

EXPRESS 201

TRAJECTORY

- **Handling URL params**
- **App.use and next**
- **Serving static resources**

URL PARAMETERS

REQ.PARAMS

GET /PUPPIES/1

```
app.get('/puppies/:puppyId', (req, res, next) => {  
  console.log(req.params.puppyId)  
  // use req.params.puppyId to find the puppy with the requested id  
  // and then send that puppy as the response!  
})
```

APP.USE AND NEXT

```
const express = require('express')

const app = express()

app.get('/', (req, res, next) => {
  res.send(`<h1>Welcome to the main page</h1>`)
})

app.get('/puppies', (req, res, next) => {
  res.send(`<h1>Welcome to the puppies page</h1>`)
})

app.listen(3000)
```

YOUR EXPRESS APP

```
app.get( '/' )
```

```
app.get( '/puppies' )
```

GET /puppies HTTP /1.1

HTTP/1.1 200 OK

“Are you a handler for GET /puppies”?
Date: Wed, 03 Jan 2018 17:29:52 GMT

Connection: keep-alive

Transfer-Encoding: chunked

<h1>Welcome to the puppies page!</h1>


```
app.get('*', (req, res, next) => {  
  console.log(req.method, req.url)  
})
```

```
app.get('/', (req, res, next) => {  
  res.send(`<h1>Welcome to the main page</h1>`)  
})
```

```
app.get('/puppies', (req, res, next) => {  
  res.send(`<h1>Welcome to the puppies page</h1>`)  
})
```

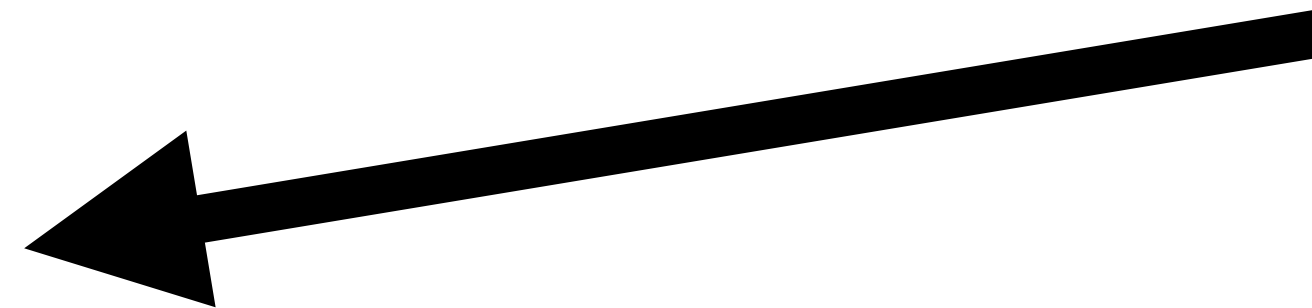
YOUR EXPRESS APP

```
app.get( '*' )
```

```
app.get( ' / ' )
```

```
app.get( ' /puppies ' )
```

GET /puppies HTTP /1.1



“Are you a handler for GET /puppies”?

“I’m a handler for everything! I’ll log something to the console and...”

“ ”

```
app.get('*', (req, res, next) => {  
  console.log(req.method, req.url)  
  next()  
})
```

```
app.get('/', (req, res, next) => {  
  res.send(`<h1>Welcome to the main page</h1>`)  
})
```

```
app.get('/puppies', (req, res, next) => {  
  res.send(`<h1>Welcome to the puppies page</h1>`)  
})
```

```
app.get('*', (req, res, next) => {  
  console.log(req.method, req.url)  
  next()  
})
```

```
app.get('/', (req, res, next) => {  
  res.send(`<h1>Welcome to the main page</h1>`)  
})
```

```
app.get('/puppies', (req, res, next) => {  
  res.send(`<h1>Welcome to the puppies page</h1>`)  
})
```



```
app.use('*', (req, res, next) => {  
  console.log(req.method, req.url)  
  next()  
})
```

```
app.get('/', (req, res, next) => {  
  res.send(`<h1>Welcome to the main page</h1>`)  
})
```

```
app.get('/puppies', (req, res, next) => {  
  res.send(`<h1>Welcome to the puppies page</h1>`)  
})
```

```
app.use((req, res, next) => {  
  console.log(req.method, req.url)  
  next()  
})
```

```
app.get('/', (req, res, next) => {  
  res.send(`<h1>Welcome to the main page</h1>`)  
})
```

```
app.get('/puppies', (req, res, next) => {  
  res.send(`<h1>Welcome to the puppies page</h1>`)  
})
```

APP.GET

- Only for GET requests
- Exact match of verb and URL

APP.USE

- Handles all verbs
- Fuzzy match of url (we'll explore this more later)



EXPRESS MIDDLEWARE

A function that handles responding to requests.

*Of the form: **function(req, res, next){...}***



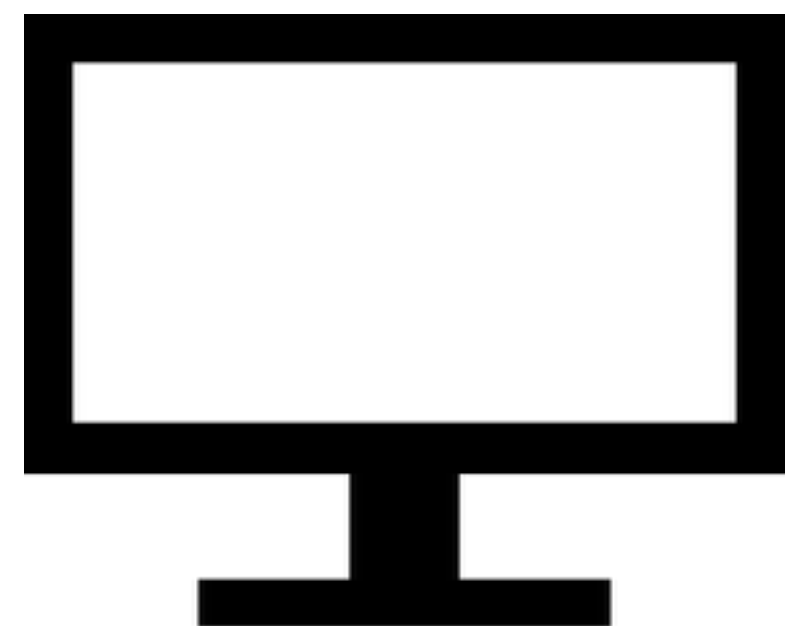
```
app.use((req, res, next) => {  
  console.log(req.method, req.url)  
  next()  
})
```



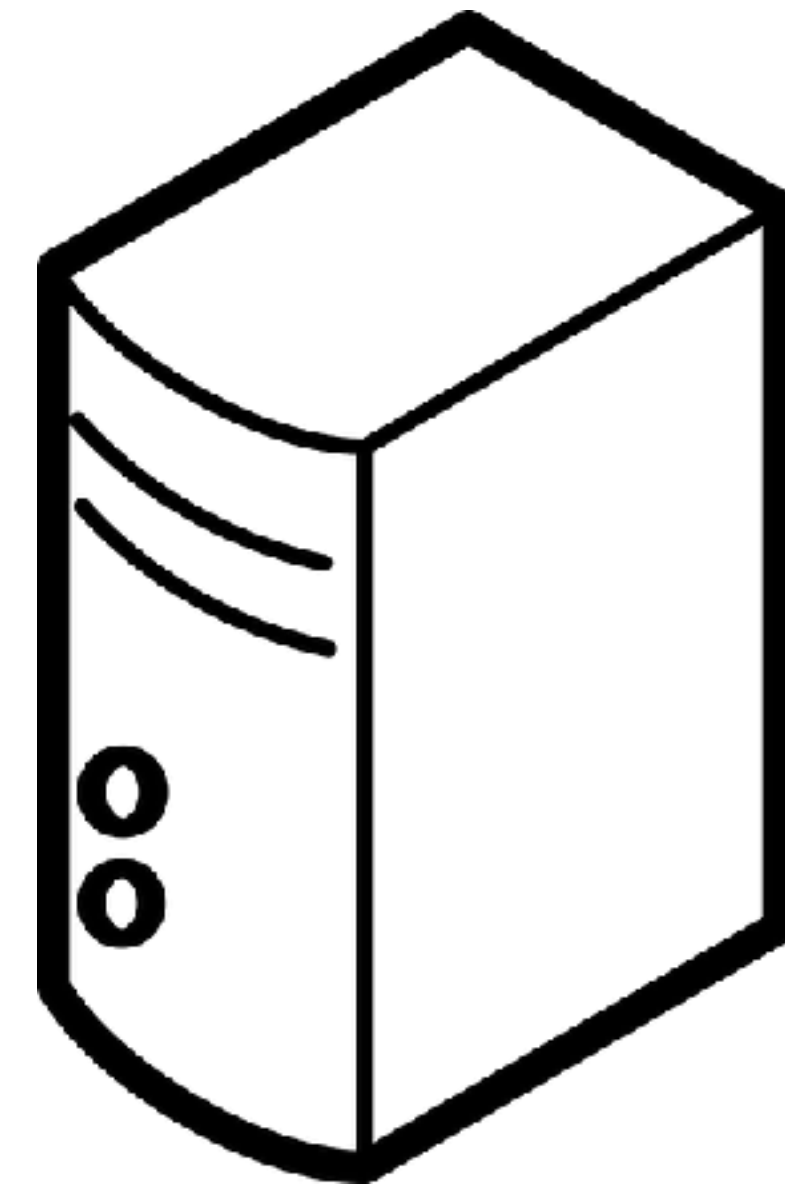
```
const morgan = require('morgan')  
app.use(morgan('dev'))
```

EXPRESS.STATIC

```
/server
  app.js
/public
  styleA.css
  styleB.css
  clientSideScript.js
```



GET / HTTP /1.1



HTTP/1.1 200 OK

Date: Wed, 03 Jan 2018 17:29:52 GMT

Connection: keep-alive

Transfer-Encoding: chunked

<!doctype html>

<!doctype html>

<head>

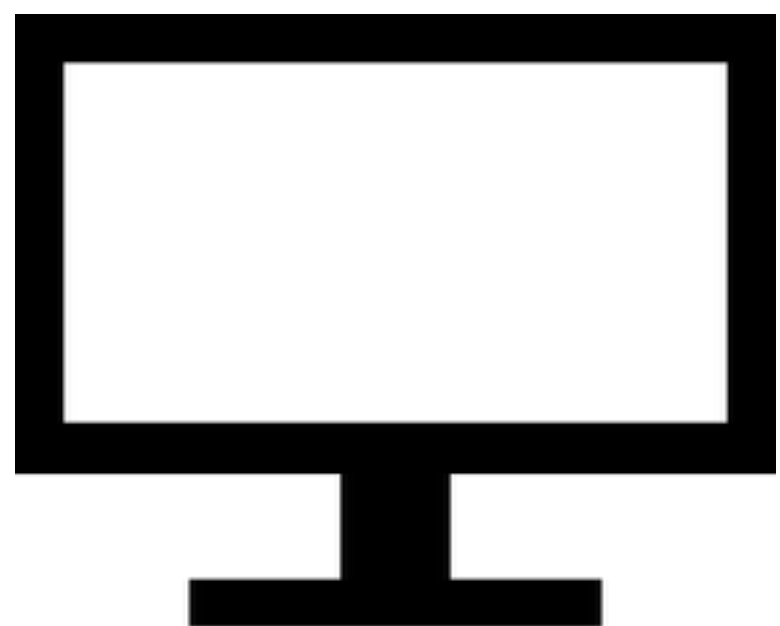
<link rel="stylesheet" href="/style.css" />

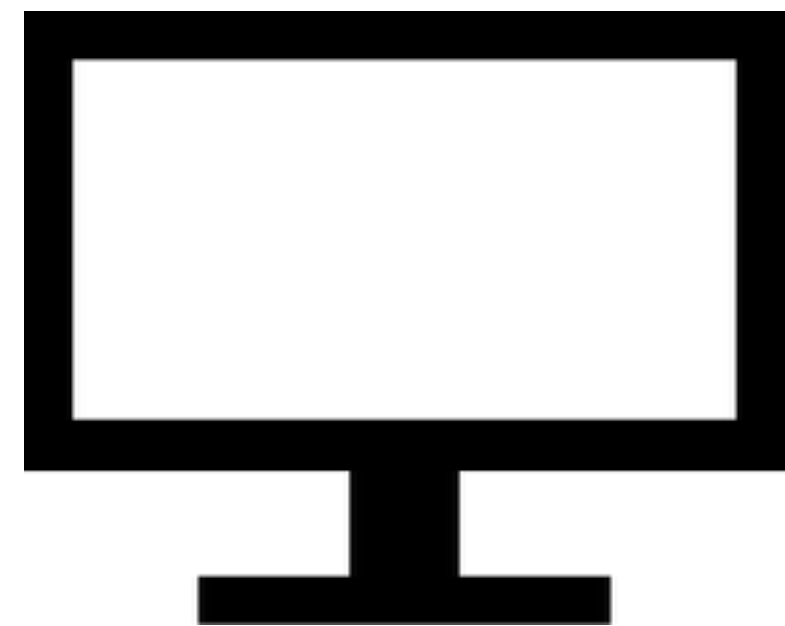
</head>

<body>

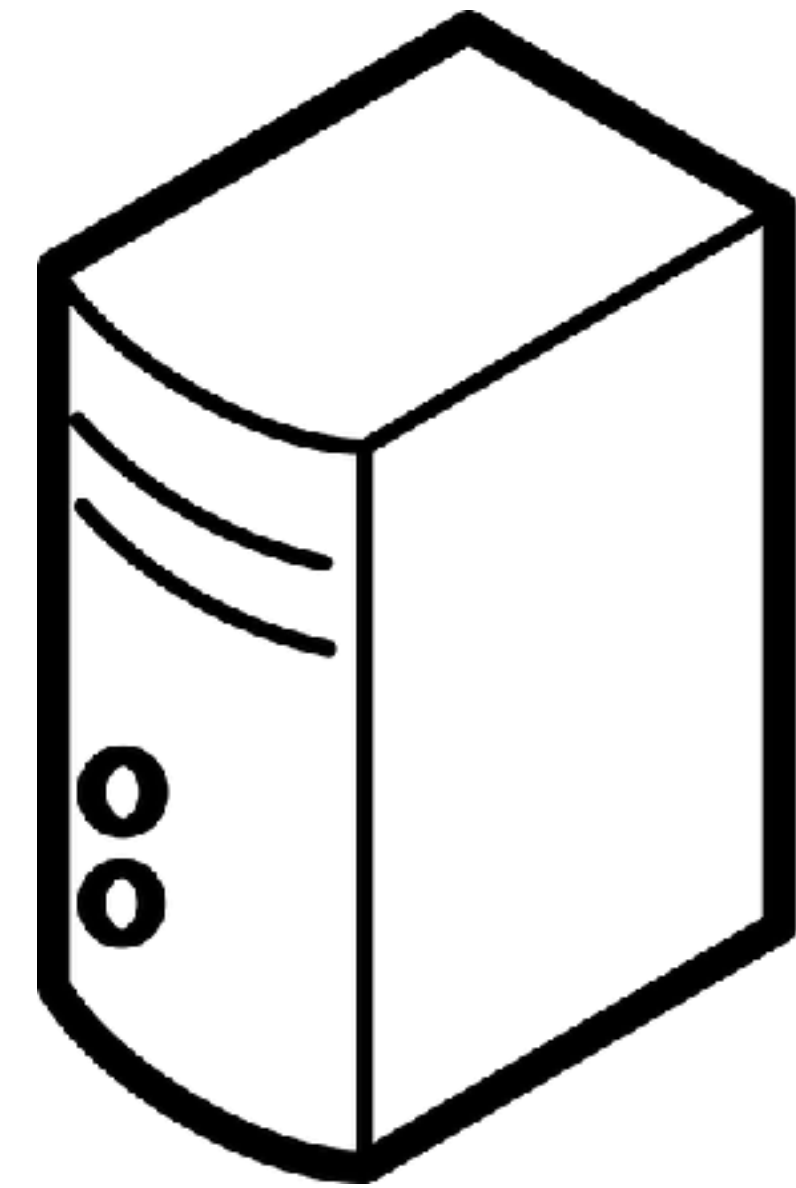
</body>

</html>





GET /style.css HTTP /1.1



```
app.get('/style.css', (req, res, next) => {  
  res.sendFile(path.join(__dirname, 'public', 'style.css'))  
})
```



```
app.get('/styleA.css')  
  res.sendFile(  
  })
```

```
app.get('/styleB.css')  
  res.sendFile(  
  })
```

```
app.get('/clientSideScript.js')  
  res.sendFile(  
  })
```

```
styleA.css'))
```

```
styleB.css'))
```

```
⇒ {  
  clientSideScript.js'))
```



```
const staticMiddleware = express.static(path.join(__dirname, 'public'))  
app.use(staticMiddleware)
```