CORS Explained Like You’re 5 (But With Code)

@ sanuj bansal

Meet the Problem

You're building a website: http://frontend.com
Your data is on: http://backend.com
You try to fetch data from the backend...
And boom — you get this:

```
Access to fetch at 'http://backend.com'
from origin 'http://frontend.com'
has been blocked by CORS policy
```

@ sanuj bansal

What Just Happened?

Think of it like this:
You go to your friend’s house to get cookies.
But the friend’s mom (the browser) stops you and says:

“You’re not from this house. Do you have permission?”
That’s CORS — Cross-Origin Resource Sharing.

The browser is just protecting your friend’s cookies.

@ sanuj bansal

What is an “Origin”?

Origin = Protocol + Domain + Port

Same origin:
http://site.com:80 → http://site.com:80

Cross origin:
http://site.com → http://api.site.com

http://localhost:3000 → http://localhost:5000
Different origins trigger CORS checks!

@ sanuj bansal

→

How Does CORS Work?

1. You send a request from frontend.com
2. Browser asks the server:

“Hey, can this origin access your resources?”

1. Server must respond with this header:

```
Access-Control-Allow-Origin: http://frontend.com
```

If yes — request goes through
If no — blocked!

@ sanuj bansal

What If It’s a POST or Custom Header?

1. Now things get a little fancy…
2. The browser sends a preflight request using OPTIONS:

```
OPTIONS /data HTTP/1.1
Origin: http://frontend.com
Access-Control-Request-Method: POST
```

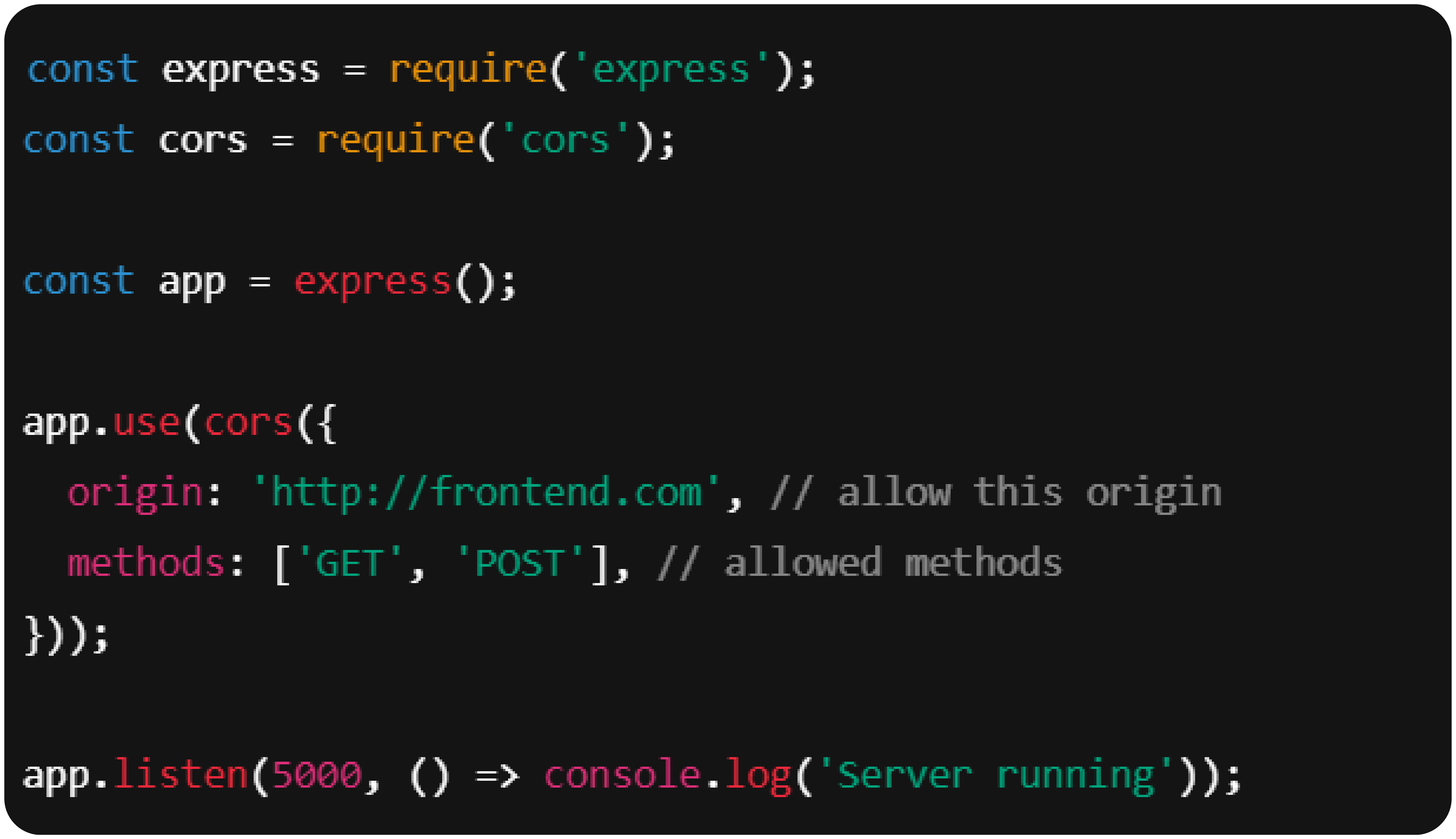
The server must respond like:

```
Access-Control-Allow-Origin: http://frontend.com
Access-Control-Allow-Methods: POST
```

Then the real request goes through.

@ sanuj bansal

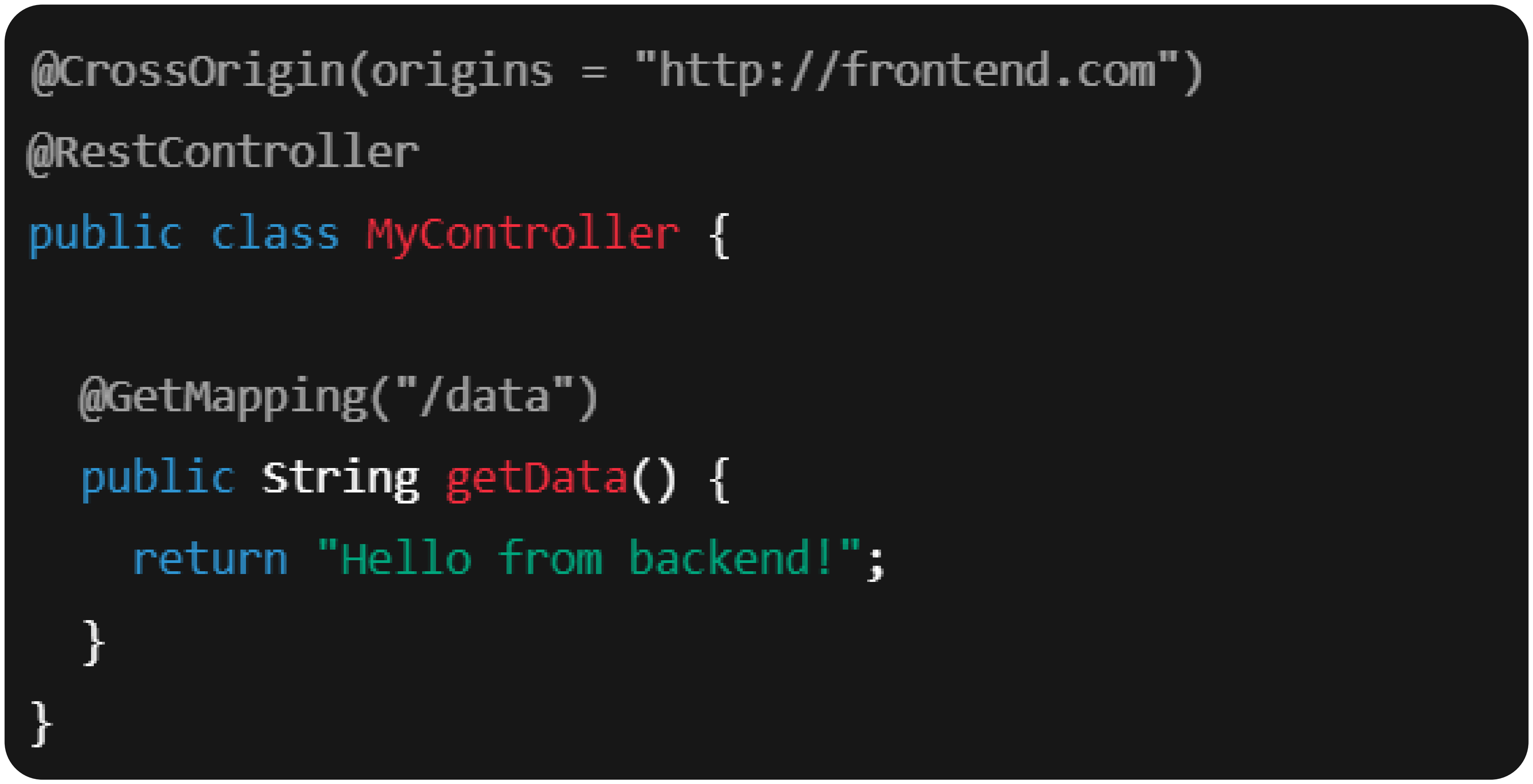
Fixing CORS in Code (Express.js Example)



```javascript
const express = require('express');
const cors = require('cors');

const app = express();

app.use(cors({
  origin: 'http://frontend.com', // allow this origin
  methods: ['GET', 'POST'], // allowed methods
}));

app.listen(5000, () => console.log('Server running'));
```

@ sanuj bansal

Fixing CORS in Spring Boot



```java
@CrossOrigin(origins = "http://frontend.com")
@RestController
public class MyController {

  @GetMapping("/data")
  public String getData() {
    return "Hello from backend!";
  }
}
```

@ sanuj bansal

Quick CORS Tips

✅ CORS is a browser-side security feature
✅ CORS is not a server bug
✅ Don’t use * in production (wildcard = risky)
✅ Use proxy during development (like Vite/React proxy option)
✅ Backend must explicitly allow frontend origin
✅CORS = Browser asking “Do you have permission to access this origin?”

You fix CORS by:
✓ Configuring backend to allow the right origin
✓ Using proper headers
✓ Testing with browser tools (check network tab)

@ sanuj bansal

→