NATIONAL INSTITUTE OF TECHNOLOGY SURATHKAL

IT-254 WTA Lab

Server Side Programming - Using MongoDB

Submitted By-
Akshay B Bistagond
191EC203

Develop an online library website for your department. Design an admin page where the
admin can list, add, remove, update books available in the library. Consider details such
as: book title, author name, edition, subject, number of copies available.

● The admin must make a list of books and add it to the online library.

● Admin must be able to retrieve all the books in the library.

● Admin must be able to search and list the books based on author as well as subject. Example: Web technologies must display a list of Web technologies related books.

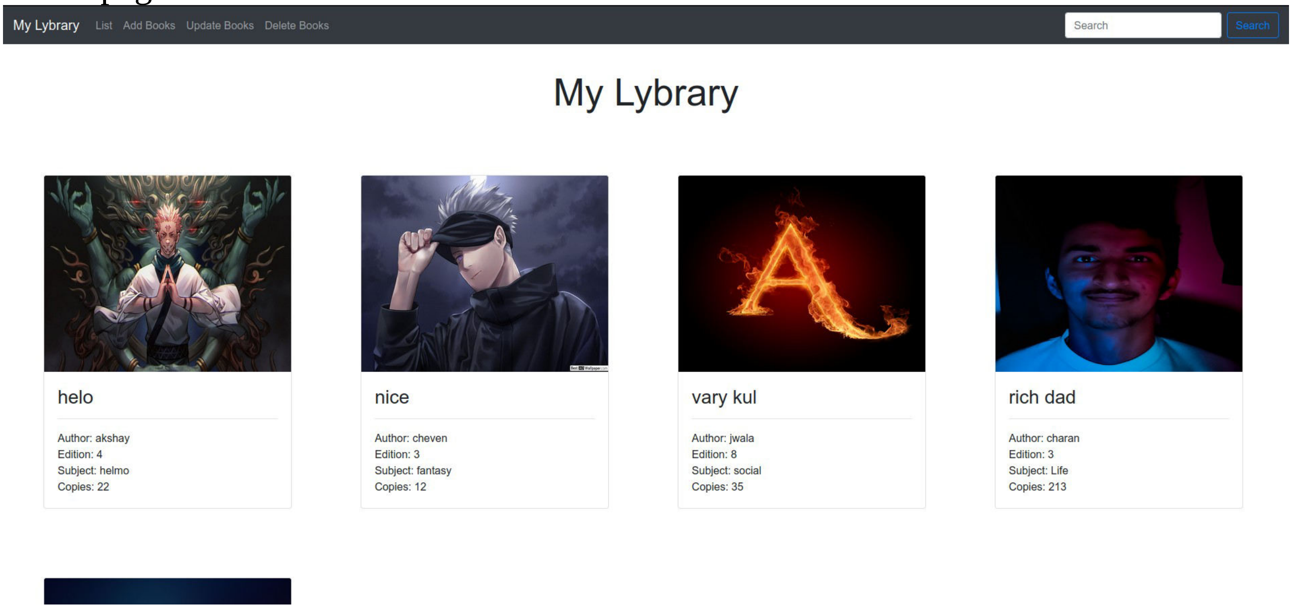
● Admin must be able to delete any book.

● Admin must be able to update the details of any book.

It is required to follow the Model View and Controller approach for this assignment. Make an appealing frontend for user interaction

Pages:

Home page:



My Lybrary

helo

Author: akshay
Edition: 4
Subject: helmo
Copies: 22

nice

Author: cheven
Edition: 3
Subject: fantasy
Copies: 12

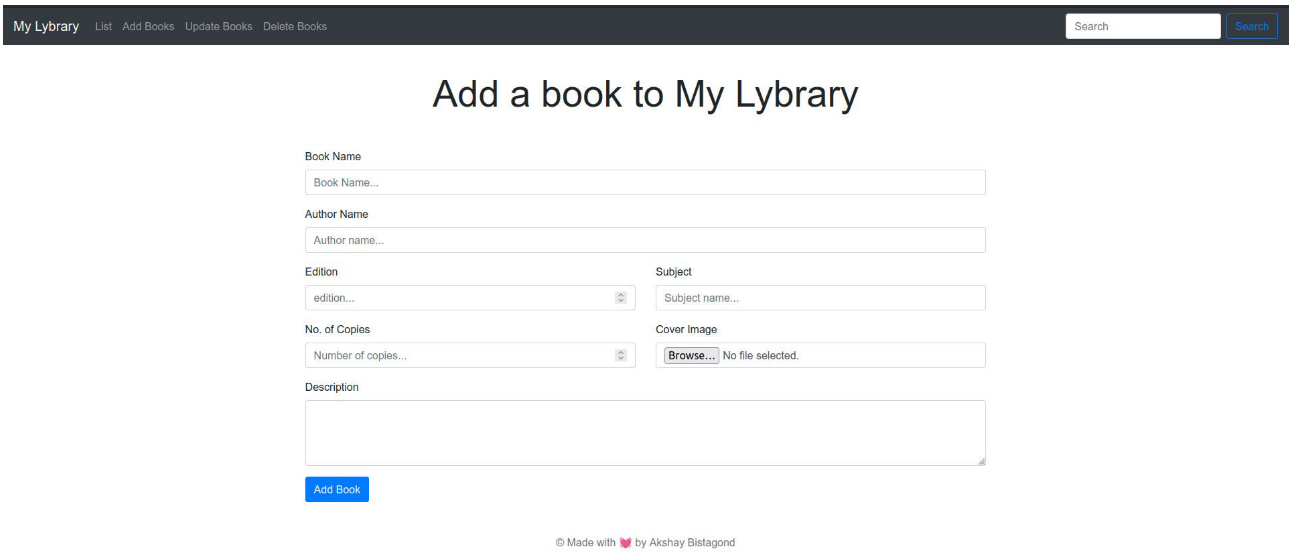
vary kul

Author: jwala
Edition: 8
Subject: social
Copies: 35

rich dad

Author: charan
Edition: 3
Subject: Life
Copies: 213

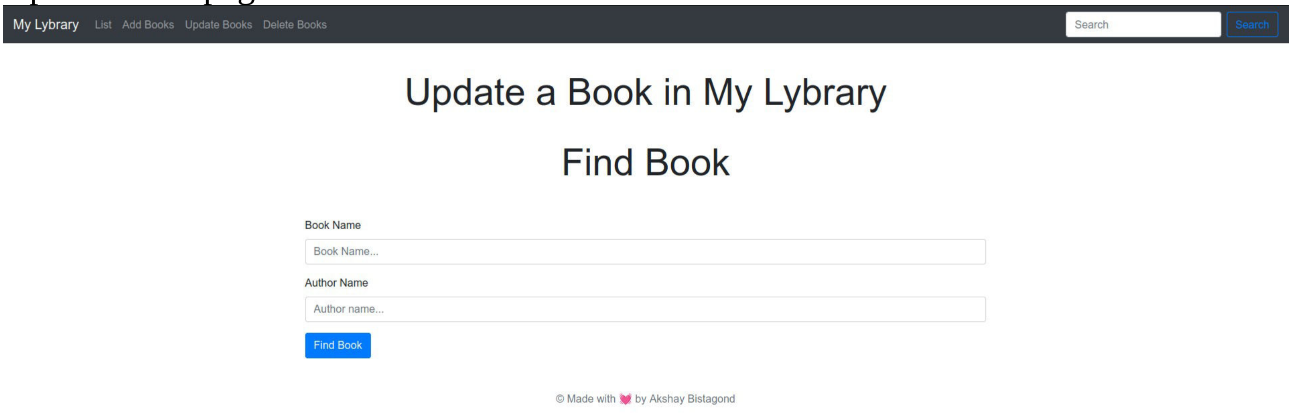
Add Book Page:



Add a book to My Lybrary

© Made with ❤ by Akshay Bistagond

Update Book page:

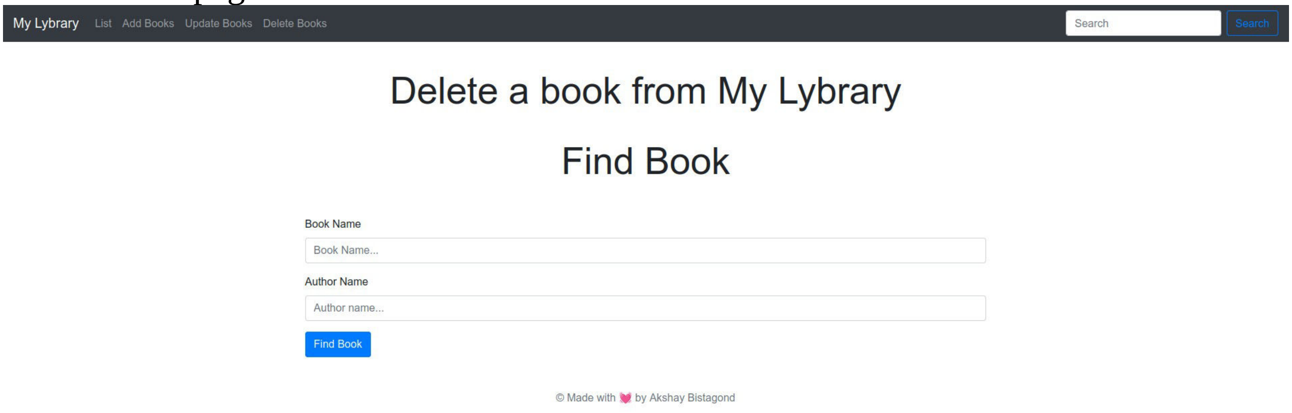


Update a Book in My Lybrary

Find Book

© Made with ❤ by Akshay Bistagond

Delete Book page:

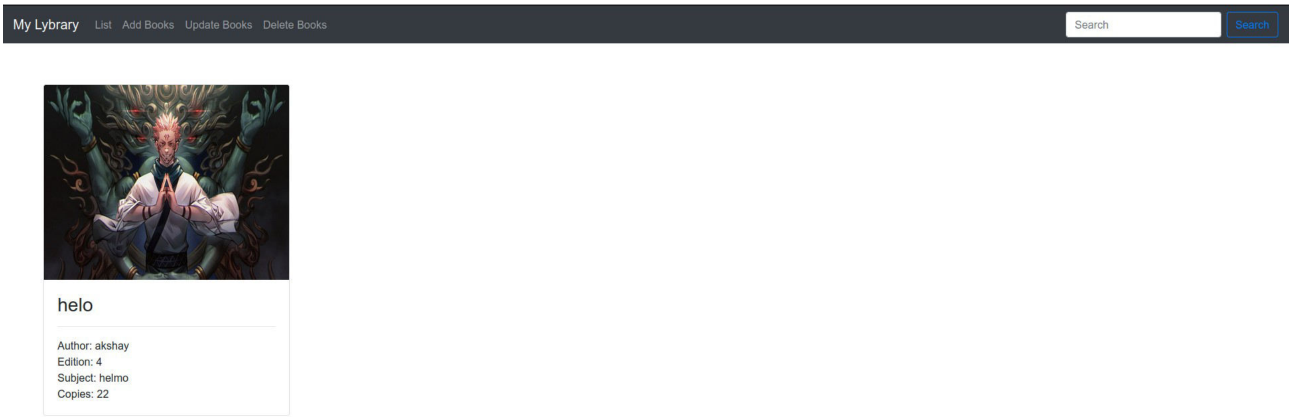


Delete a book from My Lybrary

Find Book

© Made with ❤ by Akshay Bistagond

Search Result page:

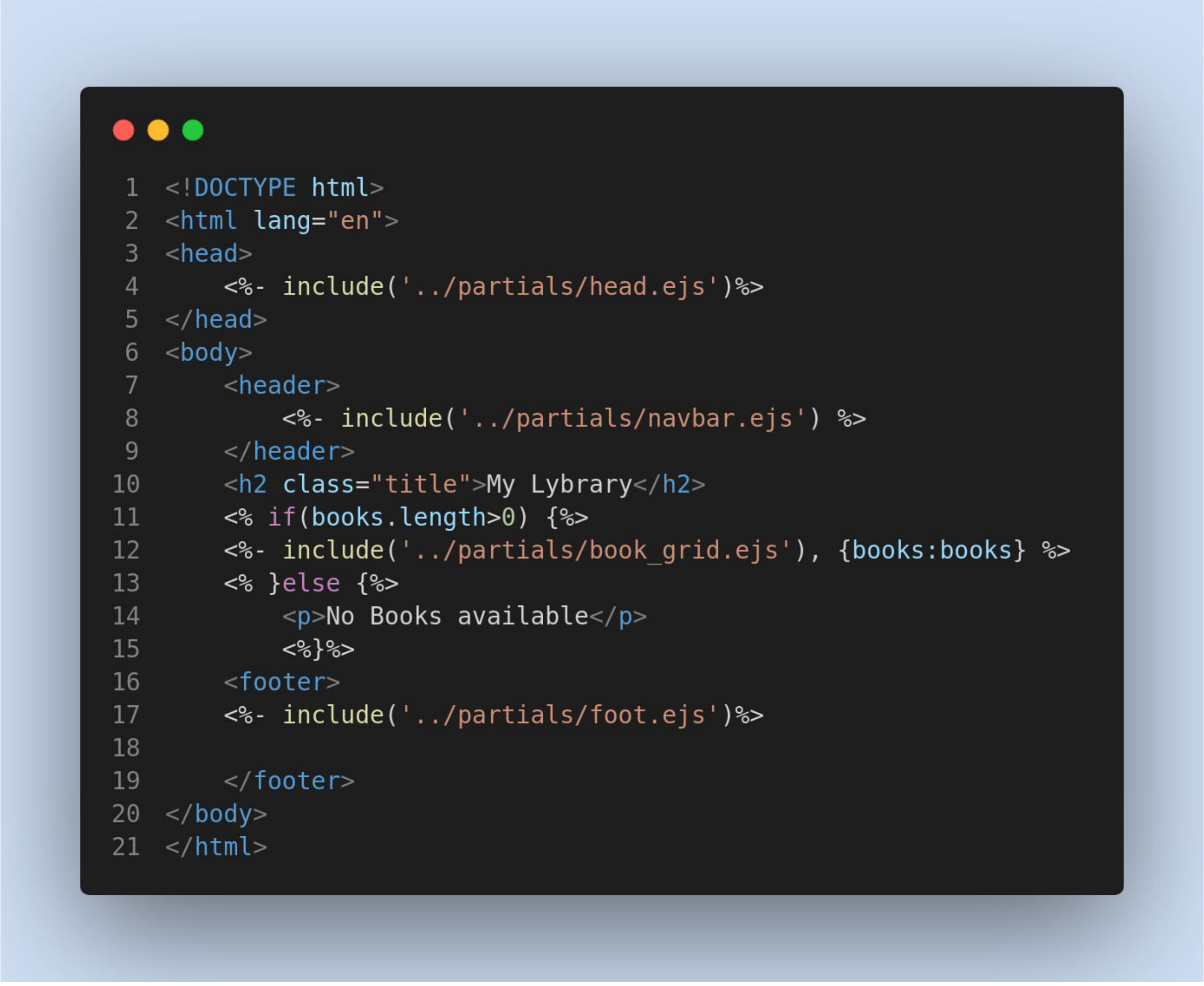


helo

Author: akshay
Edition: 4
Subject: helmo
Copies: 22

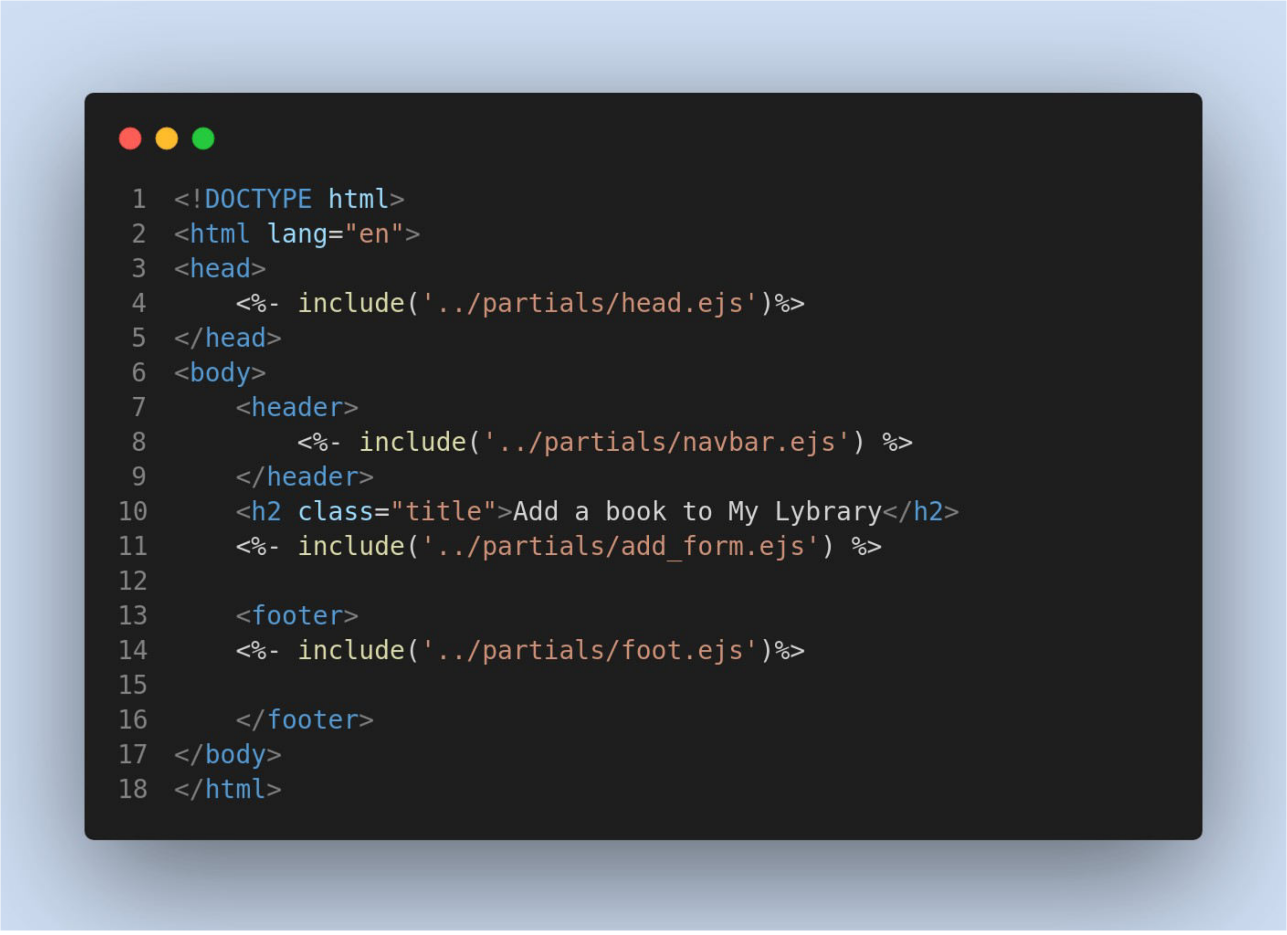
Views used (ejs):

Home page:



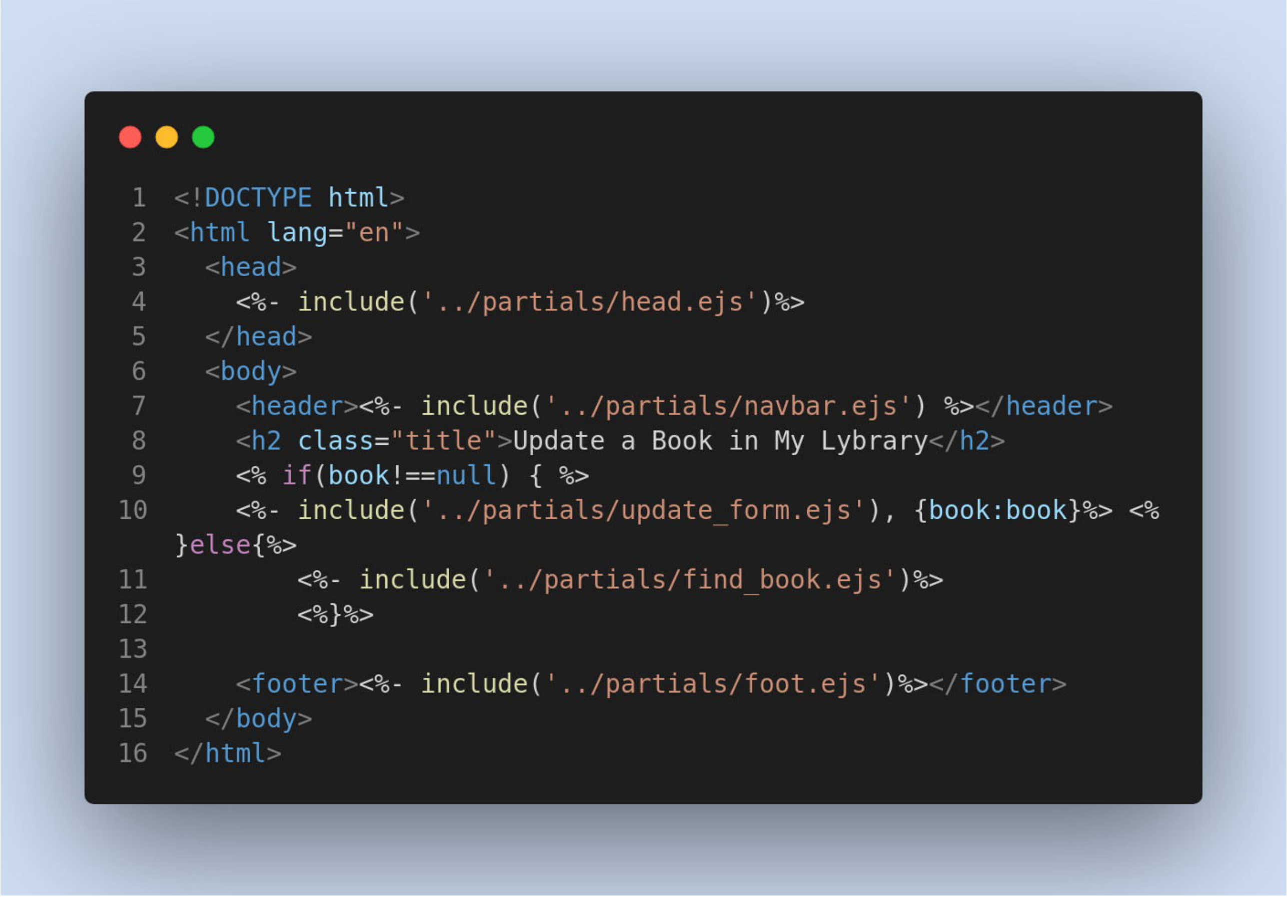
```
<!DOCTYPE html>
<html lang="en">
<head>
    <%- include('../partials/head.ejs')%>
</head>
<body>
    <header>
        <%- include('../partials/navbar.ejs') %>
    </header>
    <h2 class="title">My Lybrary</h2>
    <% if(books.length>0) {%>
    <%- include('../partials/book_grid.ejs'), {books:books} %>
    <% }else {%>
        <p>No Books available</p>
        <%}%>
    <footer>
    <%- include('../partials/foot.ejs')%>

    </footer>
</body>
</html>
```

Add Book Page:



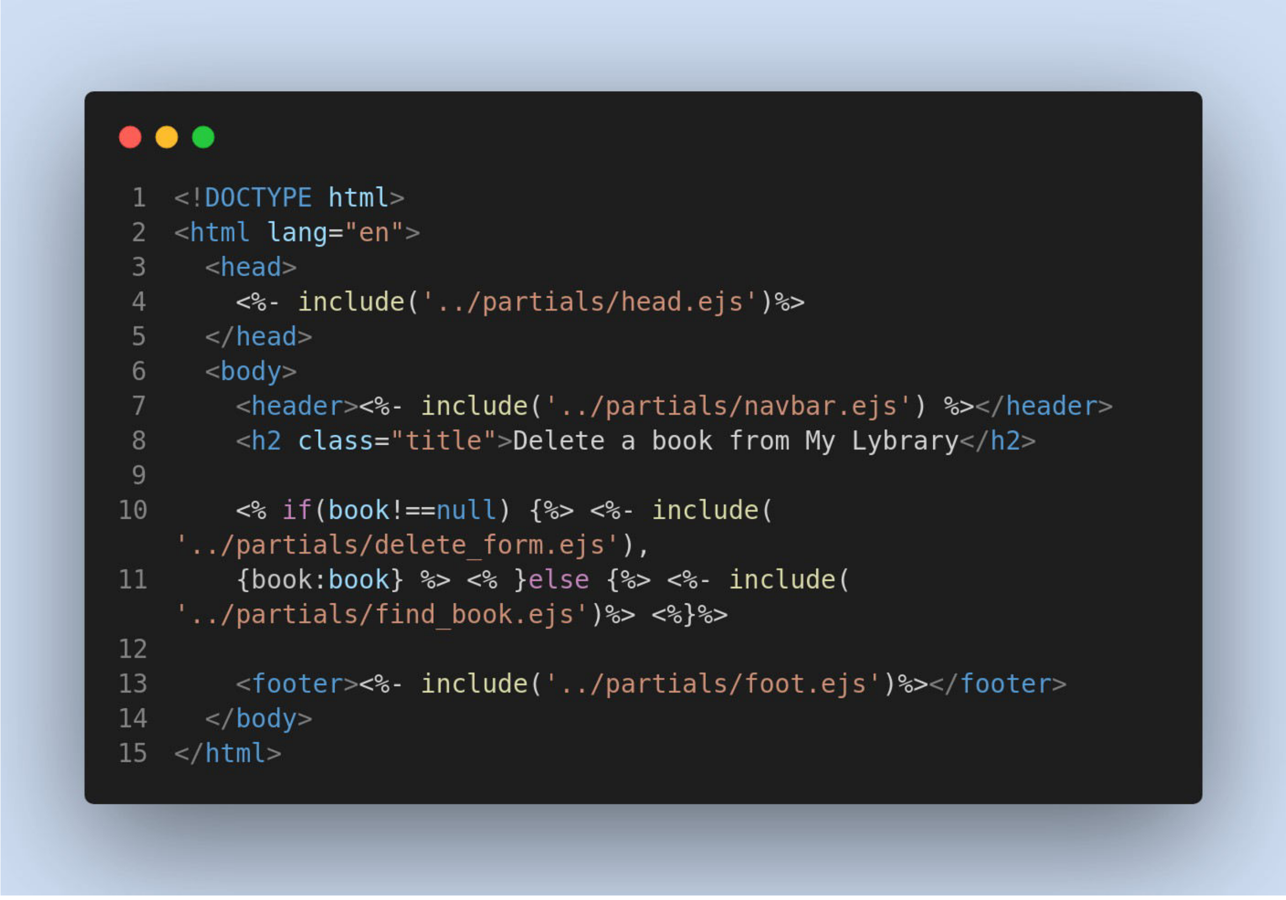
```
<!DOCTYPE html>
<html lang="en">
<head>
    <%- include('../partials/head.ejs')%>
</head>
<body>
    <header>
        <%- include('../partials/navbar.ejs') %>
    </header>
    <h2 class="title">Add a book to My Lybrary</h2>
    <%- include('../partials/add_form.ejs') %>

    <footer>
    <%- include('../partials/foot.ejs')%>

    </footer>
</body>
</html>
```

Update Book Page:



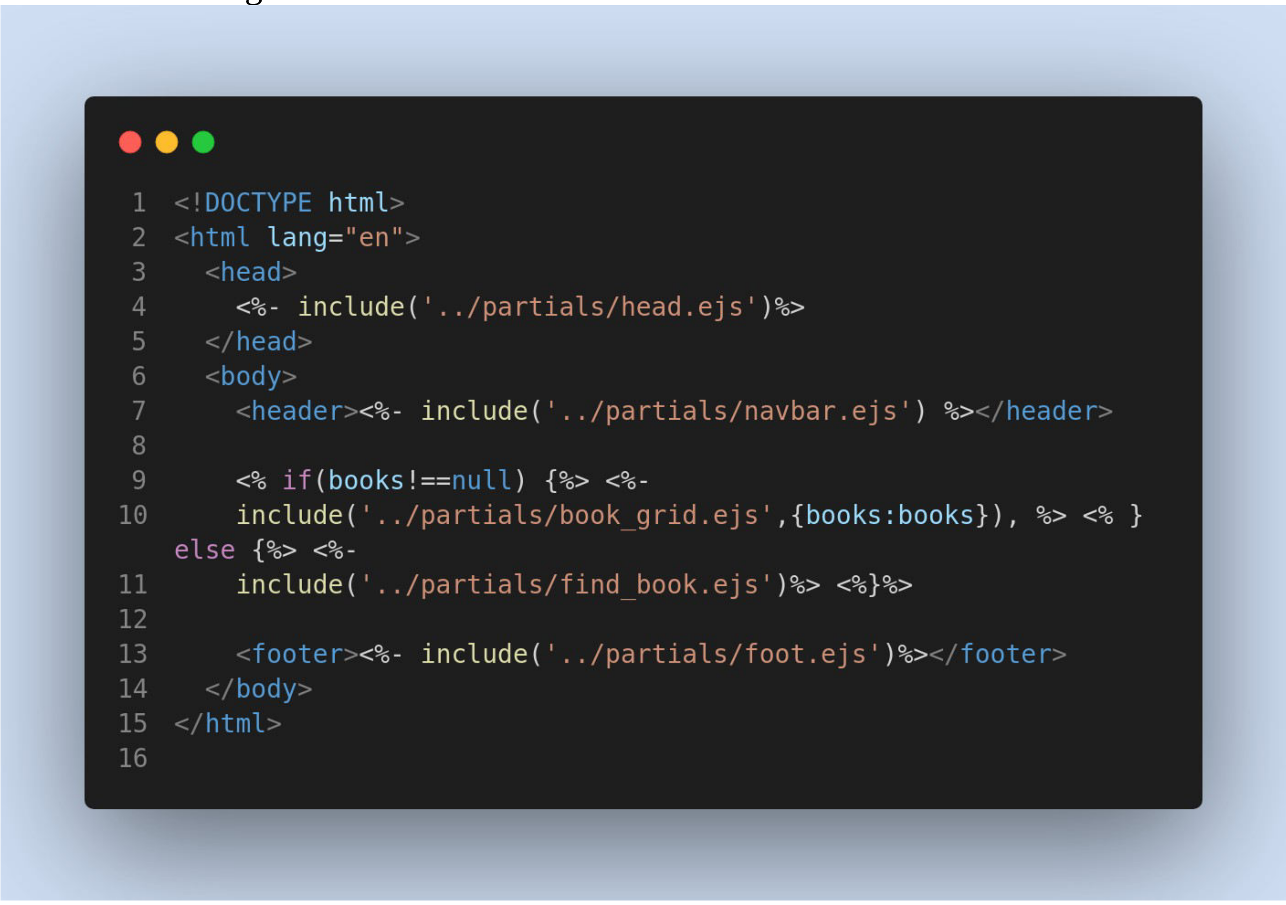
```
<!DOCTYPE html>
<html lang="en">
  <head>
    <%- include('../partials/head.ejs')%>
  </head>
  <body>
    <header><%- include('../partials/navbar.ejs') %></header>
    <h2 class="title">Update a Book in My Lybrary</h2>
    <% if(book!==null) { %>
    <%- include('../partials/update_form.ejs'), {book:book}%> <%
}else{%>
        <%- include('../partials/find_book.ejs')%>
        <%}%>

    <footer><%- include('../partials/foot.ejs')%></footer>
  </body>
</html>
```

Delete Book Page:



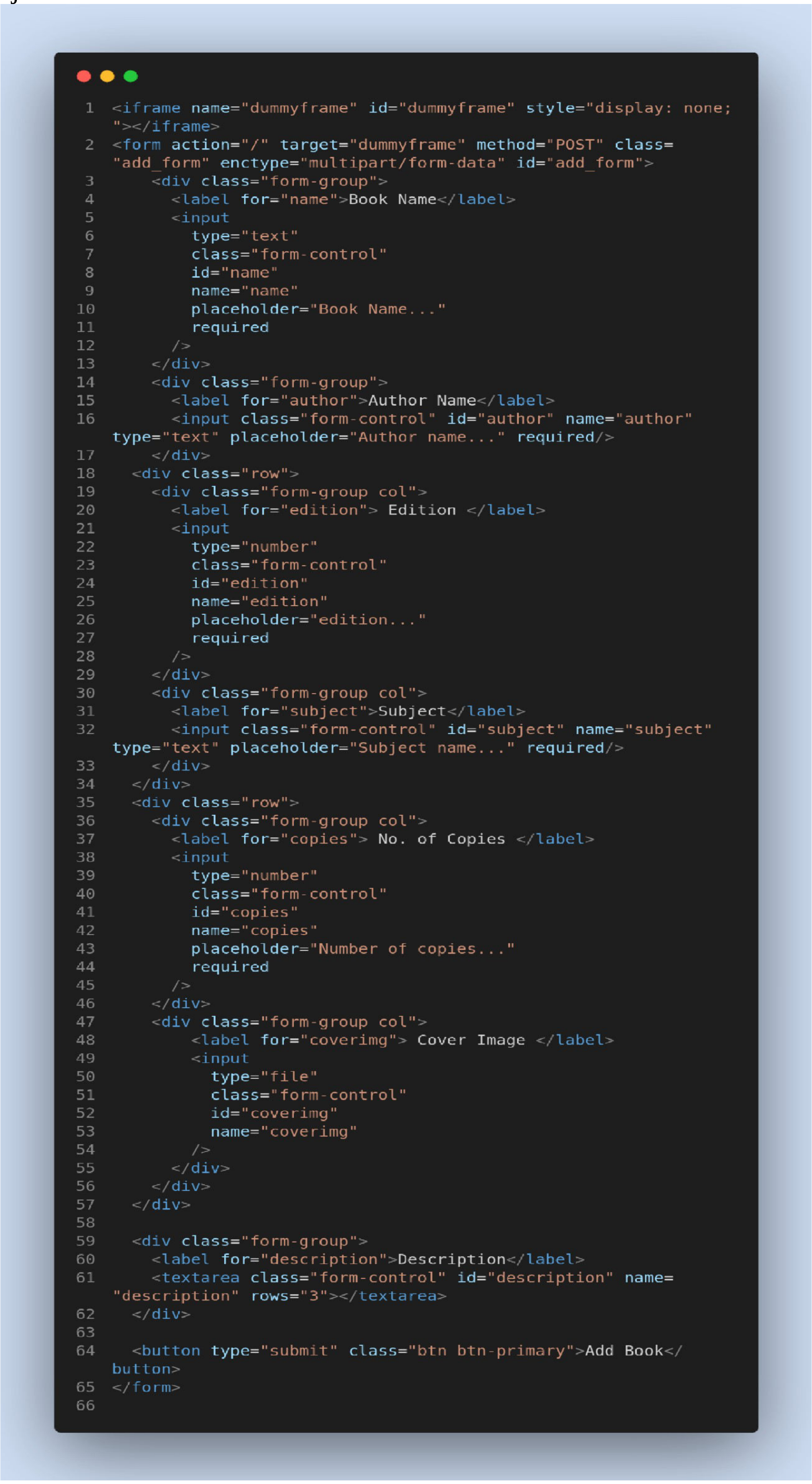
```
<!DOCTYPE html>
<html lang="en">
  <head>
    <%- include('../partials/head.ejs')%>
  </head>
  <body>
    <header><%- include('../partials/navbar.ejs') %></header>
    <h2 class="title">Delete a book from My Lybrary</h2>

    <% if(book!==null) {%> <%- include('../partials/delete_form.ejs'),
    {book:book} %> <% }else {%> <%- include('../partials/find_book.ejs')%> <%}%>

    <footer><%- include('../partials/foot.ejs')%></footer>
  </body>
</html>
```

Search Result Page:



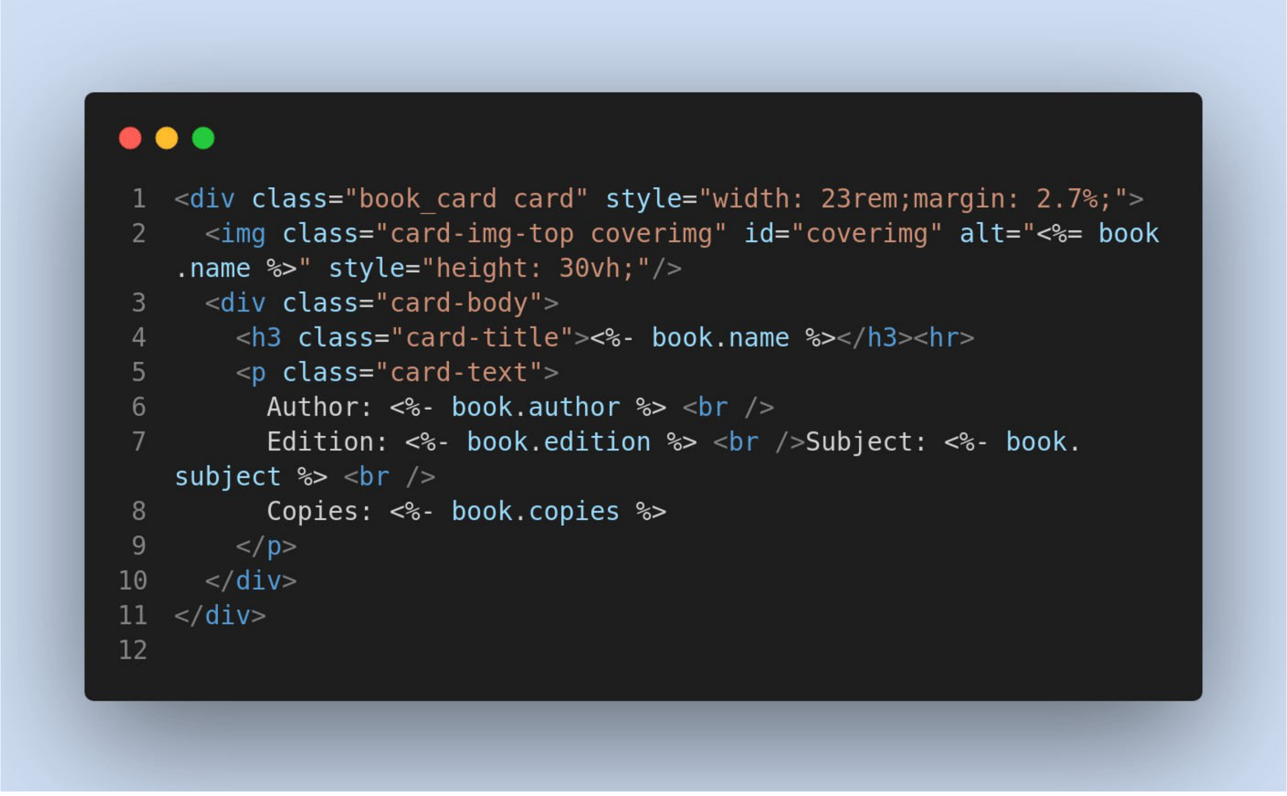
```
<!DOCTYPE html>
<html lang="en">
  <head>
    <%- include('../partials/head.ejs')%>
  </head>
  <body>
    <header><%- include('../partials/navbar.ejs') %></header>

    <% if(books!==null) {%> <%-
    include('../partials/book_grid.ejs',{books:books}), %> <% } else {%> <%-
    include('../partials/find_book.ejs')%> <%}%>

    <footer><%- include('../partials/foot.ejs')%></footer>
  </body>
</html>

```

Various Partial ejs files used:
add_form.js



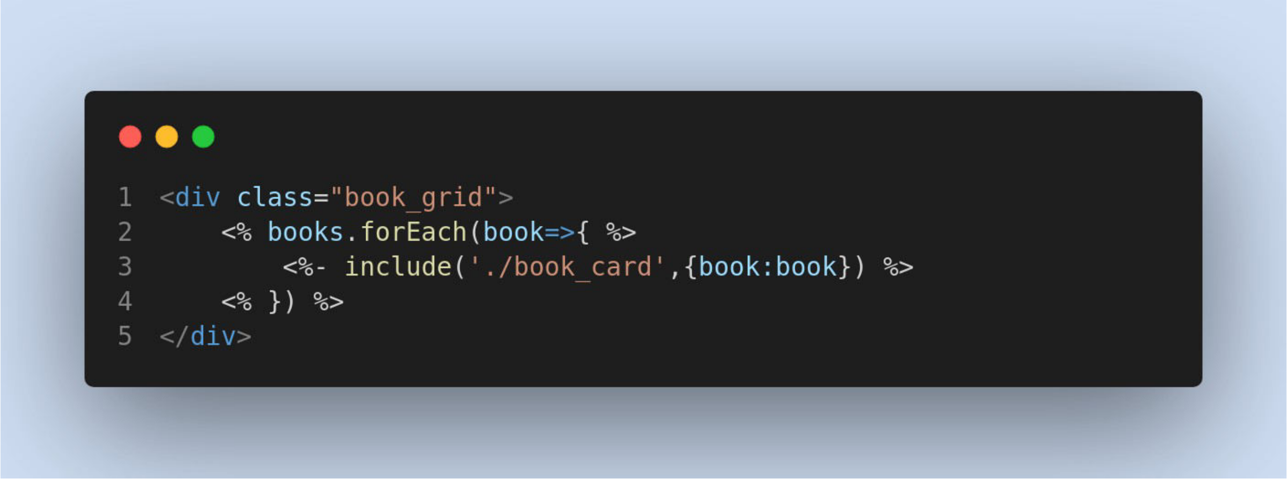
```
<iframe name="dummyframe" id="dummyframe" style="display: none;"></iframe>
<form action="/" target="dummyframe" method="POST" class="add_form" enctype="multipart/form-data" id="add_form">
    <div class="form-group">
      <label for="name">Book Name</label>
      <input
        type="text"
        class="form-control"
        id="name"
        name="name"
        placeholder="Book Name..."
        required
      />
    </div>
    <div class="form-group">
      <label for="author">Author Name</label>
      <input class="form-control" id="author" name="author" type="text" placeholder="Author name..." required/>
    </div>
  <div class="row">
    <div class="form-group col">
      <label for="edition"> Edition </label>
      <input
        type="number"
        class="form-control"
        id="edition"
        name="edition"
        placeholder="edition..."
        required
      />
    </div>
    <div class="form-group col">
      <label for="subject">Subject</label>
      <input class="form-control" id="subject" name="subject" type="text" placeholder="Subject name..." required/>
    </div>
  </div>
  <div class="row">
    <div class="form-group col">
      <label for="copies"> No. of Copies </label>
      <input
        type="number"
        class="form-control"
        id="copies"
        name="copies"
        placeholder="Number of copies..."
        required
      />
    </div>
    <div class="form-group col">
        <label for="coverimg"> Cover Image </label>
        <input
          type="file"
          class="form-control"
          id="coverimg"
          name="coverimg"
        />
      </div>
    </div>
  </div>

  <div class="form-group">
    <label for="description">Description</label>
    <textarea class="form-control" id="description" name="description" rows="3"></textarea>
  </div>

  <button type="submit" class="btn btn-primary">Add Book</button>
</form>

```

book_card.js



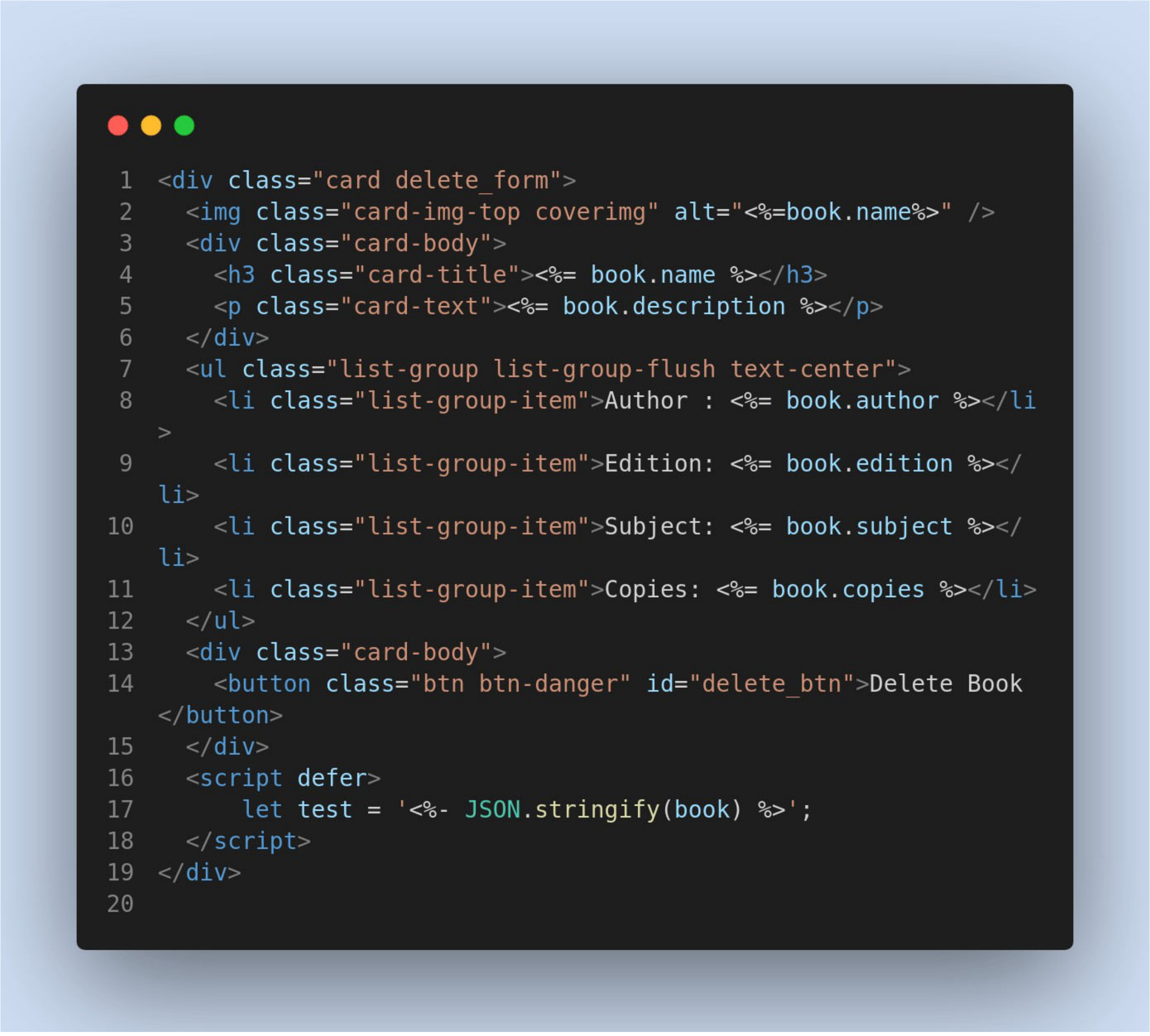
```
<div class="book_card card" style="width: 23rem;margin: 2.7%;">
  <img class="card-img-top coverimg" id="coverimg" alt="<%= book.name %>" style="height: 30vh;"/>
  <div class="card-body">
    <h3 class="card-title"><%- book.name %></h3><hr>
    <p class="card-text">
      Author: <%- book.author %> <br />
      Edition: <%- book.edition %> <br />Subject: <%- book.subject %> <br />
      Copies: <%- book.copies %>
    </p>
  </div>
</div>

```

book_grid.js



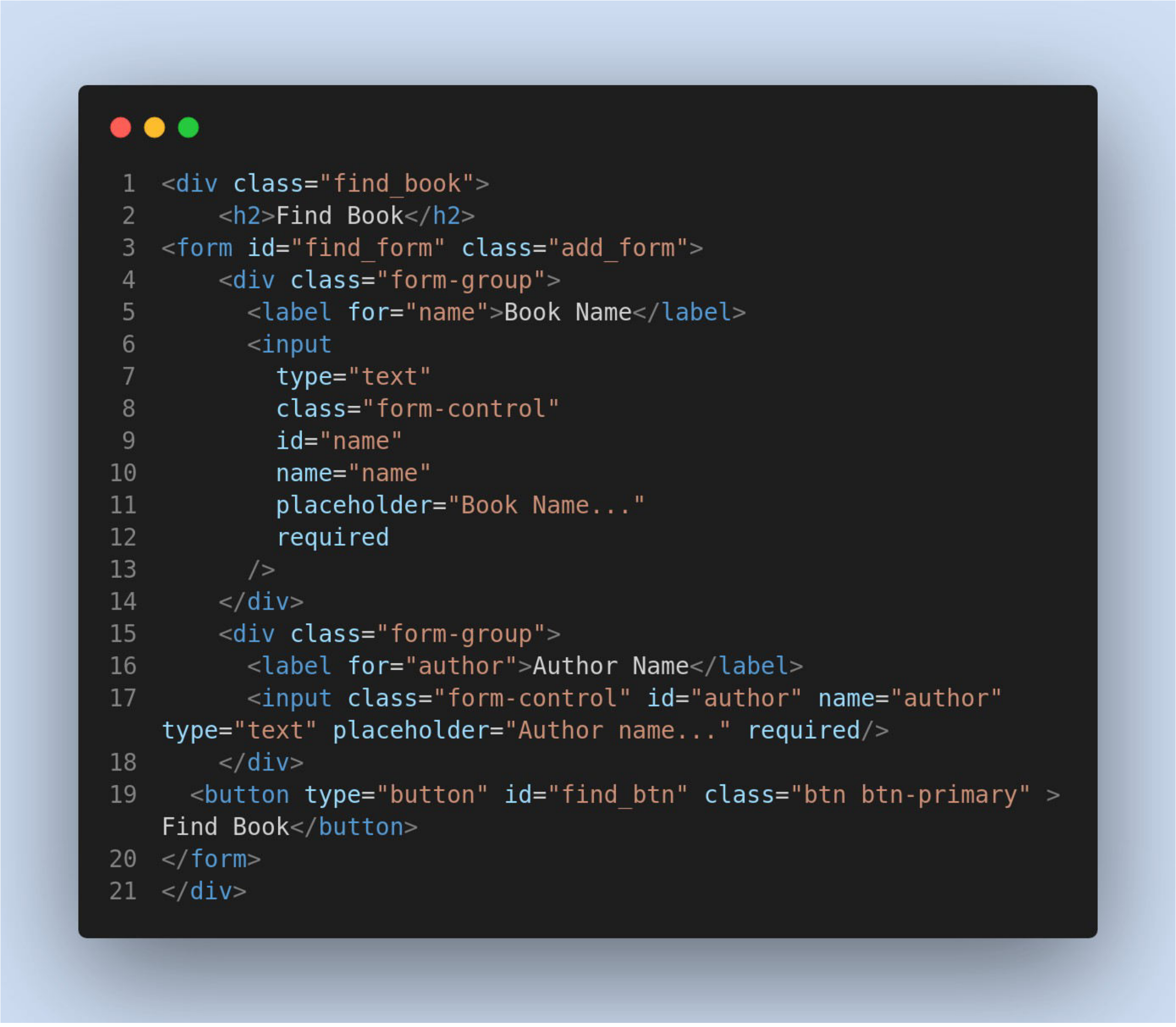
```
<div class="book_grid">
    <% books.forEach(book=>{ %>
        <%- include('./book_card',{book:book}) %>
    <% }) %>
</div>
```

delete_form.js



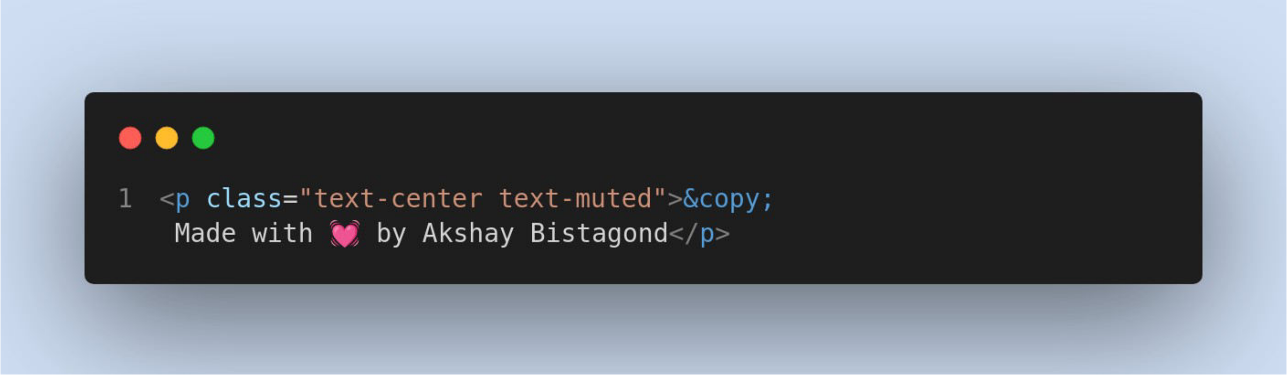
```
<div class="card delete_form">
  <img class="card-img-top coverimg" alt="<%=book.name%>" />
  <div class="card-body">
    <h3 class="card-title"><%= book.name %></h3>
    <p class="card-text"><%= book.description %></p>
  </div>
  <ul class="list-group list-group-flush text-center">
    <li class="list-group-item">Author : <%= book.author %></li>
    <li class="list-group-item">Edition: <%= book.edition %></li>
    <li class="list-group-item">Subject: <%= book.subject %></li>
    <li class="list-group-item">Copies: <%= book.copies %></li>
  </ul>
  <div class="card-body">
    <button class="btn btn-danger" id="delete_btn">Delete Book</button>
  </div>
  <script defer>
      let test = '<%- JSON.stringify(book) %>';
  </script>
</div>

```

find_book.js



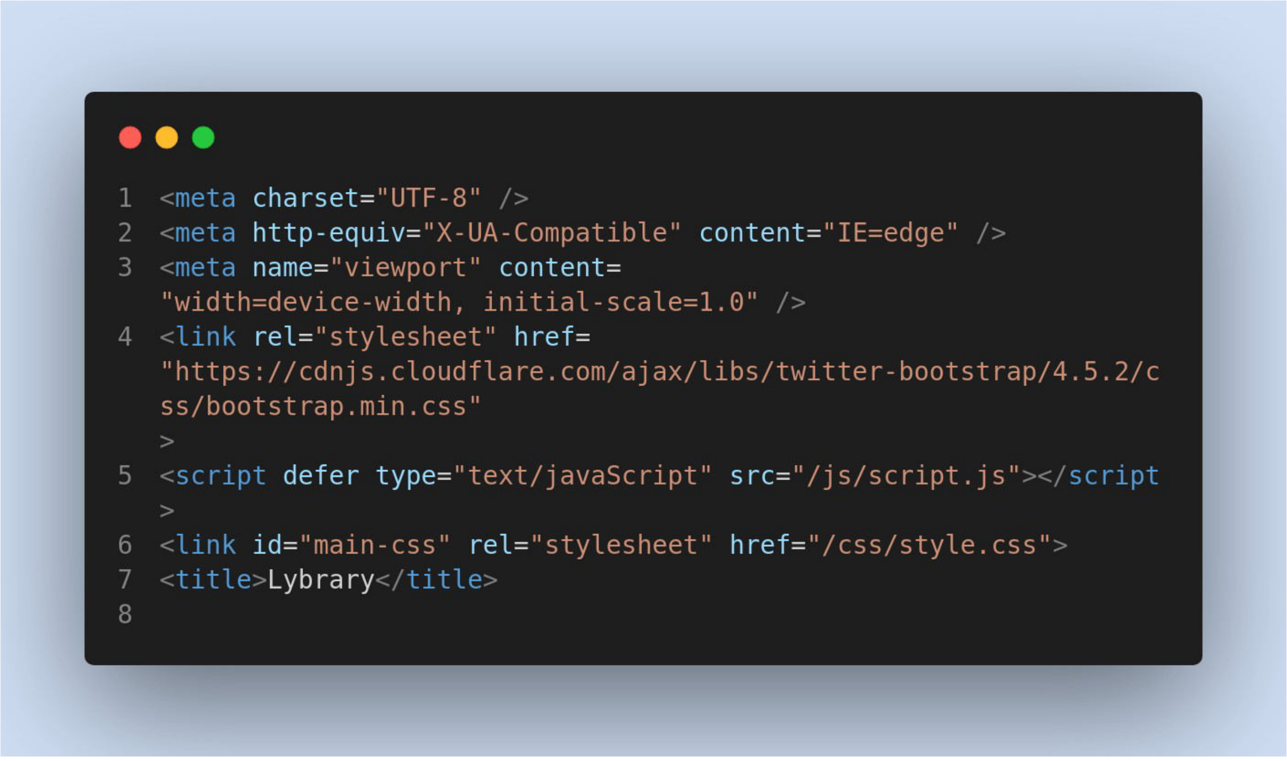
```html
<div class="find_book">
    <h2>Find Book</h2>
<form id="find_form" class="add_form">
    <div class="form-group">
      <label for="name">Book Name</label>
      <input
        type="text"
        class="form-control"
        id="name"
        name="name"
        placeholder="Book Name..."
        required
      />
    </div>
    <div class="form-group">
      <label for="author">Author Name</label>
      <input class="form-control" id="author" name="author" type="text" placeholder="Author name..." required/>
    </div>
  <button type="button" id="find_btn" class="btn btn-primary" > Find Book</button>
</form>
</div>
```

foot.js



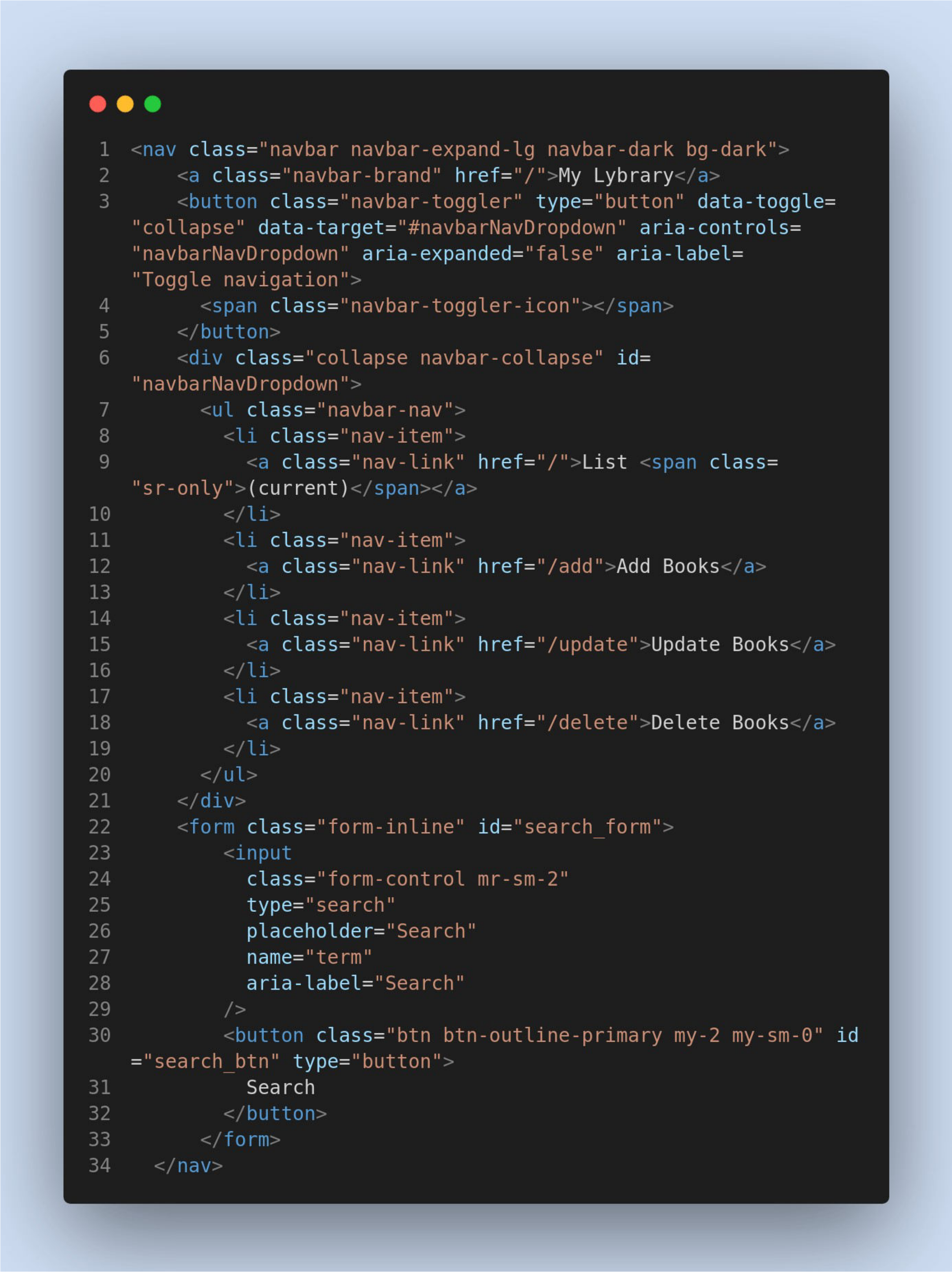
```html
<p class="text-center text-muted">&copy; Made with 💓 by Akshay Bistagond</p>
```

head.js



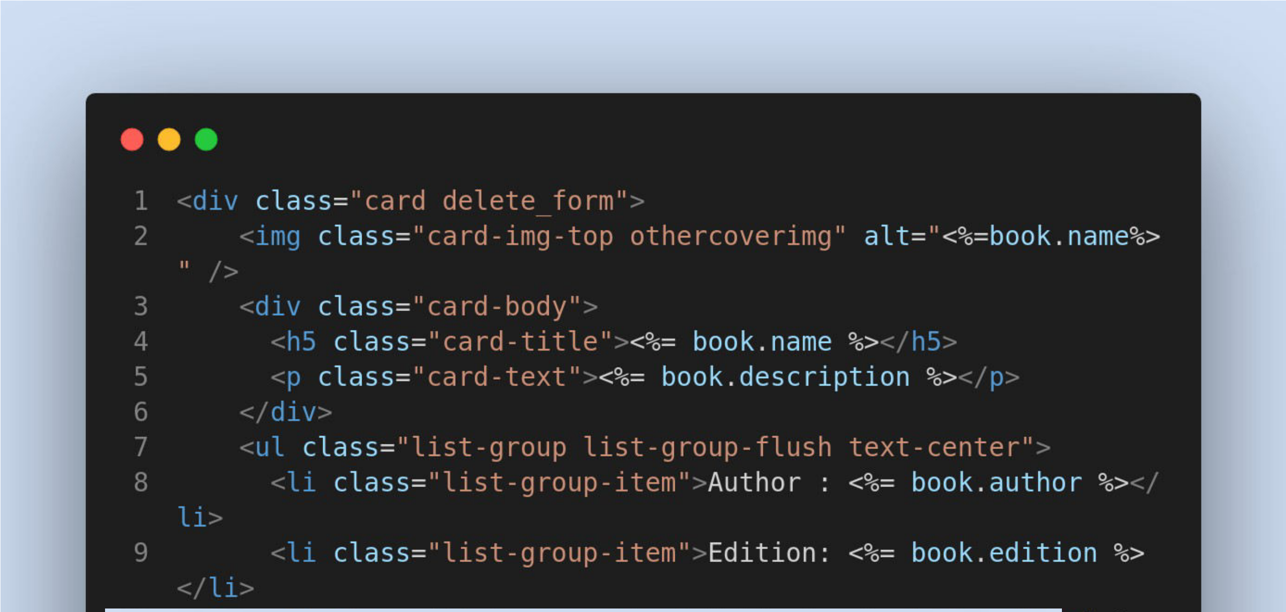
```html
<meta charset="UTF-8" />
<meta http-equiv="X-UA-Compatible" content="IE=edge" />
<meta name="viewport" content="width=device-width, initial-scale=1.0" />
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/twitter-bootstrap/4.5.2/css/bootstrap.min.css">
<script defer type="text/javaScript" src="/js/script.js"></script>
<link id="main-css" rel="stylesheet" href="/css/style.css">
<title>Lybrary</title>

```

navbar.js

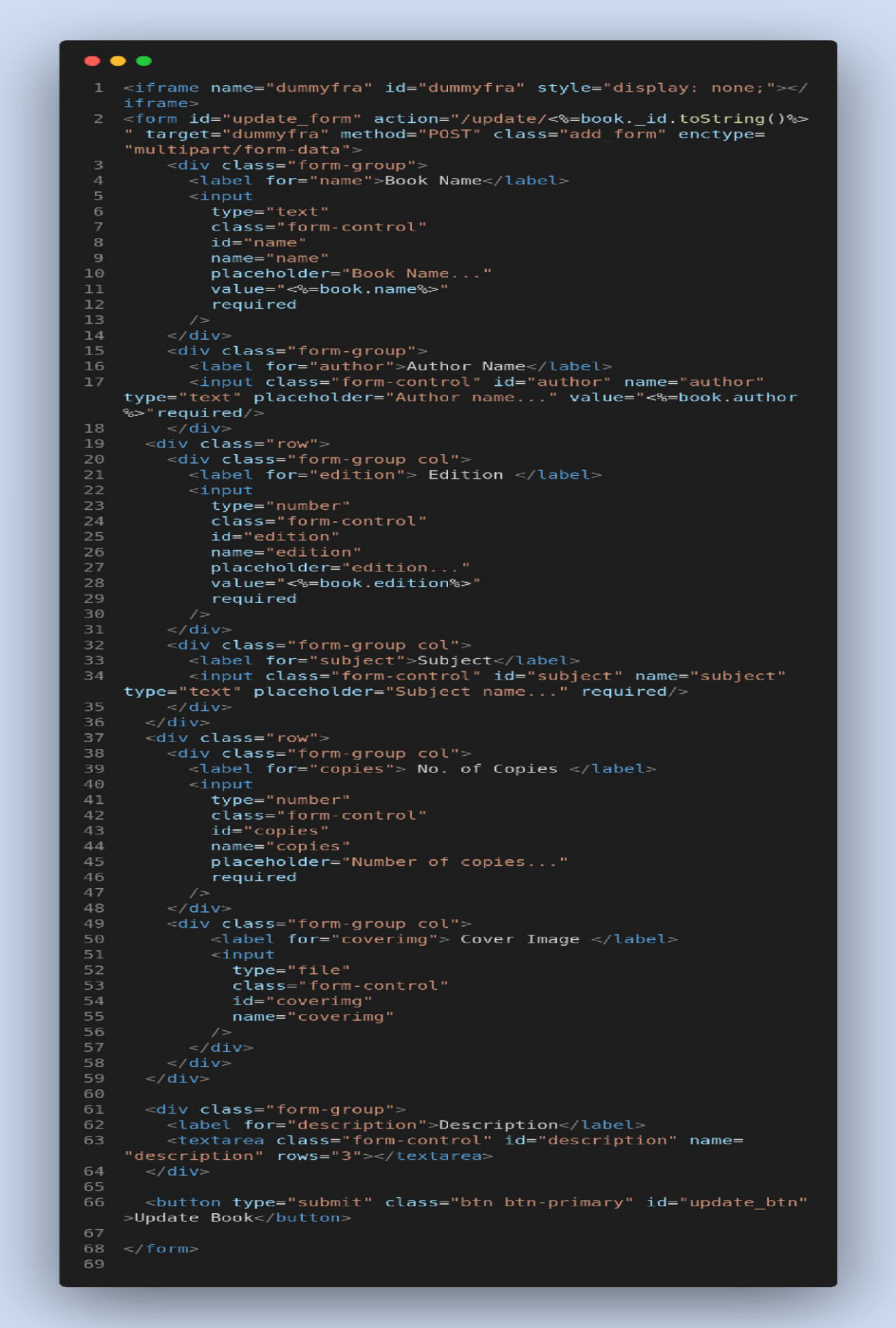


```
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
    <a class="navbar-brand" href="/">My Lybrary</a>
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNavDropdown" aria-controls="navbarNavDropdown" aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarNavDropdown">
      <ul class="navbar-nav">
        <li class="nav-item">
          <a class="nav-link" href="/">List <span class="sr-only">(current)</span></a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/add">Add Books</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/update">Update Books</a>
        </li>
        <li class="nav-item">
          <a class="nav-link" href="/delete">Delete Books</a>
        </li>
      </ul>
    </div>
    <form class="form-inline" id="search_form">
        <input
          class="form-control mr-sm-2"
          type="search"
          placeholder="Search"
          name="term"
          aria-label="Search"
        />
        <button class="btn btn-outline-primary my-2 my-sm-0" id="search_btn" type="button">
          Search
        </button>
      </form>
  </nav>
```

search_form.js



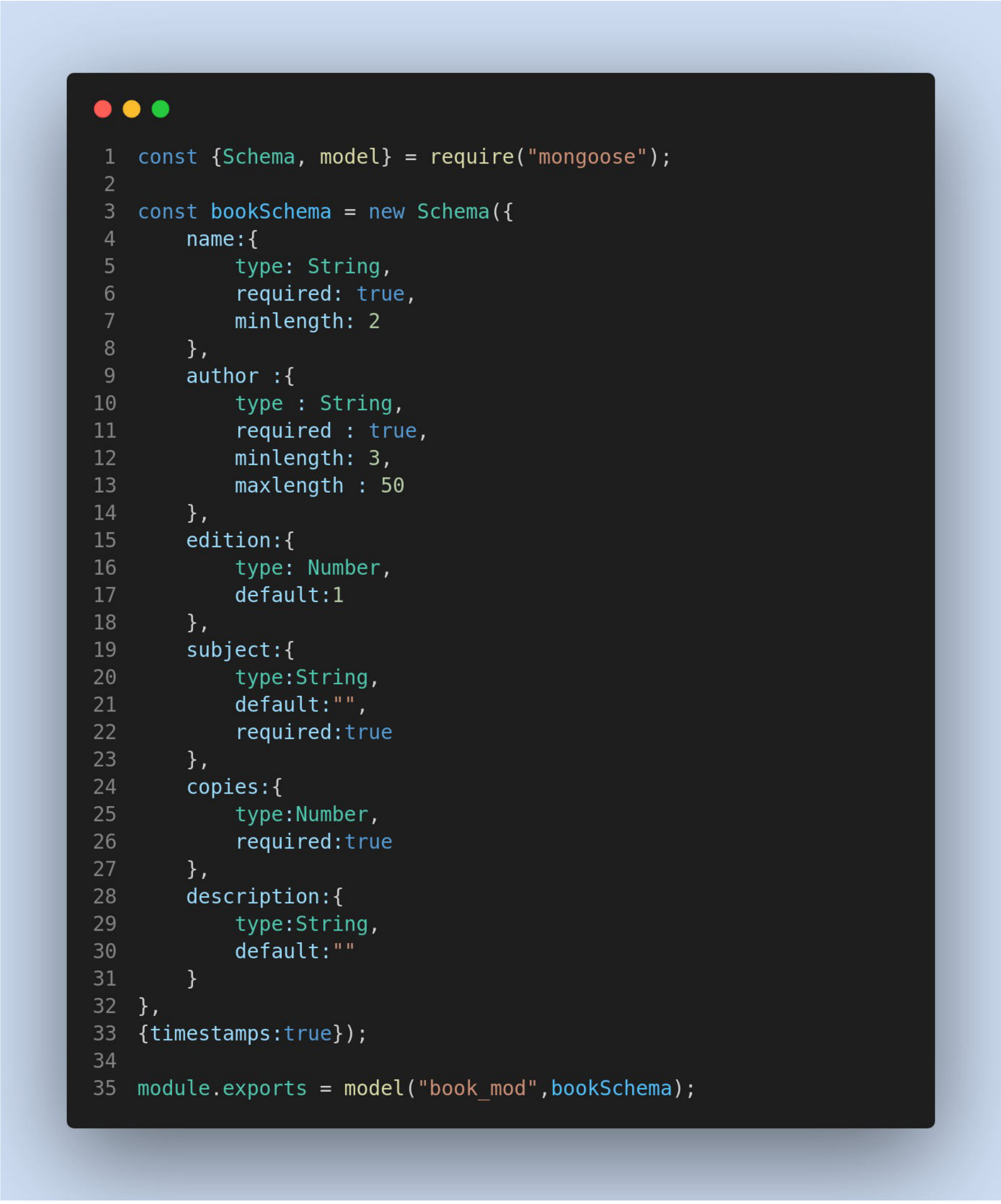
```
<div class="card delete_form">
    <img class="card-img-top othercoverimg" alt="<%=book.name%>" />
    <div class="card-body">
      <h5 class="card-title"><%= book.name %></h5>
      <p class="card-text"><%= book.description %></p>
    </div>
    <ul class="list-group list-group-flush text-center">
      <li class="list-group-item">Author : <%= book.author %></li>
      <li class="list-group-item">Edition: <%= book.edition %></li>
ect %>
s %></
```



```
<iframe name="dummyfra" id="dummyfra" style="display: none;"></iframe>
<form id="update_form" action="/update/<%=book._id.toString()%>" target="dummyfra" method="POST" class="add form" enctype="multipart/form-data">
    <div class="form-group">
      <label for="name">Book Name</label>
      <input
        type="text"
        class="form-control"
        id="name"
        name="name"
        placeholder="Book Name..."
        value="<%=book.name%>"
        required
      />
    </div>
    <div class="form-group">
      <label for="author">Author Name</label>
      <input class="form-control" id="author" name="author" type="text" placeholder="Author name..." value="<%=book.author%>"required/>
    </div>
  <div class="row">
    <div class="form-group col">
      <label for="edition"> Edition </label>
      <input
        type="number"
        class="form-control"
        id="edition"
        name="edition"
        placeholder="edition..."
        value="<%=book.edition%>"
        required
      />
    </div>
    <div class="form-group col">
      <label for="subject">Subject</label>
      <input class="form-control" id="subject" name="subject" type="text" placeholder="Subject name..." required/>
    </div>
  </div>
  <div class="row">
    <div class="form-group col">
      <label for="copies"> No. of Copies </label>
      <input
        type="number"
        class="form-control"
        id="copies"
        name="copies"
        placeholder="Number of copies..."
        required
      />
    </div>
    <div class="form-group col">
        <label for="coverimg"> Cover Image </label>
        <input
          type="file"
          class="form-control"
          id="coverimg"
          name="coverimg"
        />
      </div>
    </div>
  </div>

  <div class="form-group">
    <label for="description">Description</label>
    <textarea class="form-control" id="description" name="description" rows="3"></textarea>
  </div>

  <button type="submit" class="btn btn-primary" id="update_btn">Update Book</button>

</form>

```

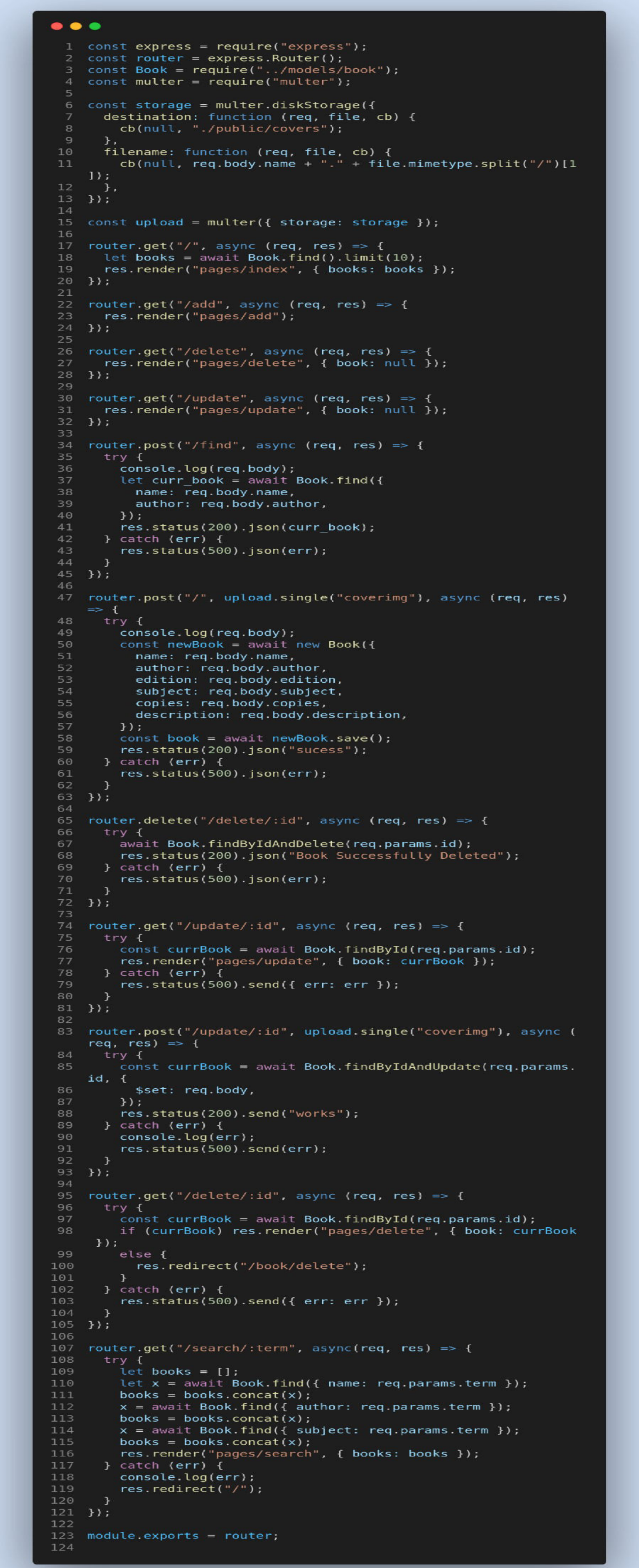
update_form.js

Database Schema to Store book details:-



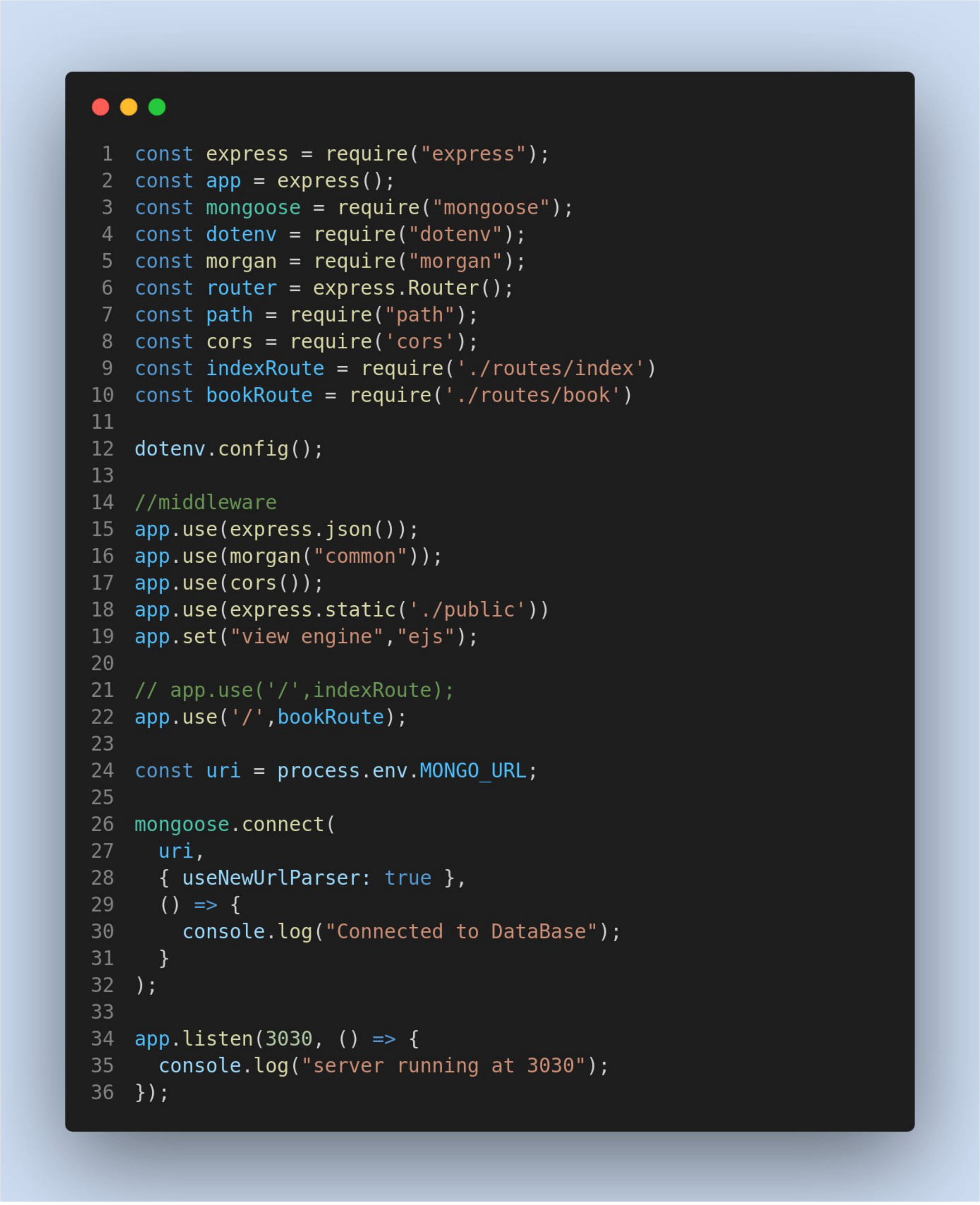
```javascript
const {Schema, model} = require("mongoose");

const bookSchema = new Schema({
    name:{
        type: String,
        required: true,
        minlength: 2
    },
    author :{
        type : String,
        required : true,
        minlength: 3,
        maxlength : 50
    },
    edition:{
        type: Number,
        default:1
    },
    subject:{
        type:String,
        default:"",
        required:true
    },
    copies:{
        type:Number,
        required:true
    },
    description:{
        type:String,
        default:""
    }
},
{timestamps:true});

module.exports = model("book_mod",bookSchema);
```

Router



```
const express = require("express");
const router = express.Router();
const Book = require("../models/book");
const multer = require("multer");

const storage = multer.diskStorage({
  destination: function (req, file, cb) {
    cb(null, "./public/covers");
  },
  filename: function (req, file, cb) {
    cb(null, req.body.name + "." + file.mimetype.split("/")[1]);
  },
});

const upload = multer({ storage: storage });

router.get("/", async (req, res) => {
  let books = await Book.find().limit(10);
  res.render("pages/index", { books: books });
});

router.get("/add", async (req, res) => {
  res.render("pages/add");
});

router.get("/delete", async (req, res) => {
  res.render("pages/delete", { book: null });
});

router.get("/update", async (req, res) => {
  res.render("pages/update", { book: null });
});

router.post("/find", async (req, res) => {
  try {
    console.log(req.body);
    let curr_book = await Book.find({
      name: req.body.name,
      author: req.body.author,
    });
    res.status(200).json(curr_book);
  } catch (err) {
    res.status(500).json(err);
  }
});

router.post("/", upload.single("coverimg"), async (req, res) => {
  try {
    console.log(req.body);
    const newBook = await new Book({
      name: req.body.name,
      author: req.body.author,
      edition: req.body.edition,
      subject: req.body.subject,
      copies: req.body.copies,
      description: req.body.description,
    });
    const book = await newBook.save();
    res.status(200).json("sucess");
  } catch (err) {
    res.status(500).json(err);
  }
});

router.delete("/delete/:id", async (req, res) => {
  try {
    await Book.findByIdAndDelete(req.params.id);
    res.status(200).json("Book Successfully Deleted");
  } catch (err) {
    res.status(500).json(err);
  }
});

router.get("/update/:id", async (req, res) => {
  try {
    const currBook = await Book.findById(req.params.id);
    res.render("pages/update", { book: currBook });
  } catch (err) {
    res.status(500).send({ err: err });
  }
});

router.post("/update/:id", upload.single("coverimg"), async (req, res) => {
  try {
    const currBook = await Book.findByIdAndUpdate(req.params.id, {
      $set: req.body,
    });
    res.status(200).send("works");
  } catch (err) {
    console.log(err);
    res.status(500).send(err);
  }
});

router.get("/delete/:id", async (req, res) => {
  try {
    const currBook = await Book.findById(req.params.id);
    if (currBook) res.render("pages/delete", { book: currBook });
    else {
      res.redirect("/book/delete");
    }
  } catch (err) {
    res.status(500).send({ err: err });
  }
});

router.get("/search/:term", async(req, res) => {
  try {
    let books = [];
    let x = await Book.find({ name: req.params.term });
    books = books.concat(x);
    x = await Book.find({ author: req.params.term });
    books = books.concat(x);
    x = await Book.find({ subject: req.params.term });
    books = books.concat(x);
    res.render("pages/search", { books: books });
  } catch (err) {
    console.log(err);
    res.redirect("/");
  }
});

module.exports = router;
```

Server.js



```javascript
const express = require("express");
const app = express();
const mongoose = require("mongoose");
const dotenv = require("dotenv");
const morgan = require("morgan");
const router = express.Router();
const path = require("path");
const cors = require('cors');
const indexRoute = require('./routes/index')
const bookRoute = require('./routes/book')

dotenv.config();

//middleware
app.use(express.json());
app.use(morgan("common"));
app.use(cors());
app.use(express.static('./public'))
app.set("view engine","ejs");

// app.use('/',indexRoute);
app.use('/',bookRoute);

const uri = process.env.MONGO_URL;

mongoose.connect(
  uri,
  { useNewUrlParser: true },
  () => {
    console.log("Connected to DataBase");
  }
);

app.listen(3030, () => {
  console.log("server running at 3030");
});
```