Server
(running Node.js)

Client

response

HTML CSS JS

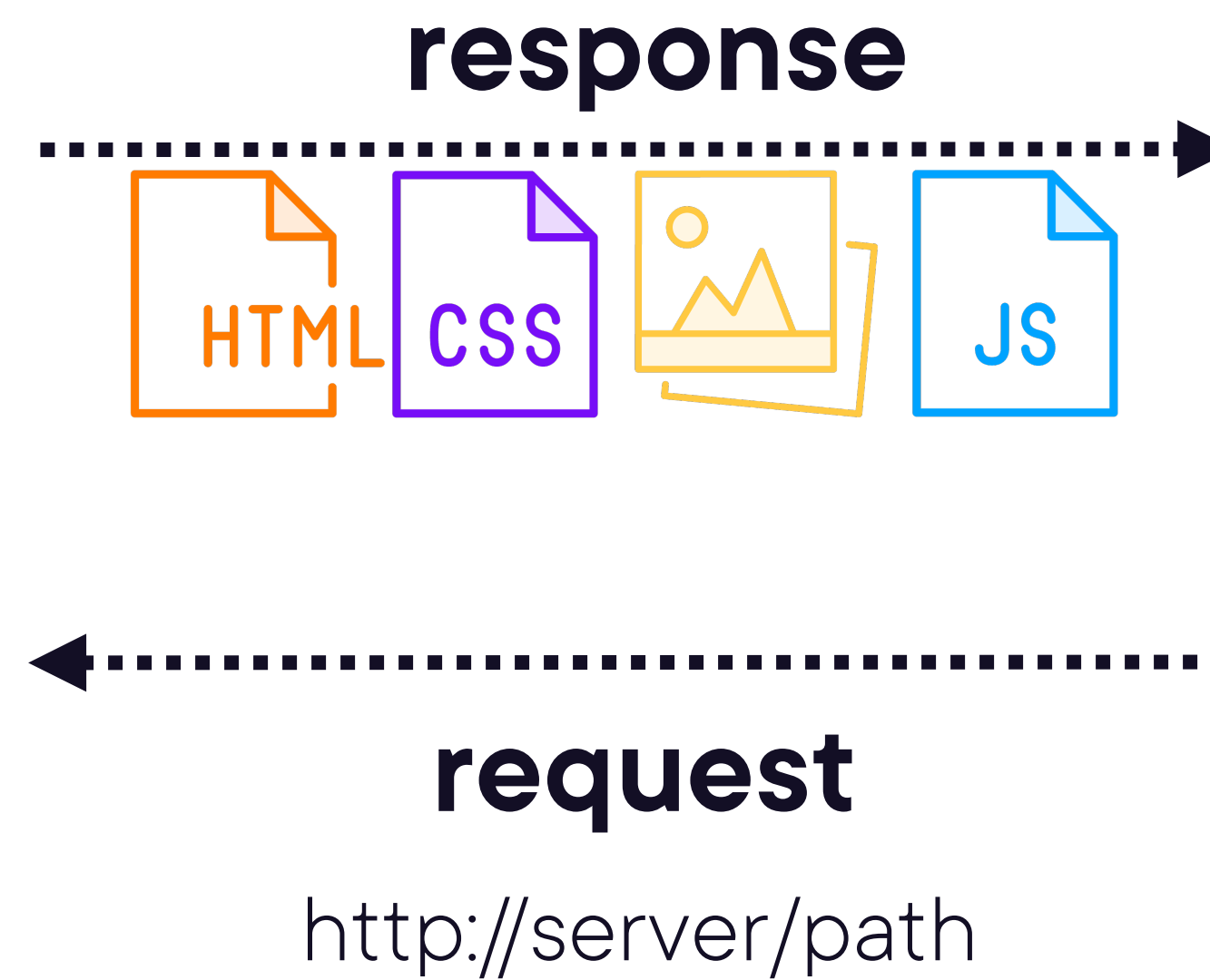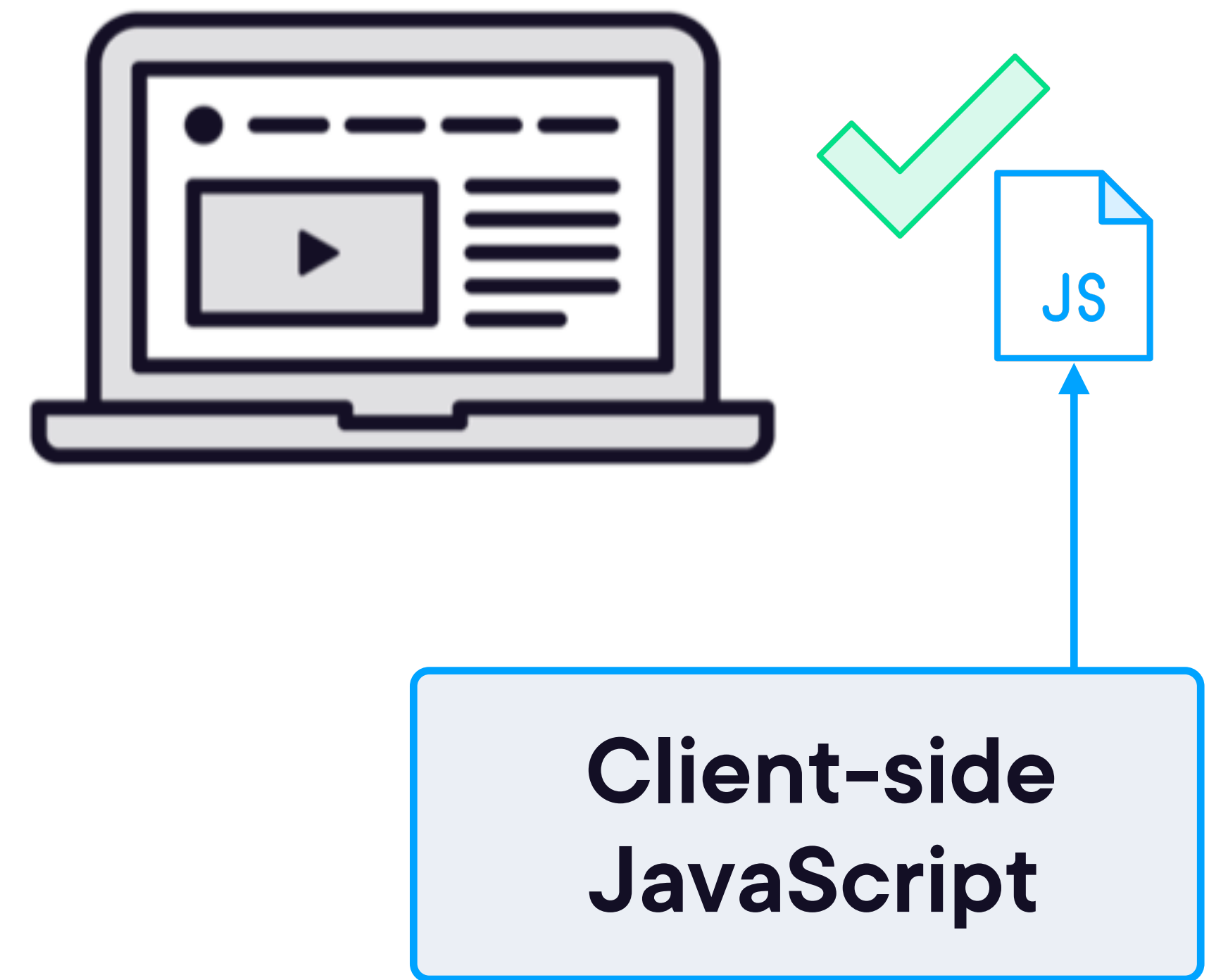http://server/path

request

Server-side JavaScript

Client-side JavaScript

A Simple Node Server

app.mjs

```javascript
import { createServer } from 'node:http'

const server = createServer()

function requestCallback(request, response) {
  response.writeHead(200)
  response.end(`request received`)
}

server.on('request', requestCallback)

server.listen(3000)
```
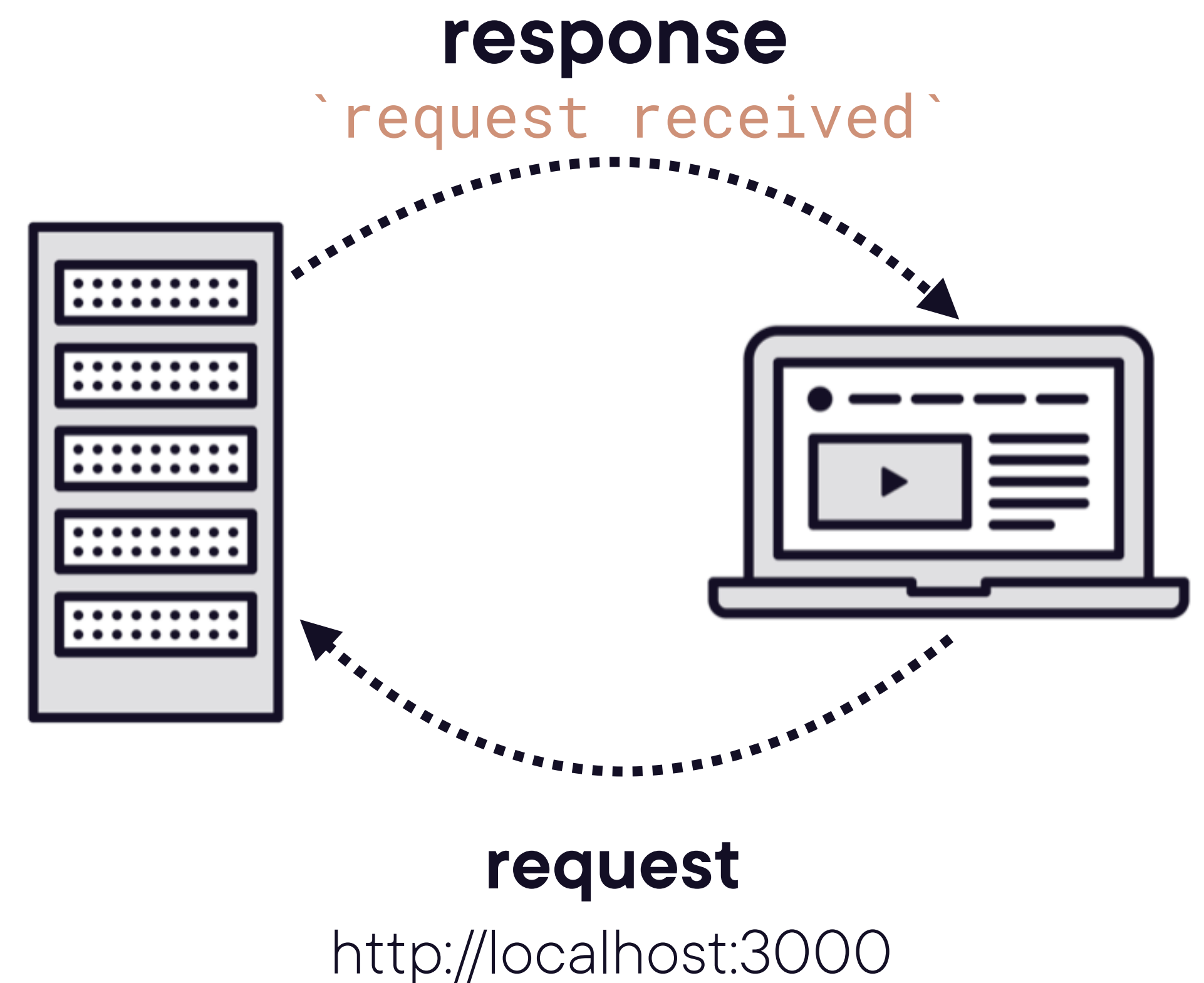
Server

Client

response
`request received`

request
http://localhost:3000

Node.js Server Creation Pattern

Three main actions to create and use a server

1 Import some code and create a server object
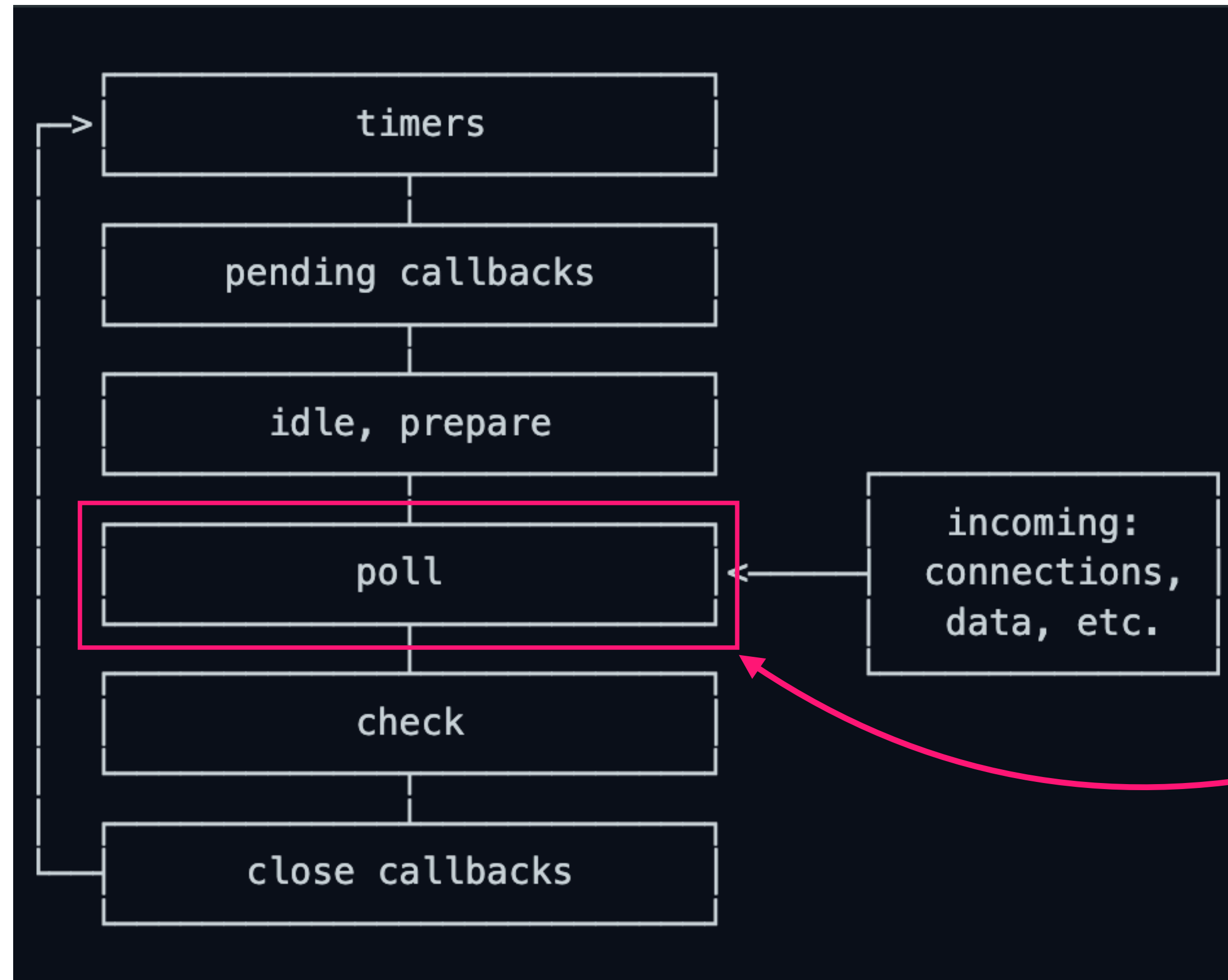
2 Listen for requests coming in to that server

3 Define callback functions that run when that request event happens

Node.js Event Loop

Node.js programs loop forever until they are stopped

Node.js Event Loop


```
    ┌─────────────┐
 ┌─>│   timers    │
 │  └─────────────┘
 │  ┌─────────────┐
 │  │   pending   │
 │  │  callbacks  │
 │  └─────────────┘
 │  ┌─────────────┐
 │  │ idle, prepare│
 │  └─────────────┘      ┌─────────────┐
 │  ┌─────────────┐      │  incoming:  │
 │  │    poll     │<─────│ connections,│
 │  └─────────────┘      │  data, etc. │
 │  ┌─────────────┐      └─────────────┘
 │  │    check    │
 │  └─────────────┘
 │  ┌─────────────┐
 └──│    close    │
    │  callbacks  │
    └─────────────┘
```

server requests and
client responses happen
in this phase

A Simple Express Server

app-express.mjs

```
import express from 'express'

const server = express()

function requestCallback(request, response) {
  response.status(200).send(`request received`)
}

server.get('/', requestCallback)

server.listen(3000)
```
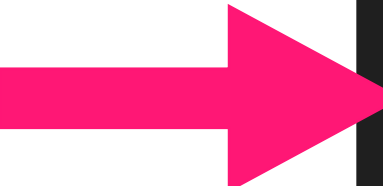
A Simple Express Server

app-express.mjs

```
import express from 'express'

const server = express()

function requestCallback(request, response) {
  response.status(200).send(`request received`)
}

server.get('/', requestCallback)

server.listen(3000)
```
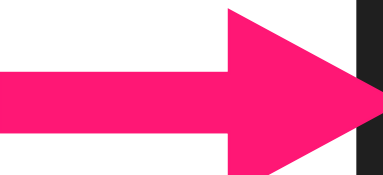
A Simple Express Server

app-express.mjs

```
import express from 'express'

const server = express()

function requestCallback(request, response) {
  response.status(200).send(`request received`)
}

server.get('/', requestCallback)

server.listen(3000)
```
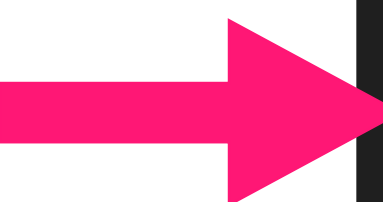
A Simple Express Server

app-express.mjs

```
import express from 'express'

const server = express()

function requestCallback(request, response) {
  response.status(200).send(`request received`)
}


server.get('/', requestCallback)

server.listen(3000)
```

A Simple Express Server

app-express.mjs

```javascript
import express from 'express'

const server = express()

function requestCallback(request, response) {
  response.status(200).send(`request received`)
}

server.get('/', requestCallback)

server.listen(3000)
```
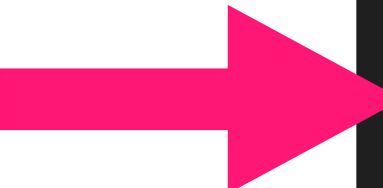
A Simple Express Server

app-express.mjs

```javascript
import express from 'express'

const server = express()

function requestCallback(request, response) {
  response.status(200).send(`request received`)
}

server.get('/', requestCallback)

server.listen(3000)
```

Server

Client

GET request

http://localhost:3000

A Simple Express Server

app-express.mjs

```javascript
import express from 'express'

const server = express()

function requestCallback(request, response) {
  response.status(200).send(`request received`)
}

server.get('/', requestCallback)

server.listen(3000)
```
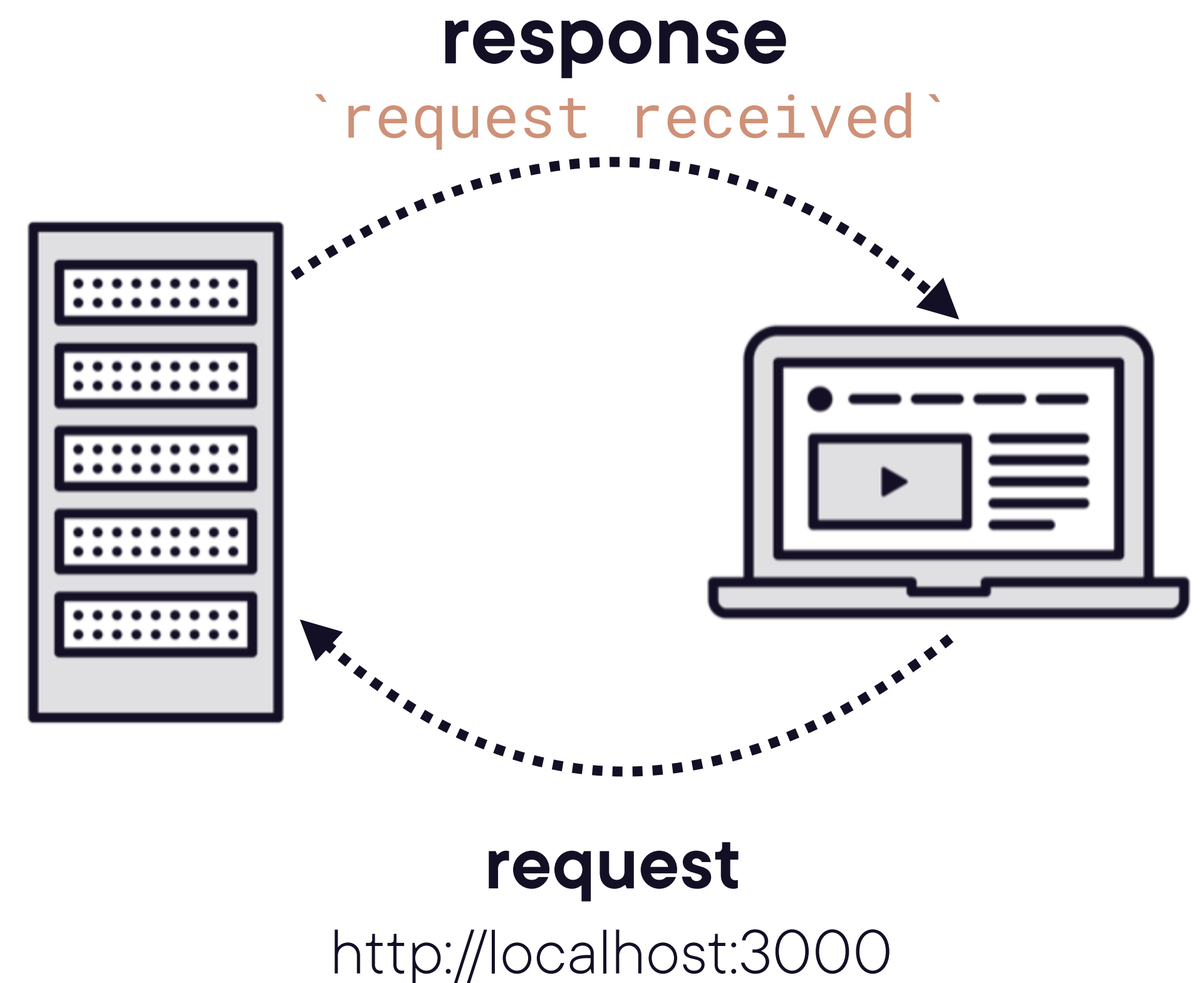
Server

Client

response
`request received`

request
http://localhost:3000

HTTP Status Code 200

"OK"

The request succeeded