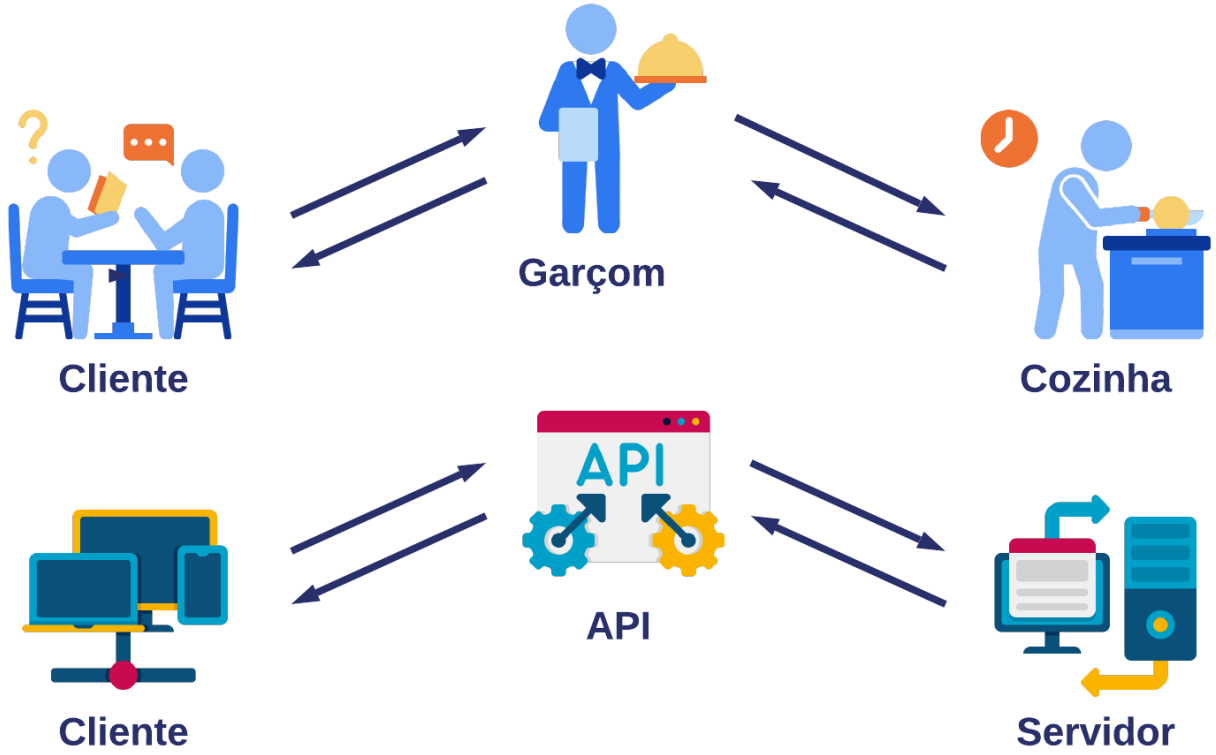
Criação de API com Node.js

Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br

Criação de API com Node.js

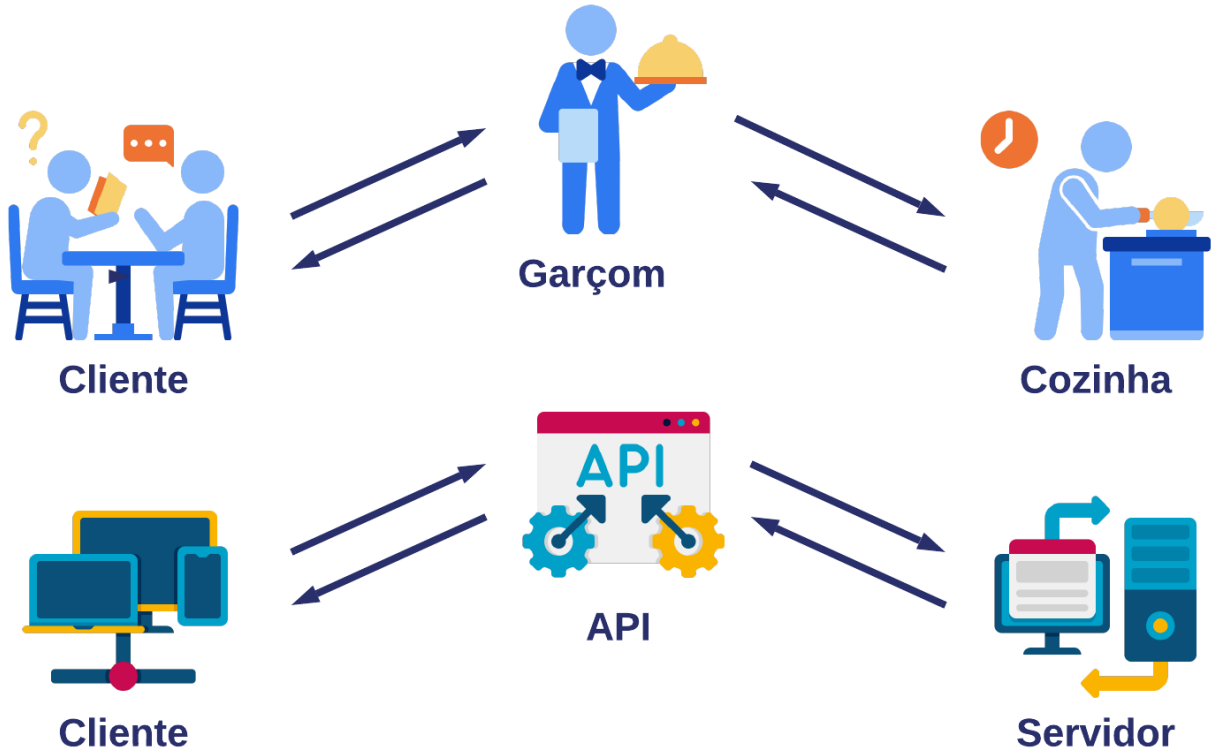
- Uma API é como um garçom em um restaurante atuando como um intermediário entre o cliente e a cozinha.
  - O cliente faz o pedido ao garçom e este repassa à cozinha.
  - A cozinha prepara o pedido para o garçom retirar e entregar ao cliente.
  - APIs funcionam basicamente da mesma forma.



Criação de API com Node.js

O que é uma API ?

- Cliente = Cliente
  - usuário solicitante, navegador, app
- Garçom = API
  - Application Programming Interface
  - Comunicação entre cliente e servidor
- Cozinha = Servidor
  - Processa pedido, e entrega uma resposta



Criação de API com Node.js

- O que é uma API ?
  - **API** (Application Programming Interface) é um conjunto de regras e definições que permite que diferentes softwares “conversem” entre si.
  - Basicamente, uma API define como você pode pedir informações ou enviar comandos para um sistema, e como esse sistema vai responder.

Criação de API com Node.js

API em Node.js

- O objetivo da aula será criar uma **API em Node.js**,
- Portanto vamos criar um **servidor** que expõe uma API,
  - Serão criados um conjunto de endpoints (URLs) que aceitam requisições HTTP:
    - GET
    - POST
    - PUT
    - DELETE
  - Estes endpoints (URLs) respondem com dados no formato JSON

Criação de API com Node.js

- O que é um JSON ?
  - **JSON** significa **JavaScript Object Notation**
    - Notação de Objeto JavaScript
  - É um formato leve e fácil de ler para representar dados estruturados.
    - É uma forma padrão de trocar informações entre sistemas (como entre um servidor e um cliente) usando texto

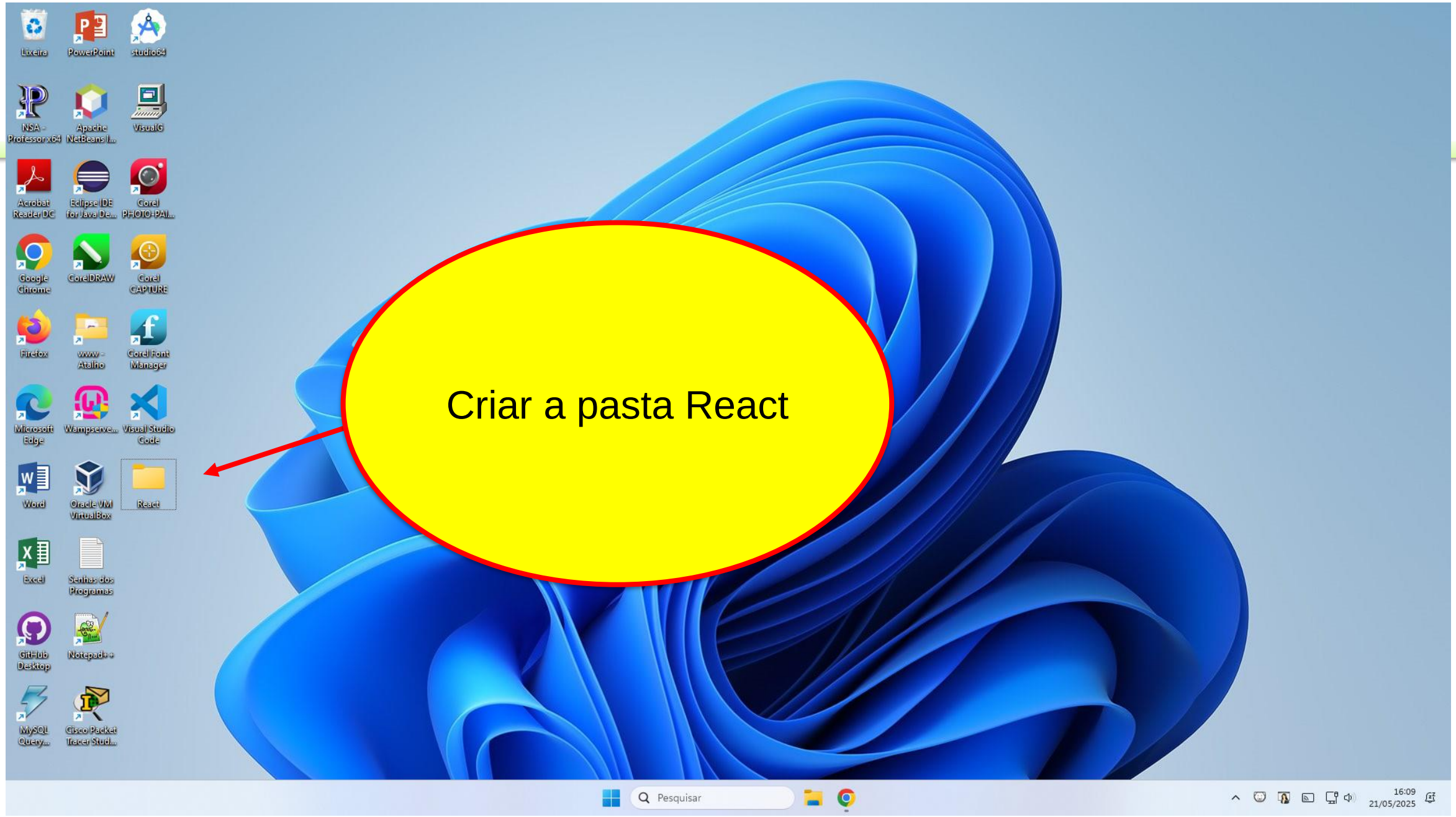
Criação de API com Node.js

- O que é um JSON ?
  - **JSON** significa **JavaScript Object Notation**
    - Notação de Objeto JavaScript

```json
{
  "nome": "Guilherme",
  "idade": 46,
  "email": "guilherme.souza@etec.sp.gov.br",
  "interesses": ["programação", "música", "carros"],
  "ativo": true
}
```

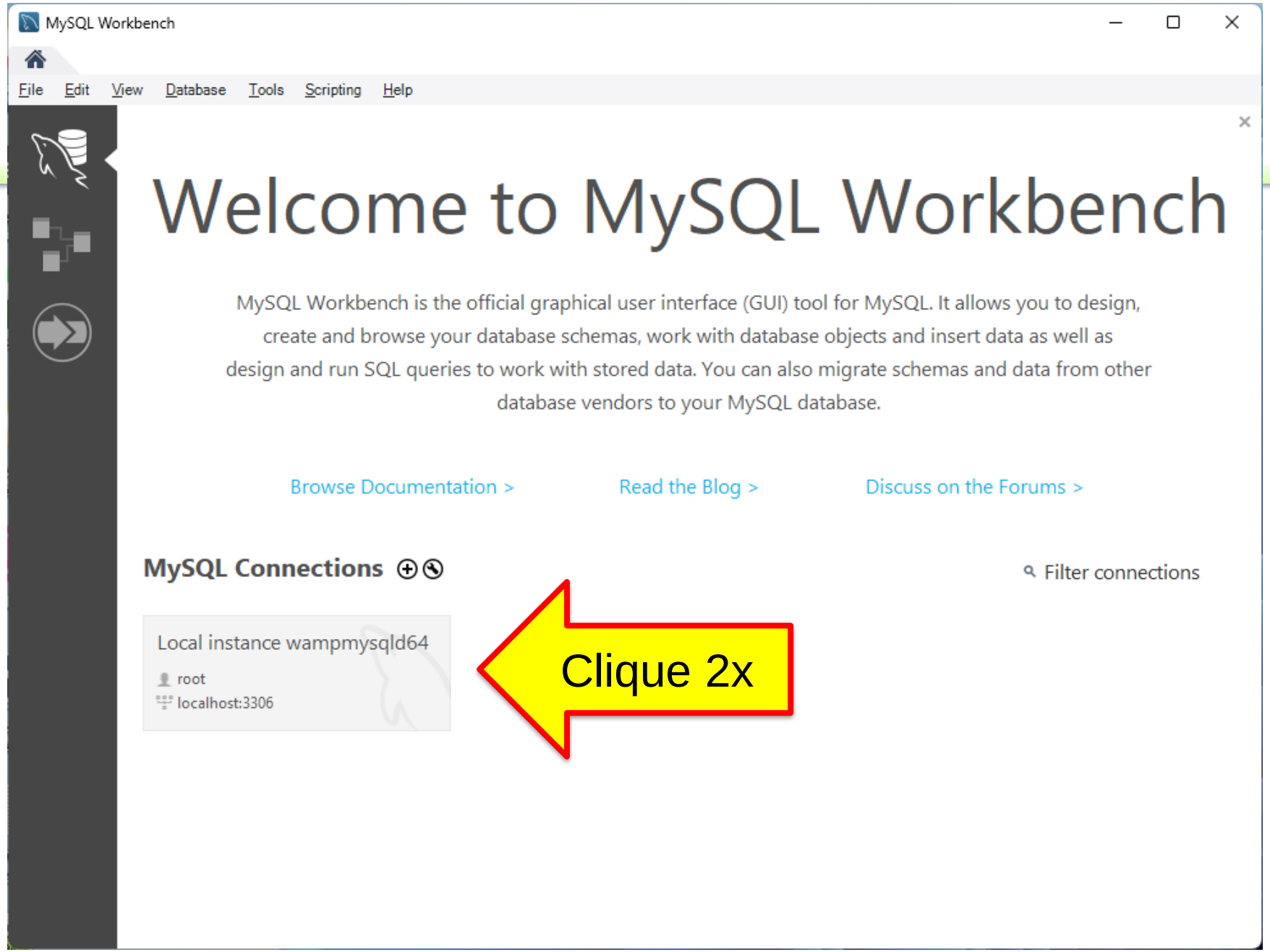
Criando a pasta “servidor”

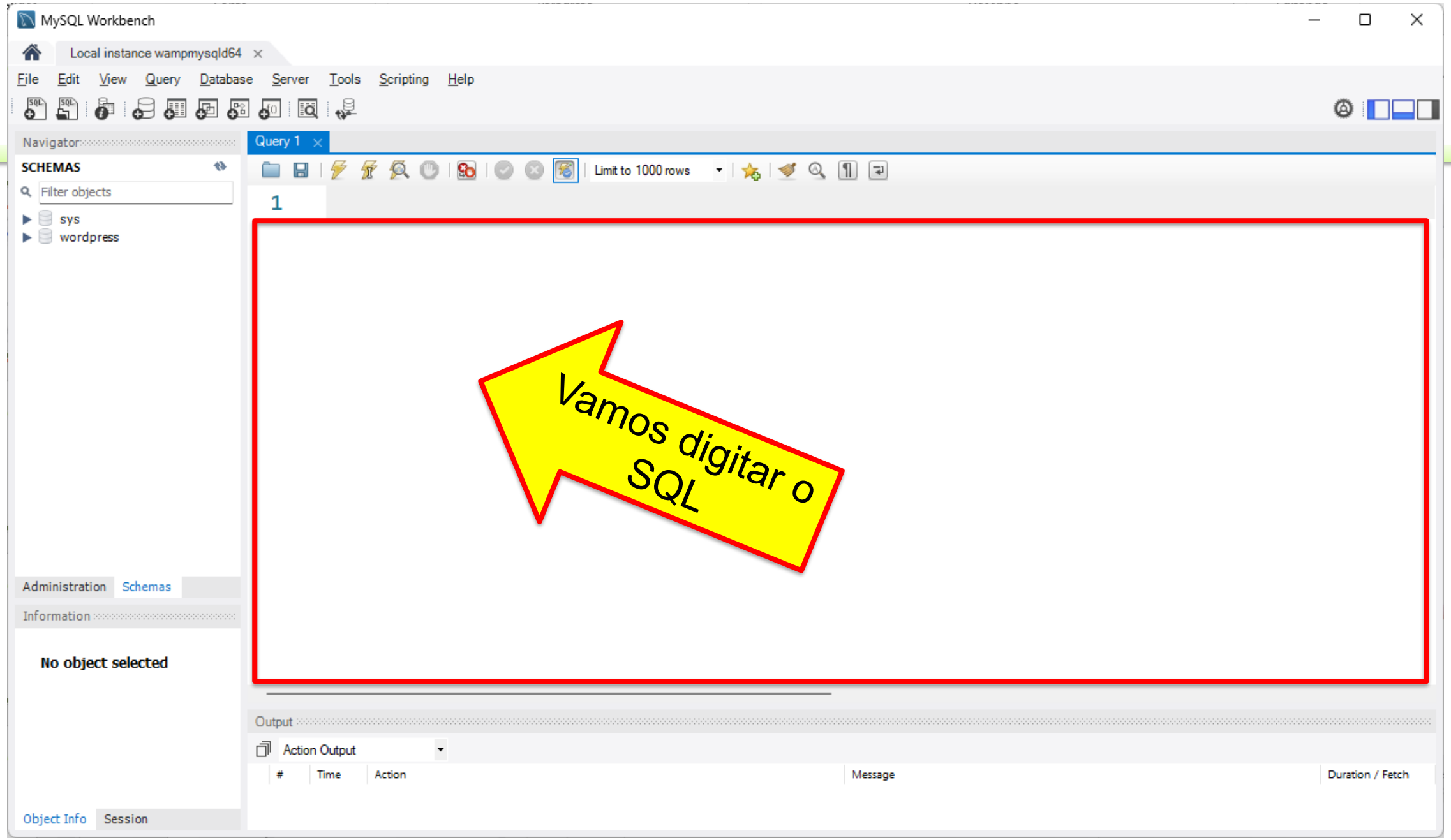
Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br


Lixeira
PowerPoint
studio64
NSA - Professor x64
Apache NetBeans I...
VisualG
Acrobat Reader DC
Eclipse IDE for Java De...
Corel PHOTO-PAI...
Google Chrome
CorelDRAW
Corel CAPTURE
Firefox
www - Atalho
Corel Font Manager
Microsoft Edge
Wampserve...
Visual Studio Code
Word
Oracle VM VirtualBox
React
Excel
Senhas dos Programas
GitHub Desktop
Notepad++
MySQL Query...
Cisco Packet Tracer Stud...
Criar a pasta React
Pesquisar
16:09
21/05/2025

1 - Criação do Banco de Dados no MySQL

Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br


MySQL Workbench
File Edit View Database Tools Scripting Help
Welcome to MySQL Workbench
MySQL Workbench is the official graphical user interface (GUI) tool for MySQL. It allows you to design, create and browse your database schemas, work with database objects and insert data as well as design and run SQL queries to work with stored data. You can also migrate schemas and data from other database vendors to your MySQL database.
Browse Documentation >
Read the Blog >
Discuss on the Forums >
MySQL Connections
Filter connections
Local instance wampmysqld64
root
localhost:3306
Clique 2x

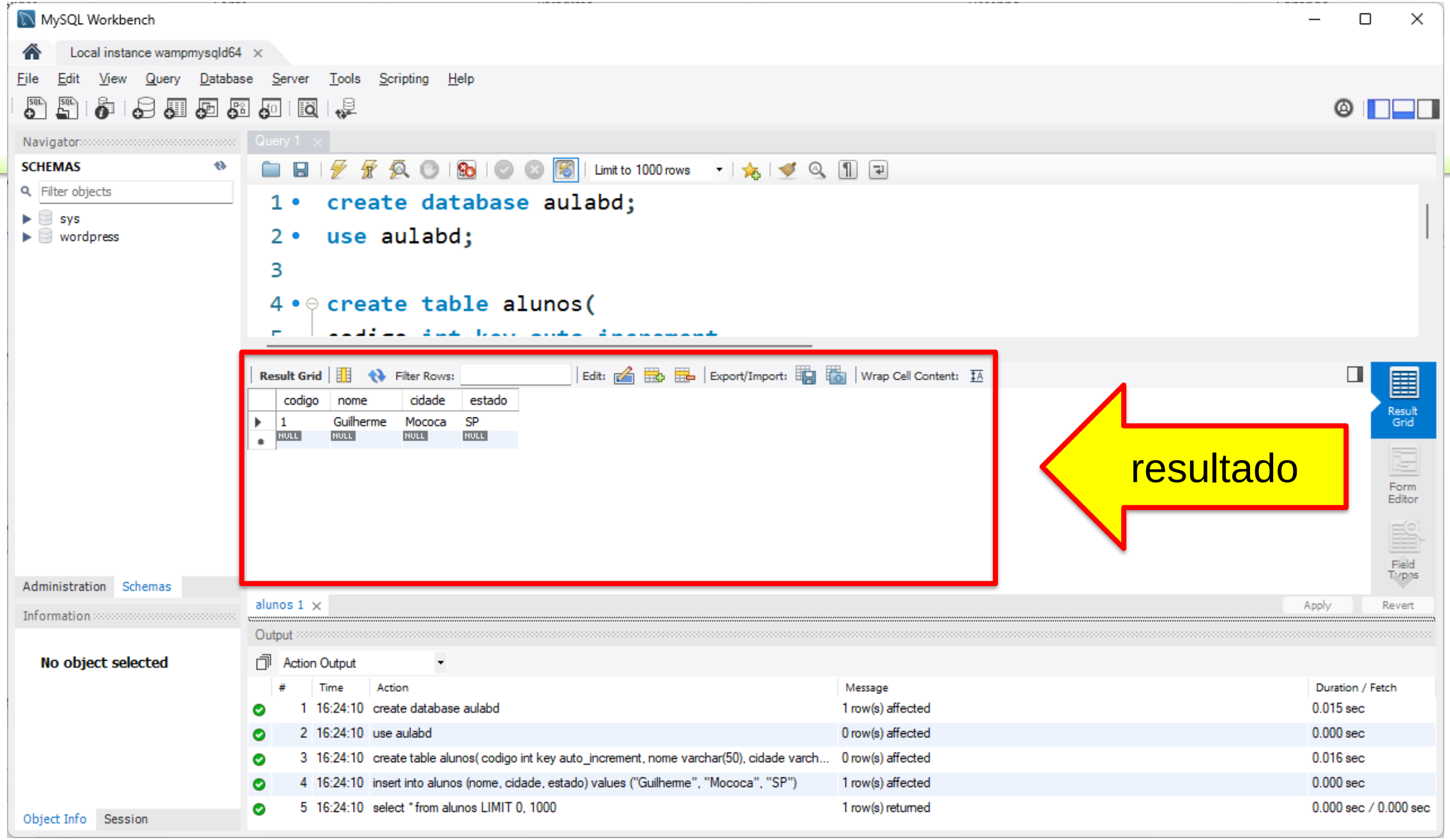

MySQL Workbench
Local instance wampmysqld64
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
sys
wordpress
Query 1
Limit to 1000 rows
1
Vamos digitar o SQL
Administration Schemas
Information
No object selected
Object Info Session
Output
Action Output
Time Action Message Duration / Fetch

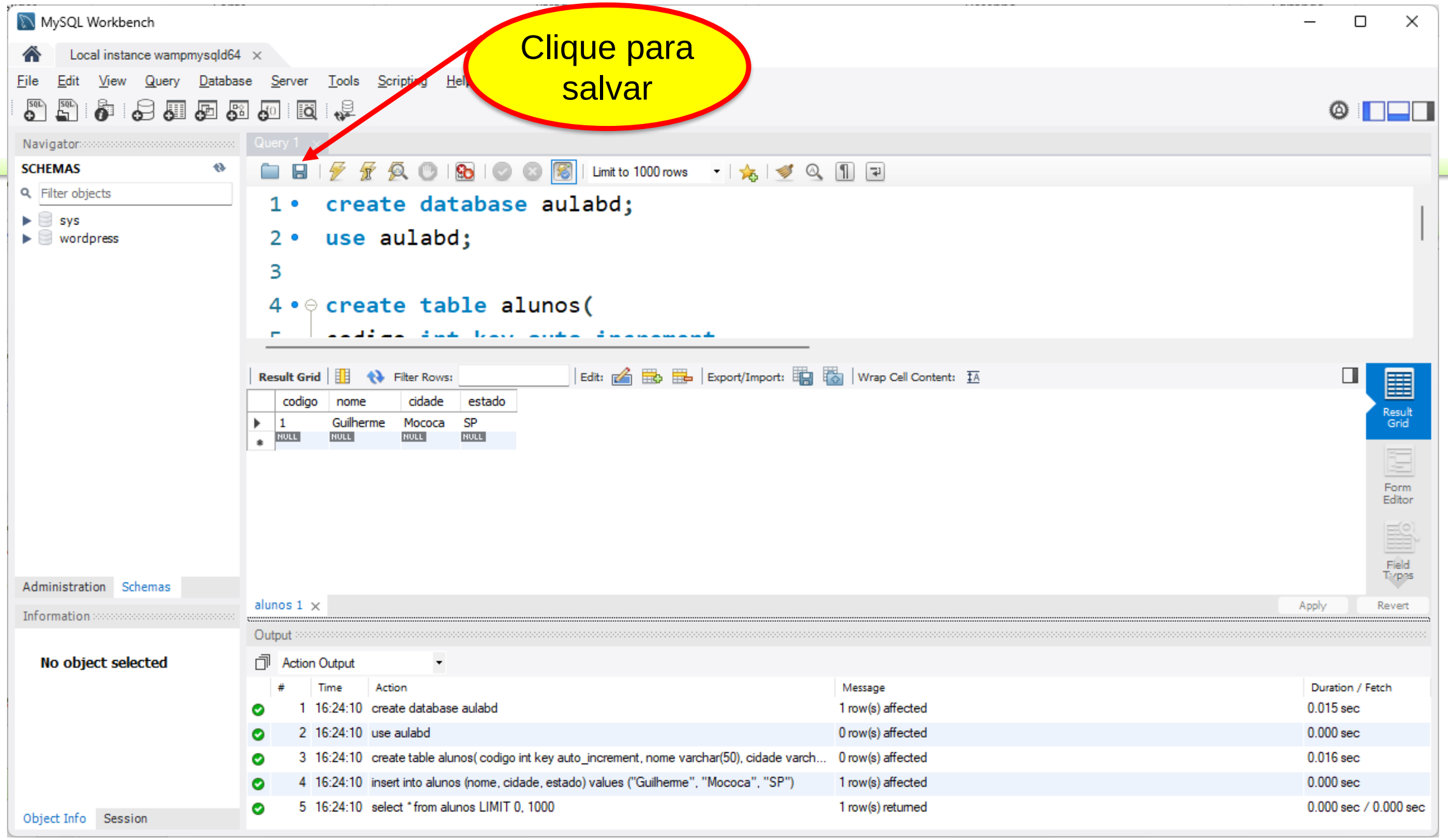
```
1  create database aulabd;
2  use aulabd;
3
4  create table alunos(
5  codigo int key auto_increment,
6  nome varchar(50),
7  cidade varchar(50),
8  estado varchar(2)
9  );
```

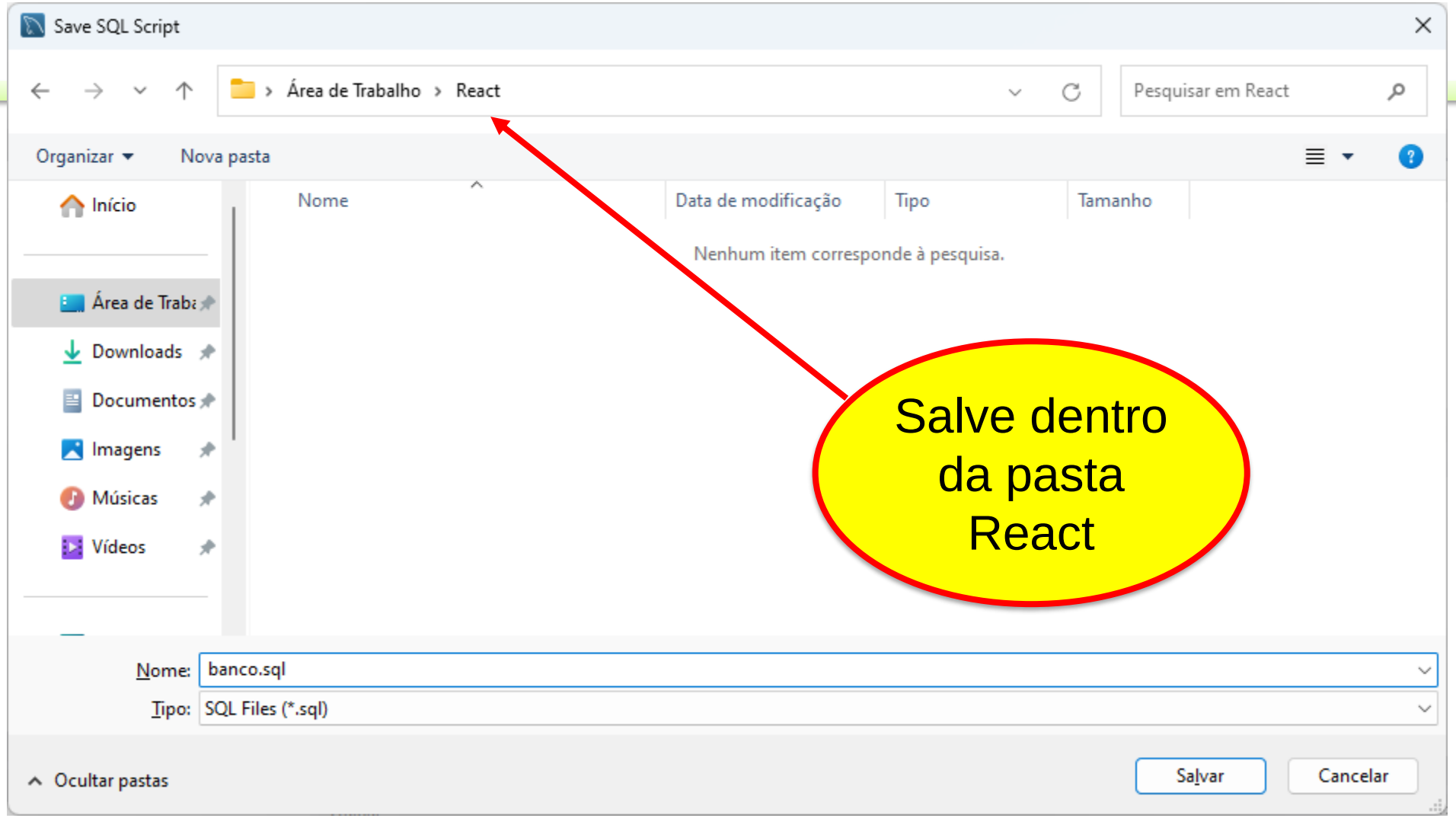
Digite

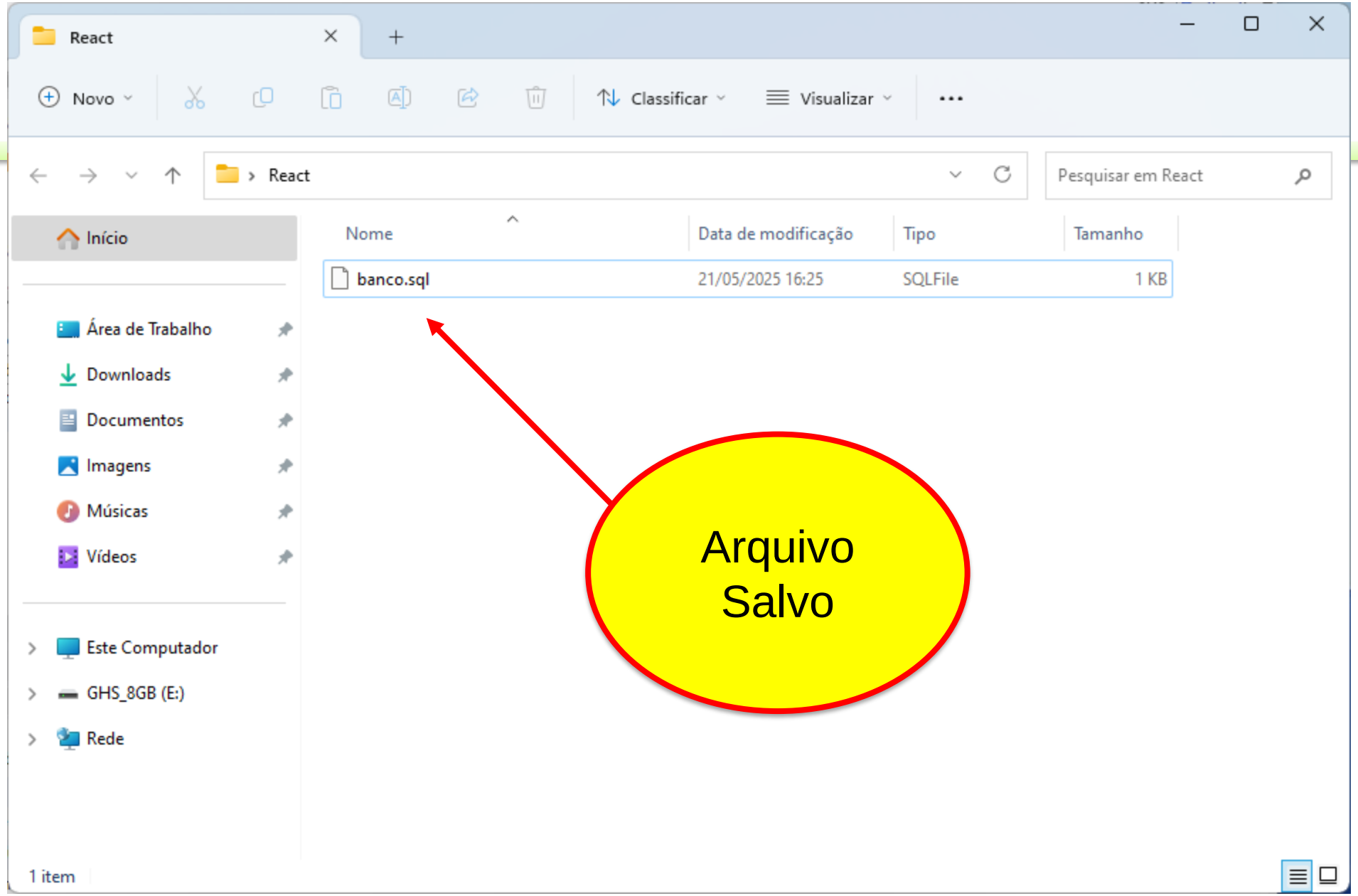
Digite

```sql
create database aulabd;
use aulabd;

create table alunos(
codigo int key auto_increment,
nome varchar(50),
cidade varchar(50),
estado varchar(2)
);

insert into alunos (nome, cidade, estado) values ("Guilherme", "Mococa", "SP");

select * from alunos;
```


MySQL Workbench
Local instance wampmysqld64
File Edit View Query Database Server Tools Scripting Help
Navigator
SCHEMAS
Filter objects
sys
wordpress
Query 1
Limit to 1000 rows
1 • create database aulabd;
2 • use aulabd;
3
4 • create table alunos(
Result Grid
Filter Rows:
Edit:
Export/Import:
Wrap Cell Content:
codigo nome cidade estado
1 Guilherme Mococa SP
NULL NULL NULL NULL
resultado
Result Grid
Form Editor
Field Types
Administration Schemas
Information
No object selected
Object Info Session
alunos 1
Apply
Revert
Output
Action Output
Time Action Message Duration / Fetch
1 16:24:10 create database aulabd 1 row(s) affected 0.015 sec
2 16:24:10 use aulabd 0 row(s) affected 0.000 sec
3 16:24:10 create table alunos(codigo int key auto_increment, nome varchar(50), cidade varch... 0 row(s) affected 0.016 sec
4 16:24:10 insert into alunos (nome, cidade, estado) values ("Guilherme", "Mococa", "SP") 1 row(s) affected 0.000 sec
5 16:24:10 select * from alunos LIMIT 0, 1000 1 row(s) returned 0.000 sec / 0.000 sec

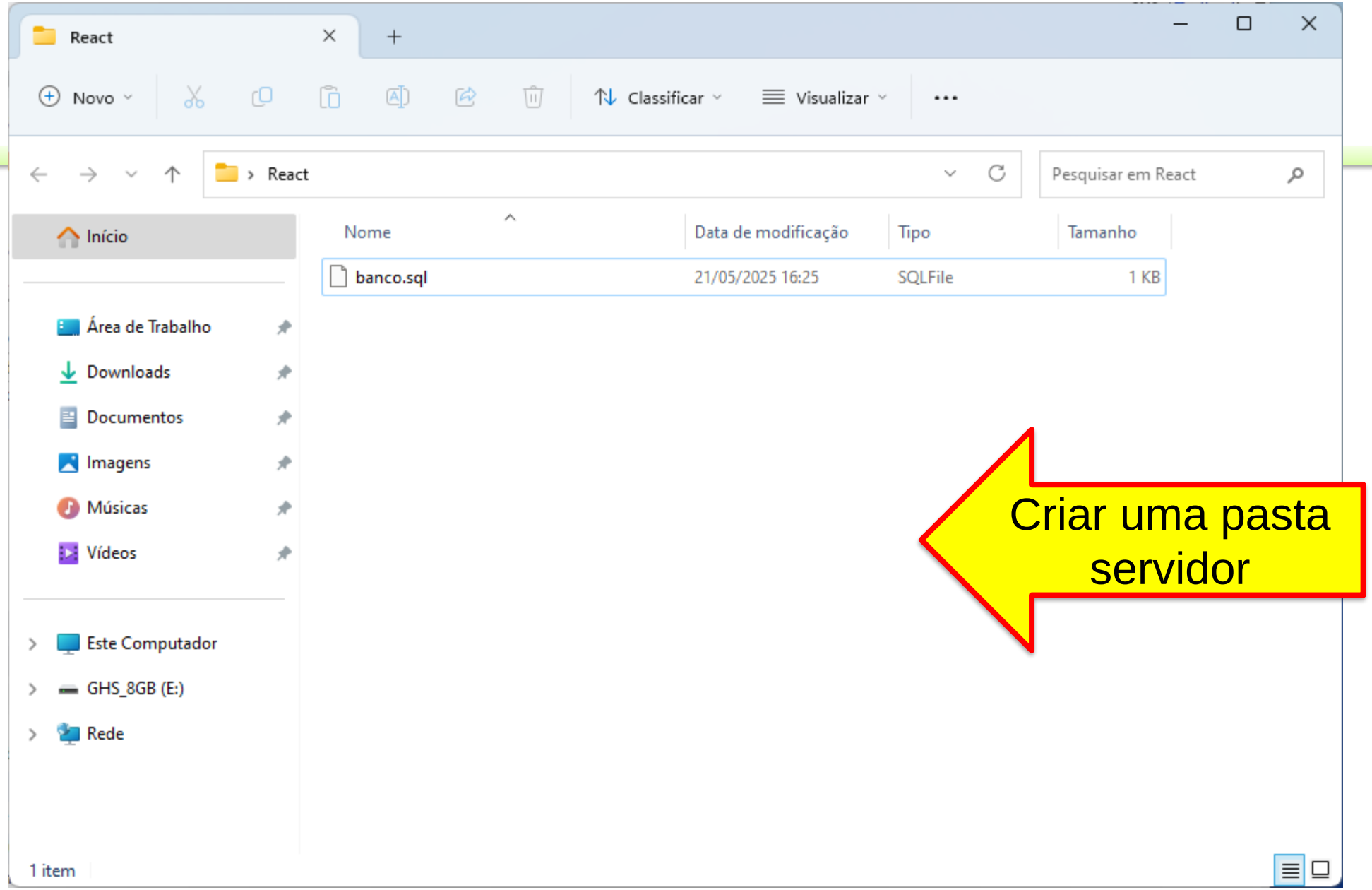

MySQL Workbench
Local instance wampmysqld64
File Edit View Query Database Server Tools Scripting Help
Clique para salvar
Navigator
SCHEMAS
Filter objects
sys
wordpress
Query 1
Limit to 1000 rows
1 • create database aulabd;
2 • use aulabd;
3
4 • create table alunos(
Result Grid
Filter Rows:
Edit:
Export/Import:
Wrap Cell Content:
codigo nome cidade estado
1 Guilherme Mococa SP
NULL NULL NULL NULL
Result Grid
Form Editor
Field Types
Administration Schemas
alunos 1
Apply
Revert
Information
No object selected
Output
Action Output
Time Action Message Duration / Fetch
1 16:24:10 create database aulabd 1 row(s) affected 0.015 sec
2 16:24:10 use aulabd 0 row(s) affected 0.000 sec
3 16:24:10 create table alunos(codigo int key auto_increment, nome varchar(50), cidade varch... 0 row(s) affected 0.016 sec
4 16:24:10 insert into alunos (nome, cidade, estado) values ("Guilherme", "Mococa", "SP") 1 row(s) affected 0.000 sec
5 16:24:10 select * from alunos LIMIT 0, 1000 1 row(s) returned 0.000 sec / 0.000 sec
Object Info Session


Save SQL Script
Área de Trabalho › React
Pesquisar em React
Organizar
Nova pasta
Início
Área de Traba
Downloads
Documentos
Imagens
Músicas
Vídeos
Nome
Data de modificação
Tipo
Tamanho
Nenhum item corresponde à pesquisa.
Salve dentro da pasta React
Nome: banco.sql
Tipo: SQL Files (*.sql)
Ocultar pastas
Salvar
Cancelar

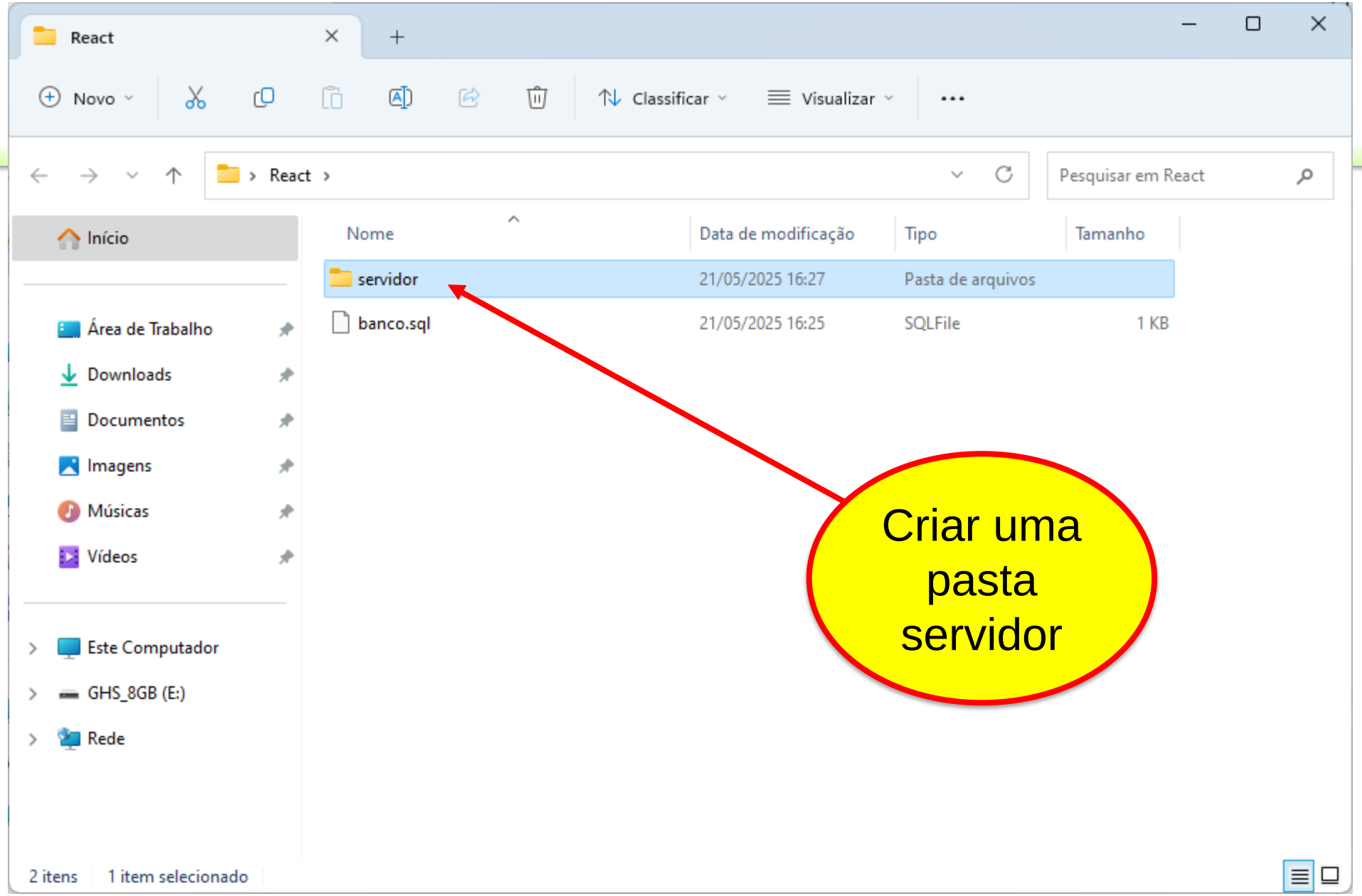

React
Novo
Classificar
Visualizar
React
Pesquisar em React
Início
Área de Trabalho
Downloads
Documentos
Imagens
Músicas
Vídeos
Este Computador
GHS_8GB (E:)
Rede
Nome
Data de modificação
Tipo
Tamanho
banco.sql
21/05/2025 16:25
SQLFile
1 KB
Arquivo Salvo
1 item

Criando a pasta “servidor”

Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br

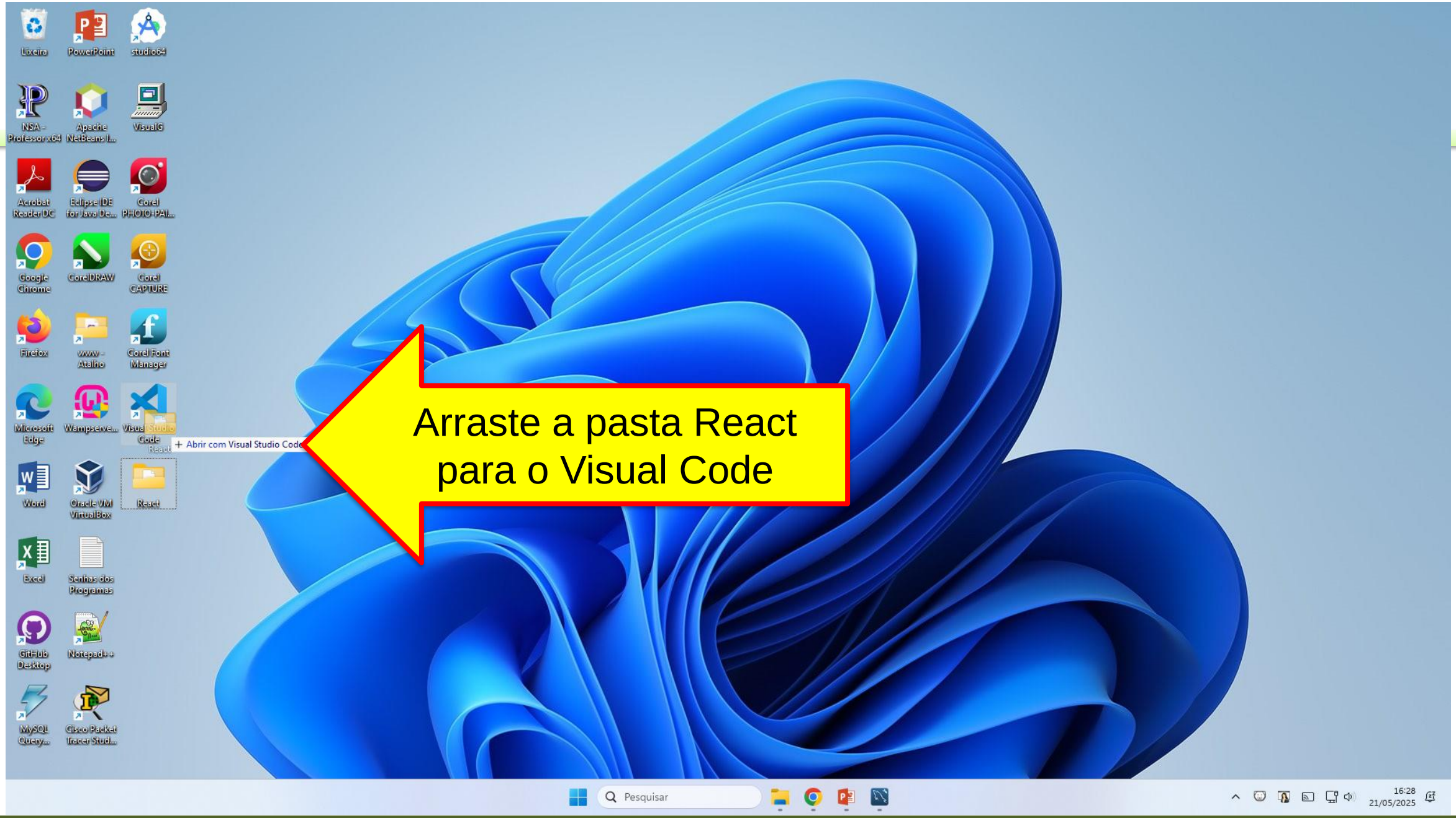


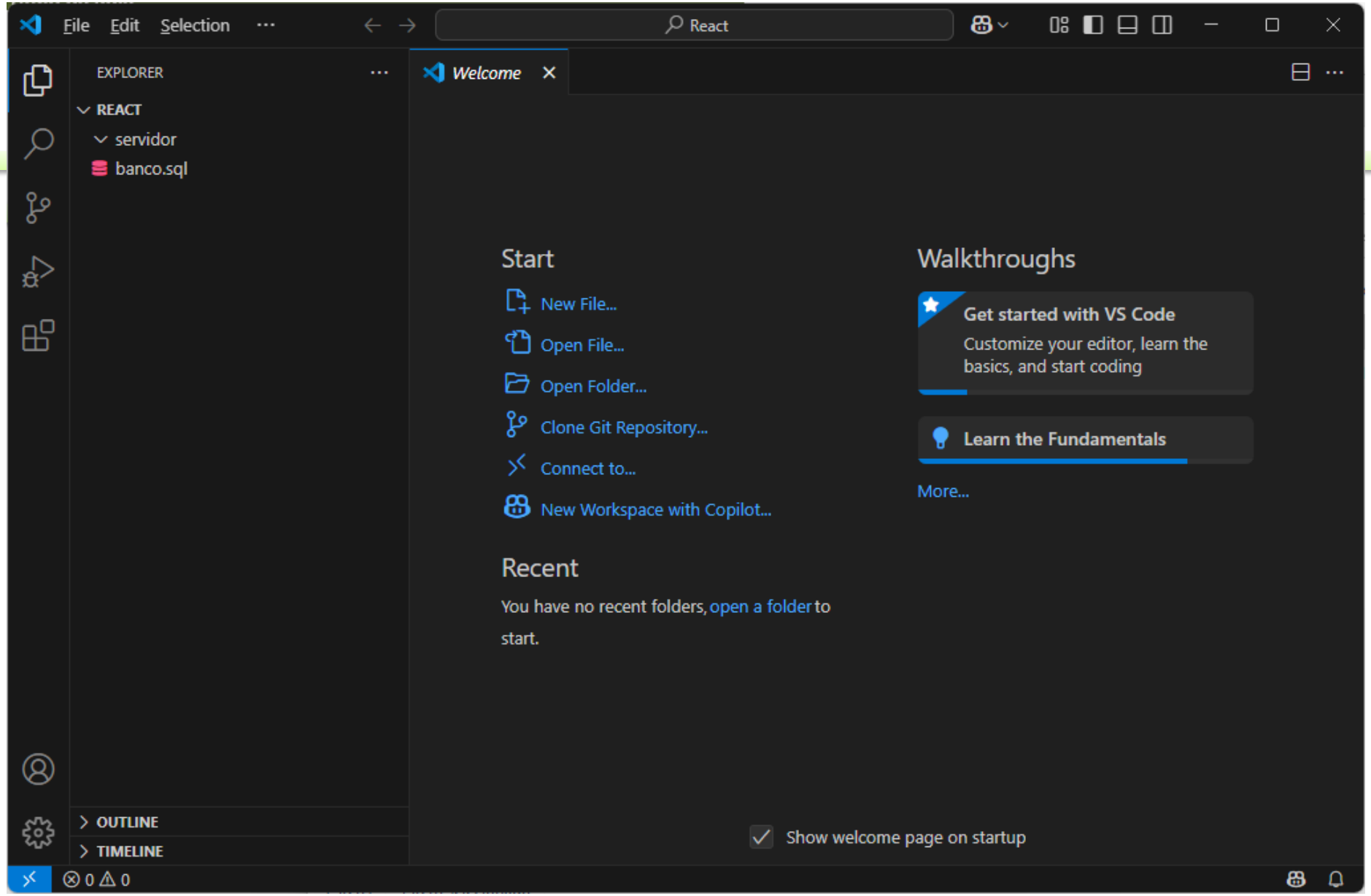
Criar uma pasta servidor

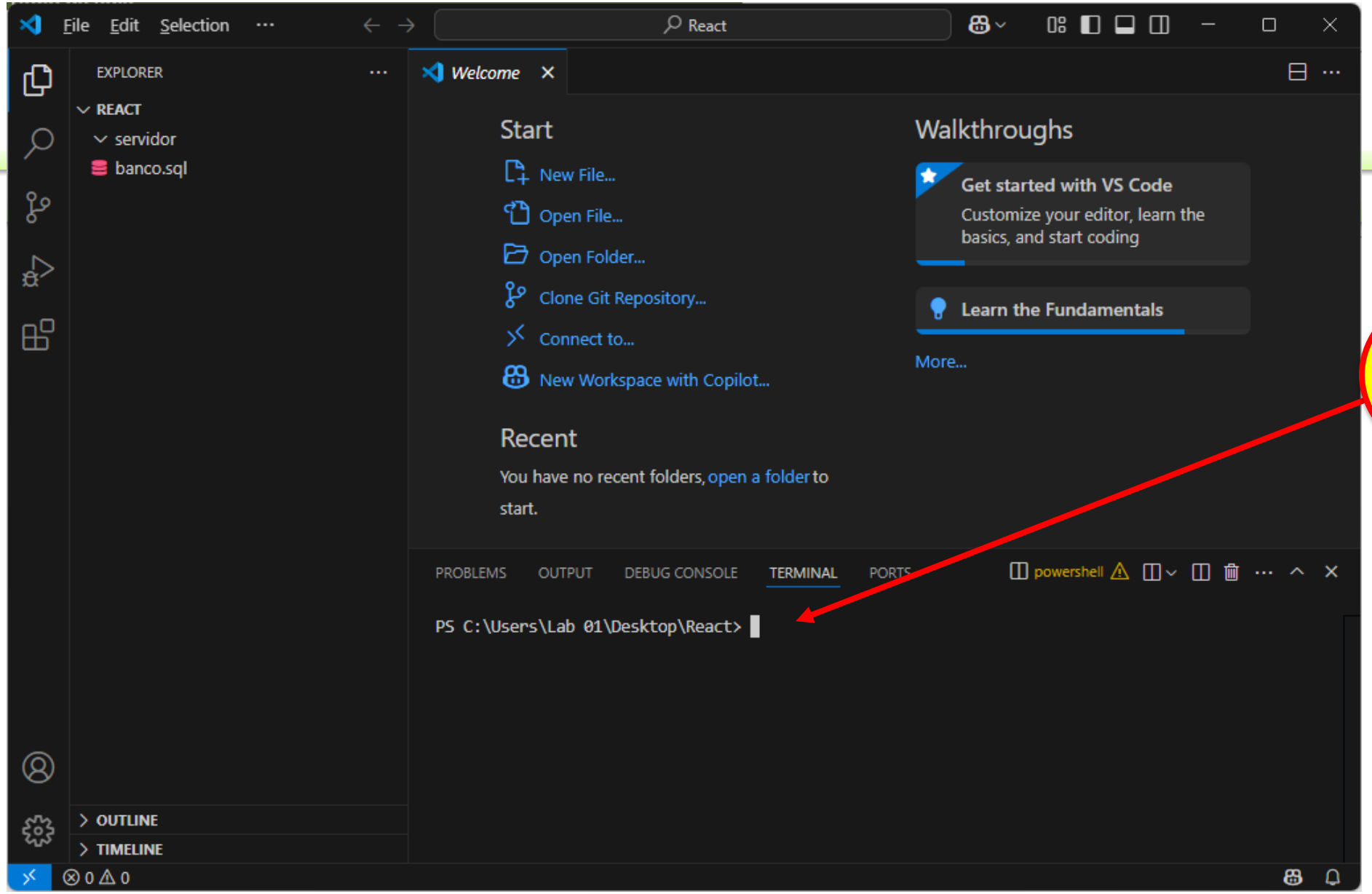

React
Novo
Classificar
Visualizar
React
Pesquisar em React
Início
Área de Trabalho
Downloads
Documentos
Imagens
Músicas
Vídeos
Este Computador
GHS_8GB (E:)
Rede
Nome
Data de modificação
Tipo
Tamanho
servidor
21/05/2025 16:27
Pasta de arquivos
banco.sql
21/05/2025 16:25
SQLFile
1 KB
Criar uma pasta servidor
2 itens
1 item selecionado

Abrindo o Visual Code

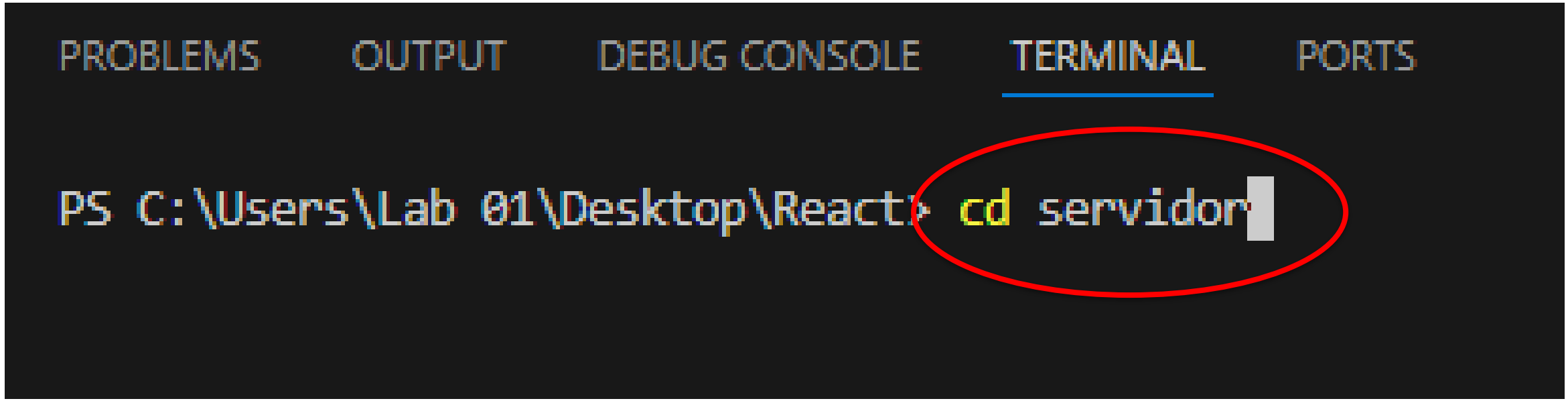
Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br

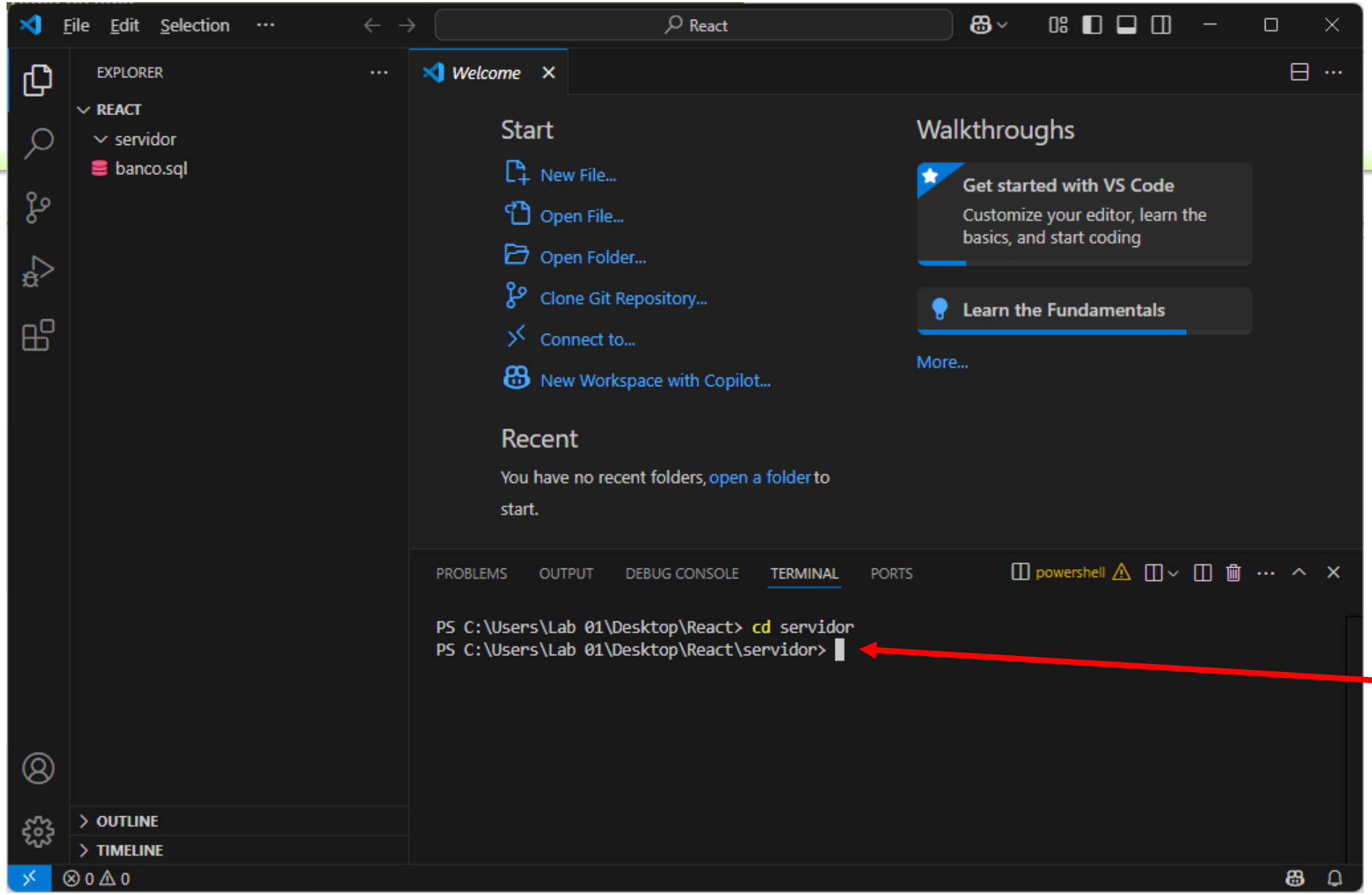

Lixeira
PowerPoint
studio64
NSA - Professor x64
Apache NetBeans I...
VisualG
Acrobat Reader DC
Eclipse IDE for Java De...
Corel PHOTO-PAI...
Google Chrome
CorelDRAW
Corel CAPTURE
Firefox
www - Atalho
Corel Font Manager
Microsoft Edge
Wampserve...
Visual Studio Code
+ Abrir com Visual Studio Code
Word
Oracle VM VirtualBox
React
Excel
Senhas dos Programas
GitHub Desktop
Notepad++
MySQL Query...
Cisco Packet Tracer Stud...
Arraste a pasta React para o Visual Code
Pesquisar
16:28
21/05/2025


File Edit Selection
React
EXPLORER
REACT
servidor
banco.sql
Welcome
Start
New File...
Open File...
Open Folder...
Clone Git Repository...
Connect to...
New Workspace with Copilot...
Recent
You have no recent folders, open a folder to start.
Walkthroughs
Get started with VS Code
Customize your editor, learn the basics, and start coding
Learn the Fundamentals
More...
Show welcome page on startup
OUTLINE
TIMELINE
0 0


File Edit Selection
React
EXPLORER
REACT
servidor
banco.sql
Welcome
Start
New File...
Open File...
Open Folder...
Clone Git Repository...
Connect to...
New Workspace with Copilot...
Walkthroughs
Get started with VS Code
Customize your editor, learn the basics, and start coding
Learn the Fundamentals
More...
Recent
You have no recent folders, open a folder to start.
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
powershell
PS C:\Users\Lab 01\Desktop\React>
OUTLINE
TIMELINE
0 0

Terminal


PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
PS C:\Users\Lab 01\Desktop\React> cd servidor

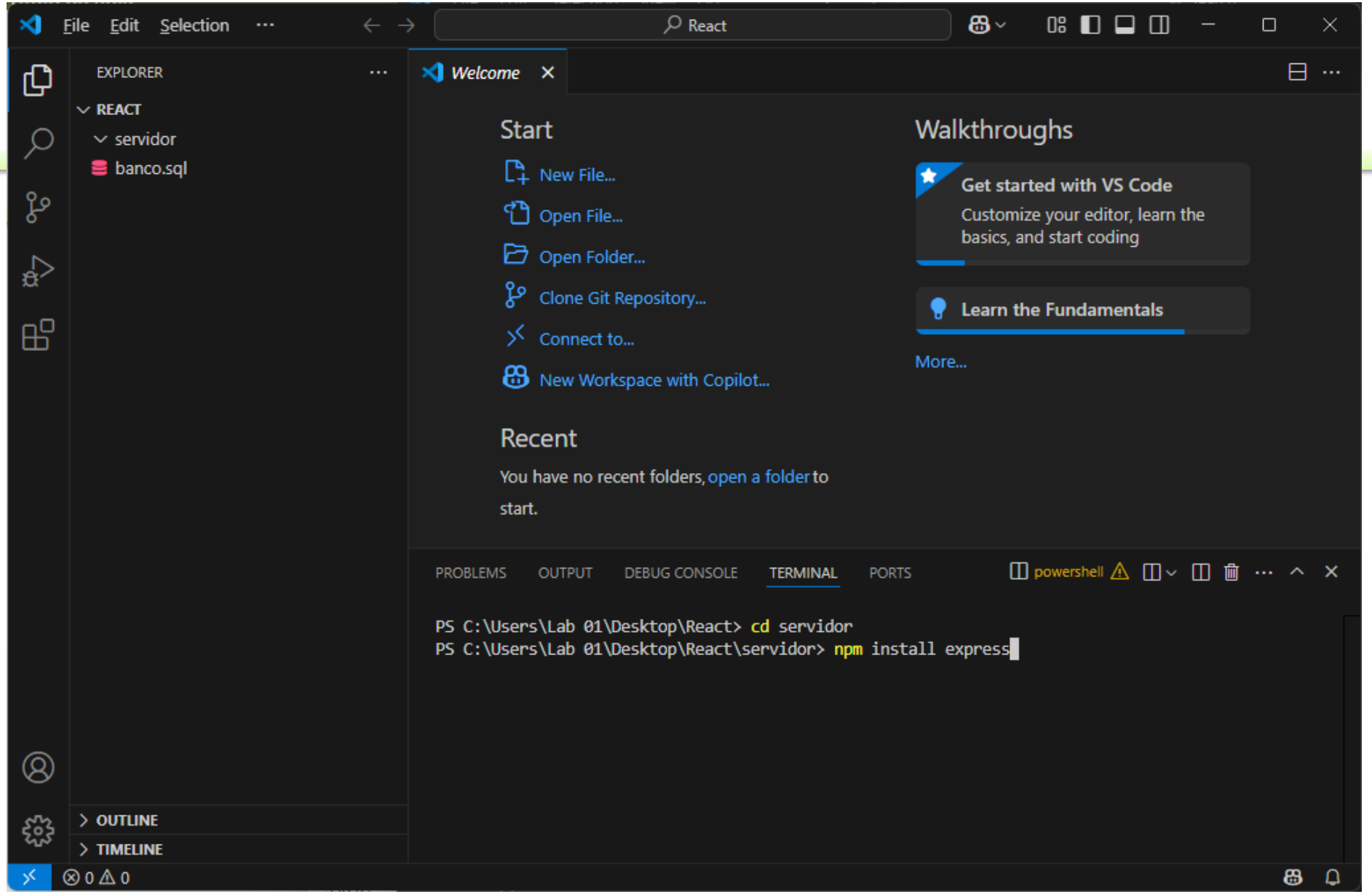


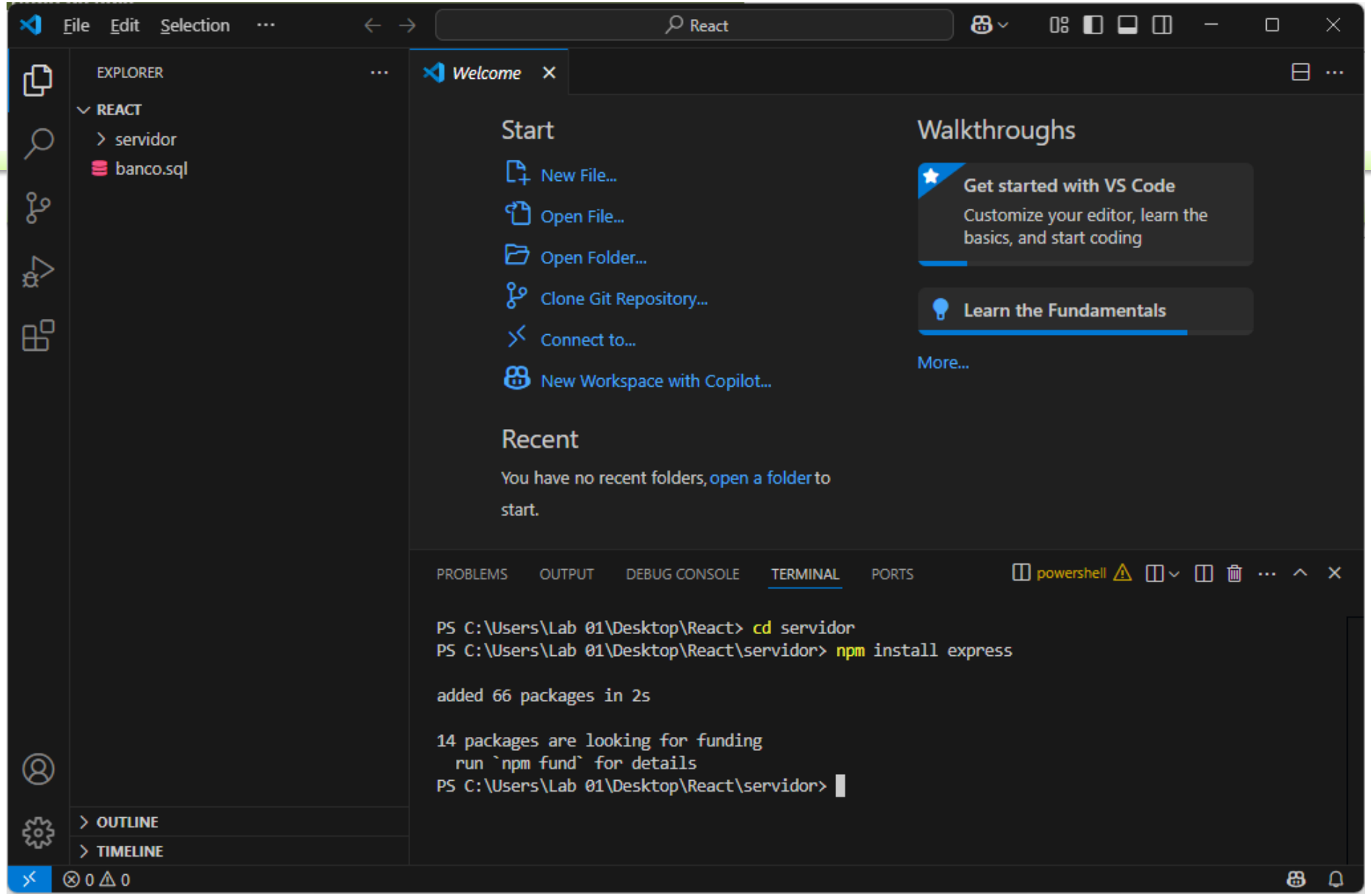
Resultado

Estamos dentro da pasta servidor

Instalando a biblioteca do “express”

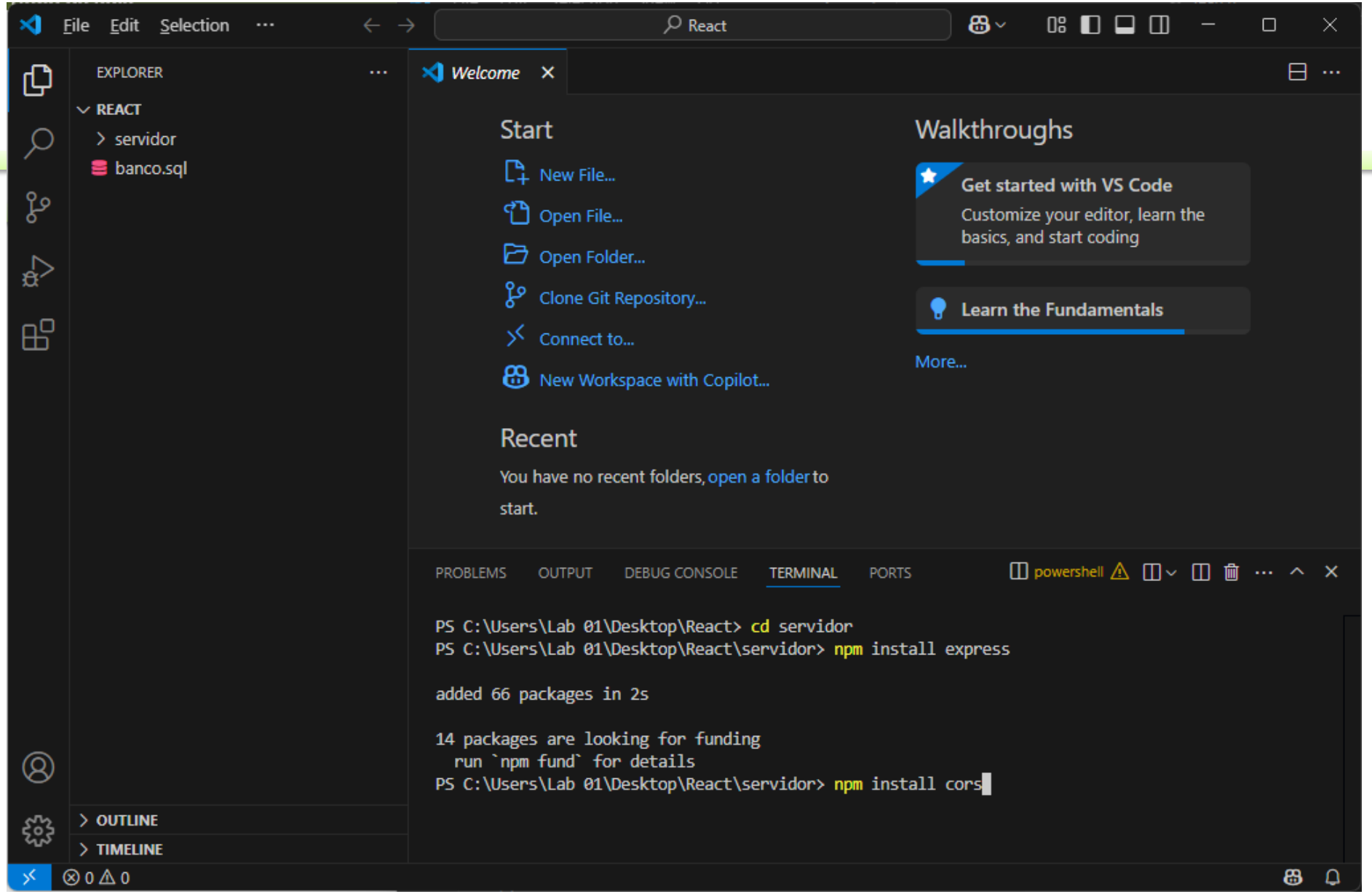
Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br

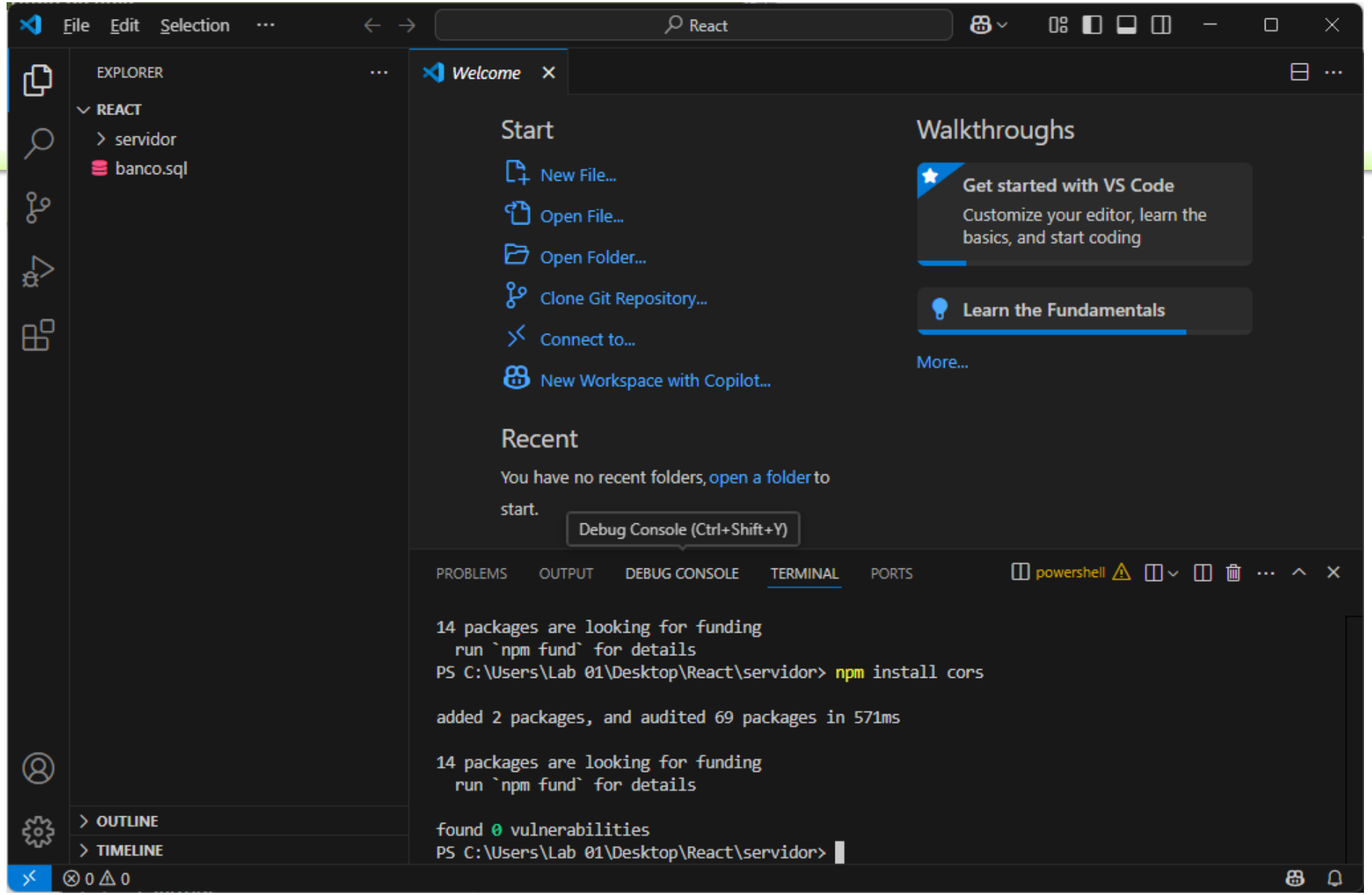

File Edit Selection
React
EXPLORER
REACT
servidor
banco.sql
Welcome
Start
New File...
Open File...
Open Folder...
Clone Git Repository...
Connect to...
New Workspace with Copilot...
Walkthroughs
Get started with VS Code
Customize your editor, learn the basics, and start coding
Learn the Fundamentals
More...
Recent
You have no recent folders, open a folder to start.
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
powershell
PS C:\Users\Lab 01\Desktop\React> cd servidor
PS C:\Users\Lab 01\Desktop\React\servidor> npm install express
OUTLINE
TIMELINE
0 0


File Edit Selection
React
EXPLORER
REACT
servidor
banco.sql
Welcome
Start
New File...
Open File...
Open Folder...
Clone Git Repository...
Connect to...
New Workspace with Copilot...
Walkthroughs
Get started with VS Code
Customize your editor, learn the basics, and start coding
Learn the Fundamentals
More...
Recent
You have no recent folders, open a folder to start.
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
PS C:\Users\Lab 01\Desktop\React> cd servidor
PS C:\Users\Lab 01\Desktop\React\servidor> npm install express
added 66 packages in 2s
14 packages are looking for funding
run `npm fund` for details
PS C:\Users\Lab 01\Desktop\React\servidor>
OUTLINE
TIMELINE
0 0

Instalando a biblioteca do “cors”

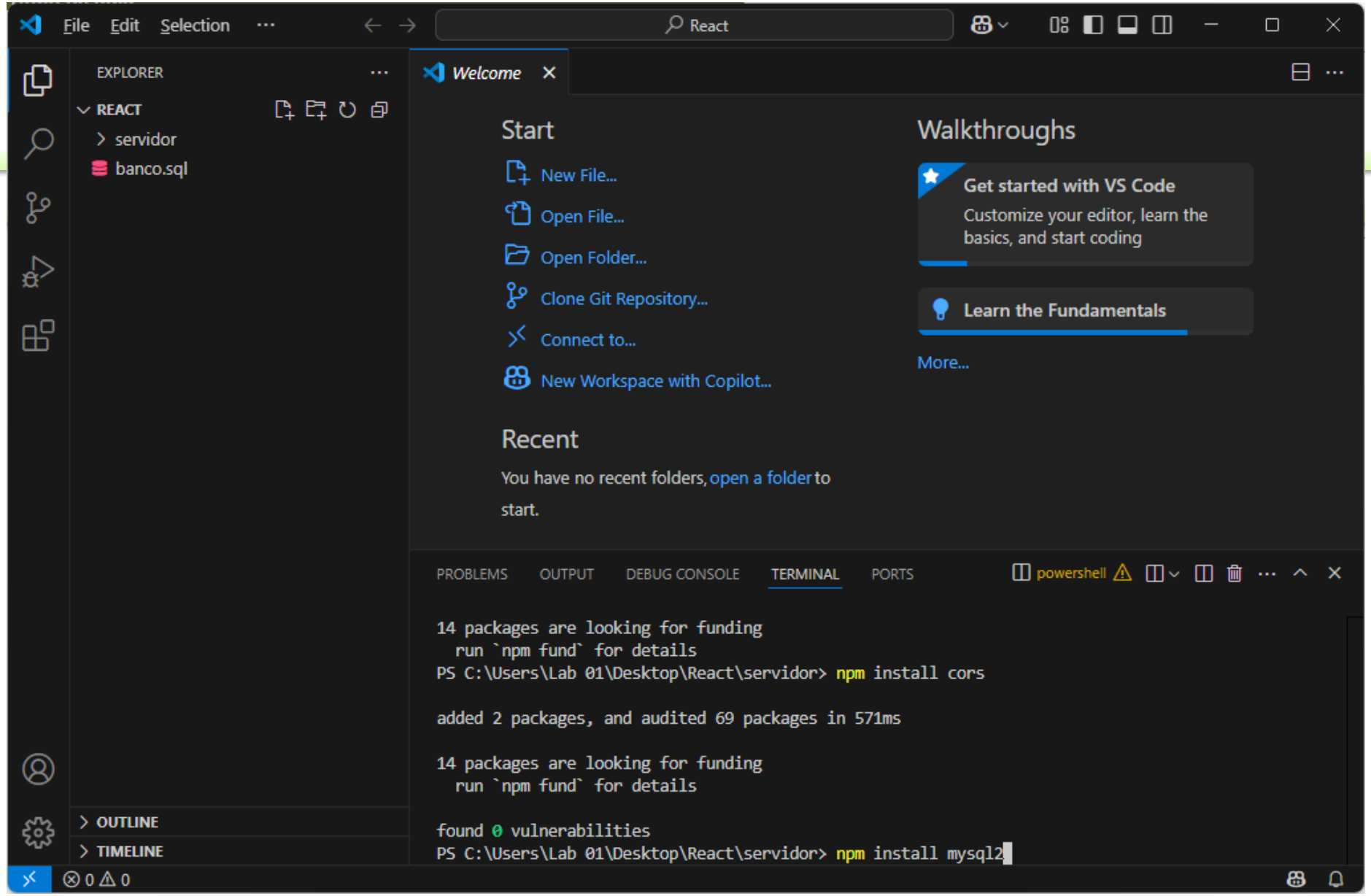
Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br

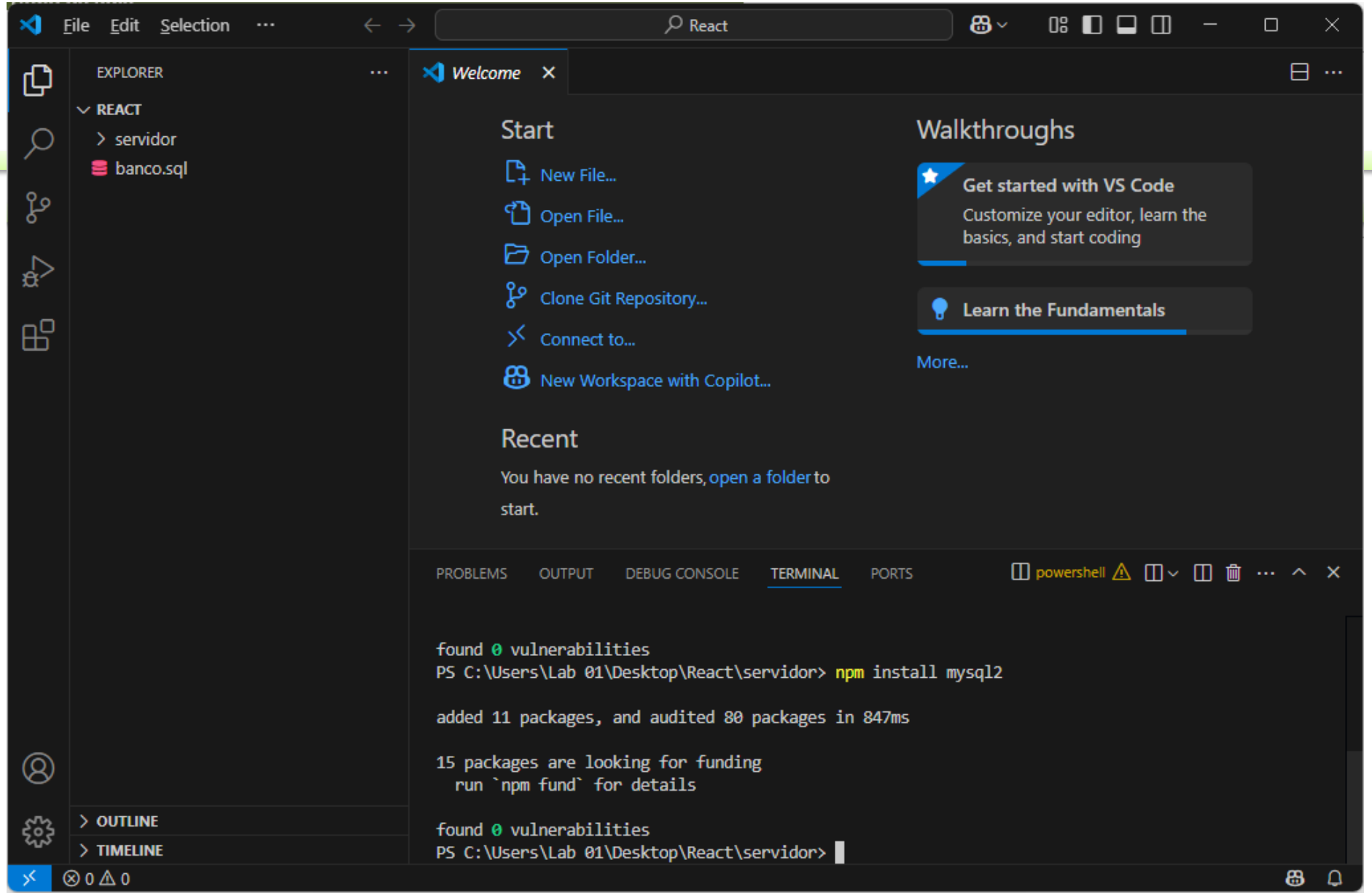

File Edit Selection
React
EXPLORER
REACT
servidor
banco.sql
Welcome
Start
New File...
Open File...
Open Folder...
Clone Git Repository...
Connect to...
New Workspace with Copilot...
Walkthroughs
Get started with VS Code
Customize your editor, learn the basics, and start coding
Learn the Fundamentals
More...
Recent
You have no recent folders, open a folder to start.
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
PS C:\Users\Lab 01\Desktop\React> cd servidor
PS C:\Users\Lab 01\Desktop\React\servidor> npm install express
added 66 packages in 2s
14 packages are looking for funding
run `npm fund` for details
PS C:\Users\Lab 01\Desktop\React\servidor> npm install cors
OUTLINE
TIMELINE
0 0


File Edit Selection
React
EXPLORER
REACT
servidor
banco.sql
Welcome
Start
New File...
Open File...
Open Folder...
Clone Git Repository...
Connect to...
New Workspace with Copilot...
Walkthroughs
Get started with VS Code
Customize your editor, learn the basics, and start coding
Learn the Fundamentals
More...
Recent
You have no recent folders, open a folder to start.
Debug Console (Ctrl+Shift+Y)
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
14 packages are looking for funding
run `npm fund` for details
PS C:\Users\Lab 01\Desktop\React\servidor> npm install cors
added 2 packages, and audited 69 packages in 571ms
14 packages are looking for funding
run `npm fund` for details
found 0 vulnerabilities
PS C:\Users\Lab 01\Desktop\React\servidor>
OUTLINE
TIMELINE
0 0

Instalando a biblioteca do “mysql2”

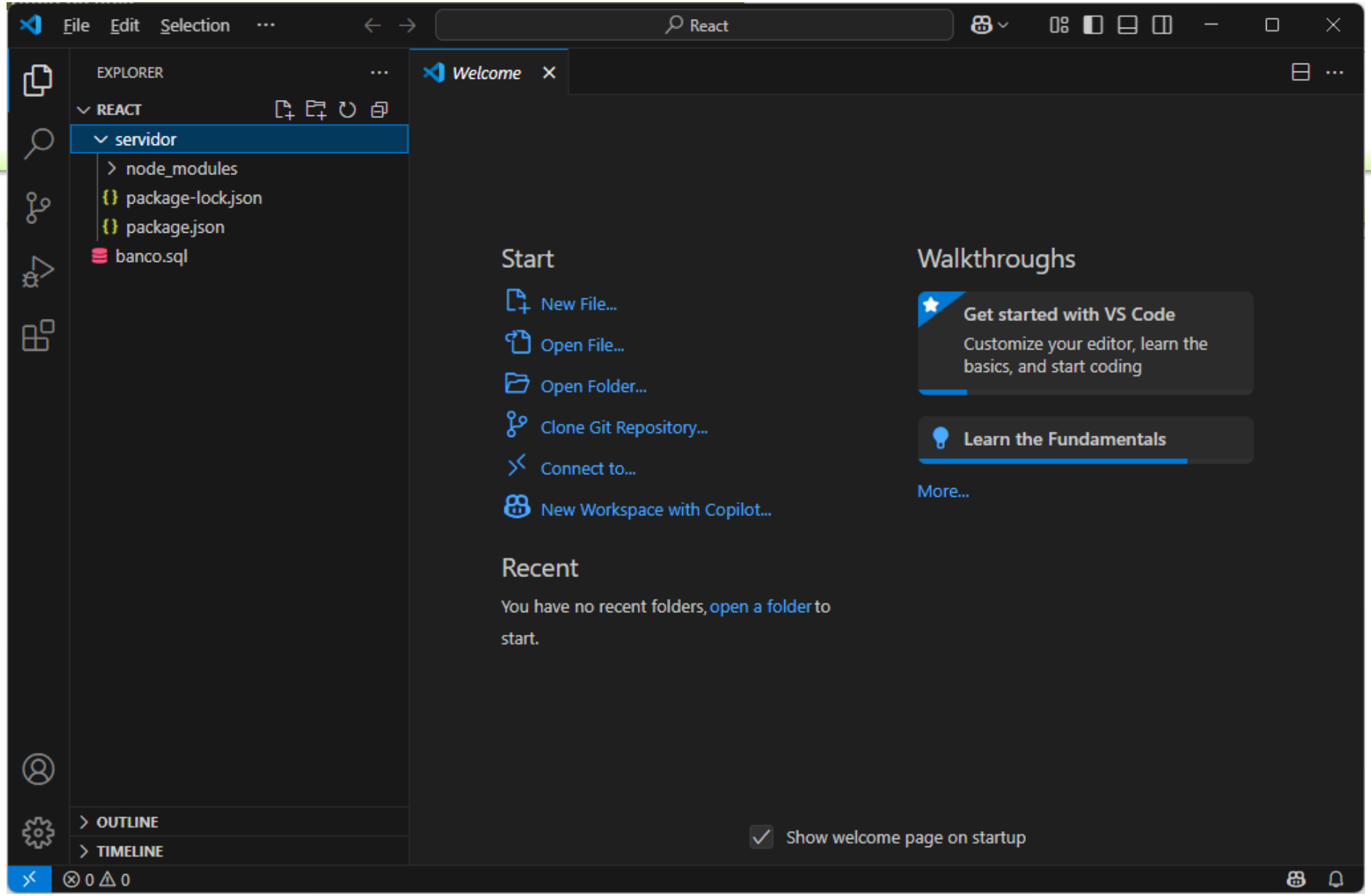
Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br

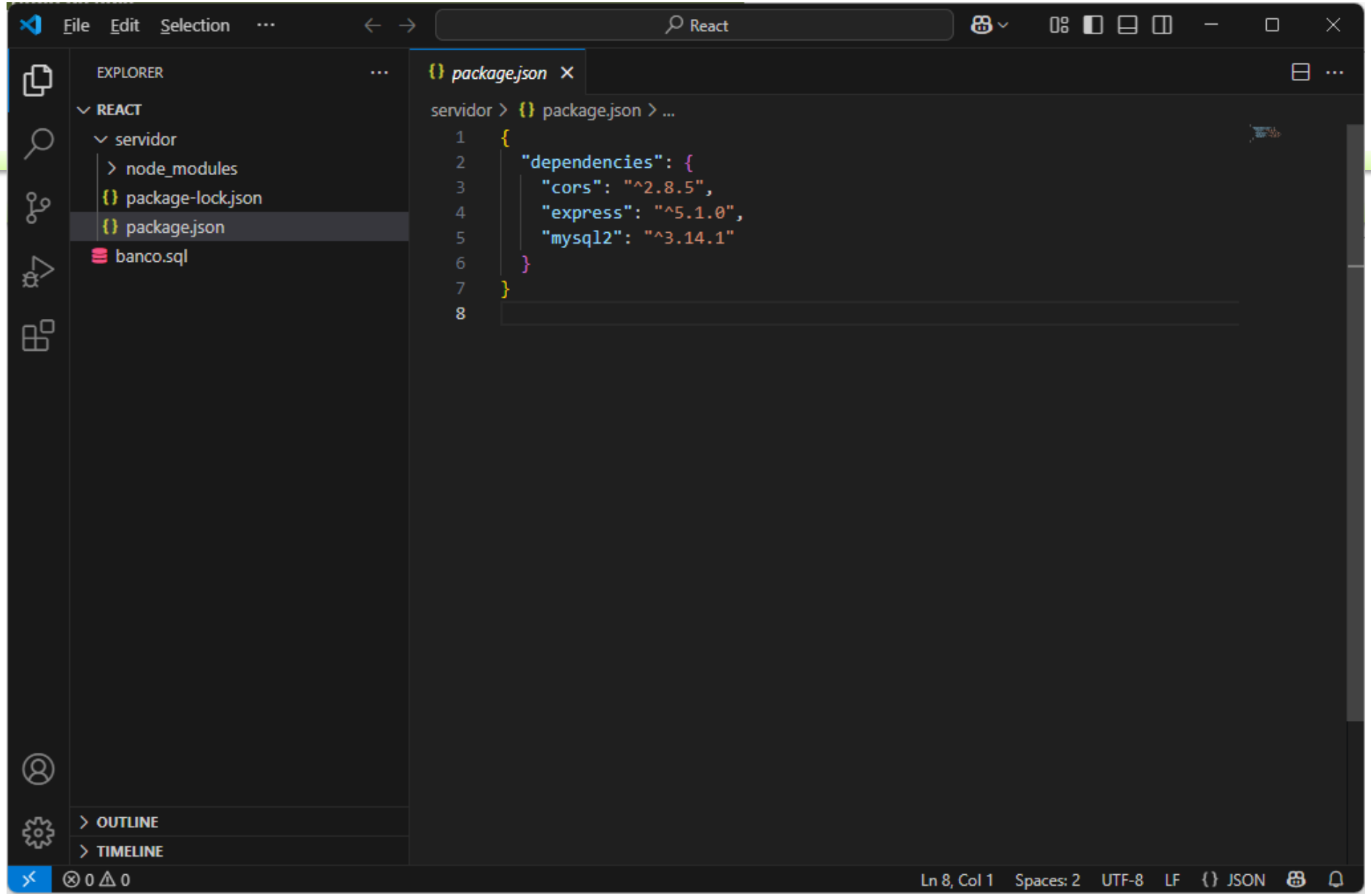

File Edit Selection
React
EXPLORER
REACT
servidor
banco.sql
Welcome
Start
New File...
Open File...
Open Folder...
Clone Git Repository...
Connect to...
New Workspace with Copilot...
Recent
You have no recent folders, open a folder to start.
Walkthroughs
Get started with VS Code
Customize your editor, learn the basics, and start coding
Learn the Fundamentals
More...
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
14 packages are looking for funding
run `npm fund` for details
PS C:\Users\Lab 01\Desktop\React\servidor> npm install cors
added 2 packages, and audited 69 packages in 571ms
14 packages are looking for funding
run `npm fund` for details
found 0 vulnerabilities
PS C:\Users\Lab 01\Desktop\React\servidor> npm install mysql2
OUTLINE
TIMELINE
0 0


File Edit Selection
React
EXPLORER
REACT
servidor
banco.sql
Welcome
Start
New File...
Open File...
Open Folder...
Clone Git Repository...
Connect to...
New Workspace with Copilot...
Walkthroughs
Get started with VS Code
Customize your editor, learn the basics, and start coding
Learn the Fundamentals
More...
Recent
You have no recent folders, open a folder to start.
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
powershell
found 0 vulnerabilities
PS C:\Users\Lab 01\Desktop\React\servidor> npm install mysql2
added 11 packages, and audited 80 packages in 847ms
15 packages are looking for funding
run `npm fund` for details
found 0 vulnerabilities
PS C:\Users\Lab 01\Desktop\React\servidor>
OUTLINE
TIMELINE
0 0

Visualizando o arquivo package.json

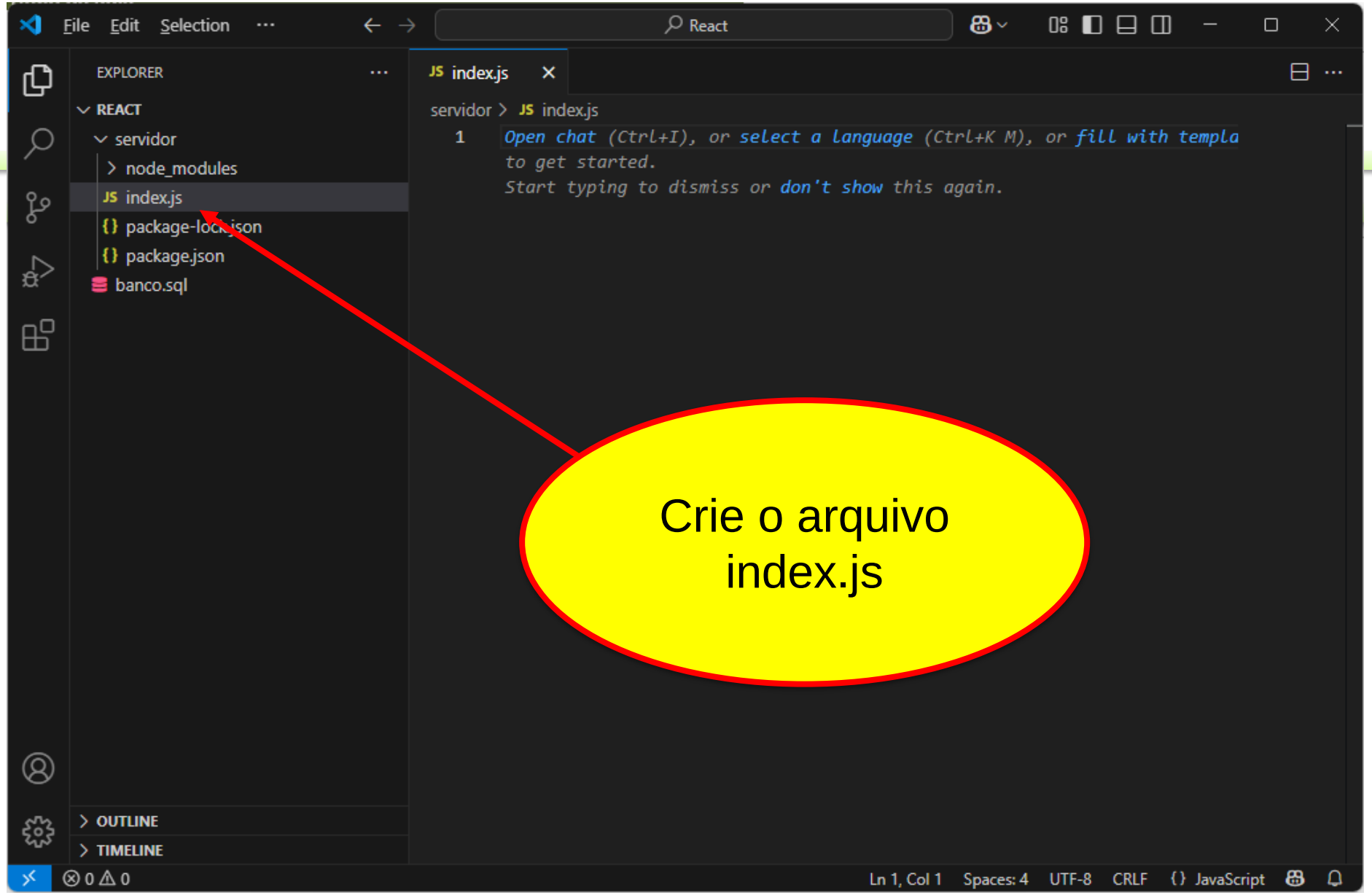
Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br

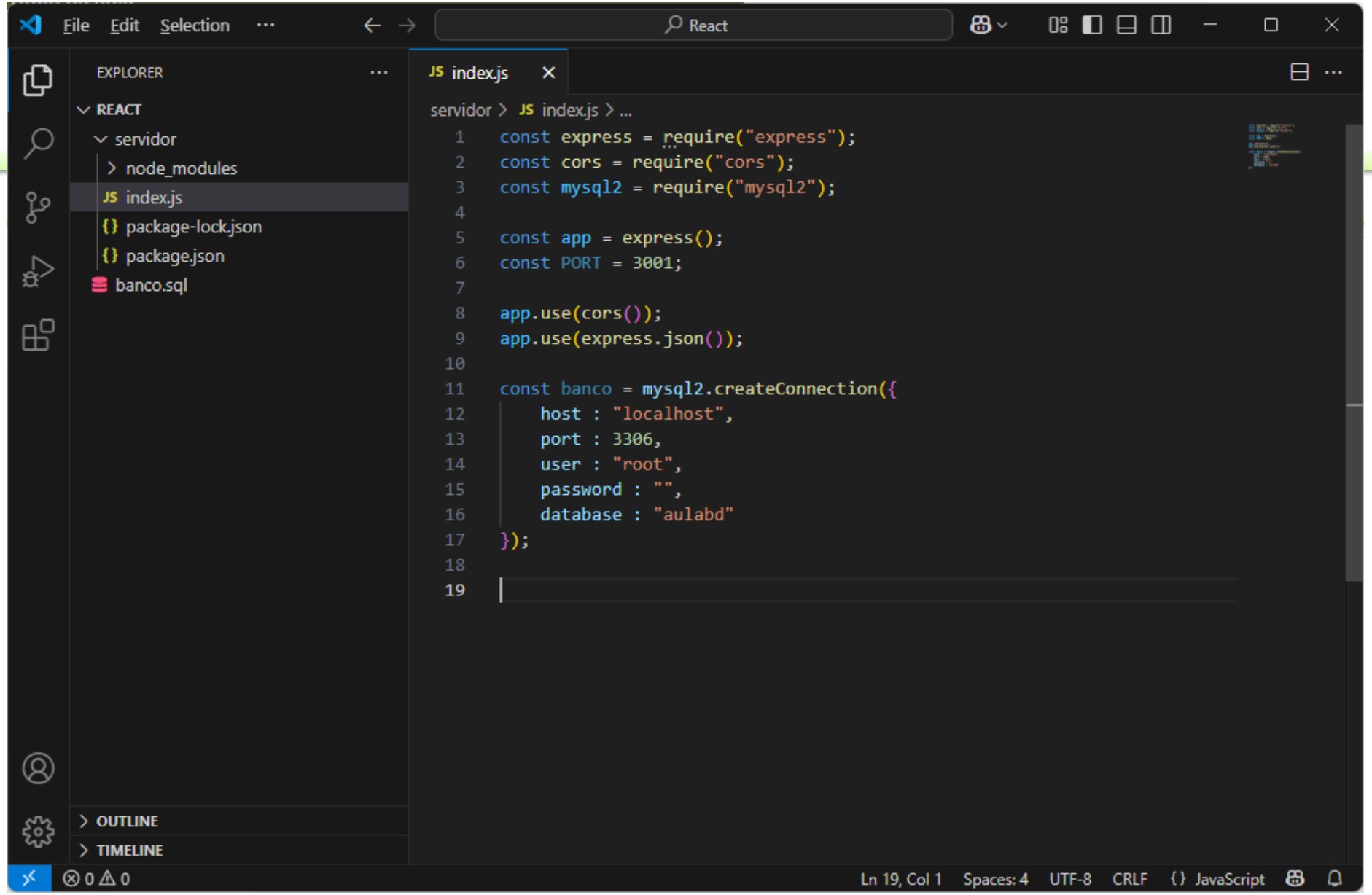

File Edit Selection
React
EXPLORER
REACT
servidor
node_modules
package-lock.json
package.json
banco.sql
Welcome
Start
New File...
Open File...
Open Folder...
Clone Git Repository...
Connect to...
New Workspace with Copilot...
Recent
You have no recent folders, open a folder to start.
Walkthroughs
Get started with VS Code
Customize your editor, learn the basics, and start coding
Learn the Fundamentals
More...
Show welcome page on startup
OUTLINE
TIMELINE
0 0


File Edit Selection
React
EXPLORER
REACT
servidor
node_modules
package-lock.json
package.json
banco.sql
package.json
servidor > package.json > ...
{
"dependencies": {
"cors": "^2.8.5",
"express": "^5.1.0",
"mysql2": "^3.14.1"
}
}
OUTLINE
TIMELINE
Ln 8, Col 1
Spaces: 2
UTF-8
LF
JSON

Criando o arquivo index.js

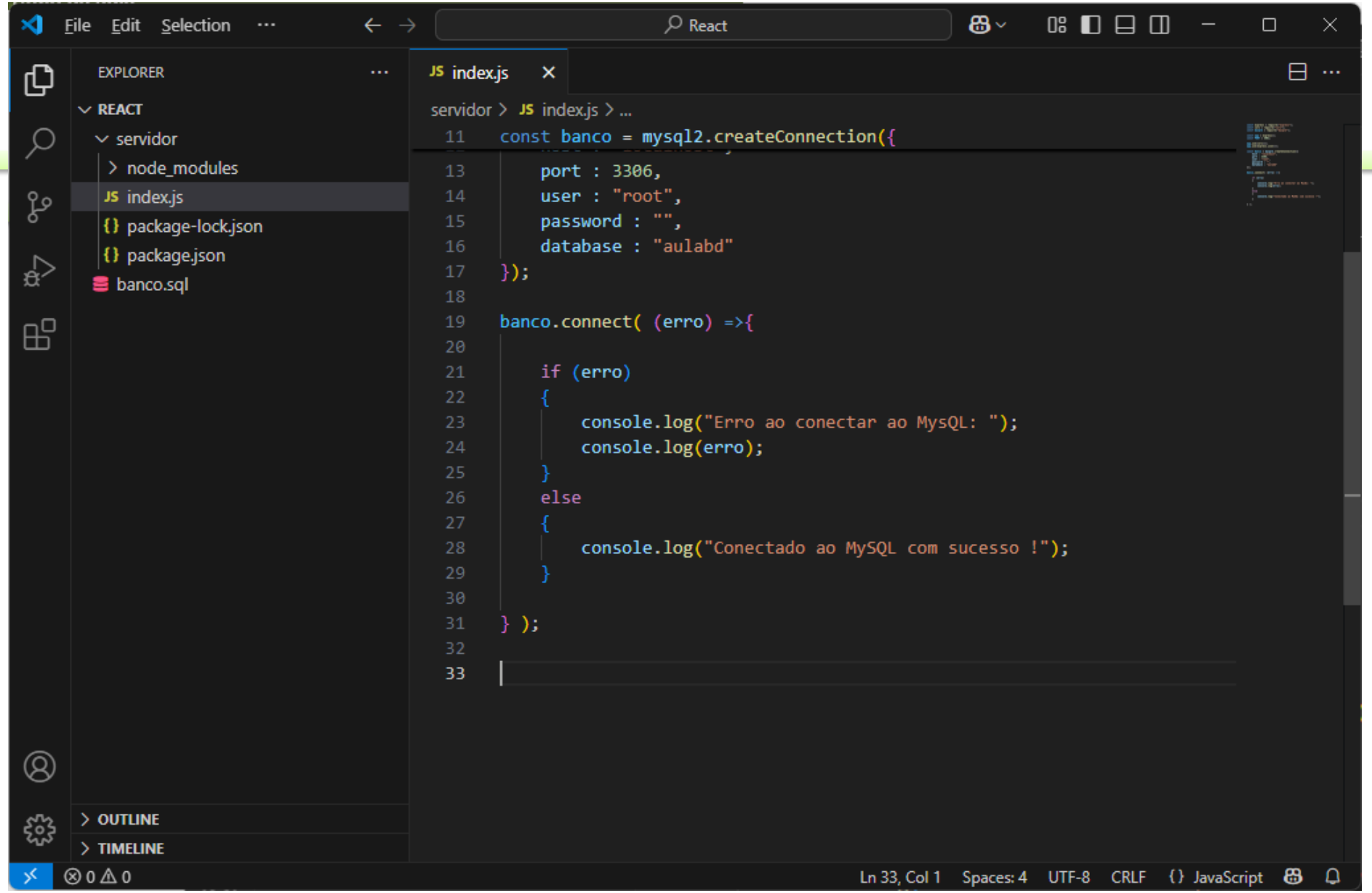
Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br


File Edit Selection
React
EXPLORER
REACT
servidor
node_modules
index.js
package-lock.json
package.json
banco.sql
index.js
servidor > index.js
1 Open chat (Ctrl+I), or select a language (Ctrl+K M), or fill with templa
to get started.
Start typing to dismiss or don't show this again.
Crie o arquivo index.js
OUTLINE
TIMELINE
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF JavaScript


File Edit Selection
React
EXPLORER
REACT
servidor
node_modules
index.js
package-lock.json
package.json
banco.sql
index.js
servidor > index.js > ...
1 const express = require("express");
2 const cors = require("cors");
3 const mysql2 = require("mysql2");
4
5 const app = express();
6 const PORT = 3001;
7
8 app.use(cors());
9 app.use(express.json());
10
11 const banco = mysql2.createConnection({
12 host : "localhost",
13 port : 3306,
14 user : "root",
15 password : "",
16 database : "aulabd"
17 });
18
19
OUTLINE
TIMELINE
0 0
Ln 19, Col 1 Spaces: 4 UTF-8 CRLF JavaScript

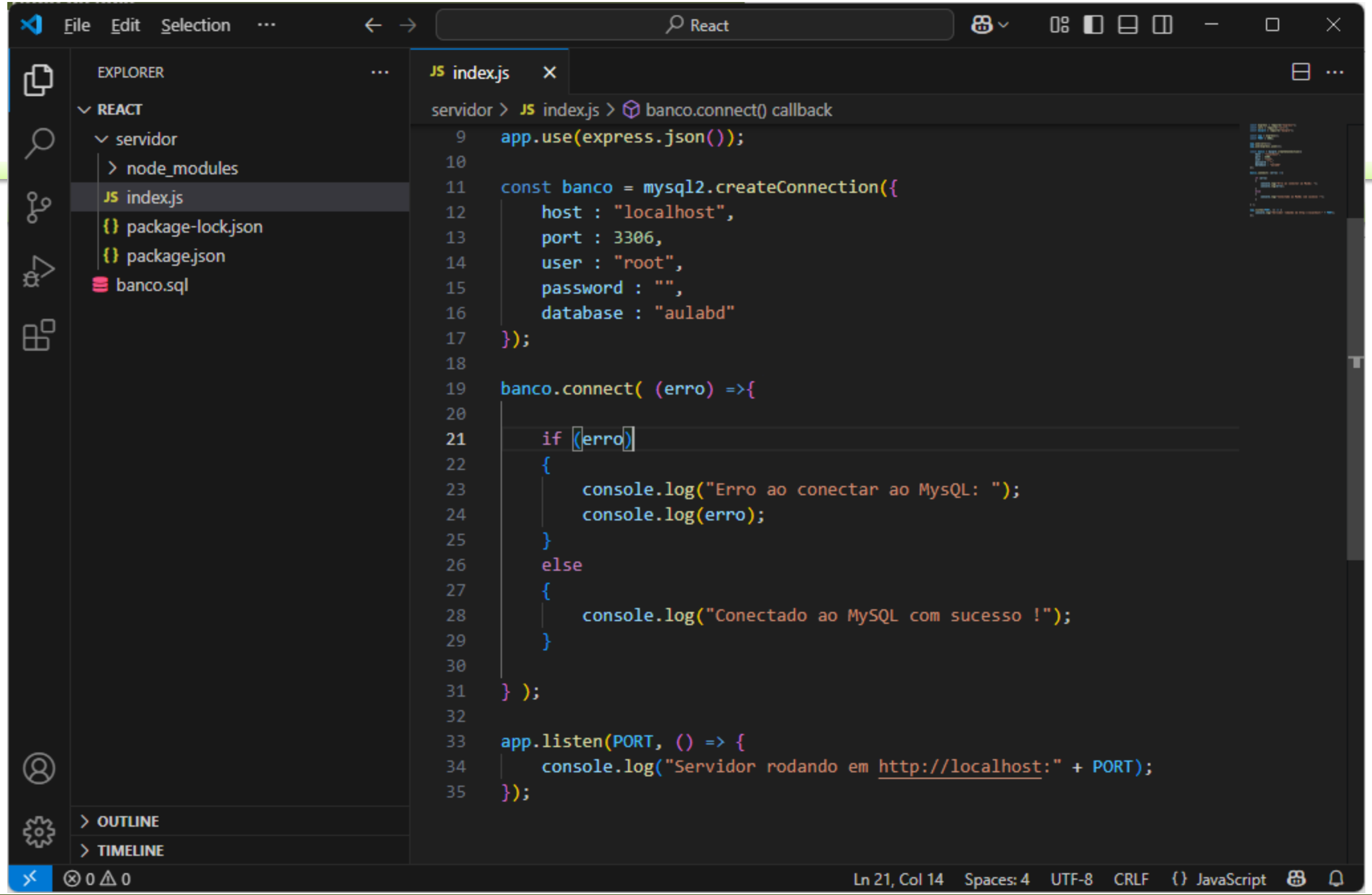
Método “connect”

Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br


File Edit Selection
React
EXPLORER
REACT
servidor
node_modules
index.js
package-lock.json
package.json
banco.sql
index.js
servidor > index.js > ...
11 const banco = mysql2.createConnection({
13 port : 3306,
14 user : "root",
15 password : "",
16 database : "aulabd"
17 });
18
19 banco.connect((erro) =>{
20
21 if (erro)
22 {
23 console.log("Erro ao conectar ao MySQL: ");
24 console.log(erro);
25 }
26 else
27 {
28 console.log("Conectado ao MySQL com sucesso !");
29 }
30
31 });
32
33
OUTLINE
TIMELINE
0 0
Ln 33, Col 1
Spaces: 4
UTF-8
CRLF
JavaScript

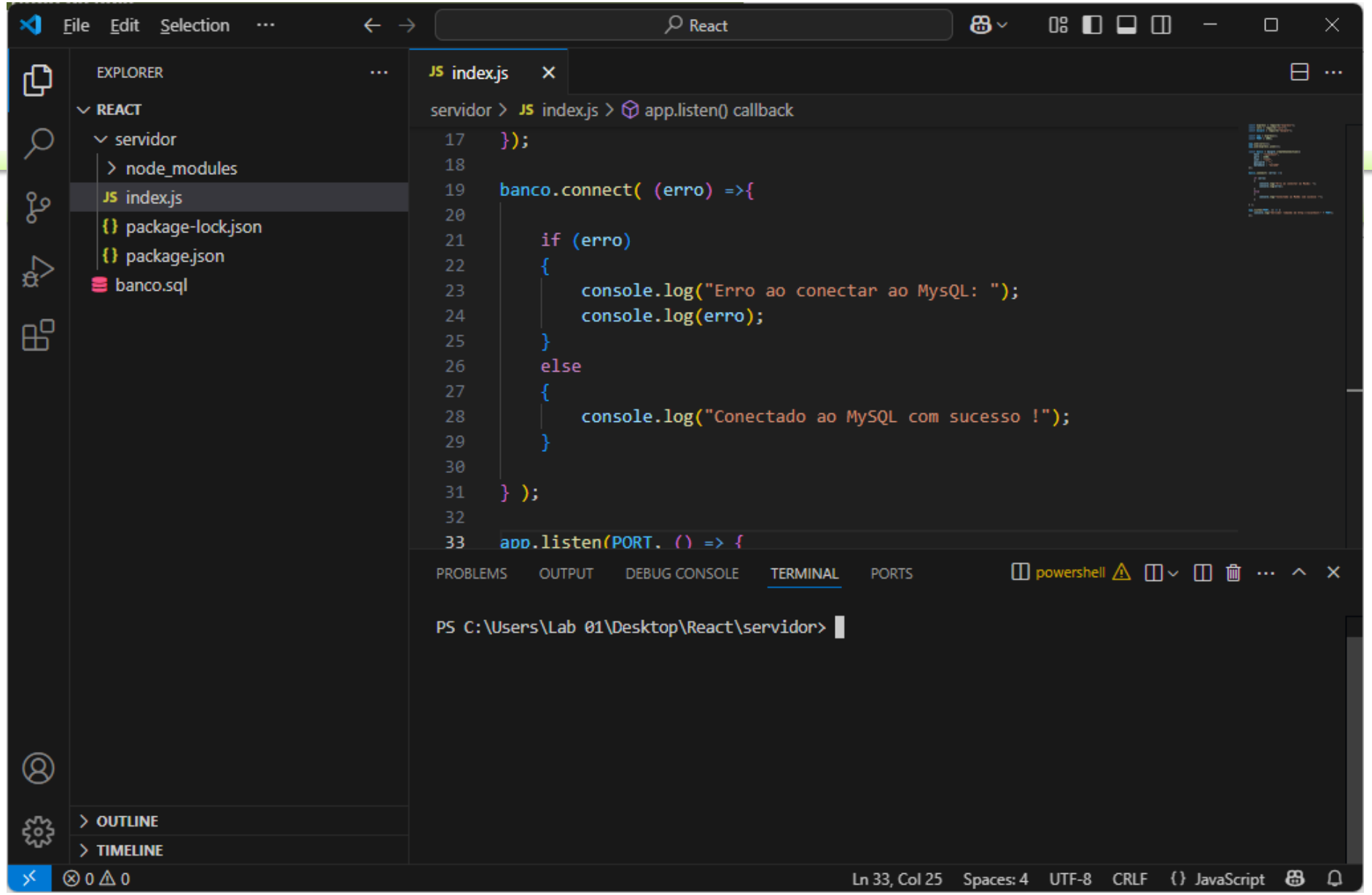
Método “listen”

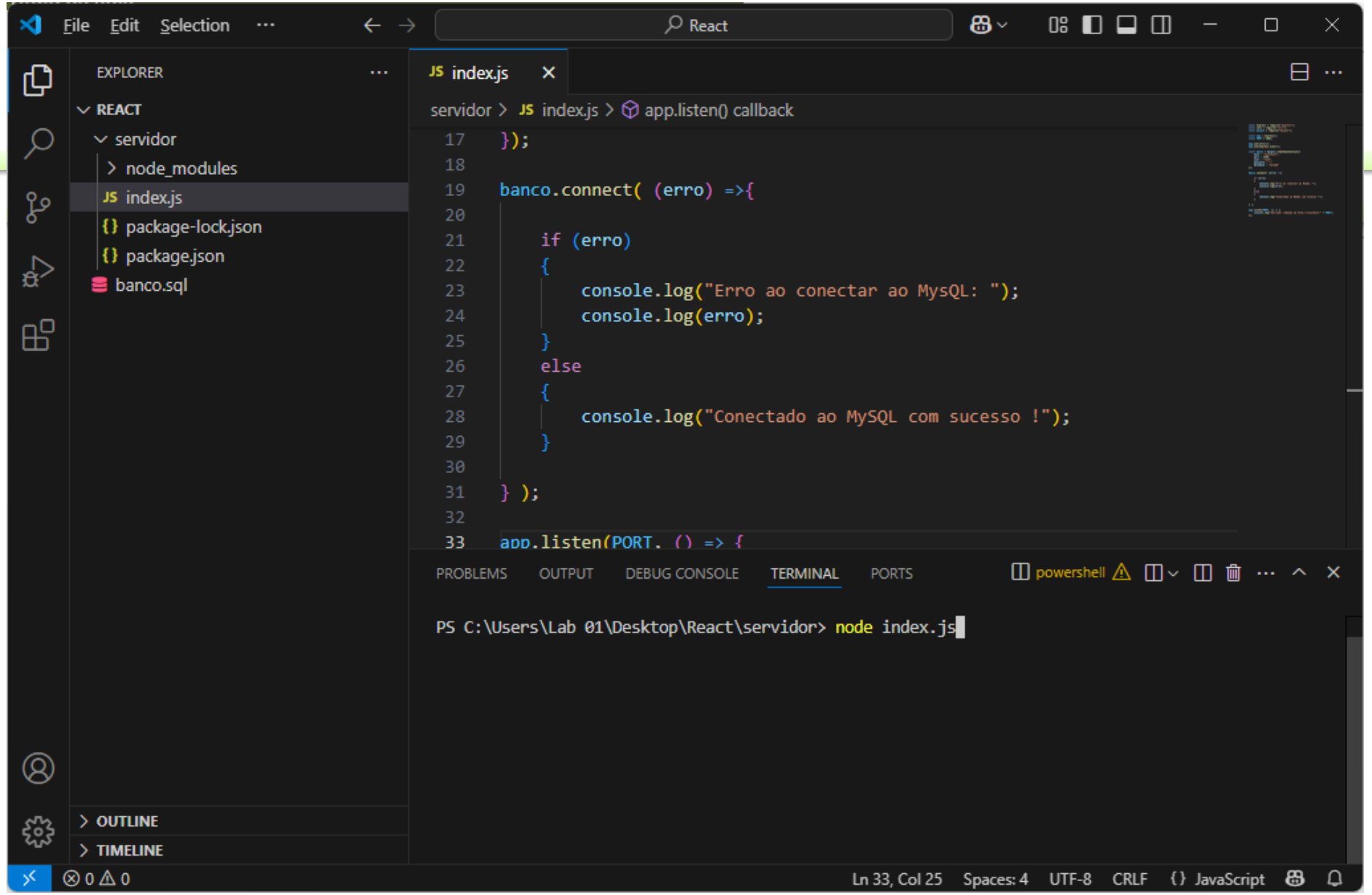
Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br

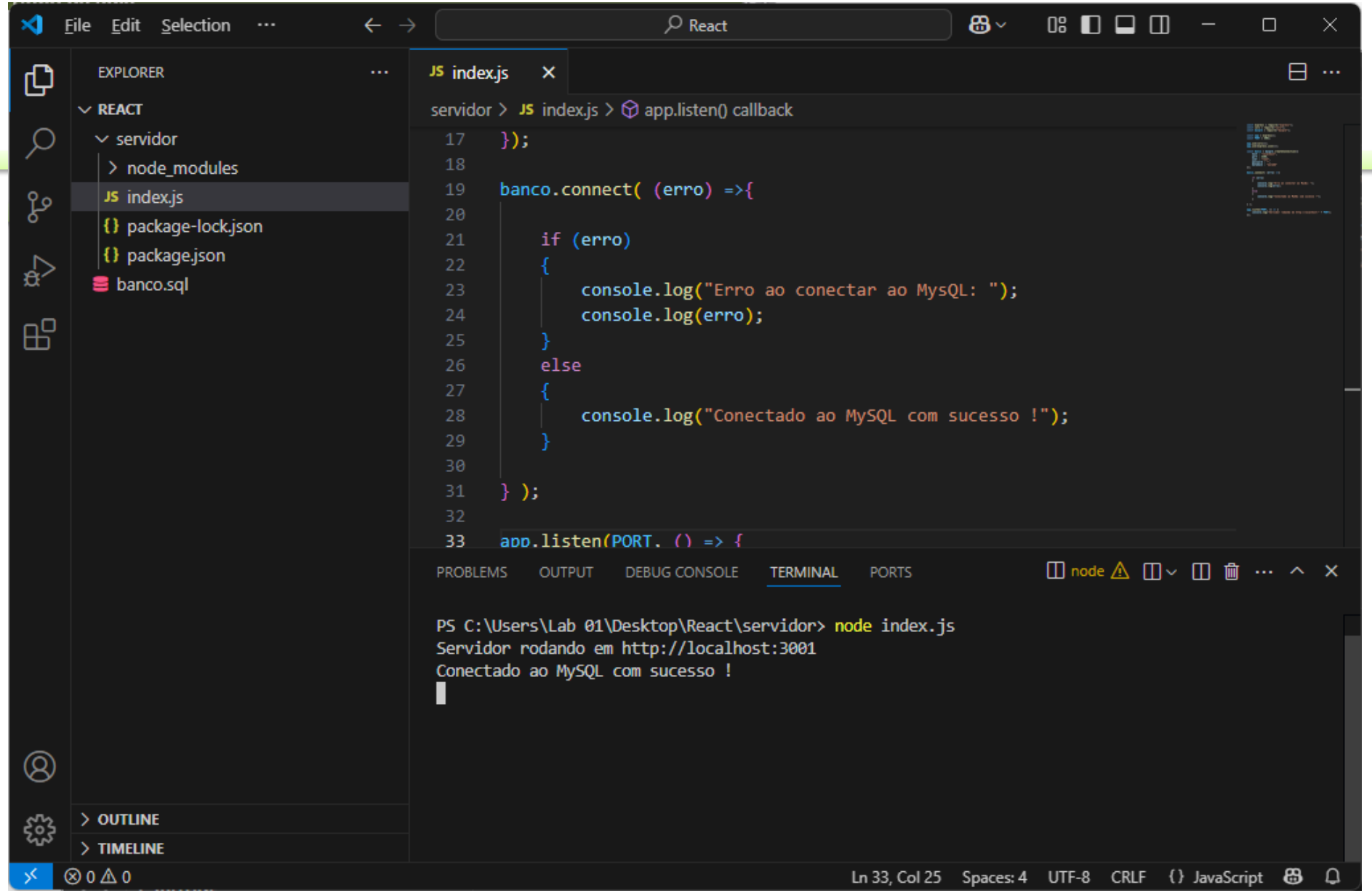

File Edit Selection
React
EXPLORER
REACT
servidor
node_modules
index.js
package-lock.json
package.json
banco.sql
OUTLINE
TIMELINE
index.js
servidor > index.js > banco.connect() callback
app.use(express.json());
const banco = mysql2.createConnection({
host : "localhost",
port : 3306,
user : "root",
password : "",
database : "aulabd"
});
banco.connect((erro) =>{
if (erro)
{
console.log("Erro ao conectar ao MySQL: ");
console.log(erro);
}
else
{
console.log("Conectado ao MySQL com sucesso !");
}
});
app.listen(PORT, () => {
console.log("Servidor rodando em http://localhost:" + PORT);
});
Ln 21, Col 14 Spaces: 4 UTF-8 CRLF JavaScript

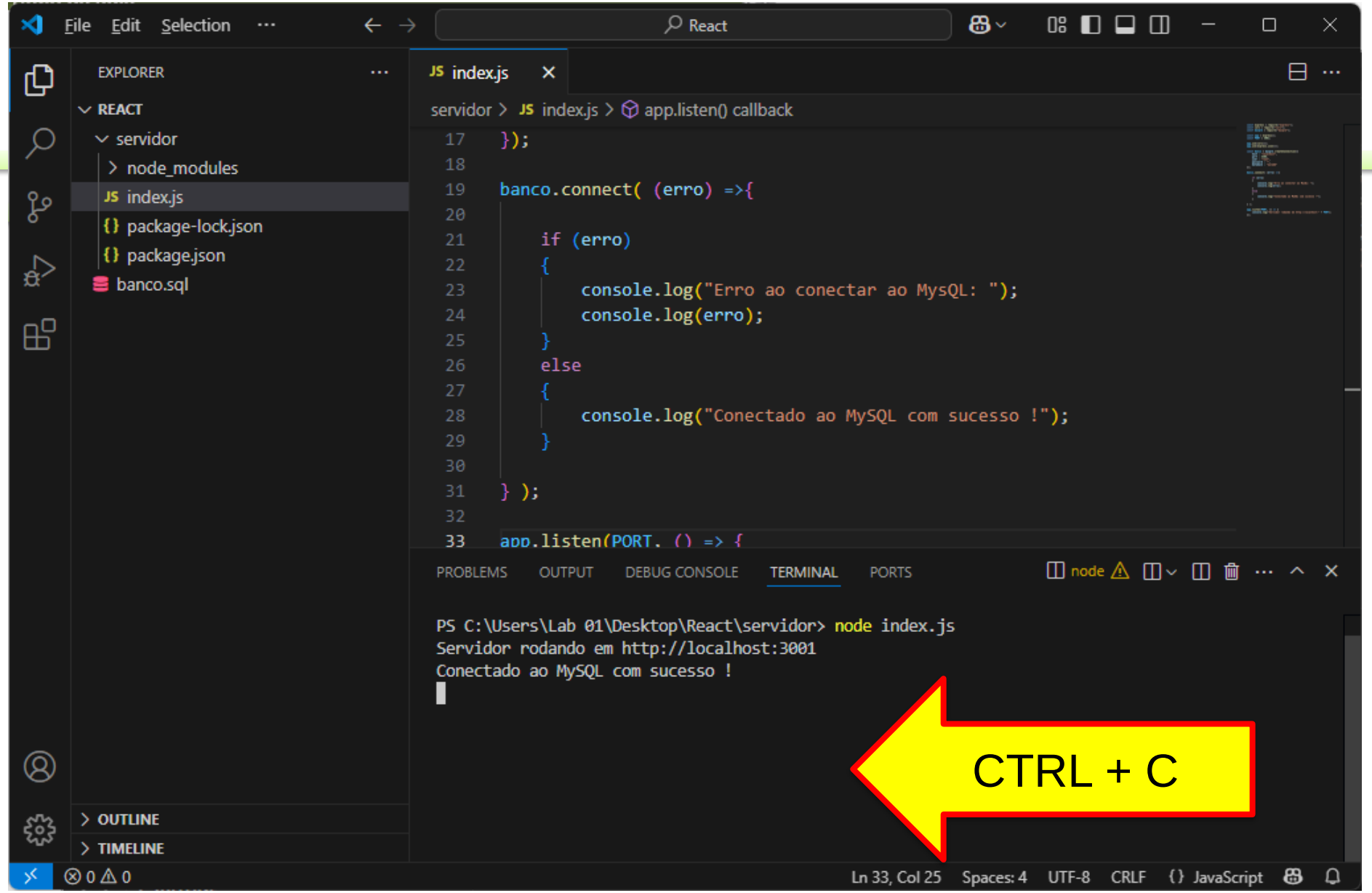
Testando

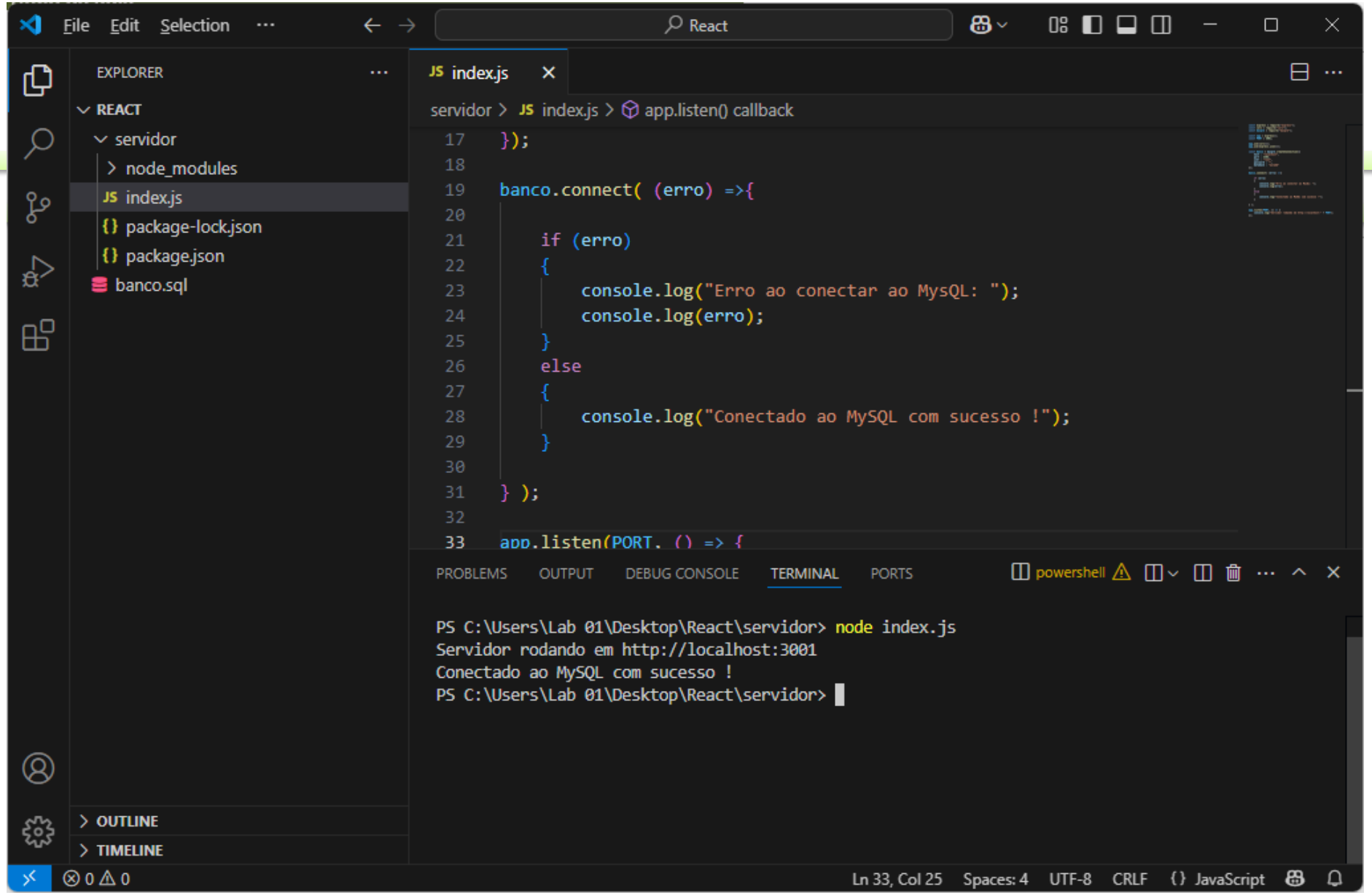
Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br


File Edit Selection
React
EXPLORER
REACT
servidor
node_modules
index.js
package-lock.json
package.json
banco.sql
index.js
servidor > index.js > app.listen() callback
17 });
18
19 banco.connect((erro) =>{
20
21 if (erro)
22 {
23 console.log("Erro ao conectar ao MySQL: ");
24 console.log(erro);
25 }
26 else
27 {
28 console.log("Conectado ao MySQL com sucesso !");
29 }
30
31 });
32
33 app.listen(PORT, () => {
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
PS C:\Users\Lab 01\Desktop\React\servidor>
OUTLINE
TIMELINE
Ln 33, Col 25 Spaces: 4 UTF-8 CRLF JavaScript


File Edit Selection
React
EXPLORER
REACT
servidor
node_modules
index.js
package-lock.json
package.json
banco.sql
index.js
servidor > index.js > app.listen() callback
17 });
18
19 banco.connect((erro) =>{
20
21 if (erro)
22 {
23 console.log("Erro ao conectar ao MySQL: ");
24 console.log(erro);
25 }
26 else
27 {
28 console.log("Conectado ao MySQL com sucesso !");
29 }
30
31 });
32
33 app.listen(PORT, () => {
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
PS C:\Users\Lab 01\Desktop\React\servidor> node index.js
OUTLINE
TIMELINE
Ln 33, Col 25 Spaces: 4 UTF-8 CRLF JavaScript

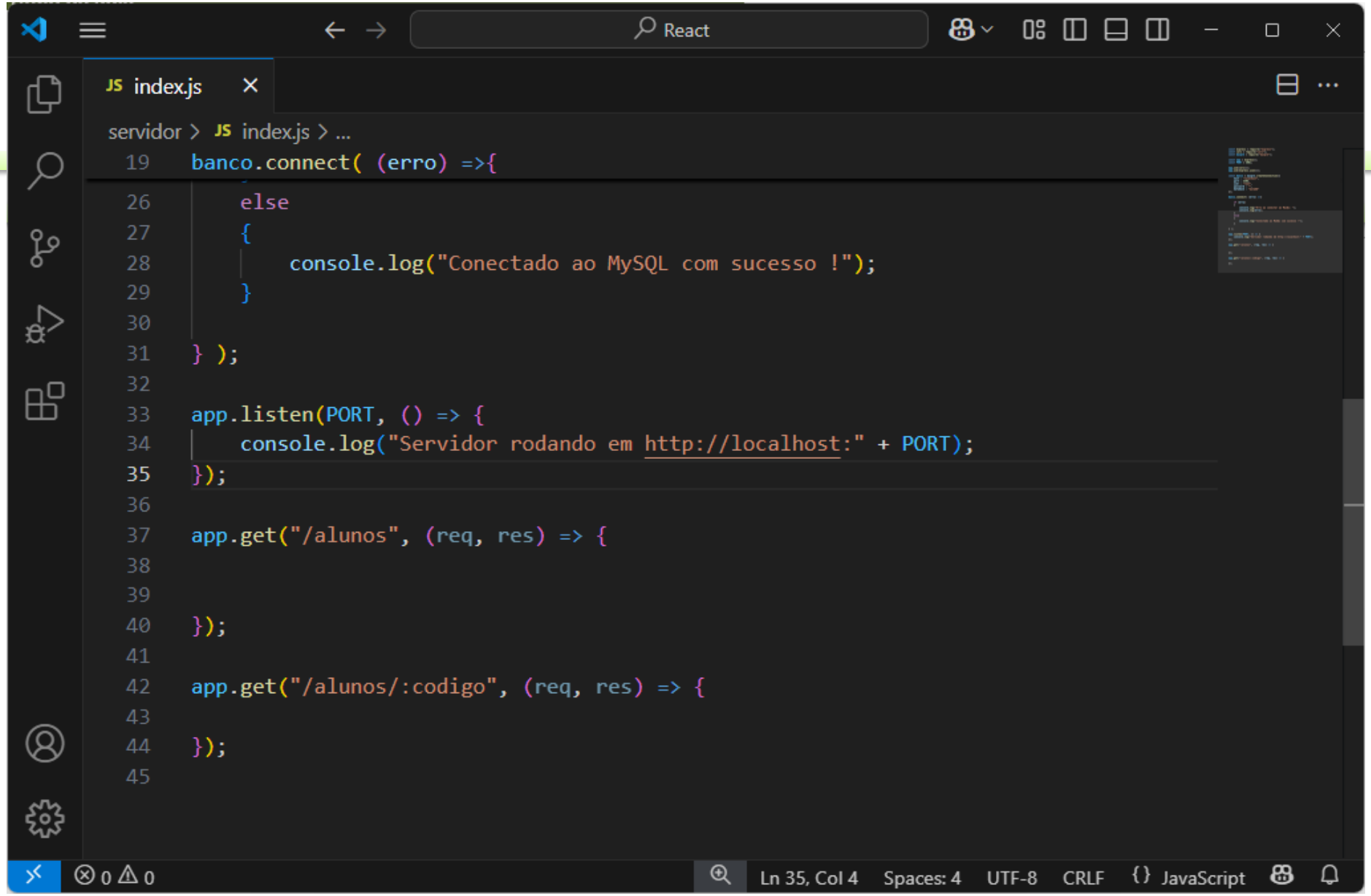

File Edit Selection
React
EXPLORER
REACT
servidor
node_modules
index.js
package-lock.json
package.json
banco.sql
index.js
servidor > index.js > app.listen() callback
17 });
18
19 banco.connect((erro) =>{
20
21 if (erro)
22 {
23 console.log("Erro ao conectar ao MySQL: ");
24 console.log(erro);
25 }
26 else
27 {
28 console.log("Conectado ao MySQL com sucesso !");
29 }
30
31 });
32
33 app.listen(PORT, () => {
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
node
PS C:\Users\Lab 01\Desktop\React\servidor> node index.js
Servidor rodando em http://localhost:3001
Conectado ao MySQL com sucesso !
OUTLINE
TIMELINE
Ln 33, Col 25 Spaces: 4 UTF-8 CRLF JavaScript

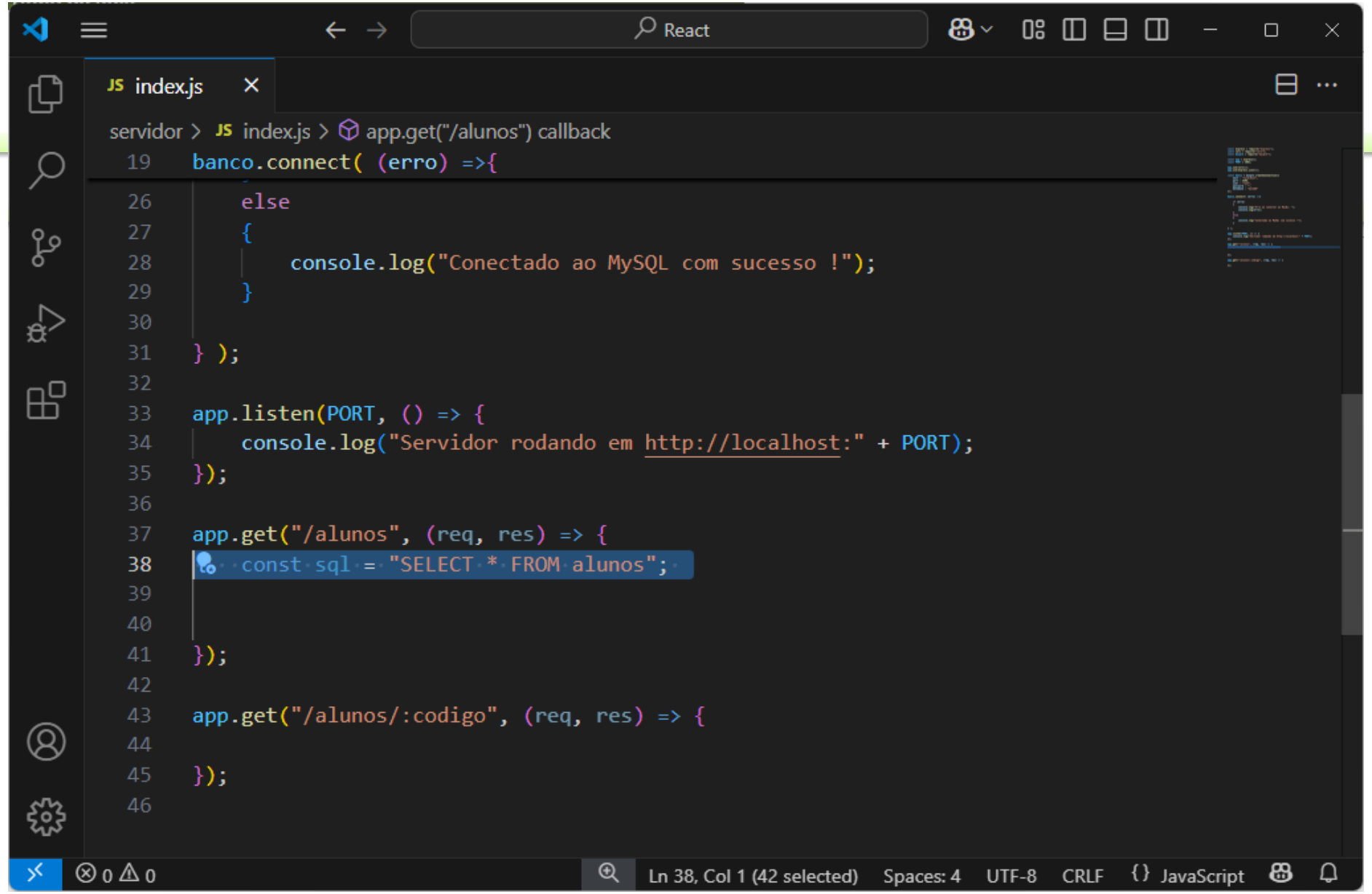

File Edit Selection
React
EXPLORER
REACT
servidor
node_modules
index.js
package-lock.json
package.json
banco.sql
index.js
servidor > index.js > app.listen() callback
17 });
18
19 banco.connect((erro) =>{
20
21 if (erro)
22 {
23 console.log("Erro ao conectar ao MySQL: ");
24 console.log(erro);
25 }
26 else
27 {
28 console.log("Conectado ao MySQL com sucesso !");
29 }
30
31 });
32
33 app.listen(PORT, () => {
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
node
PS C:\Users\Lab 01\Desktop\React\servidor> node index.js
Servidor rodando em http://localhost:3001
Conectado ao MySQL com sucesso !
CTRL + C
OUTLINE
TIMELINE
Ln 33, Col 25 Spaces: 4 UTF-8 CRLF {} JavaScript

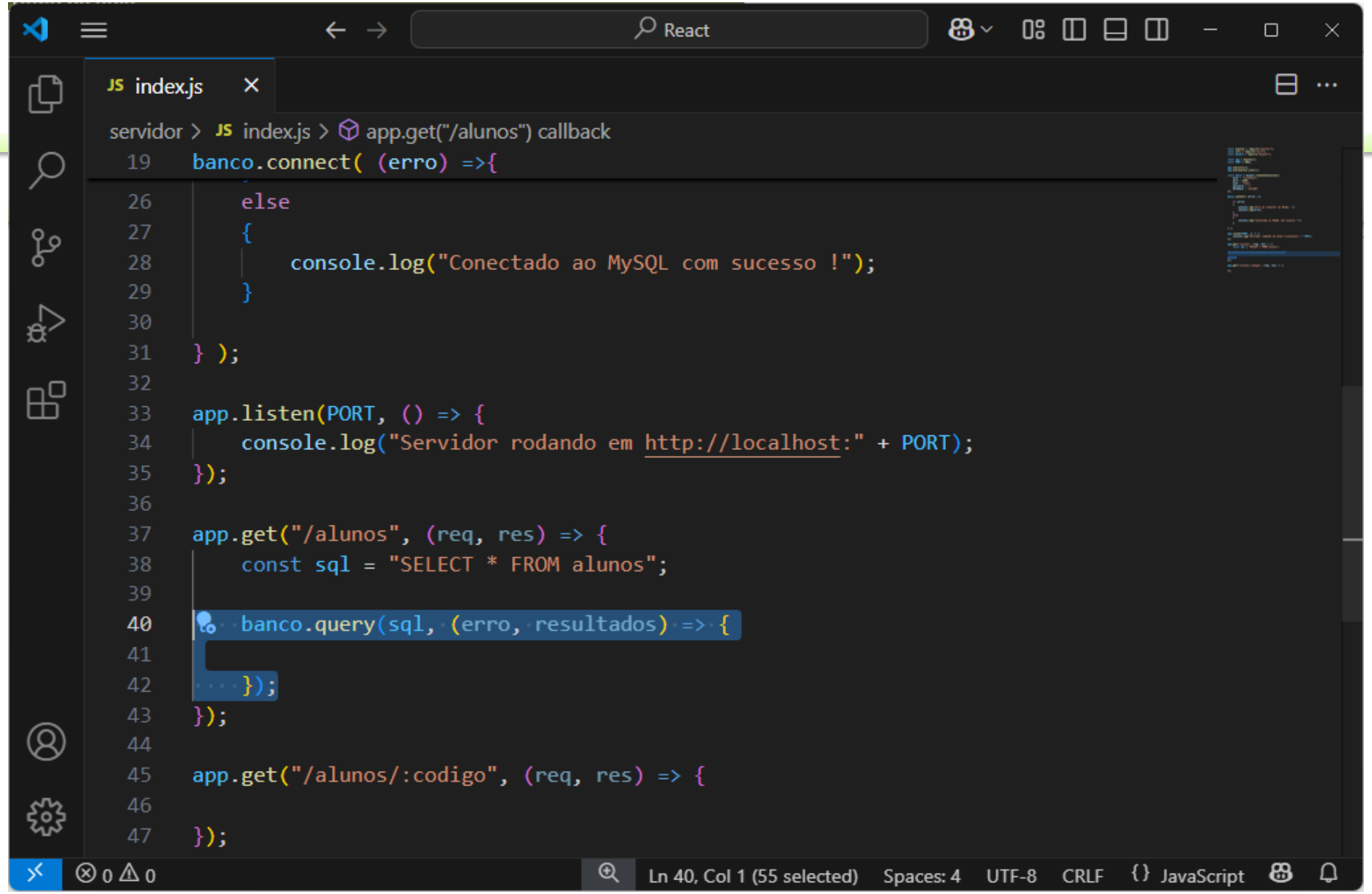

File Edit Selection
React
EXPLORER
REACT
servidor
node_modules
index.js
package-lock.json
package.json
banco.sql
index.js
servidor > index.js > app.listen() callback
17 });
18
19 banco.connect((erro) =>{
20
21 if (erro)
22 {
23 console.log("Erro ao conectar ao MySQL: ");
24 console.log(erro);
25 }
26 else
27 {
28 console.log("Conectado ao MySQL com sucesso !");
29 }
30
31 });
32
33 app.listen(PORT, () => {
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
powershell
PS C:\Users\Lab 01\Desktop\React\servidor> node index.js
Servidor rodando em http://localhost:3001
Conectado ao MySQL com sucesso !
PS C:\Users\Lab 01\Desktop\React\servidor>
OUTLINE
TIMELINE
Ln 33, Col 25 Spaces: 4 UTF-8 CRLF JavaScript

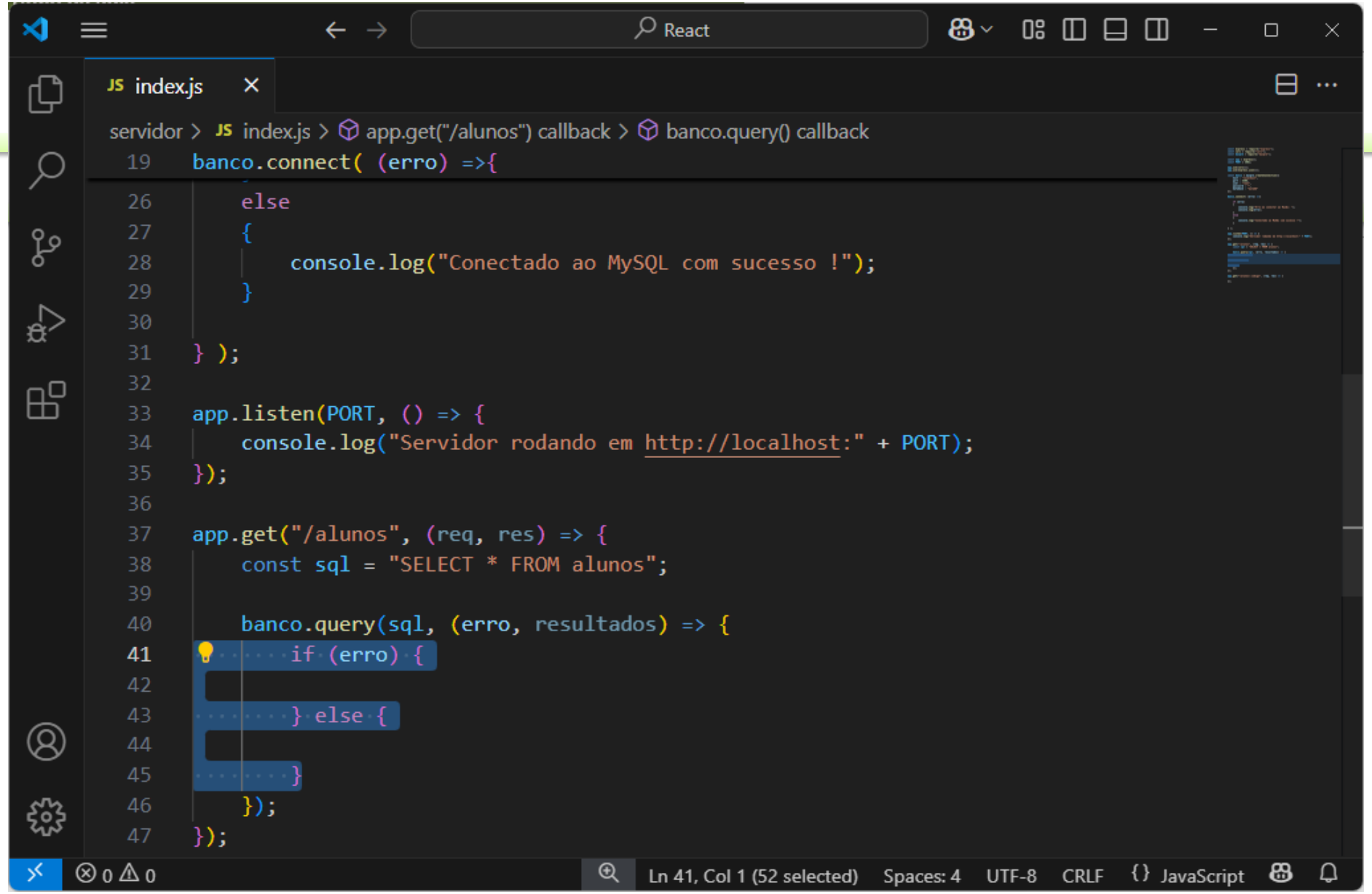
Consultar Aluno

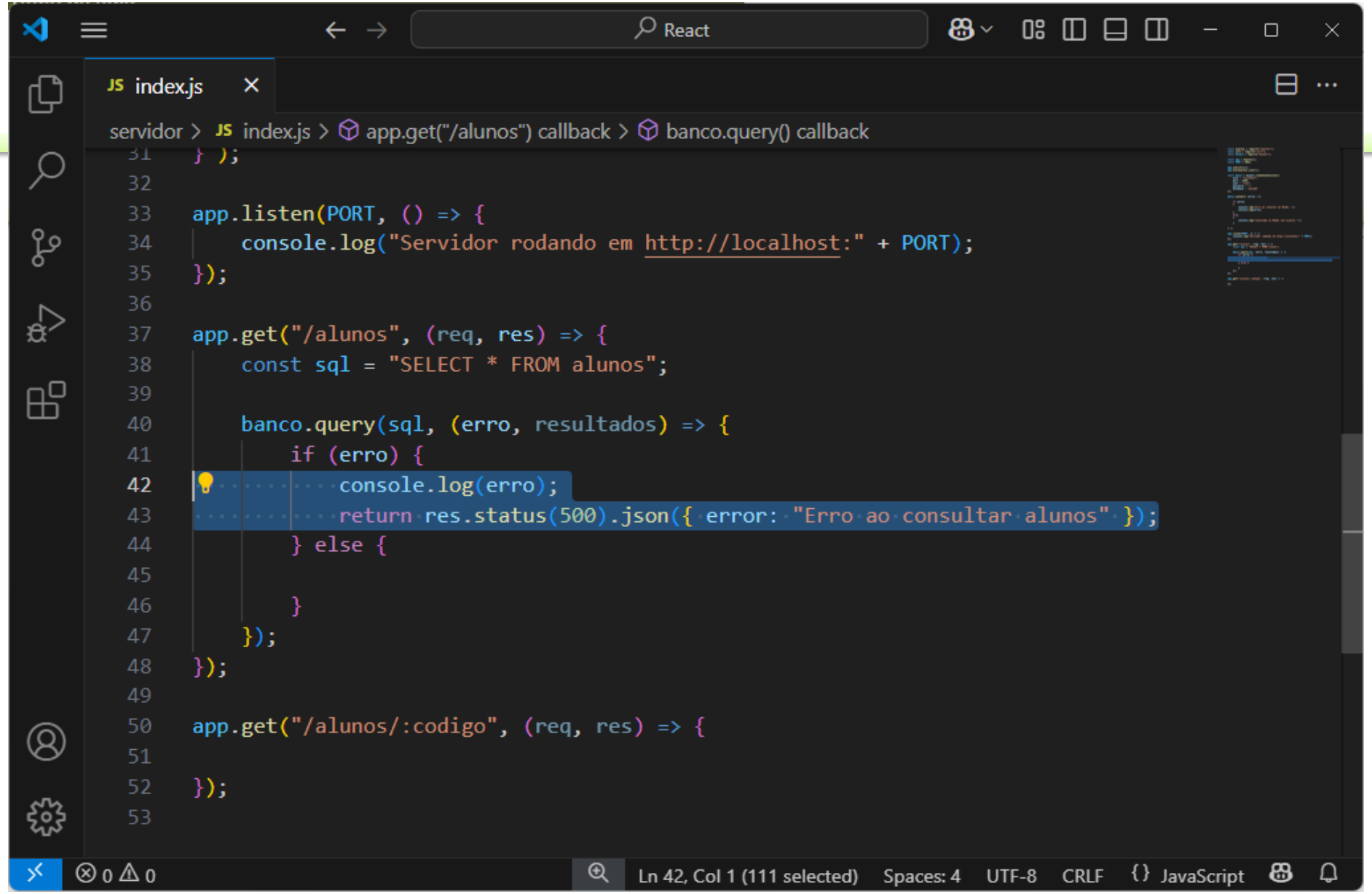
Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br

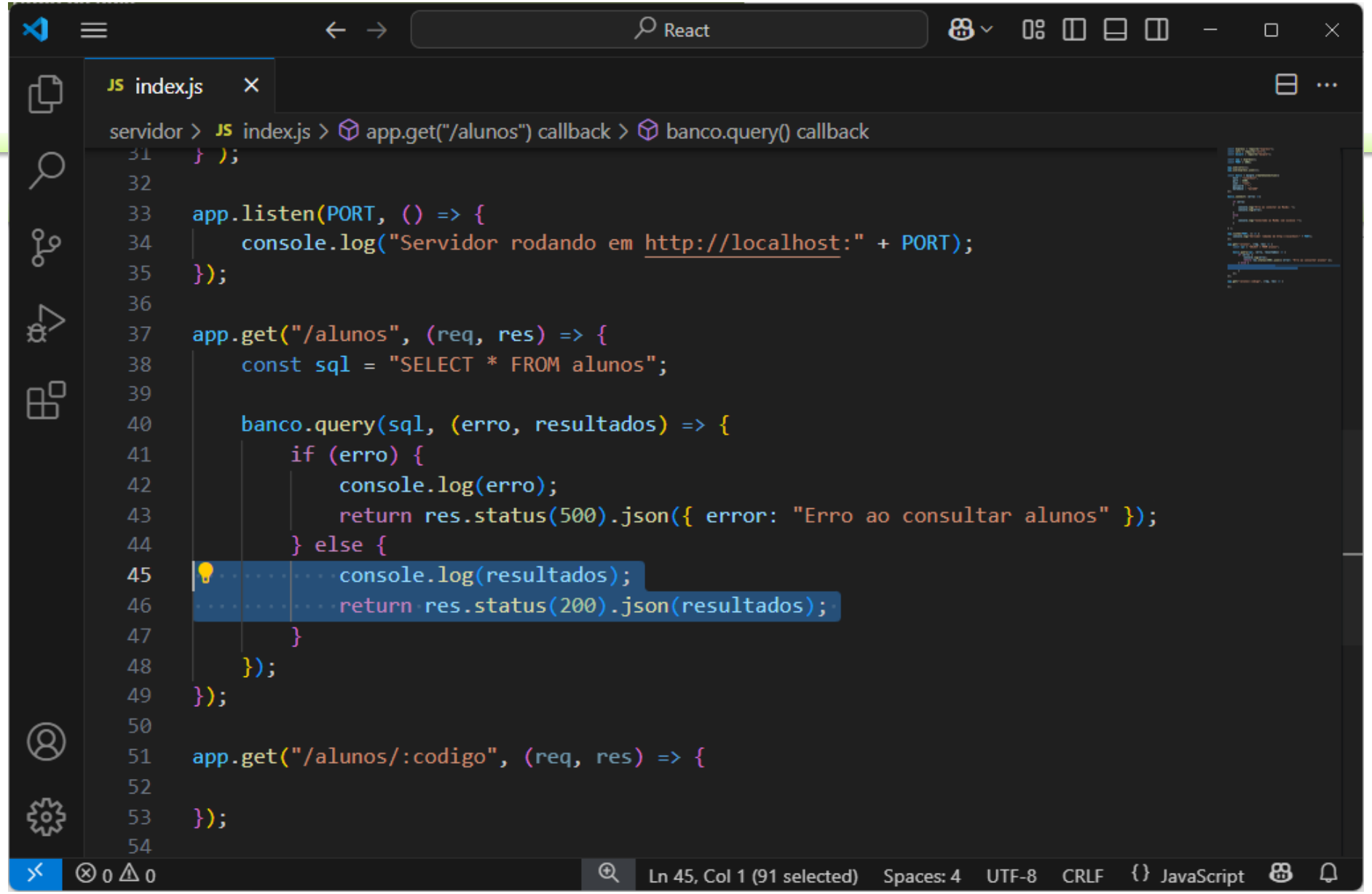

React
index.js
servidor > index.js > ...
19 banco.connect((erro) =>{
26 else
27 {
28 console.log("Conectado ao MySQL com sucesso !");
29 }
30
31 });
32
33 app.listen(PORT, () => {
34 console.log("Servidor rodando em http://localhost:" + PORT);
35 });
36
37 app.get("/alunos", (req, res) => {
38
39
40 });
41
42 app.get("/alunos/:codigo", (req, res) => {
43
44 });
45
Ln 35, Col 4
Spaces: 4
UTF-8
CRLF
JavaScript


React
index.js
servidor > index.js > app.get("/alunos") callback
banco.connect((erro) =>{
else
{
console.log("Conectado ao MySQL com sucesso !");
}
});
app.listen(PORT, () => {
console.log("Servidor rodando em http://localhost:" + PORT);
});
app.get("/alunos", (req, res) => {
const sql = "SELECT * FROM alunos";
});
app.get("/alunos/:codigo", (req, res) => {
});
Ln 38, Col 1 (42 selected)
Spaces: 4
UTF-8
CRLF
JavaScript


React
index.js
servidor > index.js > app.get("/alunos") callback
19 banco.connect((erro) =>{
26 else
27 {
28 console.log("Conectado ao MySQL com sucesso !");
29 }
30
31 });
32
33 app.listen(PORT, () => {
34 console.log("Servidor rodando em http://localhost:" + PORT);
35 });
36
37 app.get("/alunos", (req, res) => {
38 const sql = "SELECT * FROM alunos";
39
40 banco.query(sql, (erro, resultados) => {
41
42 });
43 });
44
45 app.get("/alunos/:codigo", (req, res) => {
46
47 });
0 0
Ln 40, Col 1 (55 selected)
Spaces: 4
UTF-8
CRLF
JavaScript

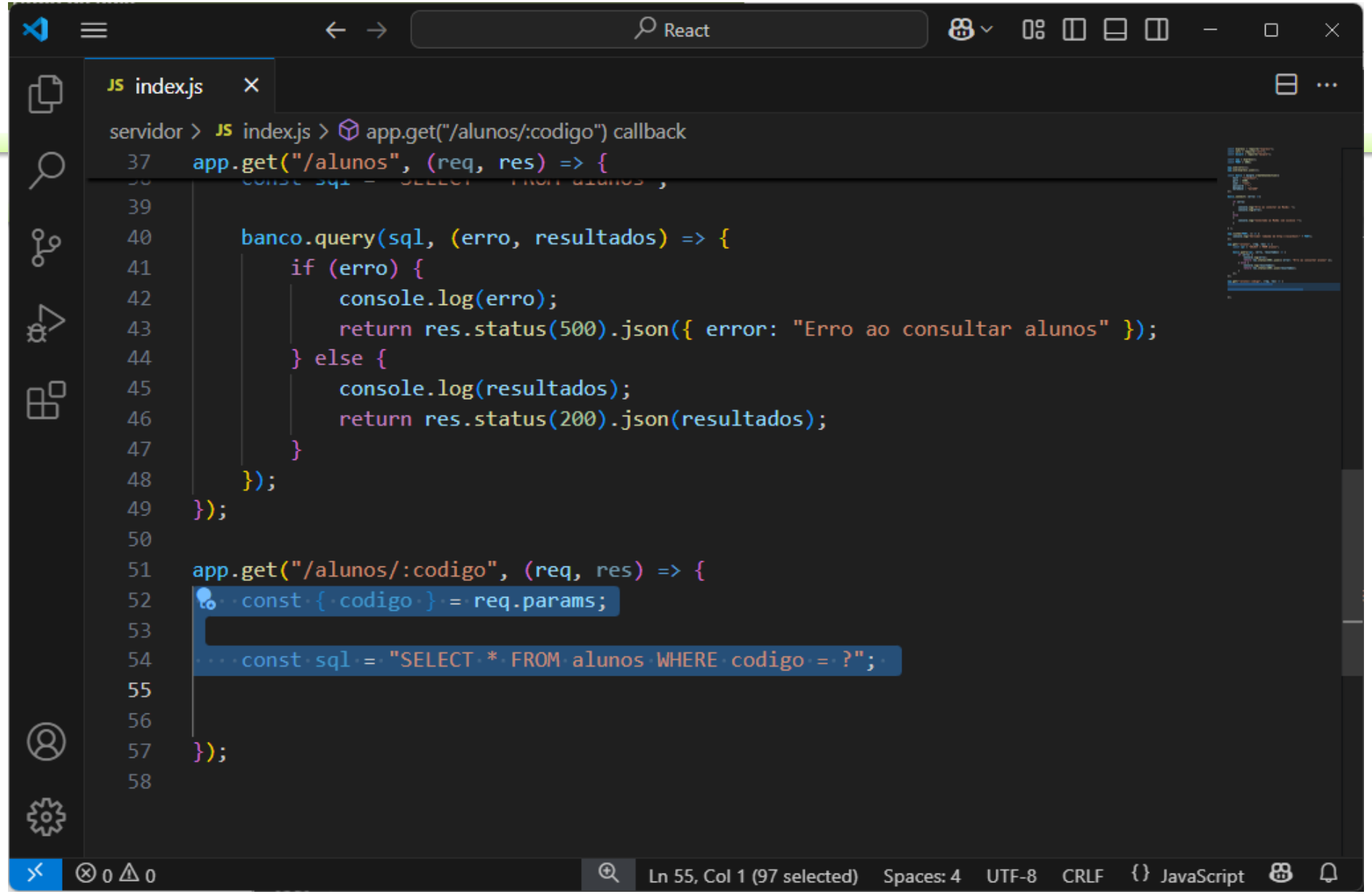

React
index.js
servidor > index.js > app.get("/alunos") callback > banco.query() callback
banco.connect((erro) =>{
else
{
console.log("Conectado ao MySQL com sucesso !");
}
});
app.listen(PORT, () => {
console.log("Servidor rodando em http://localhost:" + PORT);
});
app.get("/alunos", (req, res) => {
const sql = "SELECT * FROM alunos";
banco.query(sql, (erro, resultados) => {
if (erro) {
} else {
}
});
});
Ln 41, Col 1 (52 selected)
Spaces: 4
UTF-8
CRLF
JavaScript

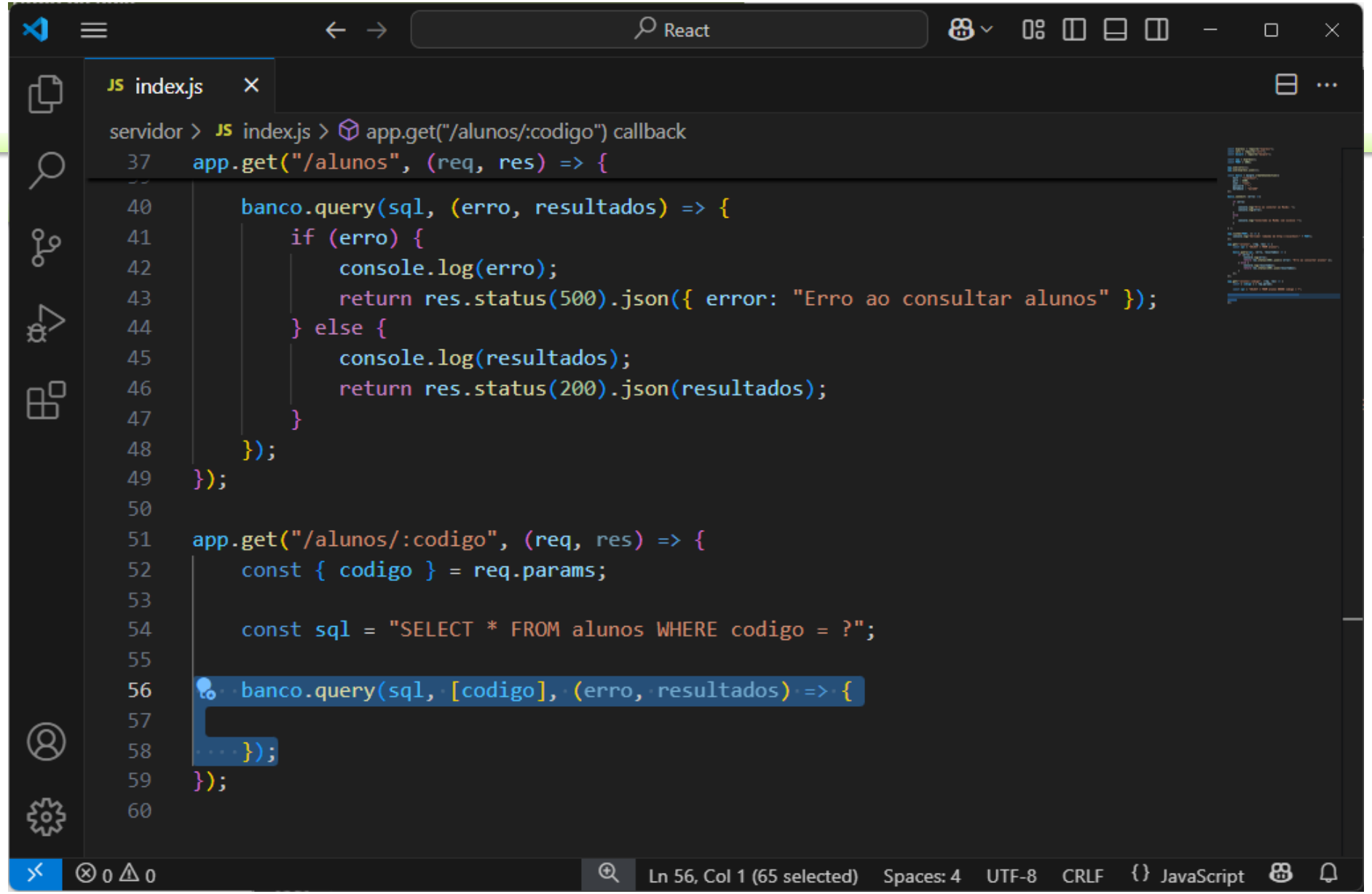

React
index.js
servidor > index.js > app.get("/alunos") callback > banco.query() callback
});
app.listen(PORT, () => {
console.log("Servidor rodando em http://localhost:" + PORT);
});
app.get("/alunos", (req, res) => {
const sql = "SELECT * FROM alunos";
banco.query(sql, (erro, resultados) => {
if (erro) {
console.log(erro);
return res.status(500).json({ error: "Erro ao consultar alunos" });
} else {
}
});
});
app.get("/alunos/:codigo", (req, res) => {
});
0 0
Ln 42, Col 1 (111 selected)
Spaces: 4
UTF-8
CRLF
JavaScript

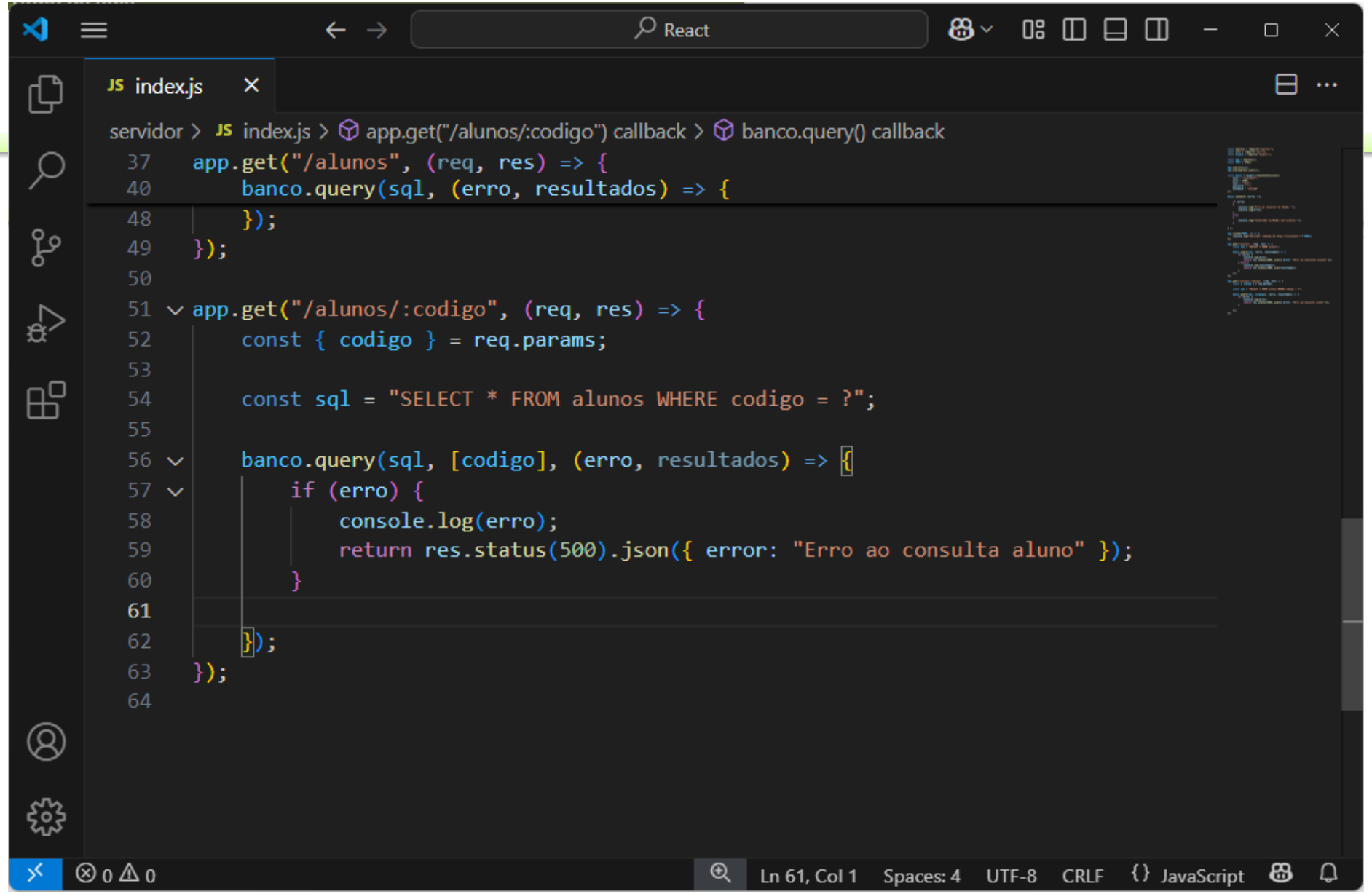

React
index.js
servidor > index.js > app.get("/alunos") callback > banco.query() callback
});
app.listen(PORT, () => {
console.log("Servidor rodando em http://localhost:" + PORT);
});
app.get("/alunos", (req, res) => {
const sql = "SELECT * FROM alunos";
banco.query(sql, (erro, resultados) => {
if (erro) {
console.log(erro);
return res.status(500).json({ error: "Erro ao consultar alunos" });
} else {
console.log(resultados);
return res.status(200).json(resultados);
}
});
});
app.get("/alunos/:codigo", (req, res) => {
});
0 0
Ln 45, Col 1 (91 selected)
Spaces: 4
UTF-8
CRLF
JavaScript

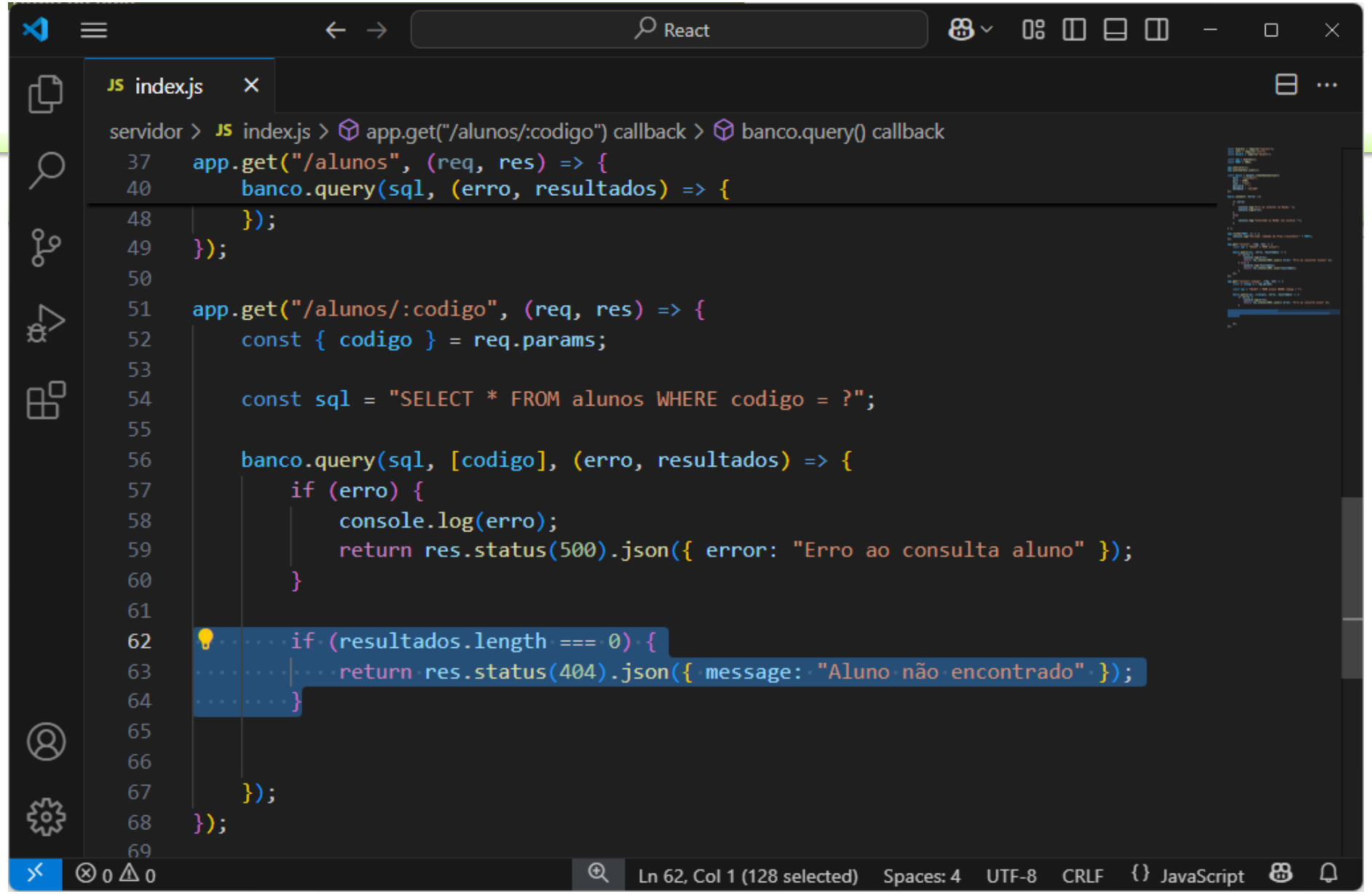
Consultar Aluno Por Código

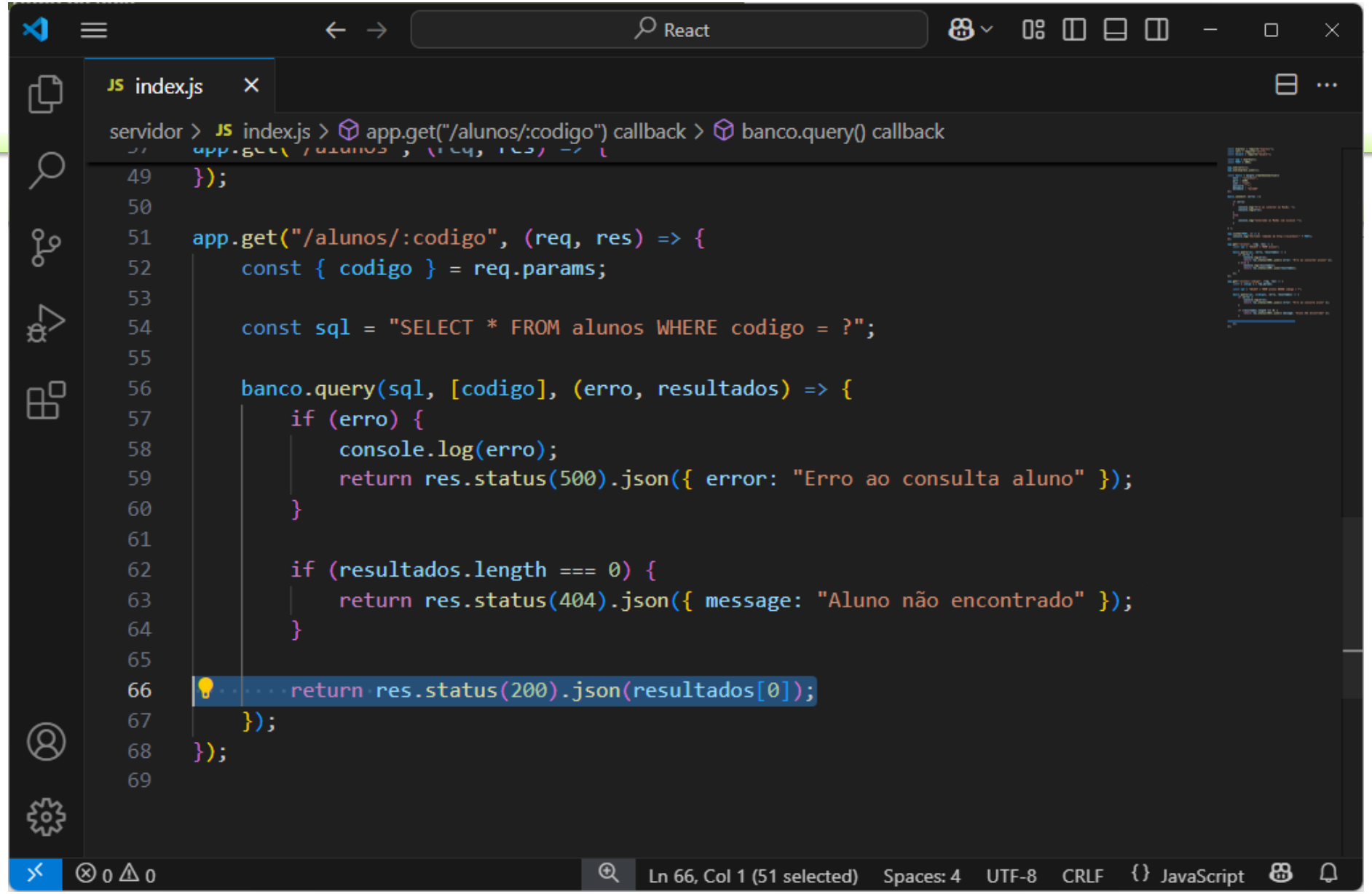
Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br


React
index.js
servidor > index.js > app.get("/alunos/:codigo") callback
app.get("/alunos", (req, res) => {
banco.query(sql, (erro, resultados) => {
if (erro) {
console.log(erro);
return res.status(500).json({ error: "Erro ao consultar alunos" });
} else {
console.log(resultados);
return res.status(200).json(resultados);
}
});
});
app.get("/alunos/:codigo", (req, res) => {
const { codigo } = req.params;
const sql = "SELECT * FROM alunos WHERE codigo = ?";
});
Ln 55, Col 1 (97 selected)
Spaces: 4
UTF-8
CRLF
JavaScript


React
index.js
servidor > index.js > app.get("/alunos/:codigo") callback
app.get("/alunos", (req, res) => {
banco.query(sql, (erro, resultados) => {
if (erro) {
console.log(erro);
return res.status(500).json({ error: "Erro ao consultar alunos" });
} else {
console.log(resultados);
return res.status(200).json(resultados);
}
});
});
app.get("/alunos/:codigo", (req, res) => {
const { codigo } = req.params;
const sql = "SELECT * FROM alunos WHERE codigo = ?";
banco.query(sql, [codigo], (erro, resultados) => {
});
});
Ln 56, Col 1 (65 selected)
Spaces: 4
UTF-8
CRLF
JavaScript

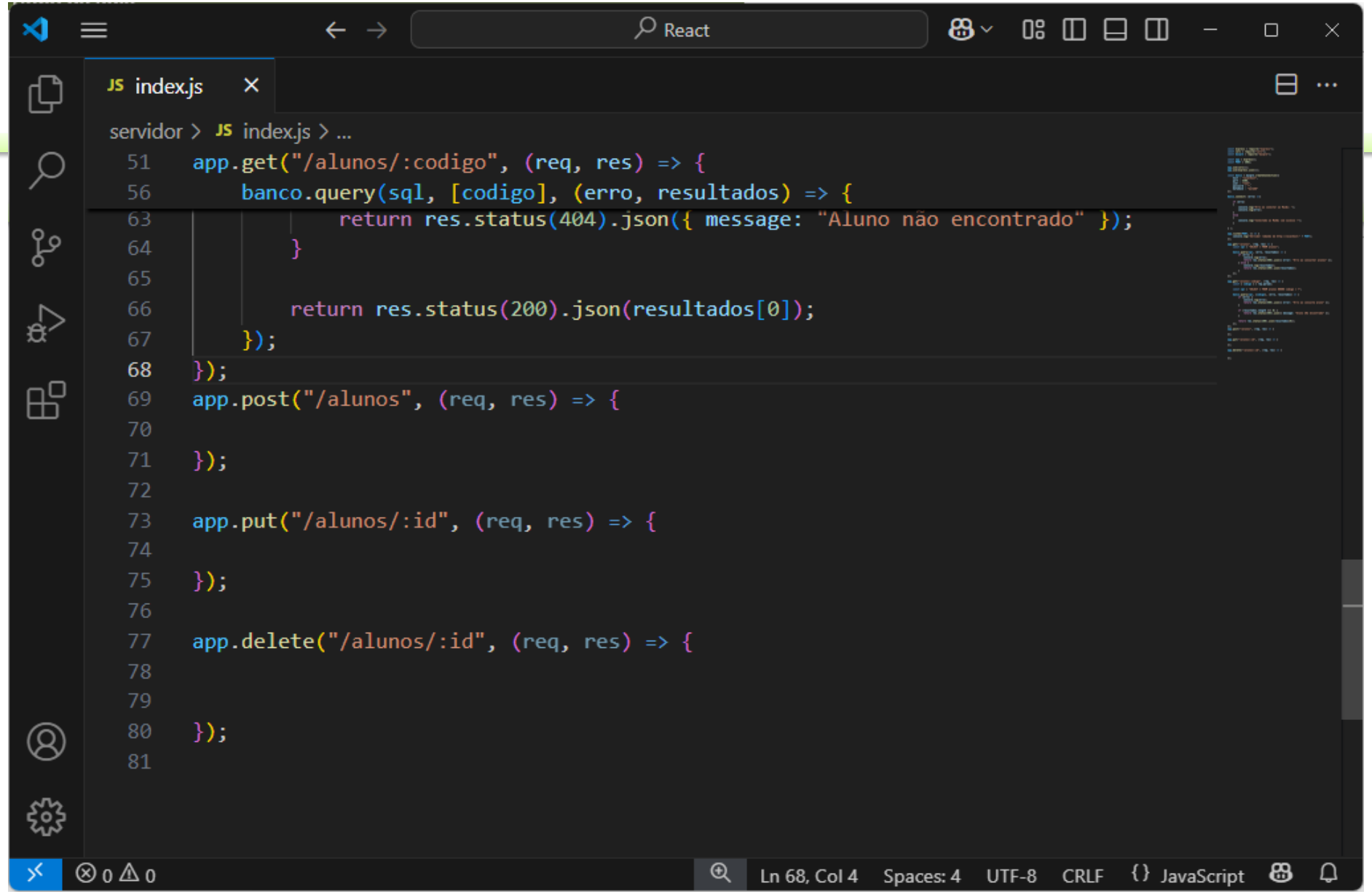

React
index.js
servidor > index.js > app.get("/alunos/:codigo") callback > banco.query() callback
app.get("/alunos", (req, res) => {
banco.query(sql, (erro, resultados) => {
});
});
app.get("/alunos/:codigo", (req, res) => {
const { codigo } = req.params;
const sql = "SELECT * FROM alunos WHERE codigo = ?";
banco.query(sql, [codigo], (erro, resultados) => {
if (erro) {
console.log(erro);
return res.status(500).json({ error: "Erro ao consulta aluno" });
}
});
});
Ln 61, Col 1
Spaces: 4
UTF-8
CRLF
JavaScript


React
index.js
servidor > index.js > app.get("/alunos/:codigo") callback > banco.query() callback
app.get("/alunos", (req, res) => {
banco.query(sql, (erro, resultados) => {
});
});
app.get("/alunos/:codigo", (req, res) => {
const { codigo } = req.params;
const sql = "SELECT * FROM alunos WHERE codigo = ?";
banco.query(sql, [codigo], (erro, resultados) => {
if (erro) {
console.log(erro);
return res.status(500).json({ error: "Erro ao consulta aluno" });
}
if (resultados.length === 0) {
return res.status(404).json({ message: "Aluno não encontrado" });
}
});
});
Ln 62, Col 1 (128 selected)
Spaces: 4
UTF-8
CRLF
JavaScript


React
index.js
servidor > index.js > app.get("/alunos/:codigo") callback > banco.query() callback
});
app.get("/alunos/:codigo", (req, res) => {
const { codigo } = req.params;
const sql = "SELECT * FROM alunos WHERE codigo = ?";
banco.query(sql, [codigo], (erro, resultados) => {
if (erro) {
console.log(erro);
return res.status(500).json({ error: "Erro ao consulta aluno" });
}
if (resultados.length === 0) {
return res.status(404).json({ message: "Aluno não encontrado" });
}
return res.status(200).json(resultados[0]);
});
});
0 0
Ln 66, Col 1 (51 selected)
Spaces: 4
UTF-8
CRLF
JavaScript

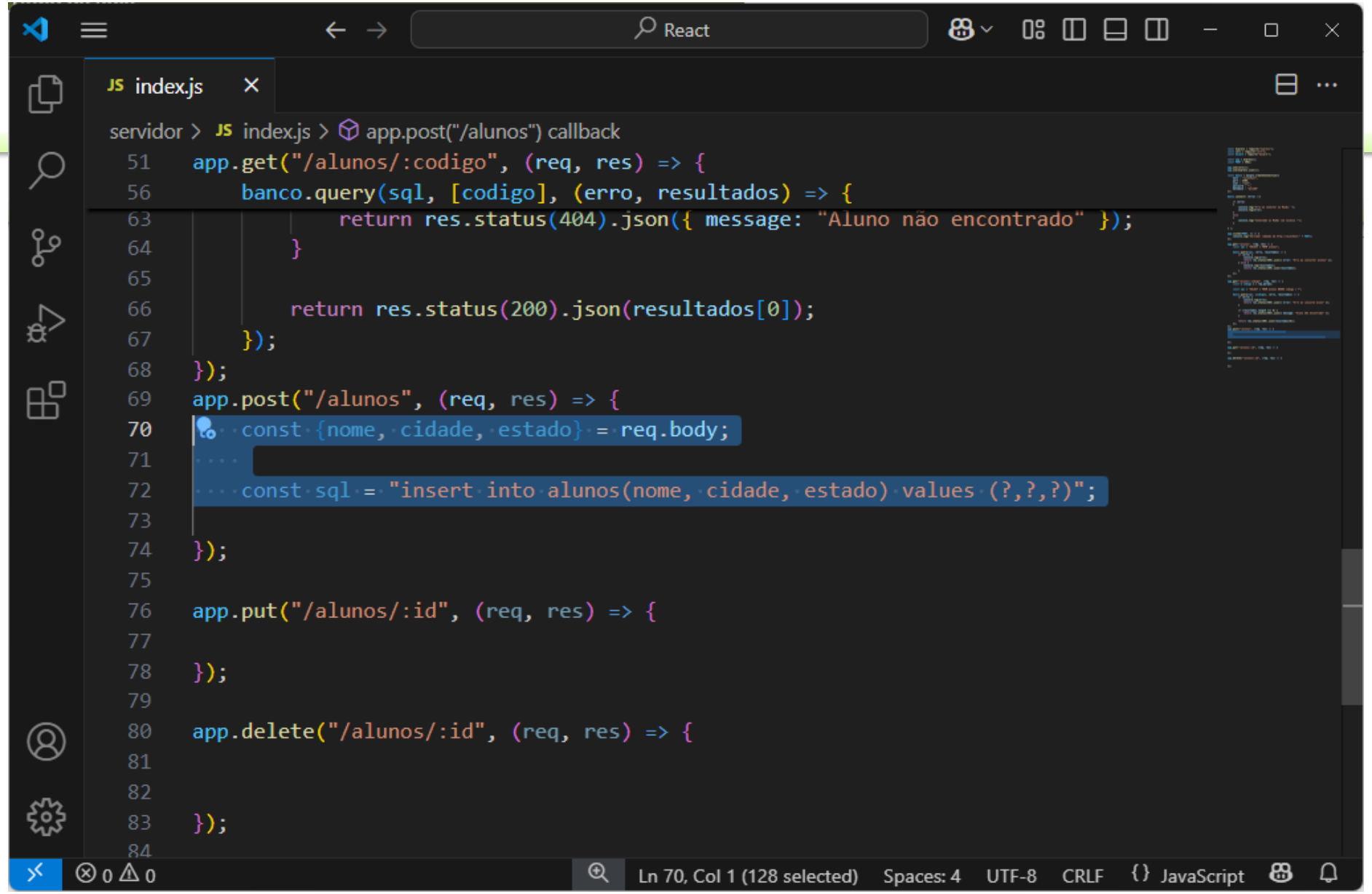
Cadastrar Alterar e Excluir

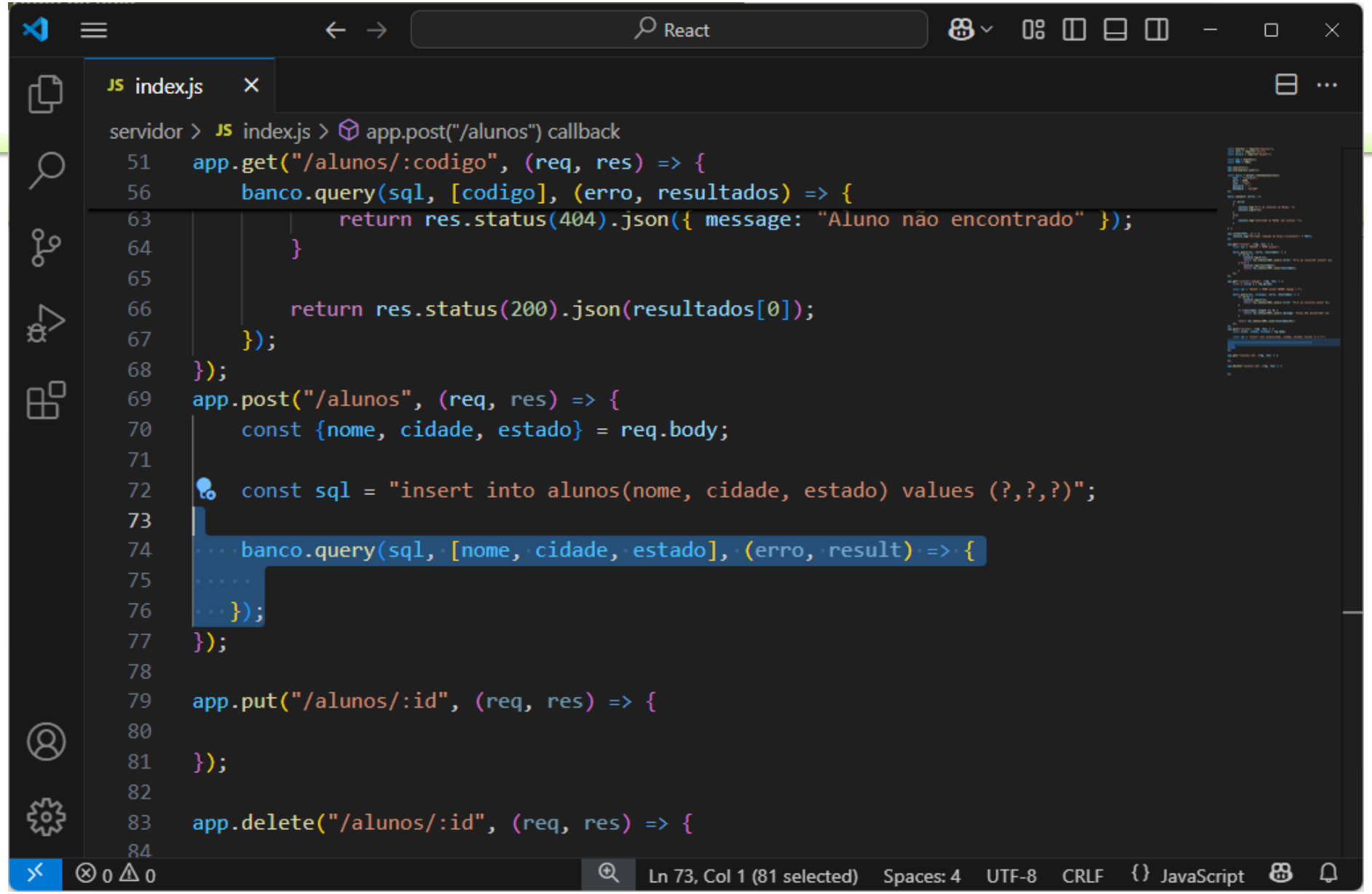
Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br

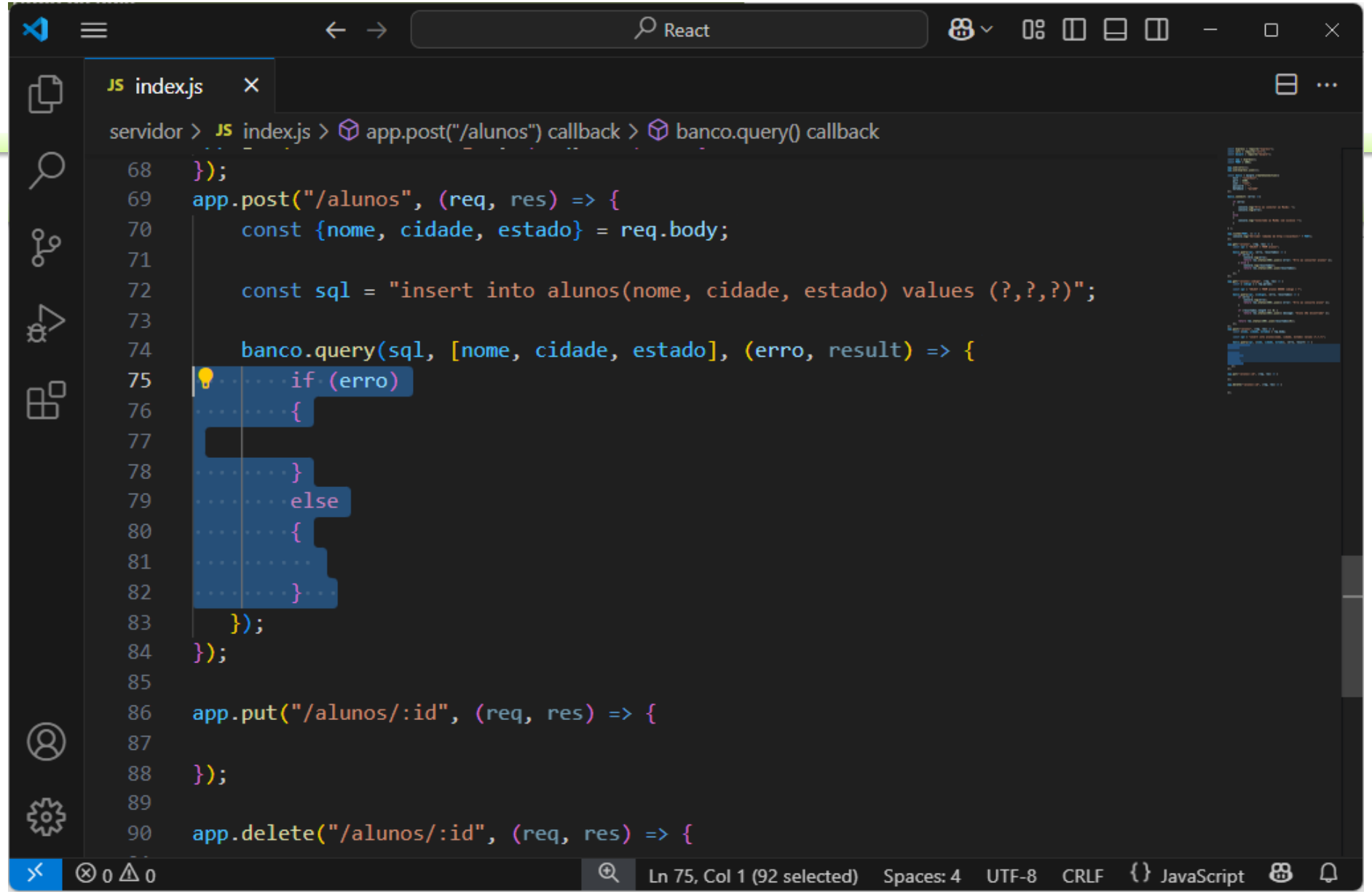

React
index.js
servidor > index.js > ...
51 app.get("/alunos/:codigo", (req, res) => {
56 banco.query(sql, [codigo], (erro, resultados) => {
63 return res.status(404).json({ message: "Aluno não encontrado" });
64 }
65
66 return res.status(200).json(resultados[0]);
67 });
68 });
69 app.post("/alunos", (req, res) => {
70
71 });
72
73 app.put("/alunos/:id", (req, res) => {
74
75 });
76
77 app.delete("/alunos/:id", (req, res) => {
78
79
80 });
81
0 0
Ln 68, Col 4
Spaces: 4
UTF-8
CRLF
JavaScript

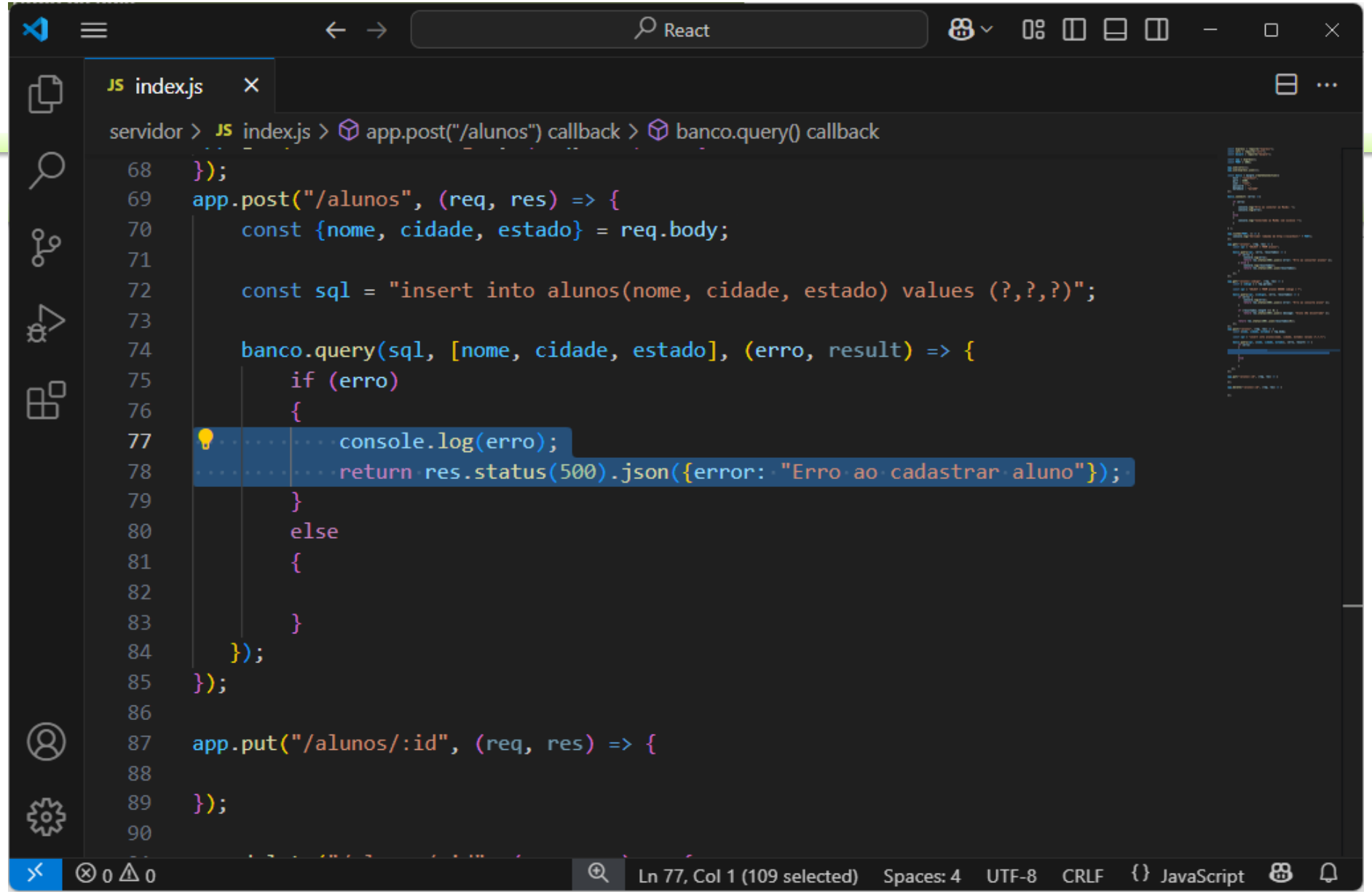
Cadastrar Aluno

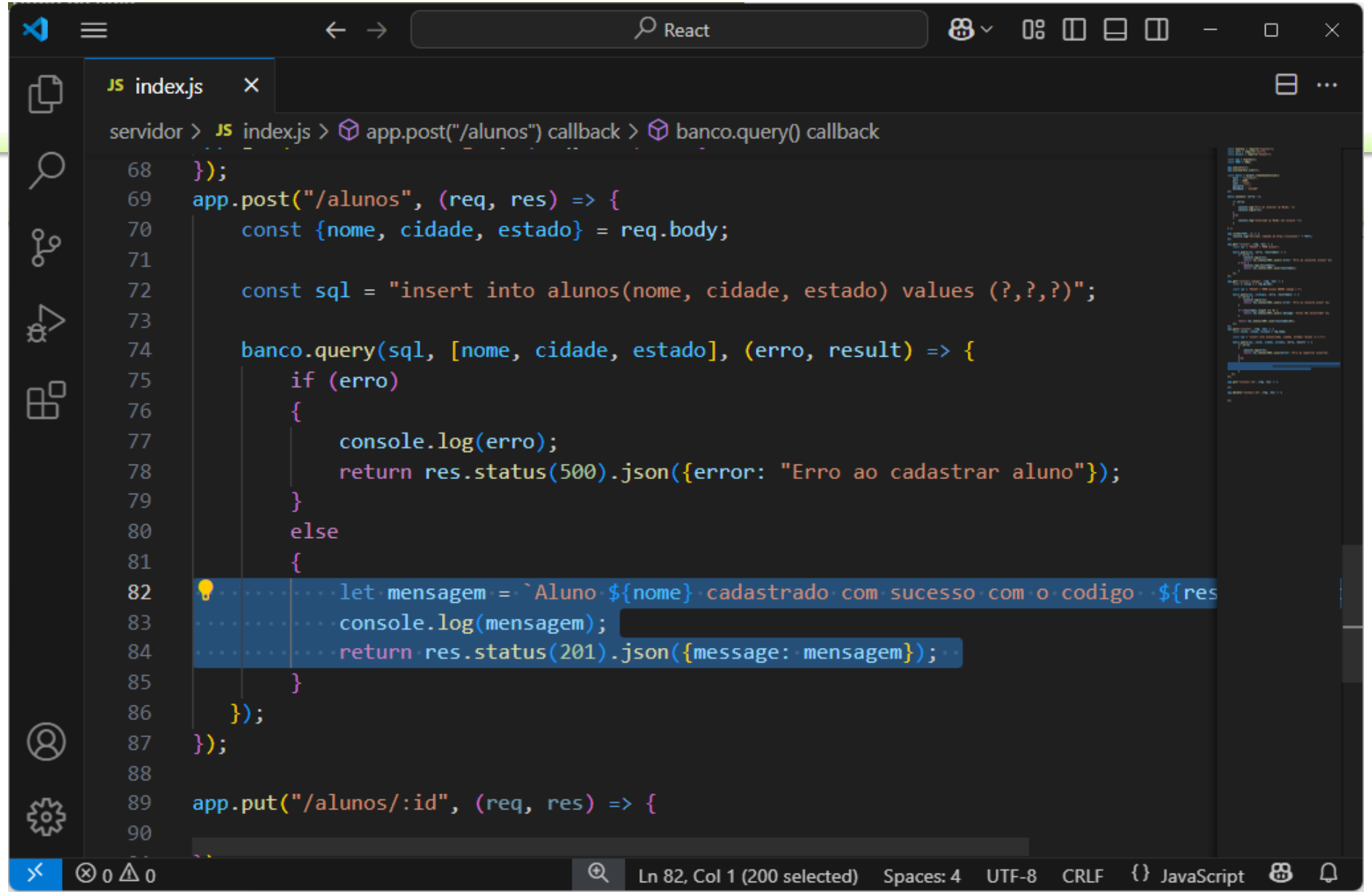
Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br


React
index.js
servidor > index.js > app.post("/alunos") callback
app.get("/alunos/:codigo", (req, res) => {
banco.query(sql, [codigo], (erro, resultados) => {
return res.status(404).json({ message: "Aluno não encontrado" });
}
return res.status(200).json(resultados[0]);
});
});
app.post("/alunos", (req, res) => {
const {nome, cidade, estado} = req.body;
const sql = "insert into alunos(nome, cidade, estado) values (?,?,?)";
});
app.put("/alunos/:id", (req, res) => {
});
app.delete("/alunos/:id", (req, res) => {
});
Ln 70, Col 1 (128 selected)
Spaces: 4
UTF-8
CRLF
JavaScript


React
index.js
servidor > index.js > app.post("/alunos") callback
app.get("/alunos/:codigo", (req, res) => {
banco.query(sql, [codigo], (erro, resultados) => {
return res.status(404).json({ message: "Aluno não encontrado" });
}
return res.status(200).json(resultados[0]);
});
});
app.post("/alunos", (req, res) => {
const {nome, cidade, estado} = req.body;
const sql = "insert into alunos(nome, cidade, estado) values (?,?,?)";
banco.query(sql, [nome, cidade, estado], (erro, result) => {
});
});
app.put("/alunos/:id", (req, res) => {
});
app.delete("/alunos/:id", (req, res) => {
Ln 73, Col 1 (81 selected)
Spaces: 4
UTF-8
CRLF
JavaScript

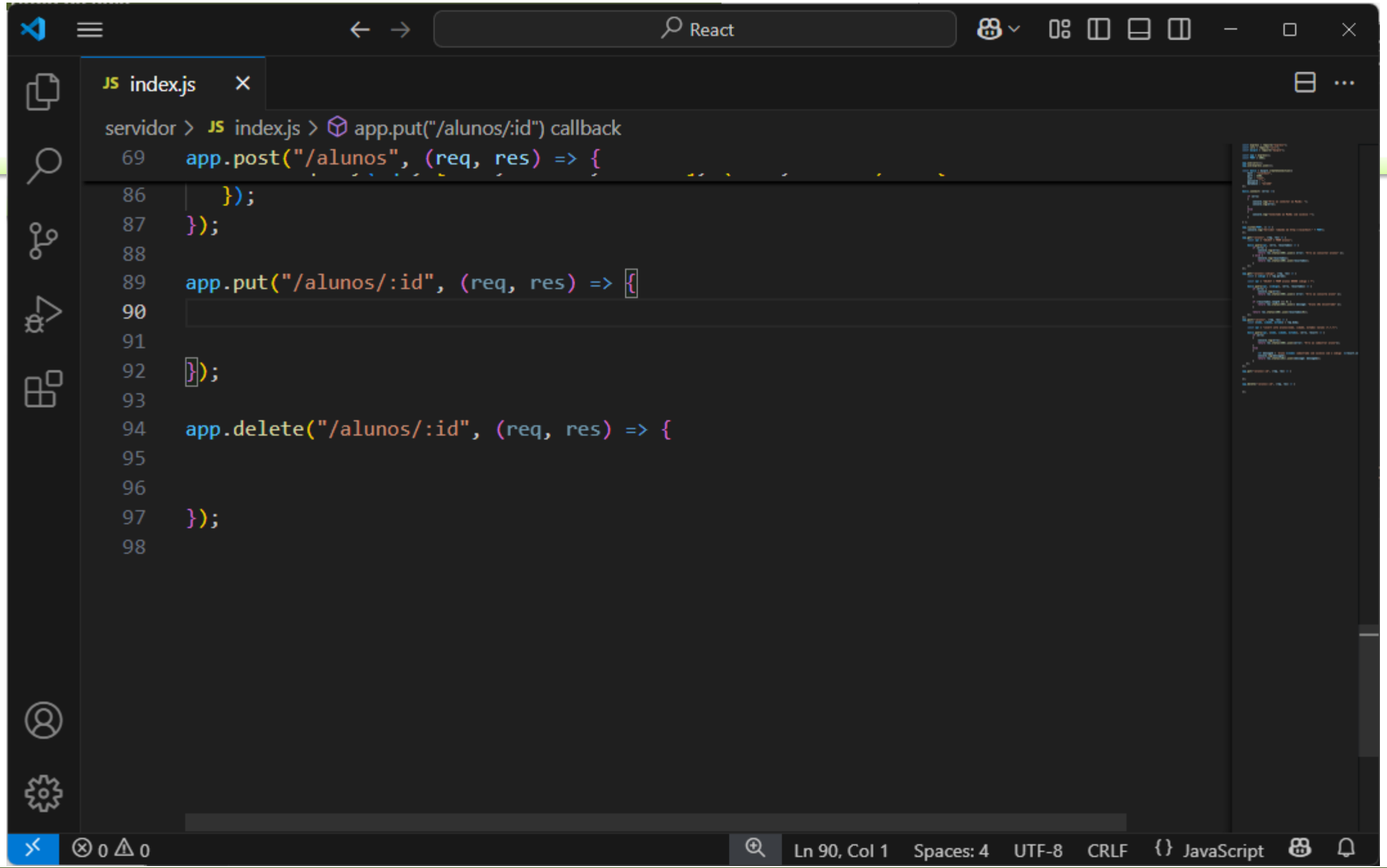

React
index.js
servidor > index.js > app.post("/alunos") callback > banco.query() callback
68 });
69 app.post("/alunos", (req, res) => {
70 const {nome, cidade, estado} = req.body;
71
72 const sql = "insert into alunos(nome, cidade, estado) values (?,?,?)";
73
74 banco.query(sql, [nome, cidade, estado], (erro, result) => {
75 if (erro)
76 {
77
78 }
79 else
80 {
81
82 }
83 });
84 });
85
86 app.put("/alunos/:id", (req, res) => {
87
88 });
89
90 app.delete("/alunos/:id", (req, res) => {
Ln 75, Col 1 (92 selected)
Spaces: 4
UTF-8
CRLF
JavaScript

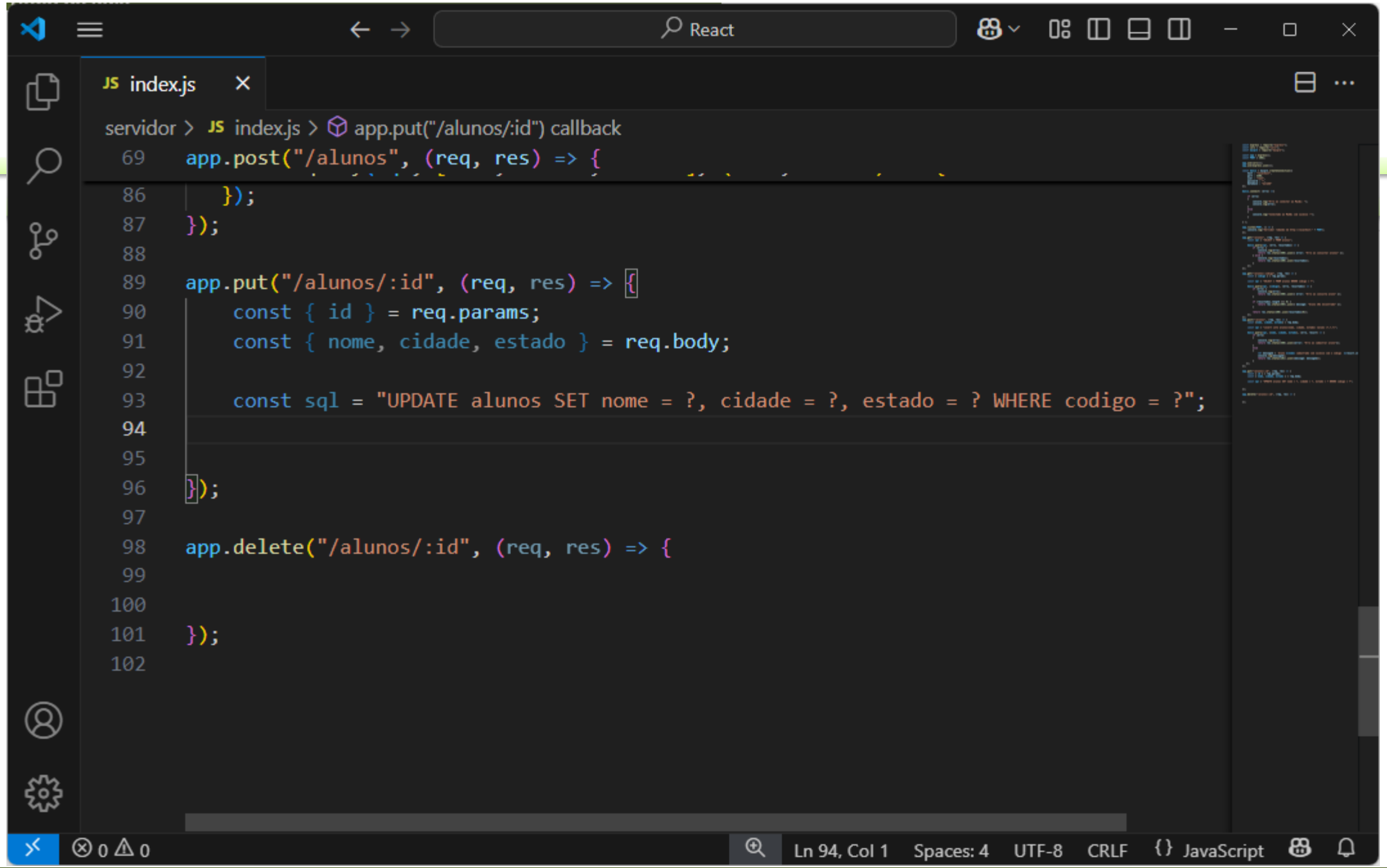

React
index.js
servidor > index.js > app.post("/alunos") callback > banco.query() callback
68 });
69 app.post("/alunos", (req, res) => {
70 const {nome, cidade, estado} = req.body;
71
72 const sql = "insert into alunos(nome, cidade, estado) values (?,?,?)";
73
74 banco.query(sql, [nome, cidade, estado], (erro, result) => {
75 if (erro)
76 {
77 console.log(erro);
78 return res.status(500).json({error: "Erro ao cadastrar aluno"});
79 }
80 else
81 {
82
83 }
84 });
85 });
86
87 app.put("/alunos/:id", (req, res) => {
88
89 });
90
0 0
Ln 77, Col 1 (109 selected)
Spaces: 4
UTF-8
CRLF
JavaScript

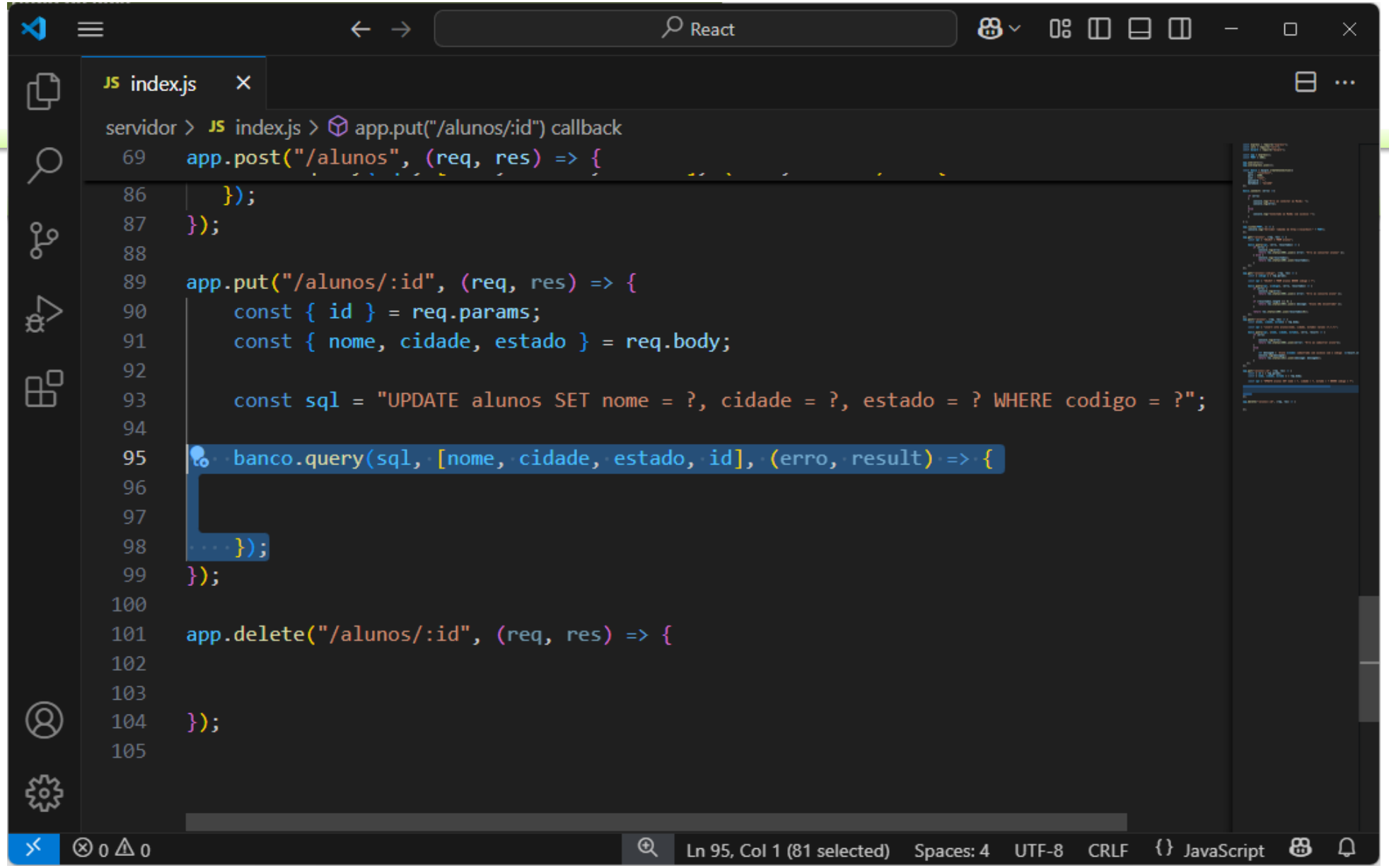

React
index.js
servidor > index.js > app.post("/alunos") callback > banco.query() callback
68 });
69 app.post("/alunos", (req, res) => {
70 const {nome, cidade, estado} = req.body;
71
72 const sql = "insert into alunos(nome, cidade, estado) values (?,?,?)";
73
74 banco.query(sql, [nome, cidade, estado], (erro, result) => {
75 if (erro)
76 {
77 console.log(erro);
78 return res.status(500).json({error: "Erro ao cadastrar aluno"});
79 }
80 else
81 {
82 let mensagem = `Aluno ${nome} cadastrado com sucesso com o codigo ${res
83 console.log(mensagem);
84 return res.status(201).json({message: mensagem});
85 }
86 });
87 });
88
89 app.put("/alunos/:id", (req, res) => {
90
0 0
Ln 82, Col 1 (200 selected)
Spaces: 4
UTF-8
CRLF
JavaScript

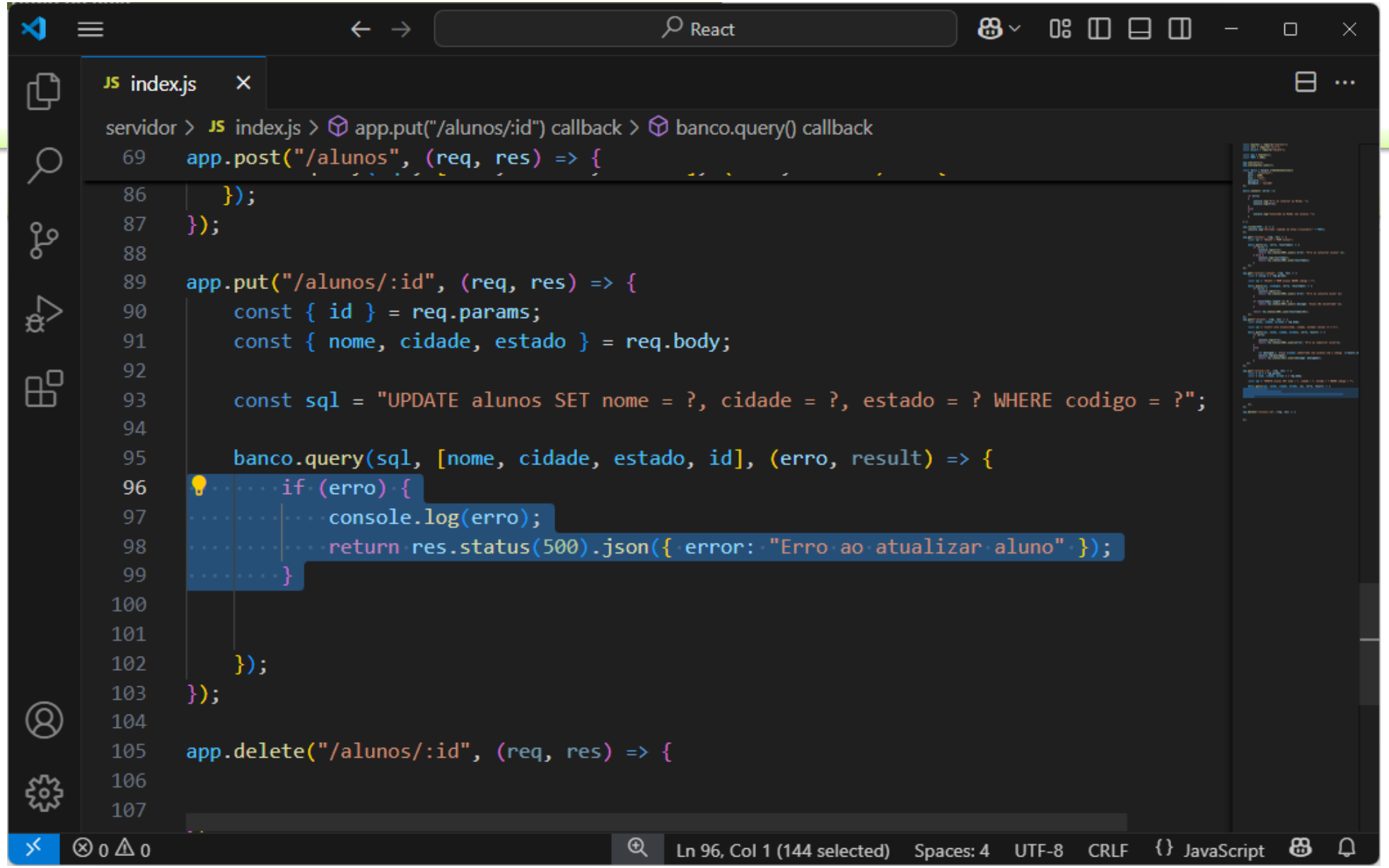
Alterar Aluno

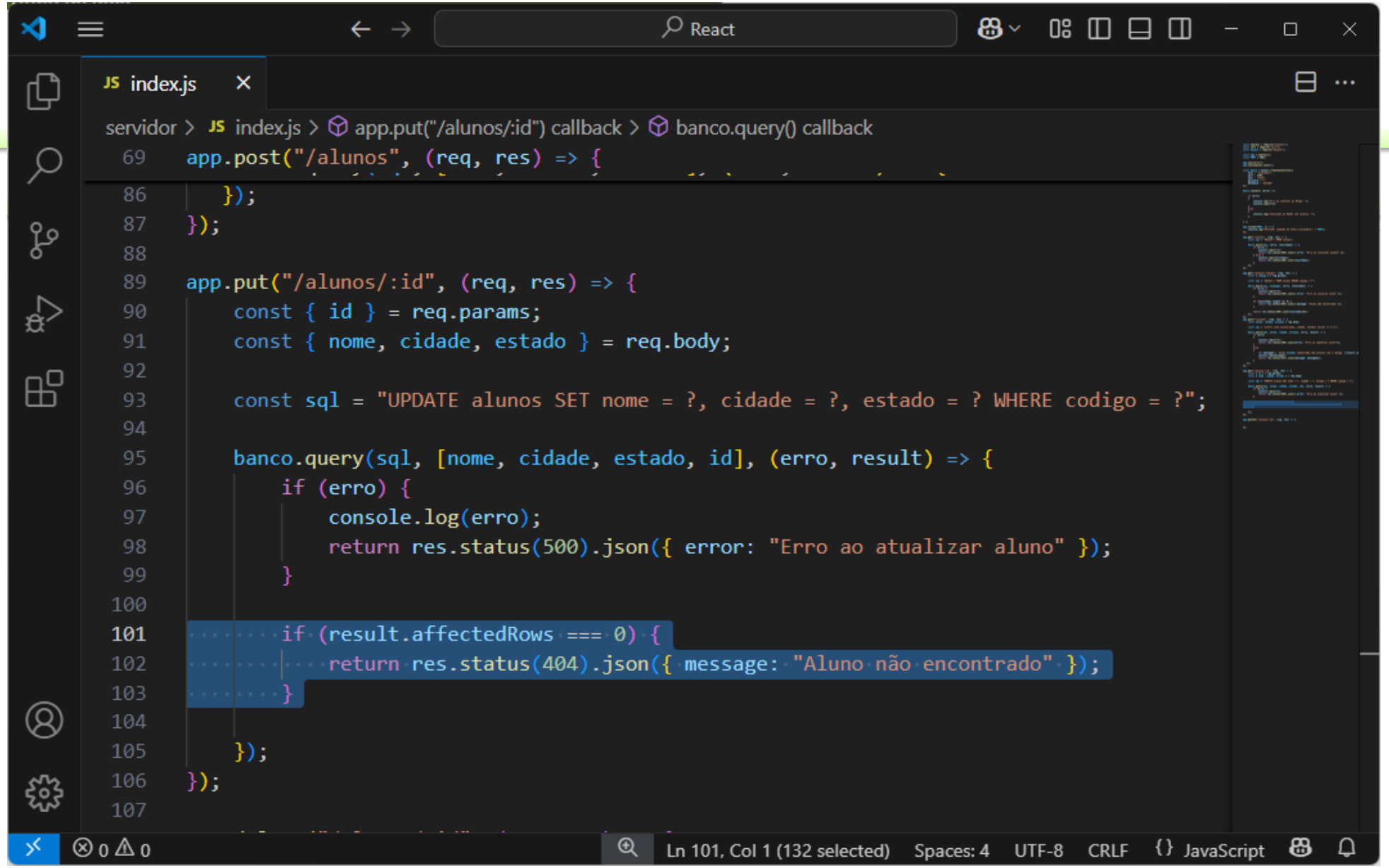
Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br

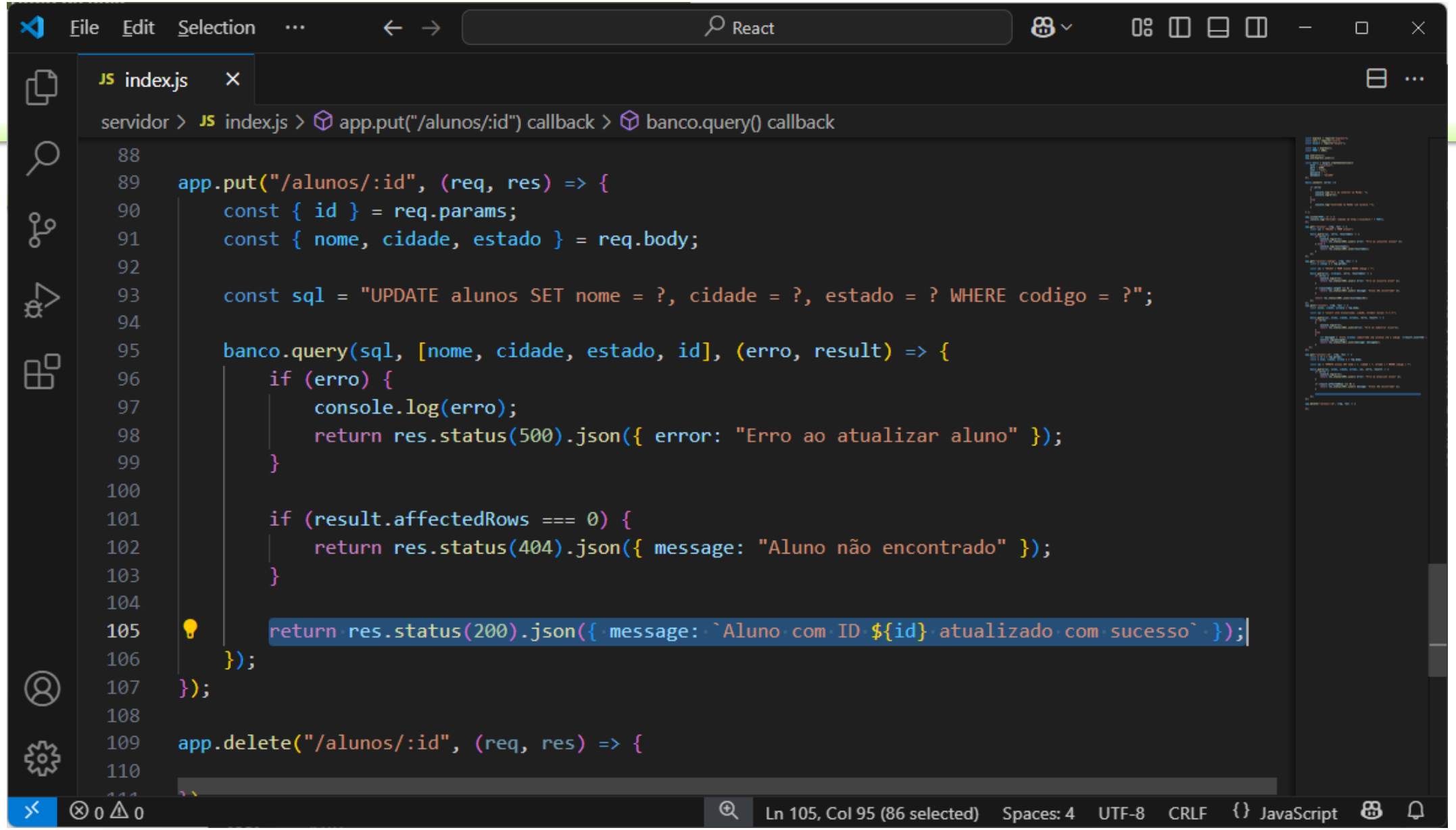

React
index.js
servidor > index.js > app.put("/alunos/:id") callback
69 app.post("/alunos", (req, res) => {
86 });
87 });
88
89 app.put("/alunos/:id", (req, res) => {
90
91
92 });
93
94 app.delete("/alunos/:id", (req, res) => {
95
96
97 });
98
0 0
Ln 90, Col 1
Spaces: 4
UTF-8
CRLF
JavaScript


React
index.js
servidor > index.js > app.put("/alunos/:id") callback
app.post("/alunos", (req, res) => {
});
});
app.put("/alunos/:id", (req, res) => {
const { id } = req.params;
const { nome, cidade, estado } = req.body;
const sql = "UPDATE alunos SET nome = ?, cidade = ?, estado = ? WHERE codigo = ?";
});
app.delete("/alunos/:id", (req, res) => {
});
Ln 94, Col 1
Spaces: 4
UTF-8
CRLF
JavaScript


React
index.js
servidor > index.js > app.put("/alunos/:id") callback
app.post("/alunos", (req, res) => {
});
app.put("/alunos/:id", (req, res) => {
const { id } = req.params;
const { nome, cidade, estado } = req.body;
const sql = "UPDATE alunos SET nome = ?, cidade = ?, estado = ? WHERE codigo = ?";
banco.query(sql, [nome, cidade, estado, id], (erro, result) => {
});
app.delete("/alunos/:id", (req, res) => {
Ln 95, Col 1 (81 selected)
Spaces: 4
UTF-8
CRLF
JavaScript

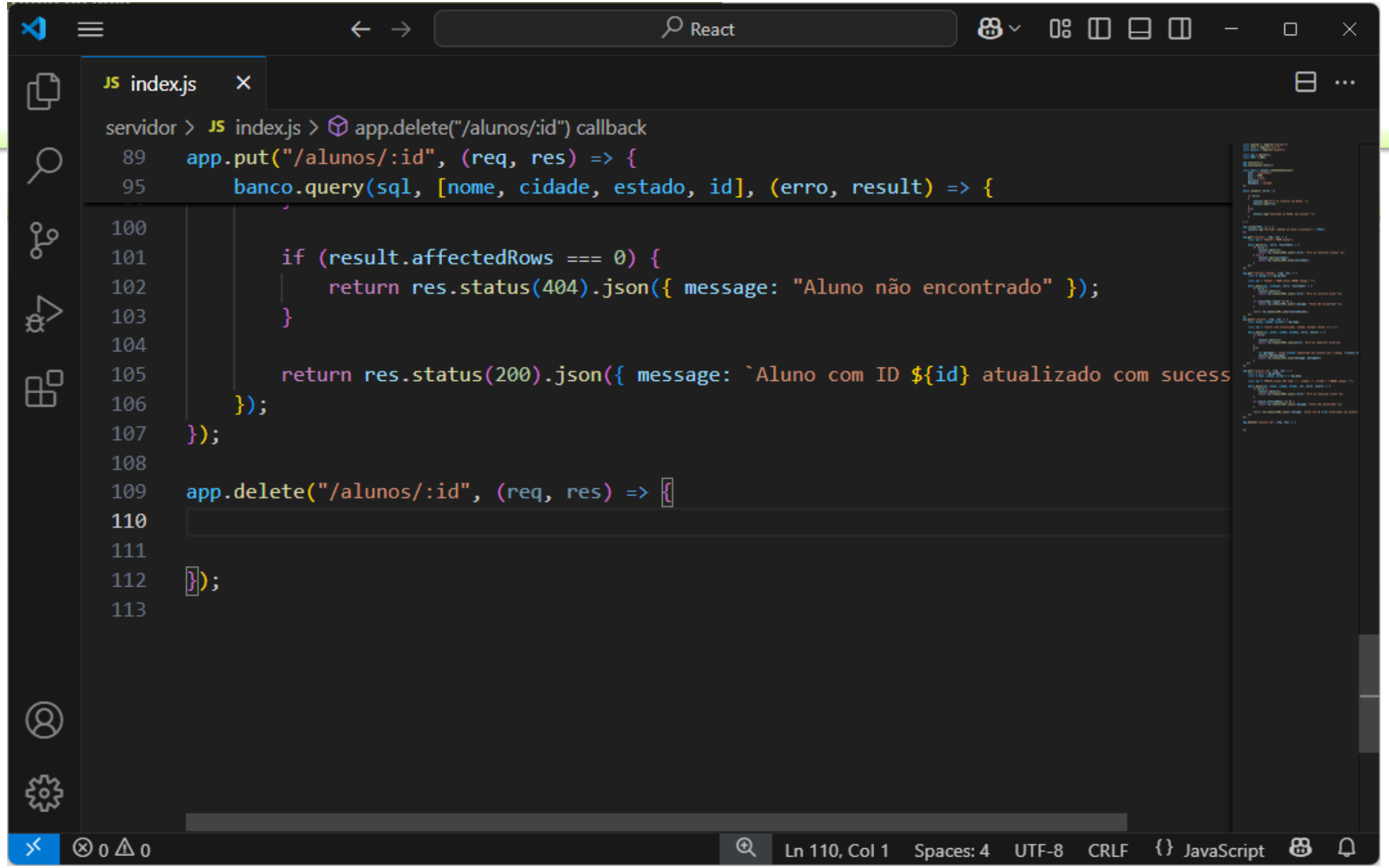

React
index.js
servidor > index.js > app.put("/alunos/:id") callback > banco.query() callback
app.post("/alunos", (req, res) => {
});
});
app.put("/alunos/:id", (req, res) => {
const { id } = req.params;
const { nome, cidade, estado } = req.body;
const sql = "UPDATE alunos SET nome = ?, cidade = ?, estado = ? WHERE codigo = ?";
banco.query(sql, [nome, cidade, estado, id], (erro, result) => {
if (erro) {
console.log(erro);
return res.status(500).json({ error: "Erro ao atualizar aluno" });
}
});
});
app.delete("/alunos/:id", (req, res) => {
0 0
Ln 96, Col 1 (144 selected)
Spaces: 4
UTF-8
CRLF
JavaScript

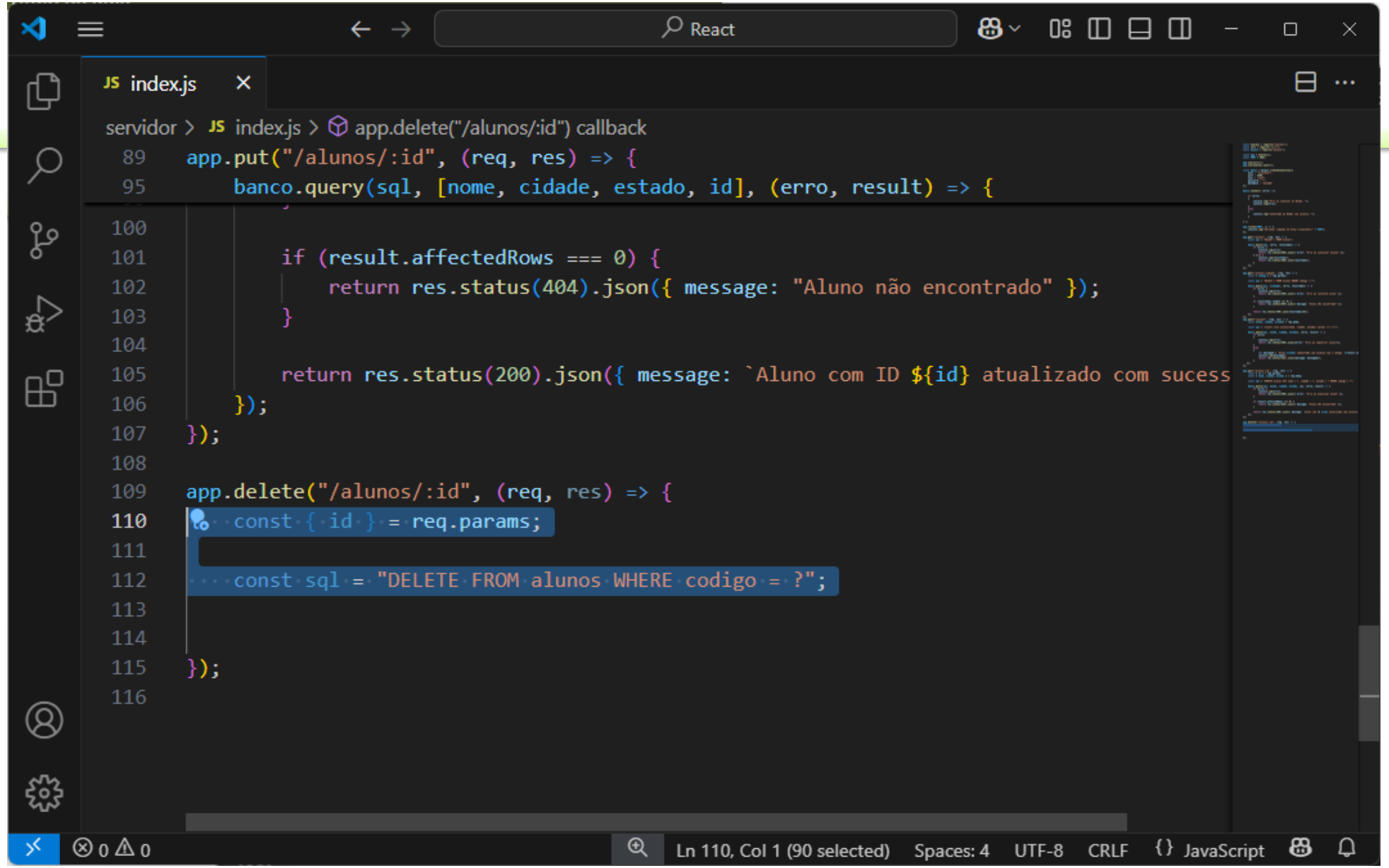

React
index.js
servidor > index.js > app.put("/alunos/:id") callback > banco.query() callback
app.post("/alunos", (req, res) => {
});
});
app.put("/alunos/:id", (req, res) => {
const { id } = req.params;
const { nome, cidade, estado } = req.body;
const sql = "UPDATE alunos SET nome = ?, cidade = ?, estado = ? WHERE codigo = ?";
banco.query(sql, [nome, cidade, estado, id], (erro, result) => {
if (erro) {
console.log(erro);
return res.status(500).json({ error: "Erro ao atualizar aluno" });
}
if (result.affectedRows === 0) {
return res.status(404).json({ message: "Aluno não encontrado" });
}
});
});
Ln 101, Col 1 (132 selected)
Spaces: 4
UTF-8
CRLF
JavaScript

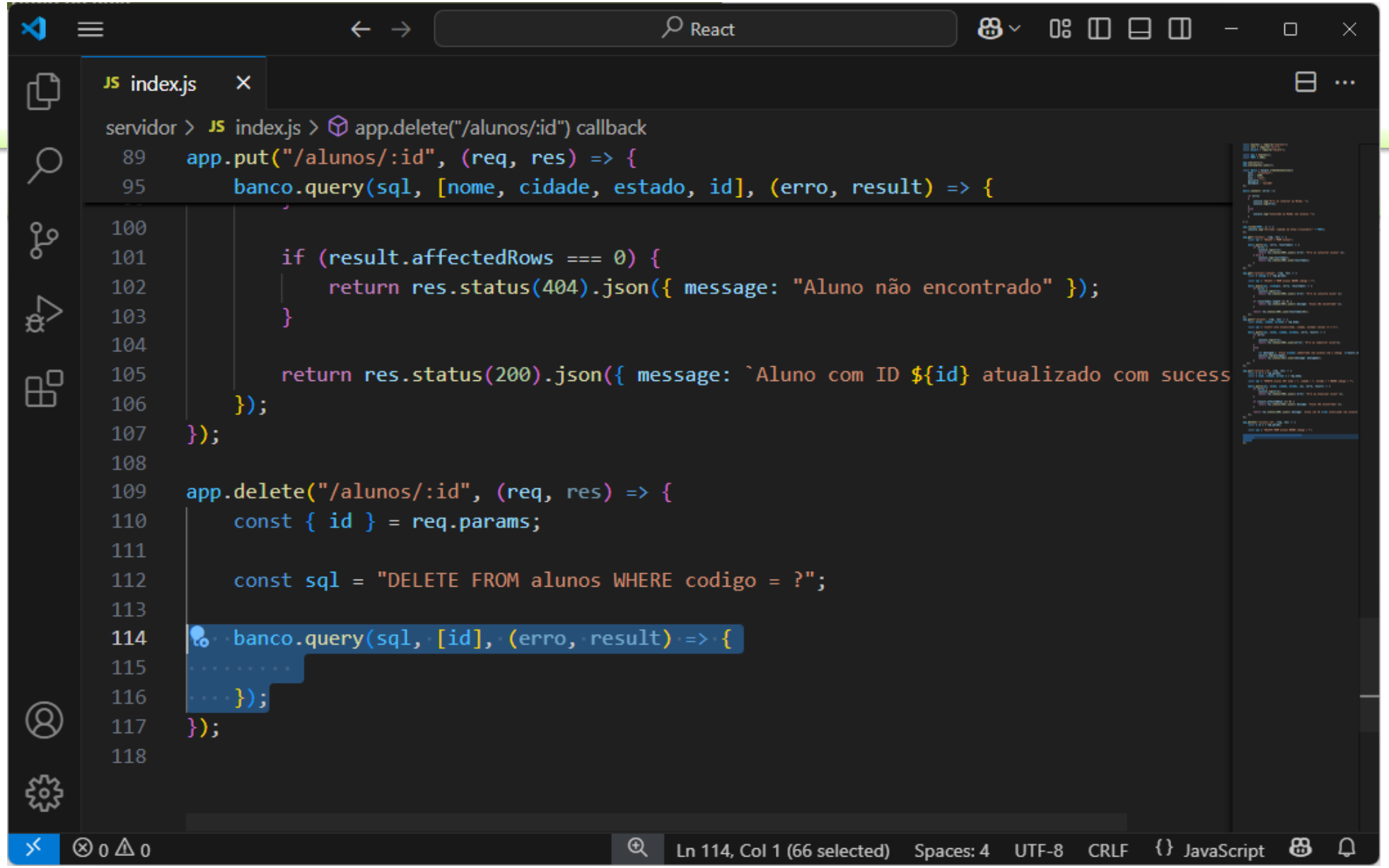

File Edit Selection ···
React
index.js
servidor > index.js > app.put("/alunos/:id") callback > banco.query() callback
app.put("/alunos/:id", (req, res) => {
const { id } = req.params;
const { nome, cidade, estado } = req.body;
const sql = "UPDATE alunos SET nome = ?, cidade = ?, estado = ? WHERE codigo = ?";
banco.query(sql, [nome, cidade, estado, id], (erro, result) => {
if (erro) {
console.log(erro);
return res.status(500).json({ error: "Erro ao atualizar aluno" });
}
if (result.affectedRows === 0) {
return res.status(404).json({ message: "Aluno não encontrado" });
}
return res.status(200).json({ message: `Aluno com ID ${id} atualizado com sucesso` });
});
});
app.delete("/alunos/:id", (req, res) => {
Ln 105, Col 95 (86 selected) Spaces: 4 UTF-8 CRLF {} JavaScript

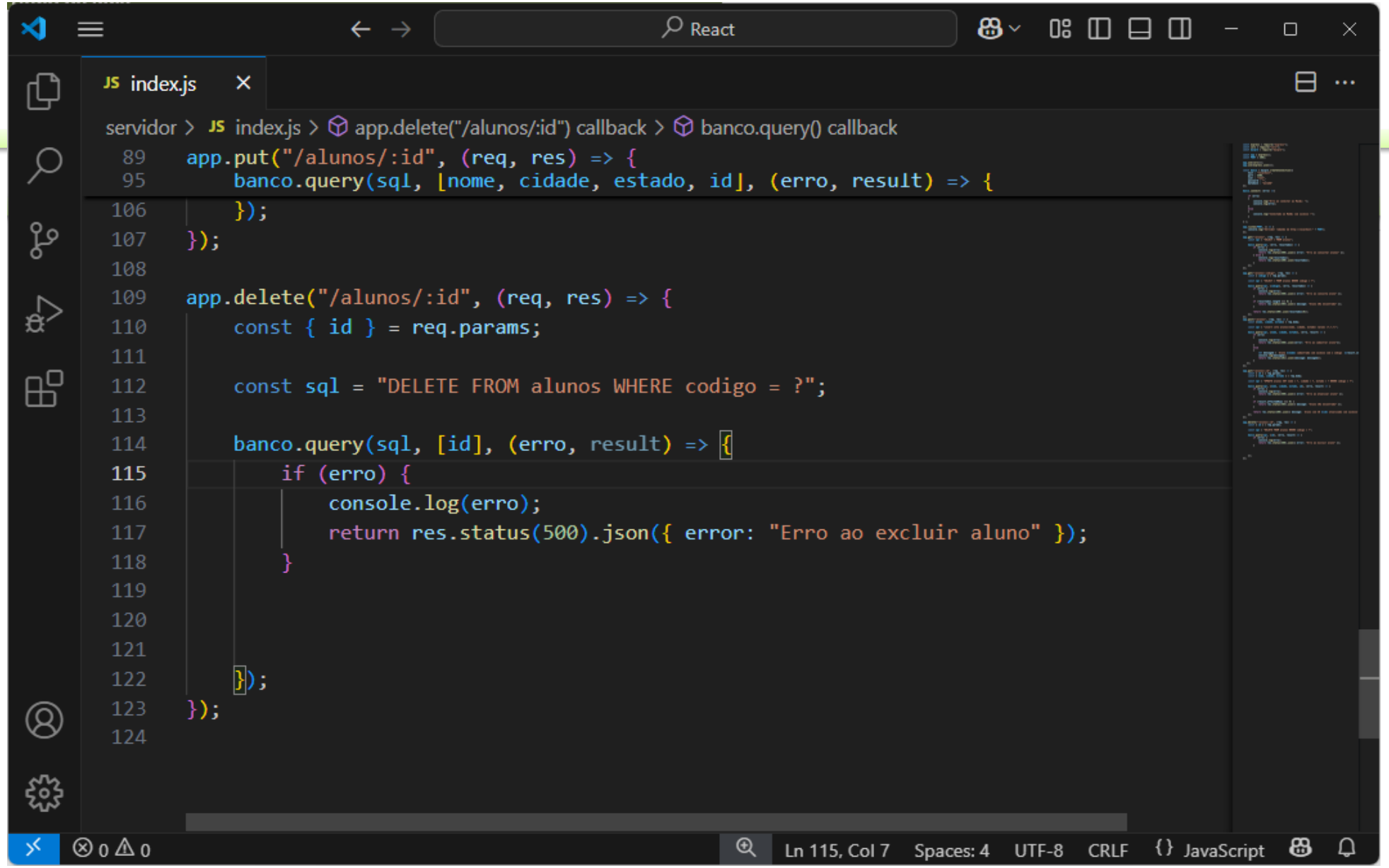
Excluir Aluno

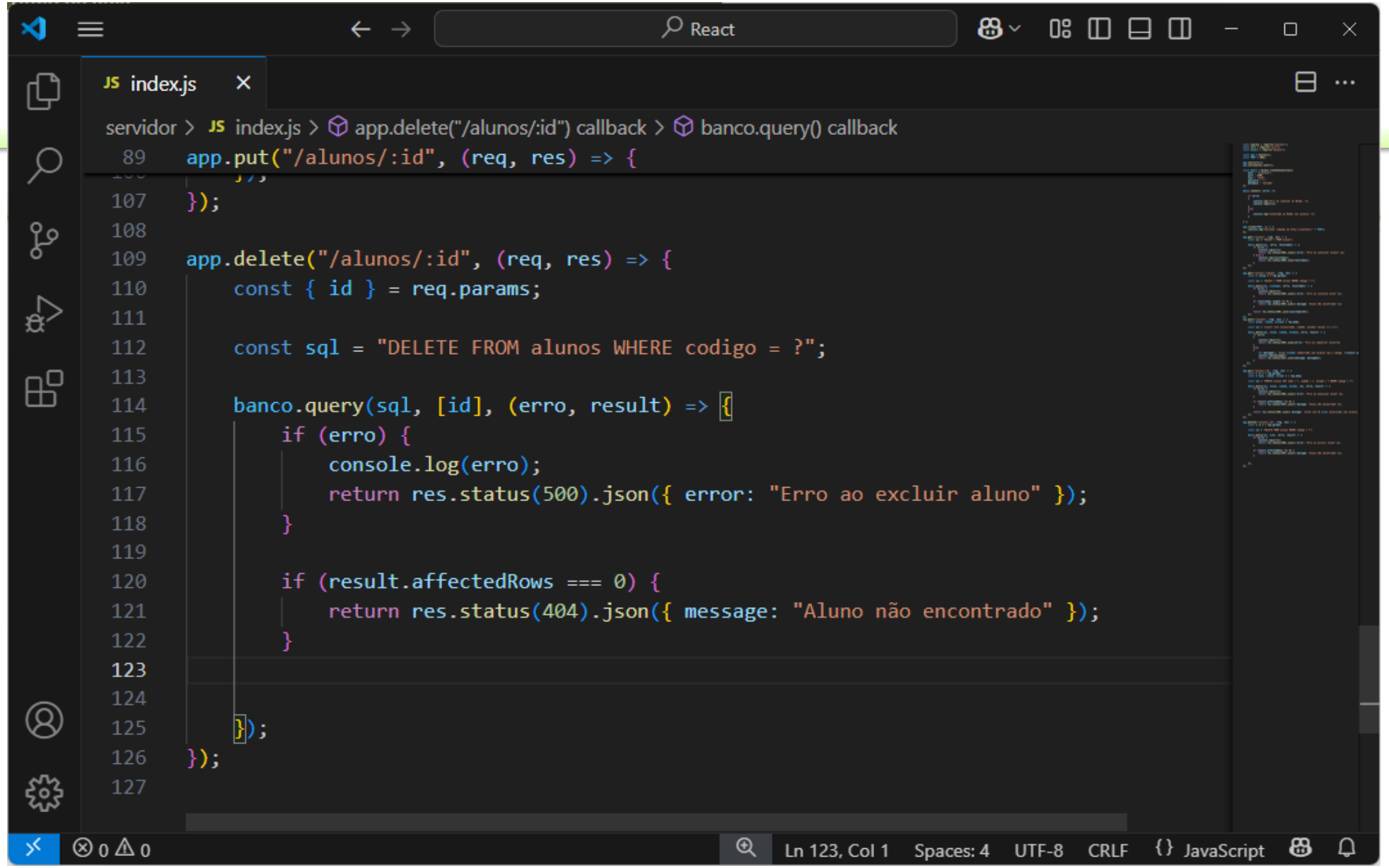
Guilherme Henrique de Souza
guilherme.souza@etec.sp.gov.br
guilherme.souza183@fatec.sp.gov.br

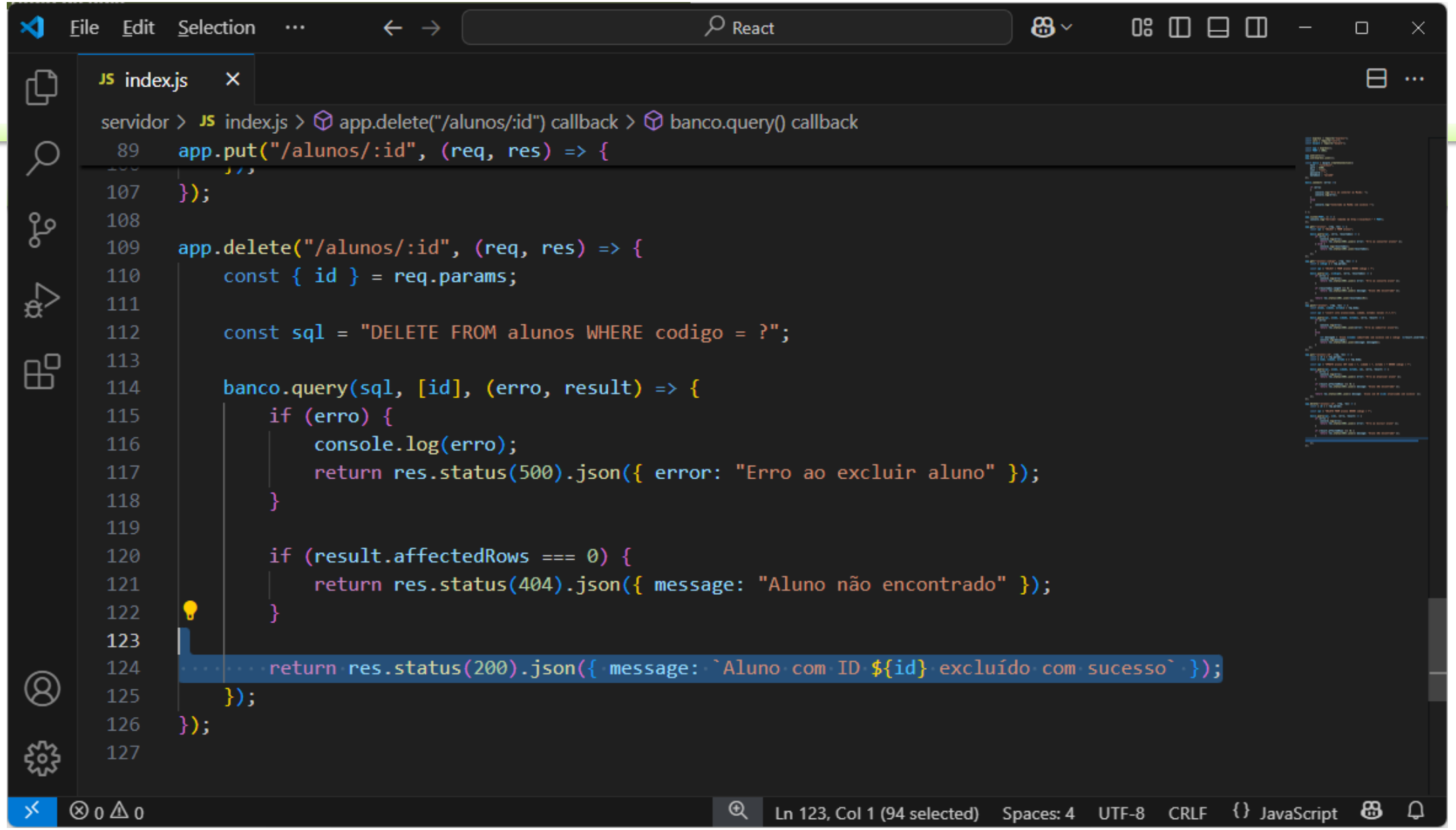

React
index.js
servidor > index.js > app.delete("/alunos/:id") callback
89 app.put("/alunos/:id", (req, res) => {
95 banco.query(sql, [nome, cidade, estado, id], (erro, result) => {
100
101 if (result.affectedRows === 0) {
102 return res.status(404).json({ message: "Aluno não encontrado" });
103 }
104
105 return res.status(200).json({ message: `Aluno com ID ${id} atualizado com sucess
106 });
107 });
108
109 app.delete("/alunos/:id", (req, res) => {
110
111
112 });
113
Ln 110, Col 1
Spaces: 4
UTF-8
CRLF
JavaScript


React
index.js
servidor > index.js > app.delete("/alunos/:id") callback
app.put("/alunos/:id", (req, res) => {
banco.query(sql, [nome, cidade, estado, id], (erro, result) => {
if (result.affectedRows === 0) {
return res.status(404).json({ message: "Aluno não encontrado" });
}
return res.status(200).json({ message: `Aluno com ID ${id} atualizado com sucess
});
});
app.delete("/alunos/:id", (req, res) => {
const { id } = req.params;
const sql = "DELETE FROM alunos WHERE codigo = ?";
});
Ln 110, Col 1 (90 selected)
Spaces: 4
UTF-8
CRLF
JavaScript


React
index.js
servidor > index.js > app.delete("/alunos/:id") callback
app.put("/alunos/:id", (req, res) => {
banco.query(sql, [nome, cidade, estado, id], (erro, result) => {
if (result.affectedRows === 0) {
return res.status(404).json({ message: "Aluno não encontrado" });
}
return res.status(200).json({ message: `Aluno com ID ${id} atualizado com sucess
});
});
app.delete("/alunos/:id", (req, res) => {
const { id } = req.params;
const sql = "DELETE FROM alunos WHERE codigo = ?";
banco.query(sql, [id], (erro, result) => {
});
Ln 114, Col 1 (66 selected)
Spaces: 4
UTF-8
CRLF
JavaScript


React
index.js
servidor > index.js > app.delete("/alunos/:id") callback > banco.query() callback
app.put("/alunos/:id", (req, res) => {
banco.query(sql, [nome, cidade, estado, id], (erro, result) => {
});
});
app.delete("/alunos/:id", (req, res) => {
const { id } = req.params;
const sql = "DELETE FROM alunos WHERE codigo = ?";
banco.query(sql, [id], (erro, result) => {
if (erro) {
console.log(erro);
return res.status(500).json({ error: "Erro ao excluir aluno" });
}
});
});
Ln 115, Col 7
Spaces: 4
UTF-8
CRLF
JavaScript


React
index.js
servidor > index.js > app.delete("/alunos/:id") callback > banco.query() callback
app.put("/alunos/:id", (req, res) => {
});
app.delete("/alunos/:id", (req, res) => {
const { id } = req.params;
const sql = "DELETE FROM alunos WHERE codigo = ?";
banco.query(sql, [id], (erro, result) => {
if (erro) {
console.log(erro);
return res.status(500).json({ error: "Erro ao excluir aluno" });
}
if (result.affectedRows === 0) {
return res.status(404).json({ message: "Aluno não encontrado" });
}
});
});
Ln 123, Col 1
Spaces: 4
UTF-8
CRLF
JavaScript


File Edit Selection ···
React
index.js
servidor > index.js > app.delete("/alunos/:id") callback > banco.query() callback
app.put("/alunos/:id", (req, res) => {
});
app.delete("/alunos/:id", (req, res) => {
const { id } = req.params;
const sql = "DELETE FROM alunos WHERE codigo = ?";
banco.query(sql, [id], (erro, result) => {
if (erro) {
console.log(erro);
return res.status(500).json({ error: "Erro ao excluir aluno" });
}
if (result.affectedRows === 0) {
return res.status(404).json({ message: "Aluno não encontrado" });
}
return res.status(200).json({ message: `Aluno com ID ${id} excluído com sucesso` });
});
});
0 0
Ln 123, Col 1 (94 selected)
Spaces: 4
UTF-8
CRLF
JavaScript

Dúvidas