

NodeJS

Part 2

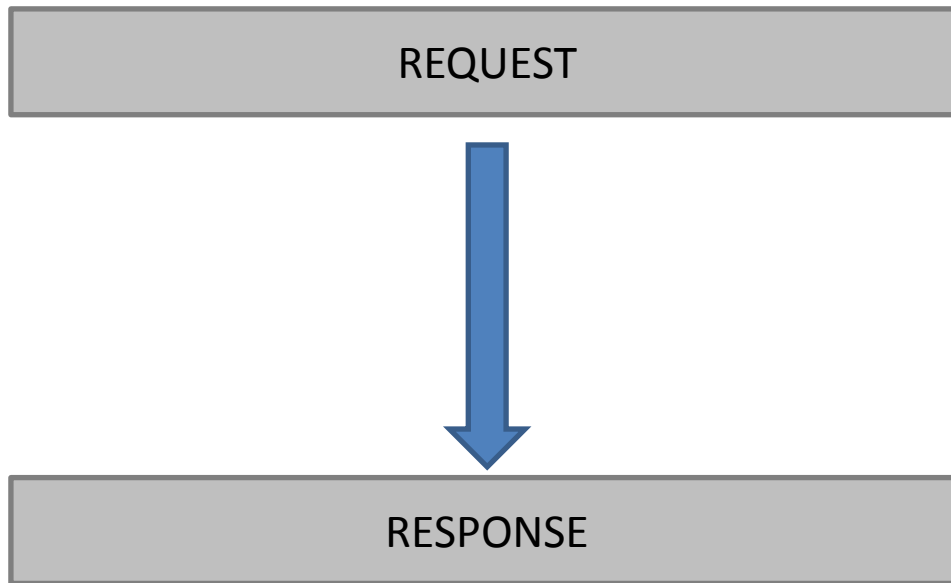
jan.schulz@devugees.org

Agenda

1. Middleware
2. Templates
3. Relational Databases: SQL

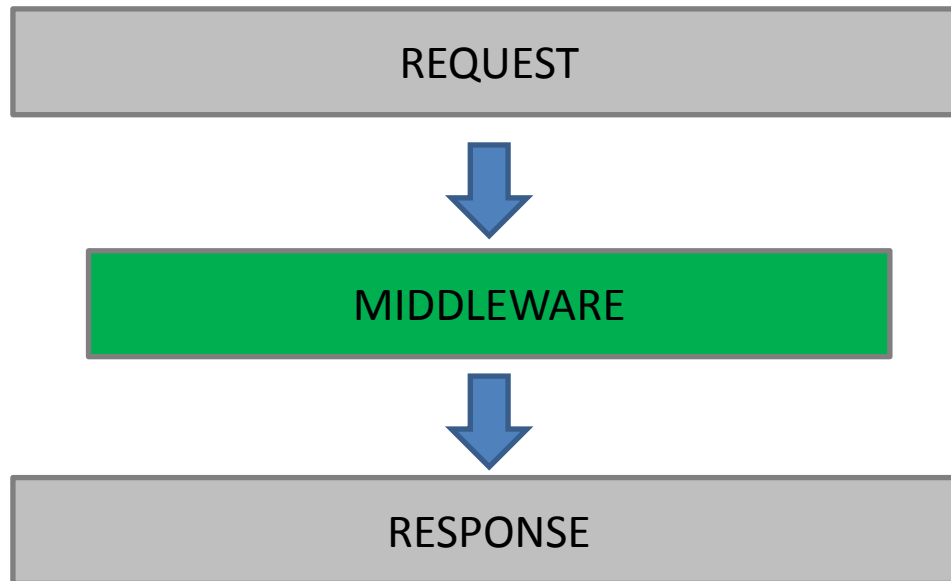
1. Middleware

- Middleware: Code that sits between two layers of software



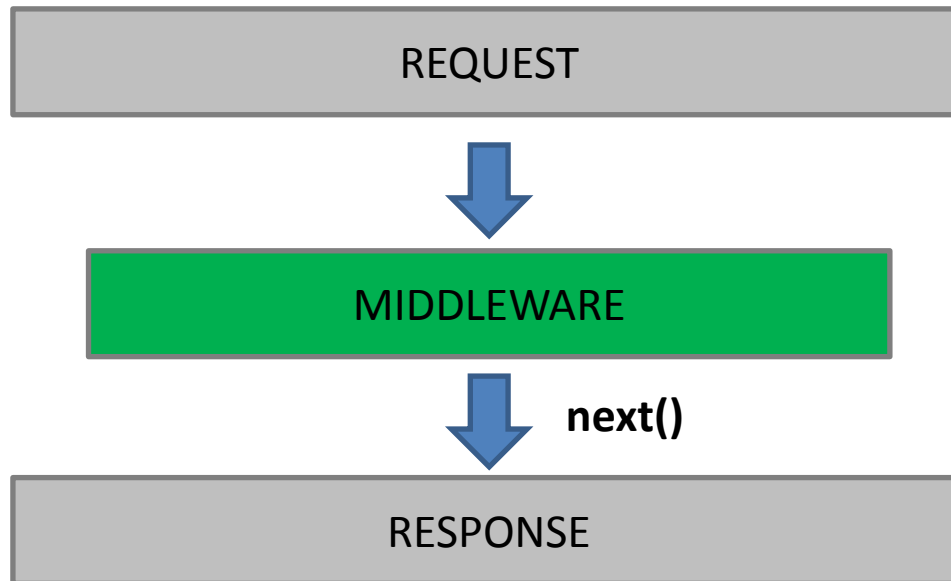
1. Middleware

- Middleware: Code that sits between two layers of software



1. Middleware

- Middleware: Code that sits between two layers of software



1. Middleware

- Middleware: creates re-usable ways of dealing with HTTP-requests

2. Templates

- A blueprint for HTML-files
- Placeholders will be replaced by JavaScript-variables when a template is rendered

```
<html>
  <head></head>
  <% if (helloworld) %>
    <h2><%=helloworld%></h2>
  <% } %>
  <body></body>
</html>
```

2. Templates

Task:

Convert your personal blog to your own NodeJS server.

- 1.) Create a static asset /public
- 2.) Implement a route GET /
- 3.) Convert your HTML file to an EJS template.

3. Relational Databases - SQL

- Relational Database:

ID	Firstname	Lastname
17	John	Doe

3. Relational Databases - SQL

- Relational Database:

ID	Firstname	Lastname
17	John	Doe

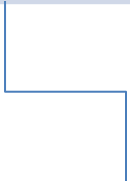


Person	Street	Number
17	Main Street	2

3. Relational Databases - SQL

- Relational Database:

ID	Firstname	Lastname
17	John	Doe



Person	Street	Number
17	Main Street	2

Task: How would this look as JavaScript Object?

3. Relational Databases - SQL

```
{  
  firstname: 'John',  
  lastname: 'Doe',  
  address: {  
    street: 'Main Street',  
    number: 2  
  }  
}
```

3. Relational Databases - SQL

Task:

Implement two routes for your personal blog NodeJS server.

GET /contact

Lists all of your contact requests

POST /contact

Creates a new contact request.

Note: Create a MySQL database and a table for this.