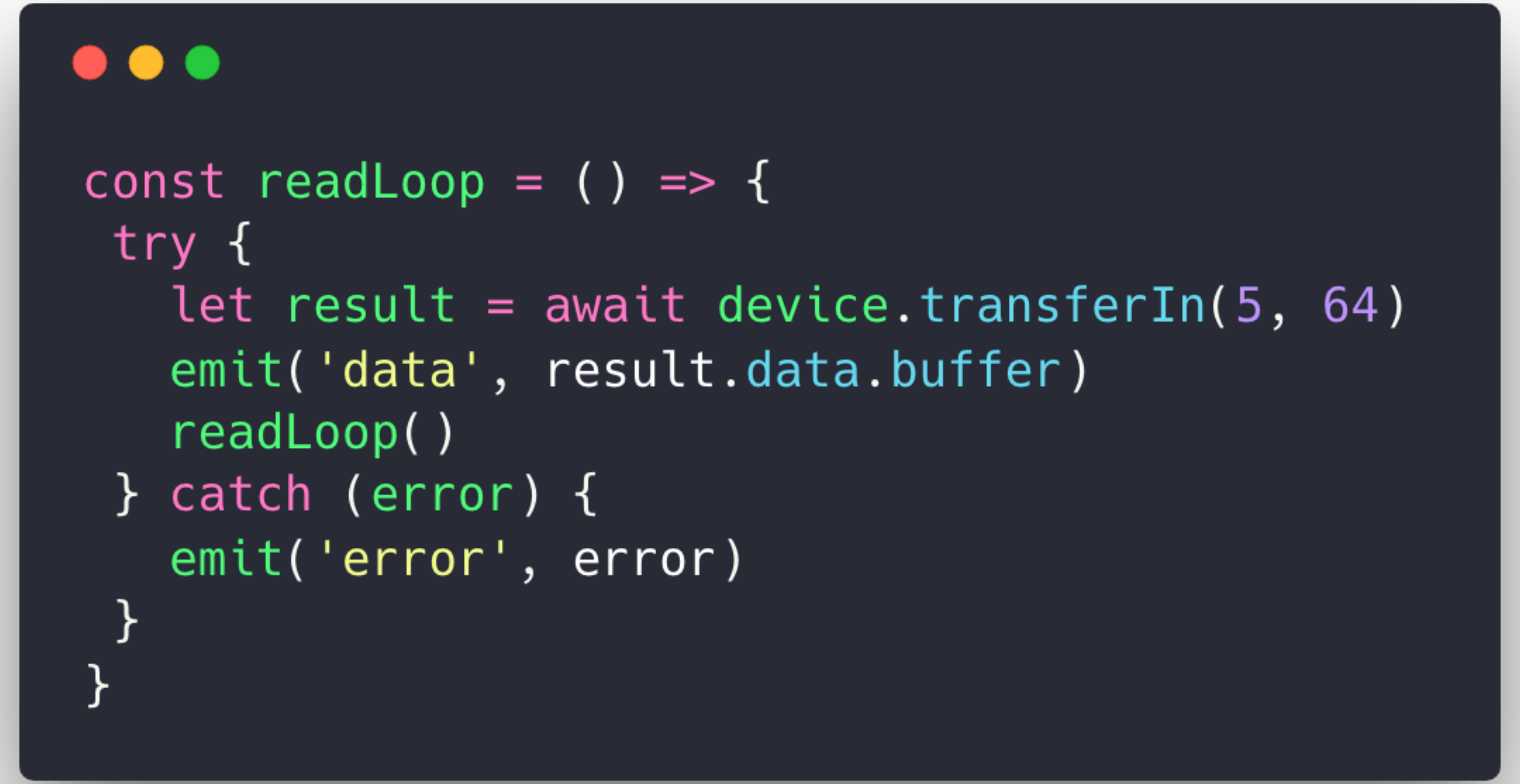
Demo

Enocean Code



```
const serialPort = new SerialPort(port , {baudrate:
57600})
serialPort.on('data', data => {
  const encoder = new Textencoder()
  // 55000a0701eba500007108018a16e50001ffffffff3400cd
  const parsed = encoder.encode(data)
  // "a1" pressed
  console.log(parsed.substring(14, 17)))
})
```

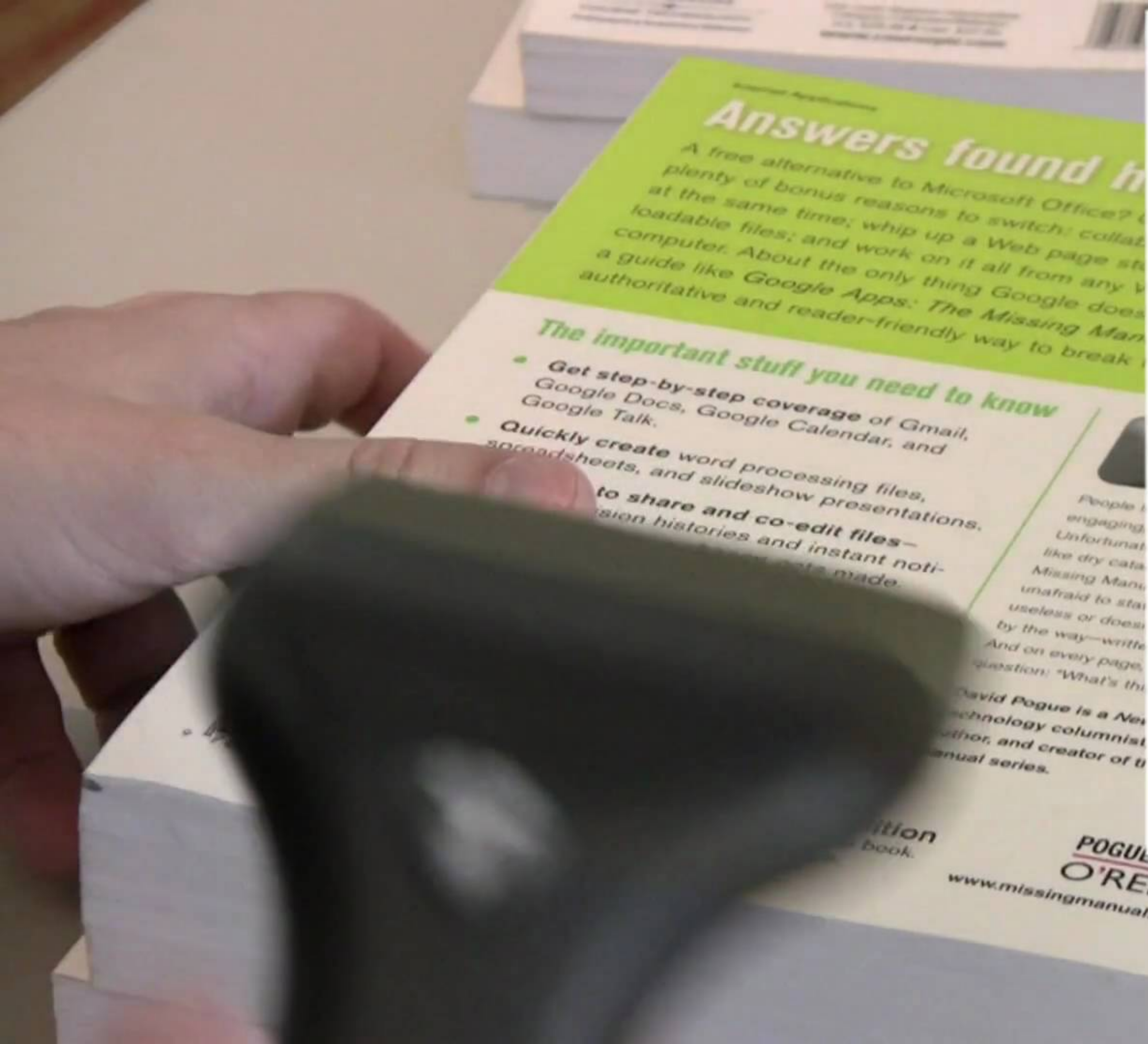
Data Polling

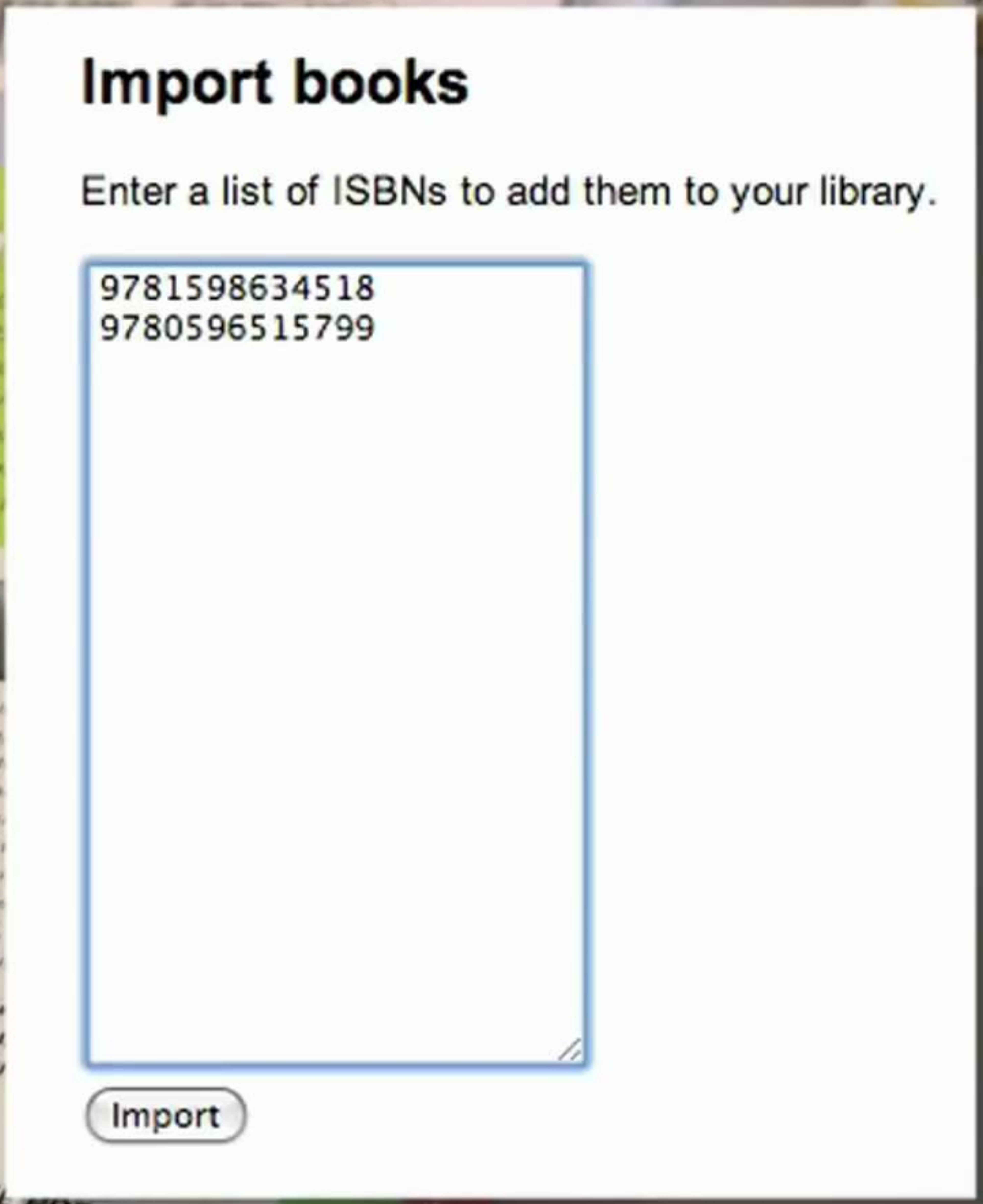


```
const readLoop = () => {
 try {
   let result = await device.transferIn(5, 64)
   emit('data', result.data.buffer)
   readLoop()
 } catch (error) {
   emit('error', error)
 }
}
```

Demos

- CREATE AN INTERACTIVE ARDUINO PLAYGROUND
- CONNECT TO AN ANDROID DEVICE & DOWNLOAD SOME DATA
- CONTROL SMART HOME DEVICES USING THE BROWSER AS THE RUNTIME
- CREATE A PUBLIC LIBRARY USING:
  RFID CARDS, A BARCODE SCANNER & A RECEIPT PRINTER





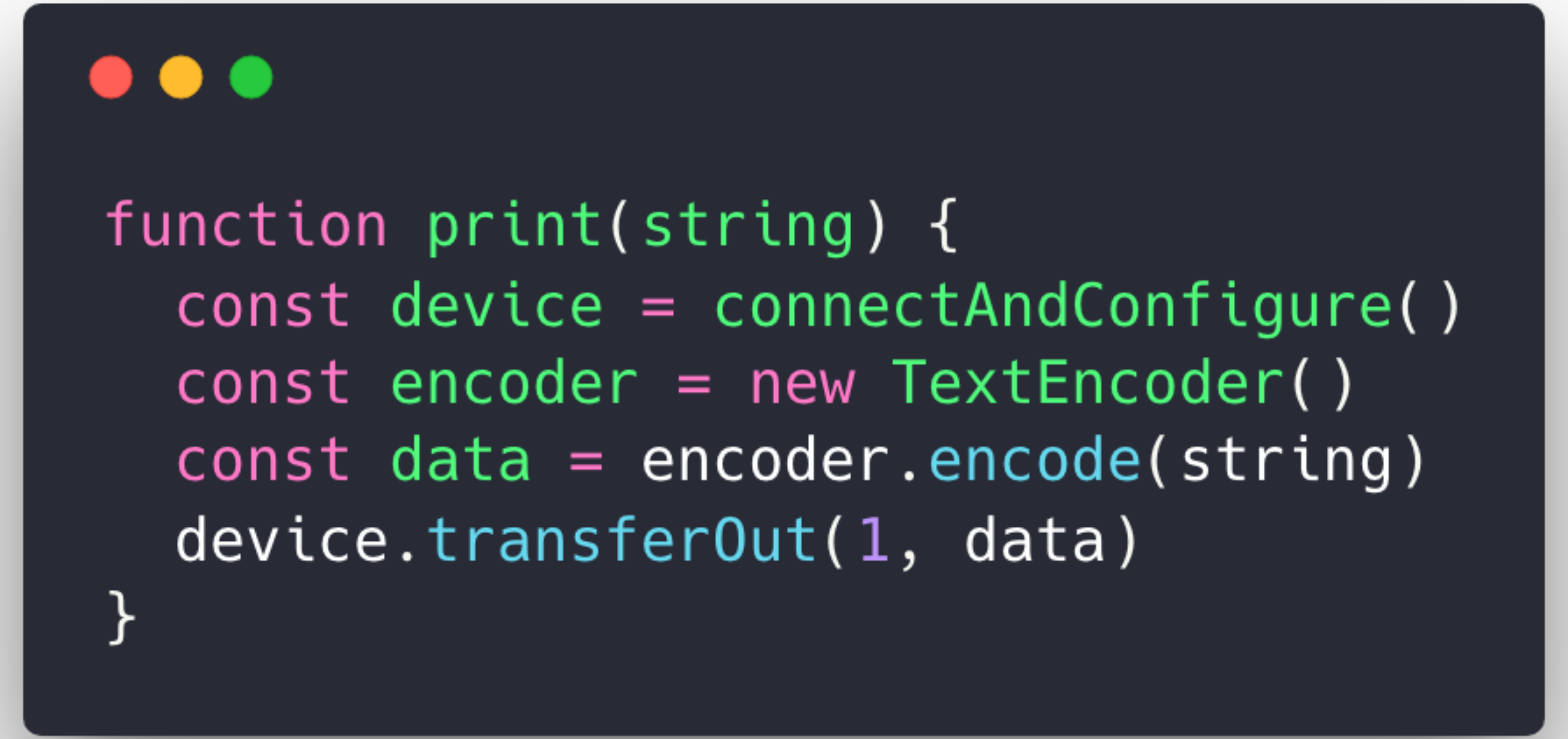
Import books

Enter a list of ISBNs to add them to your library.

```
9781598634518
9780596515799
```

Demo

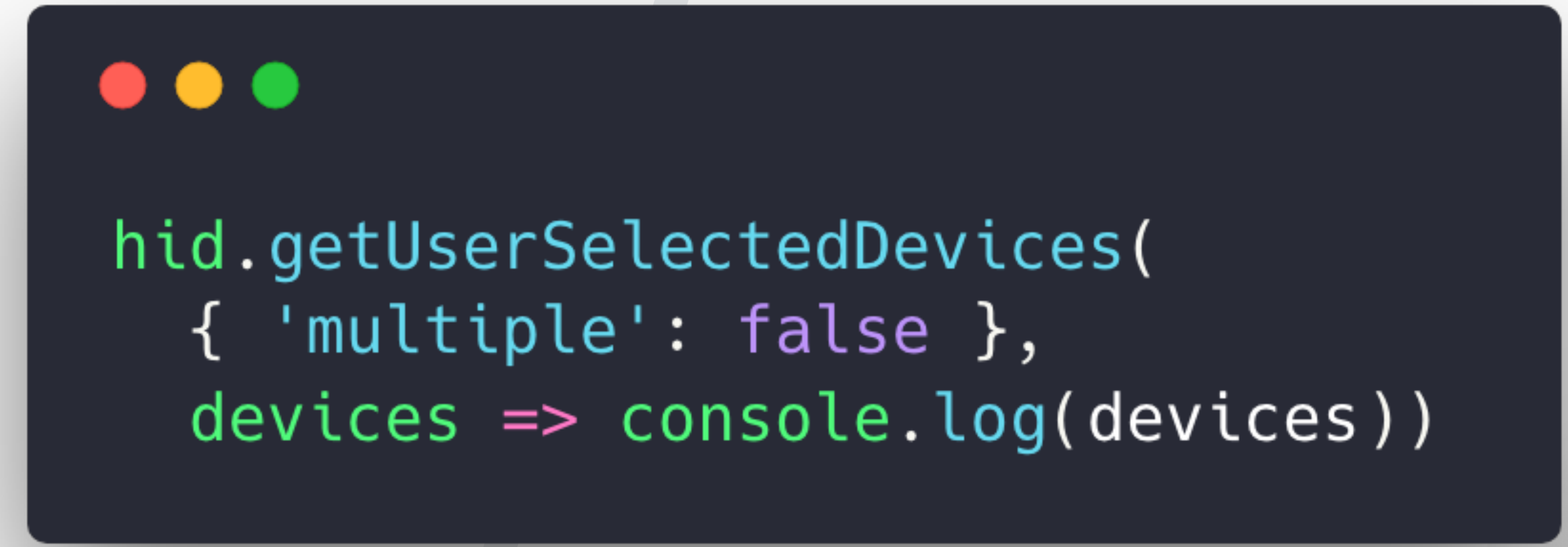
Printer Code



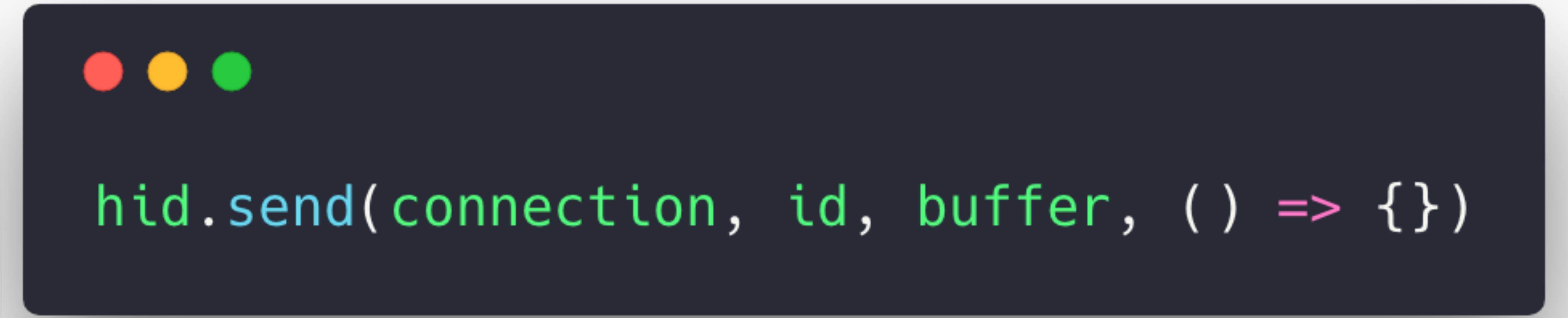
```
function print(string) {
  const device = connectAndConfigure()
  const encoder = new TextEncoder()
  const data = encoder.encode(string)
  device.transferOut(1, data)
}
```

Hid ⌨️

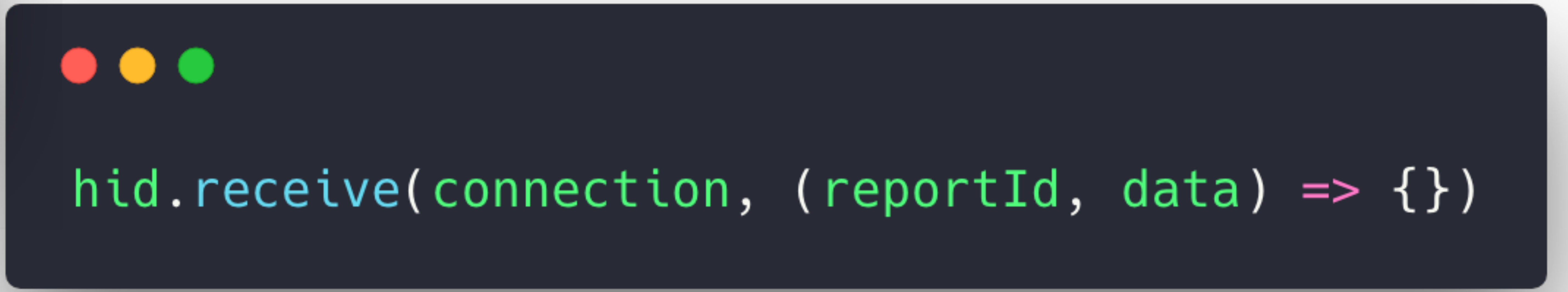
Hid Code



```
hid.getUserSelectedDevices(
  { 'multiple': false },
  devices => console.log(devices))
```

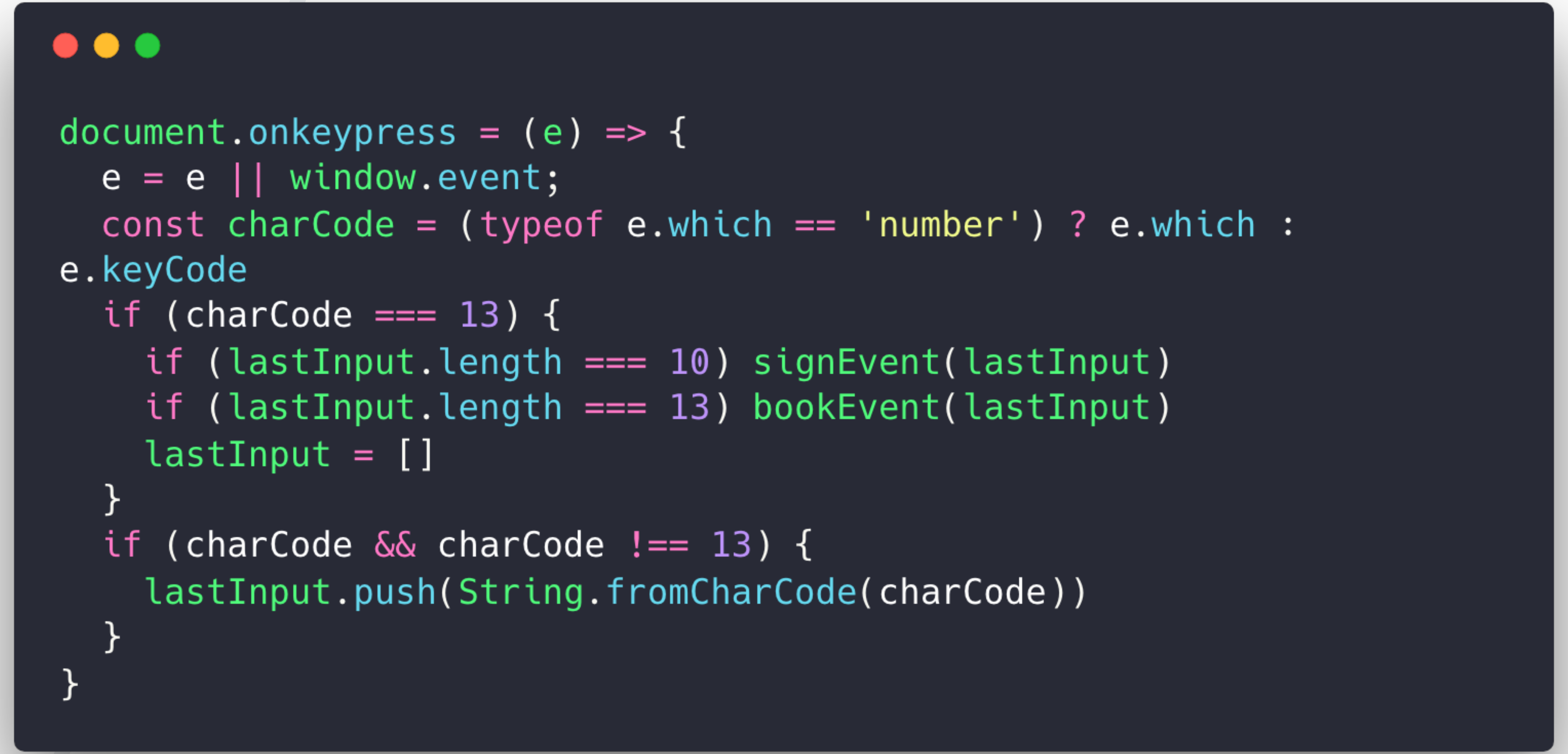


```
hid.send(connection, id, buffer, () => {})
```



```
hid.receive(connection, (reportId, data) => {})
```

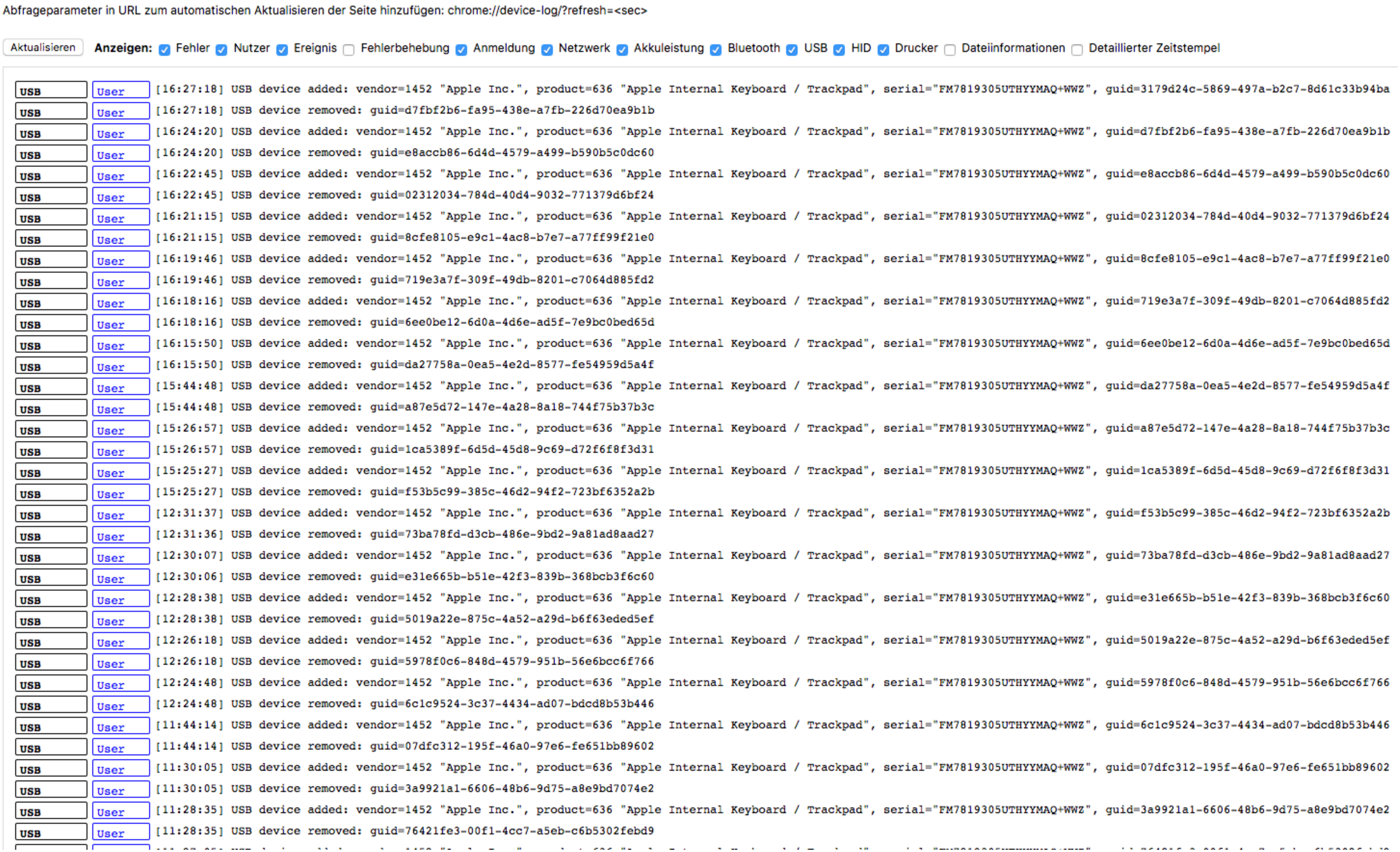
Hid Code Reality



```
document.onkeypress = (e) => {
  e = e || window.event;
  const charCode = (typeof e.which == 'number') ? e.which :
e.keyCode
  if (charCode === 13) {
    if (lastInput.length === 10) signEvent(lastInput)
    if (lastInput.length === 13) bookEvent(lastInput)
    lastInput = []
  }
  if (charCode && charCode !== 13) {
    lastInput.push(String.fromCharCode(charCode))
  }
}
```

That Worked Well 😅

Chrome://Device-Log/



Abfrageparameter in URL zum automatischen Aktualisieren der Seite hinzufügen: chrome://device-log/?refresh=<sec>

Aktualisieren **Anzeigen:** ☑ Fehler ☑ Nutzer ☑ Ereignis ☐ Fehlerbehebung ☑ Anmeldung ☑ Netzwerk ☑ Akkuleistung ☑ Bluetooth ☑ USB ☑ HID ☑ Drucker ☐ Dateiinformationen ☐ Detaillierter Zeitstempel

```
USB  User  [16:27:18] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=3179d24c-5869-497a-b2c7-8d61c33b94ba
USB  User  [16:27:18] USB device removed: guid=d7fbf2b6-fa95-438e-a7fb-226d70ea9b1b
USB  User  [16:24:20] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=d7fbf2b6-fa95-438e-a7fb-226d70ea9b1b
USB  User  [16:24:20] USB device removed: guid=e8accb86-6d4d-4579-a499-b590b5c0dc60
USB  User  [16:22:45] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=e8accb86-6d4d-4579-a499-b590b5c0dc60
USB  User  [16:22:45] USB device removed: guid=02312034-784d-40d4-9032-771379d6bf24
USB  User  [16:21:15] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=02312034-784d-40d4-9032-771379d6bf24
USB  User  [16:21:15] USB device removed: guid=8cfe8105-e9c1-4ac8-b7e7-a77ff99f21e0
USB  User  [16:19:46] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=8cfe8105-e9c1-4ac8-b7e7-a77ff99f21e0
USB  User  [16:19:46] USB device removed: guid=719e3a7f-309f-49db-8201-c7064d885fd2
USB  User  [16:18:16] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=719e3a7f-309f-49db-8201-c7064d885fd2
USB  User  [16:18:16] USB device removed: guid=6ee0be12-6d0a-4d6e-ad5f-7e9bc0bed65d
USB  User  [16:15:50] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=6ee0be12-6d0a-4d6e-ad5f-7e9bc0bed65d
USB  User  [16:15:50] USB device removed: guid=da27758a-0ea5-4e2d-8577-fe54959d5a4f
USB  User  [15:44:48] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=da27758a-0ea5-4e2d-8577-fe54959d5a4f
USB  User  [15:44:48] USB device removed: guid=a87e5d72-147e-4a28-8a18-744f75b37b3c
USB  User  [15:26:57] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=a87e5d72-147e-4a28-8a18-744f75b37b3c
USB  User  [15:26:57] USB device removed: guid=1ca5389f-6d5d-45d8-9c69-d72f6f8f3d31
USB  User  [15:25:27] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=1ca5389f-6d5d-45d8-9c69-d72f6f8f3d31
USB  User  [15:25:27] USB device removed: guid=f53b5c99-385c-46d2-94f2-723bf6352a2b
USB  User  [12:31:37] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=f53b5c99-385c-46d2-94f2-723bf6352a2b
USB  User  [12:31:36] USB device removed: guid=73ba78fd-d3cb-486e-9bd2-9a81ad8aad27
USB  User  [12:30:07] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=73ba78fd-d3cb-486e-9bd2-9a81ad8aad27
USB  User  [12:30:06] USB device removed: guid=e31e665b-b51e-42f3-839b-368bcb3f6c60
USB  User  [12:28:38] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=e31e665b-b51e-42f3-839b-368bcb3f6c60
USB  User  [12:28:38] USB device removed: guid=5019a22e-875c-4a52-a29d-b6f63eded5ef
USB  User  [12:26:18] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=5019a22e-875c-4a52-a29d-b6f63eded5ef
USB  User  [12:26:18] USB device removed: guid=5978f0c6-848d-4579-951b-56e6bcc6f766
USB  User  [12:24:48] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=5978f0c6-848d-4579-951b-56e6bcc6f766
USB  User  [12:24:48] USB device removed: guid=6c1c9524-3c37-4434-ad07-bdcd8b53b446
USB  User  [11:44:14] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=6c1c9524-3c37-4434-ad07-bdcd8b53b446
USB  User  [11:44:14] USB device removed: guid=07dfc312-195f-46a0-97e6-fe651bb89602
USB  User  [11:30:05] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=07dfc312-195f-46a0-97e6-fe651bb89602
USB  User  [11:30:05] USB device removed: guid=3a9921a1-6606-48b6-9d75-a8e9bd7074e2
USB  User  [11:28:35] USB device added: vendor=1452 "Apple Inc.", product=636 "Apple Internal Keyboard / Trackpad", serial="FM7819305UTHYYMAQ+WWZ", guid=3a9921a1-6606-48b6-9d75-a8e9bd7074e2
USB  User  [11:28:35] USB device removed: guid=76421fe3-00f1-4cc7-a5eb-c6b5302febd9
```

Chrome://Usb-Internals/

Name	Serial number	Landing page	
Fake Device	AABBCCDD	https://localhost:5000/	Remove

Add a test device:

Name: Fake Device

Serial number: AABBCCDD

Landing page: https://localhost:5000

Add Added.

Nagging Mario's Princess
npm Enterprise
Features
Pricing
Docs
Support
npm
Search packages
log in or sign up

webusb

1.0.13 • Public • Published 2 months ago

Readme
1 Dependencies
0 Dependents
18 Versions

Node WebUSB

Node.js implementation of the WebUSB Specification

circleci passing
downloads 109/m
licence MIT

Prerequisites

Node.js > v4.8.0, which includes `npm`.

Installation

```
$ npm install webusb
```

Getting Started

See the examples in examples or view the API documentation at:

https://thegecko.github.io/webusb/

Specification

install

```
> npm i webusb
```

weekly downloads
17

version
1.0.13

license
MIT

open issues
5

pull requests
1

homepage
github.com

repository
github

last publish
2 months ago

collaborators

Take Aways 👜

Webusb Lets Us Write Our Own Drivers For Devices

We Can Build Plug & Play Systems:
Like A Checkout System For Shops

It Makes It Easy To Start Tinkering With Hardware (Think Schools), No Hard Setup Steps For Beginners

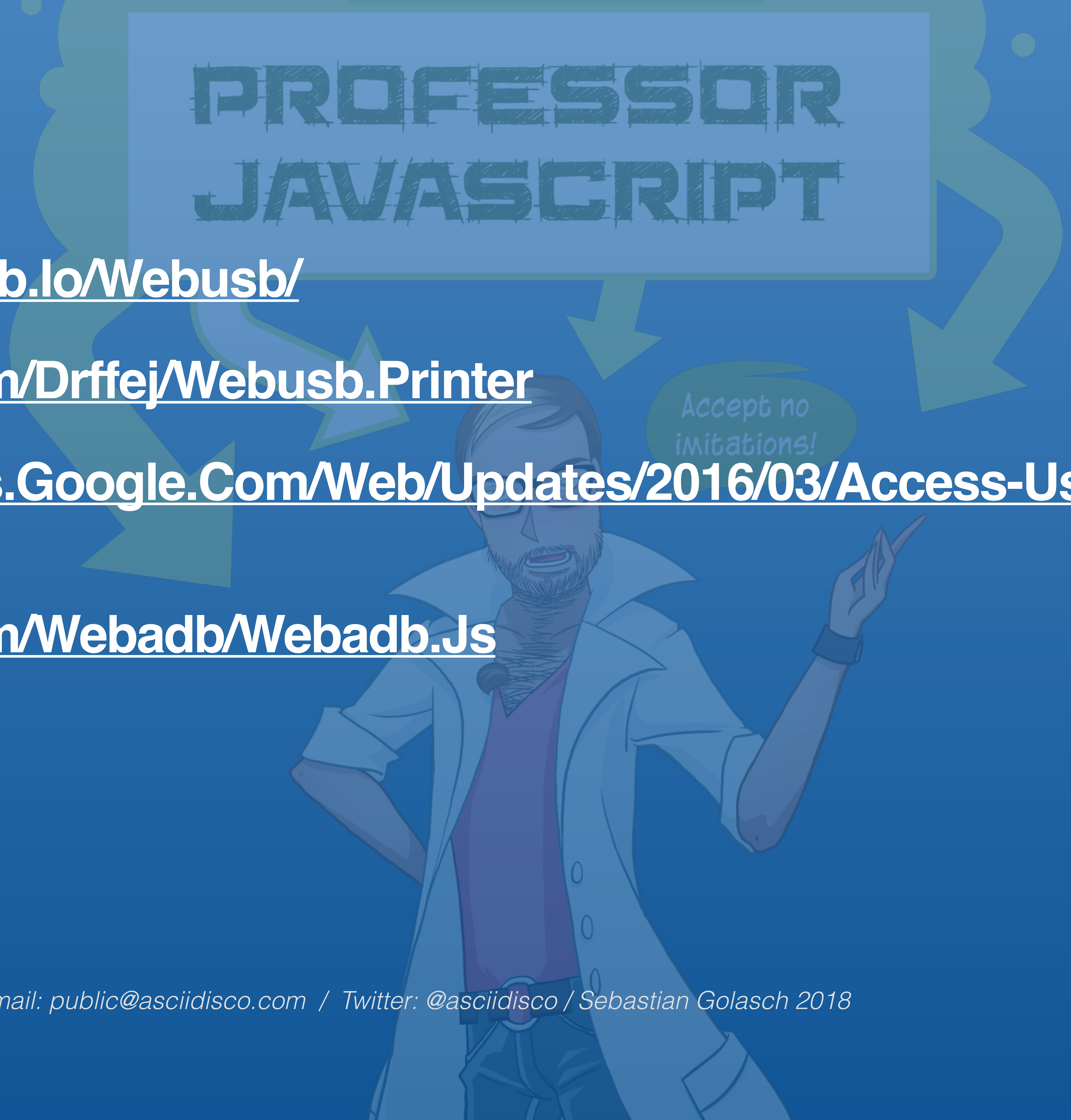
We Can Write Drivers That Work In The Browser & In Backend Systems

We´re Improving The Lives Of The End-User

We Can Have Lots Of Fun 🍾
And Built Shitty Robots 🤖

Спаси́бо Moscow

https://asciidisco.com / E-mail: public@asciidisco.com / Twitter: @asciidisco / Sebastian Golasch 2018



Useful Links:

Https://Wicg.Github.Io/Webusb/

Https://Github.Com/Drffej/Webusb.Printer

Https://Developers.Google.Com/Web/Updates/2016/03/Access-Usb-Devices-On-The-Web

Https://Github.Com/Webadb/Webadb.Js