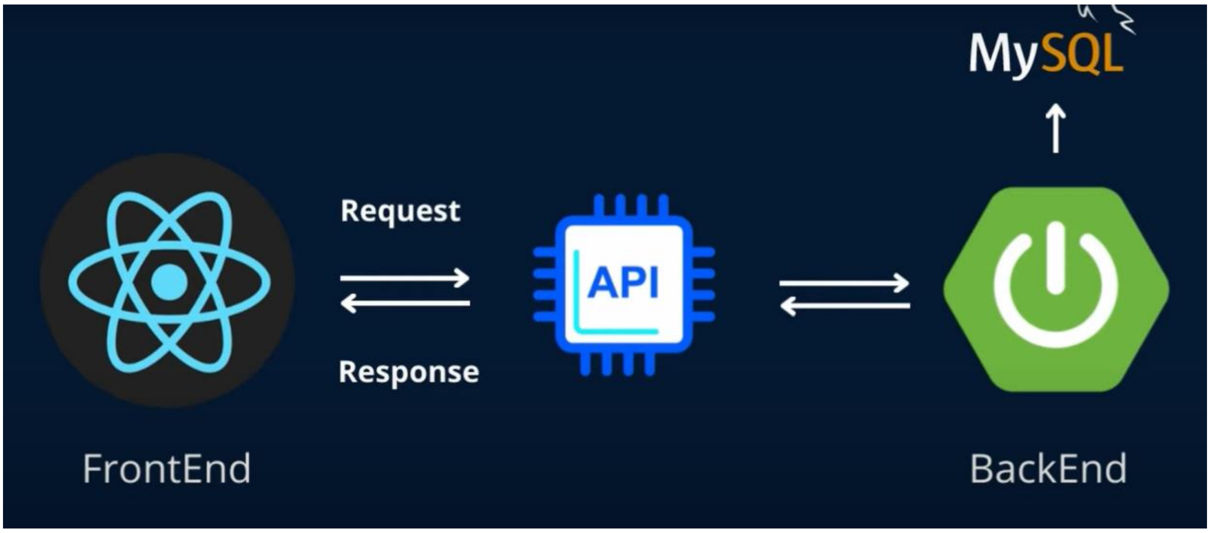
Fullstack Project -Guildelines (installation)

Building a Fullstack Application using spring boot and react



Prerequisites to run the project :

Java jdk and Node js

IDE : Intellij IDEA (spring boot)

VS code (react)

MySql Database

Postman (for testing API)

Detailed Description and Deployment about project :

Created a Project “StationProjectFullStack” where my backend code is present.

I have followed the MVC pattern.

In order to run the application Tomcat should run on port :8080.

Model Package – contains Station Model

Here we create a model class and connects with sql

In order to connect with your database, go to

“src -> main -> resources -> application.properties”

Update the datasource url , username and passwords with your specified url , usename and password .

In this application database is “fullstack”

DB commands:

```
create database fullstack;

use fullstack;
```

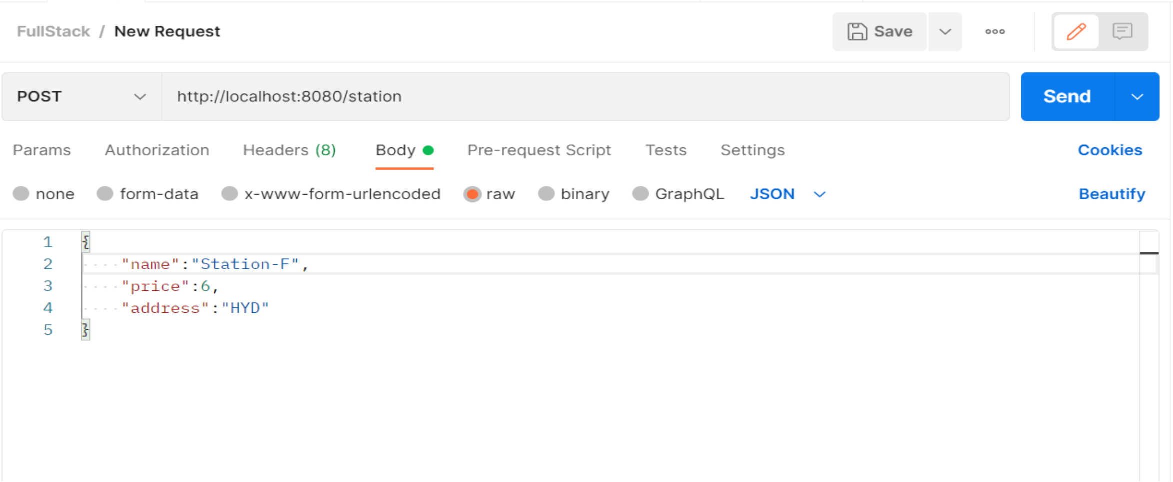
Repository package – Contains Station repository

Controller Package - contains StationController

Exception Package – To handle custom exceptions.

Testing with Postman:

For posting Data using postman:

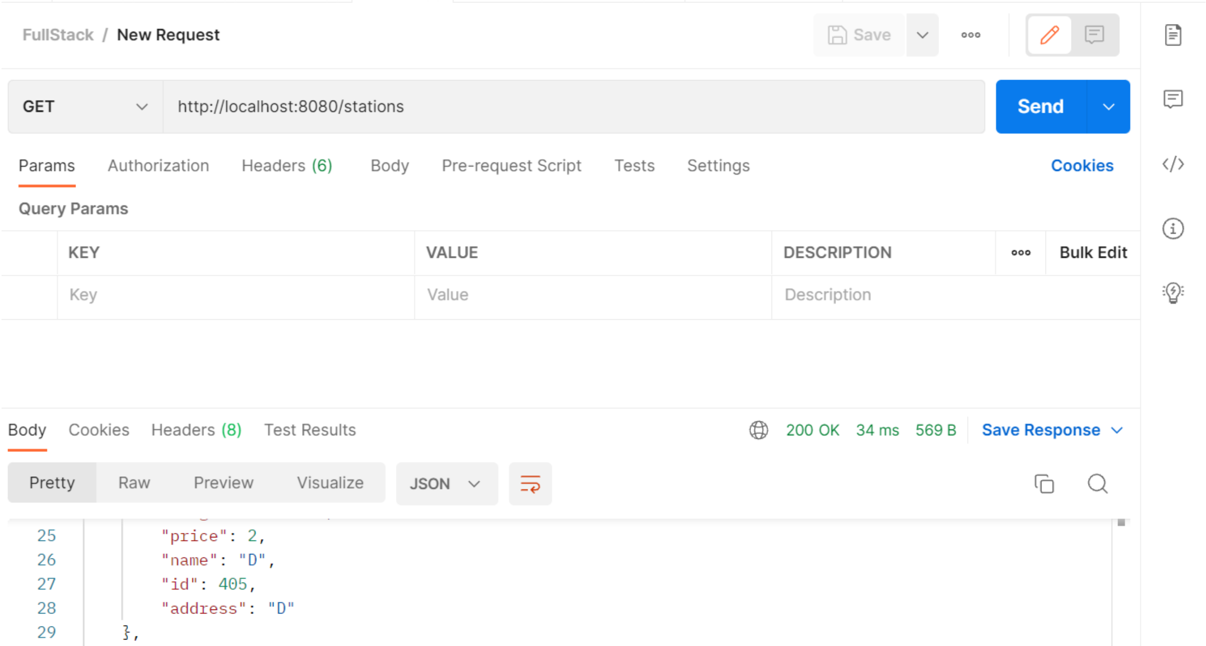


FullStack / New Request

POST http://localhost:8080/station

```
{
    "name":"Station-F",
    "price":6,
    "address":"HYD"
}
```

To get all the stations:



FullStack / New Request

GET http://localhost:8080/stations

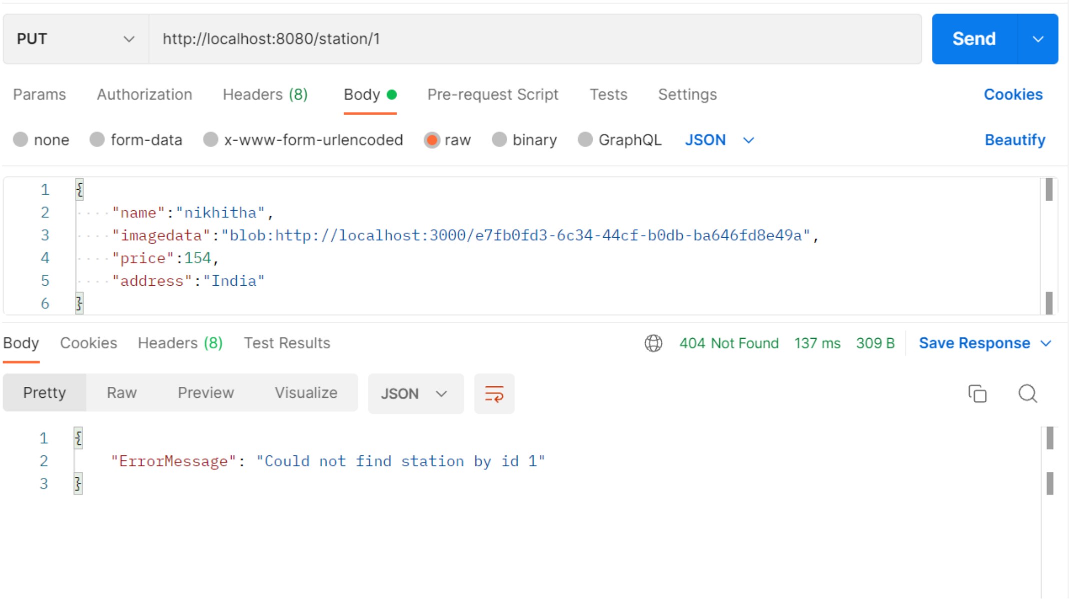
Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

200 OK 34 ms 569 B

```
        "price": 2,
        "name": "D",
        "id": 405,
        "address": "D"
    },
```

To update the data: if no id is present an error message is displayed by using custom exception.



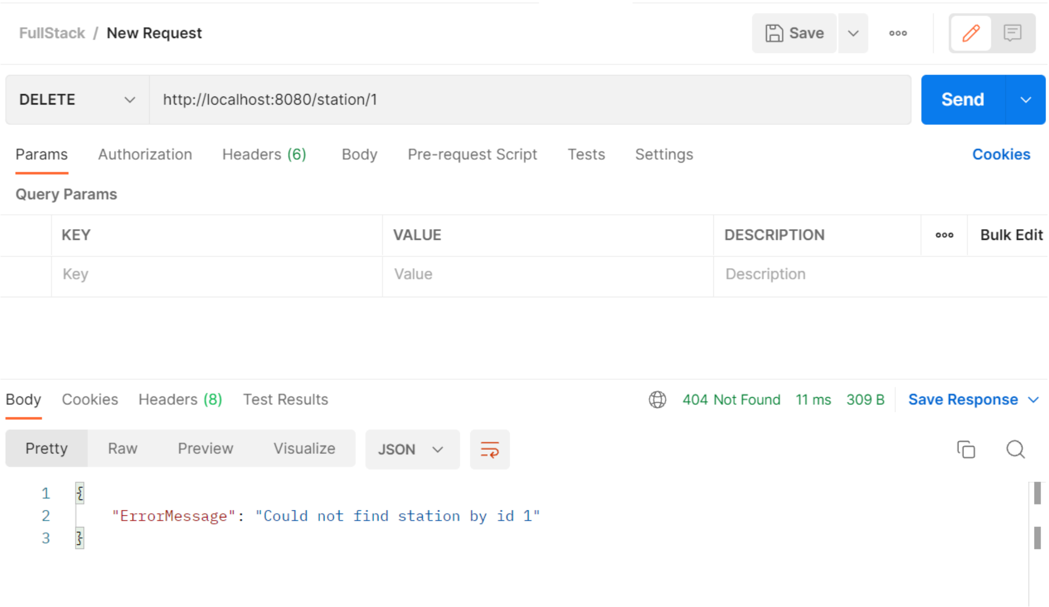
PUT http://localhost:8080/station/1

```
{
····"name":"nikhitha",
····"imagedata":"blob:http://localhost:3000/e7fb0fd3-6c34-44cf-b0db-ba646fd8e49a",
····"price":154,
····"address":"India"
}
```

404 Not Found 137 ms 309 B

```
{
    "ErrorMessage": "Could not find station by id 1"
}
```

To delete the specific station data:



DELETE http://localhost:8080/station/1

KEY	VALUE	DESCRIPTION
Key	Value	Description

404 Not Found 11 ms 309 B

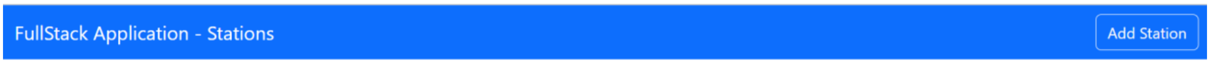
```
{
    "ErrorMessage": "Could not find station by id 1"
}
```

Frontend code:

For frontend, I created the project "FULLSTACK-FRONTEND"

Which contains modules like

Layout – which contains the navbar

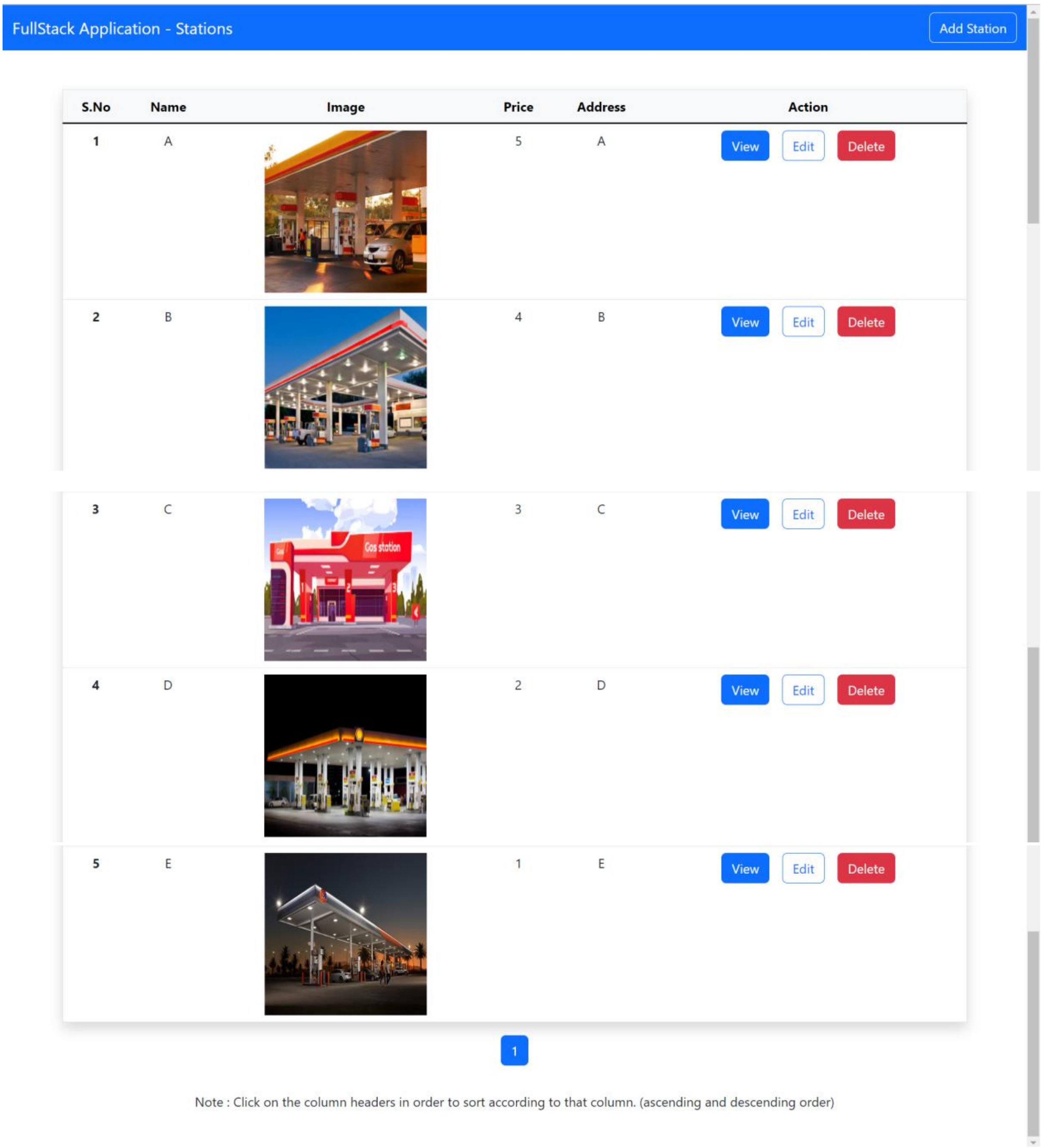


Both frontend and backend applications should be connected. In order to connect go to Project “StationProjectFullStack” in “StationController.java” I have added “@CrossOrigin("http://localhost:3000")” because my react application is running on port : 3000.

At first we need to start the spring boot application - “StationProjectFullStack”

later react application - “FULLSTACK-FRONTEND”

Pages – which contains the Home.js



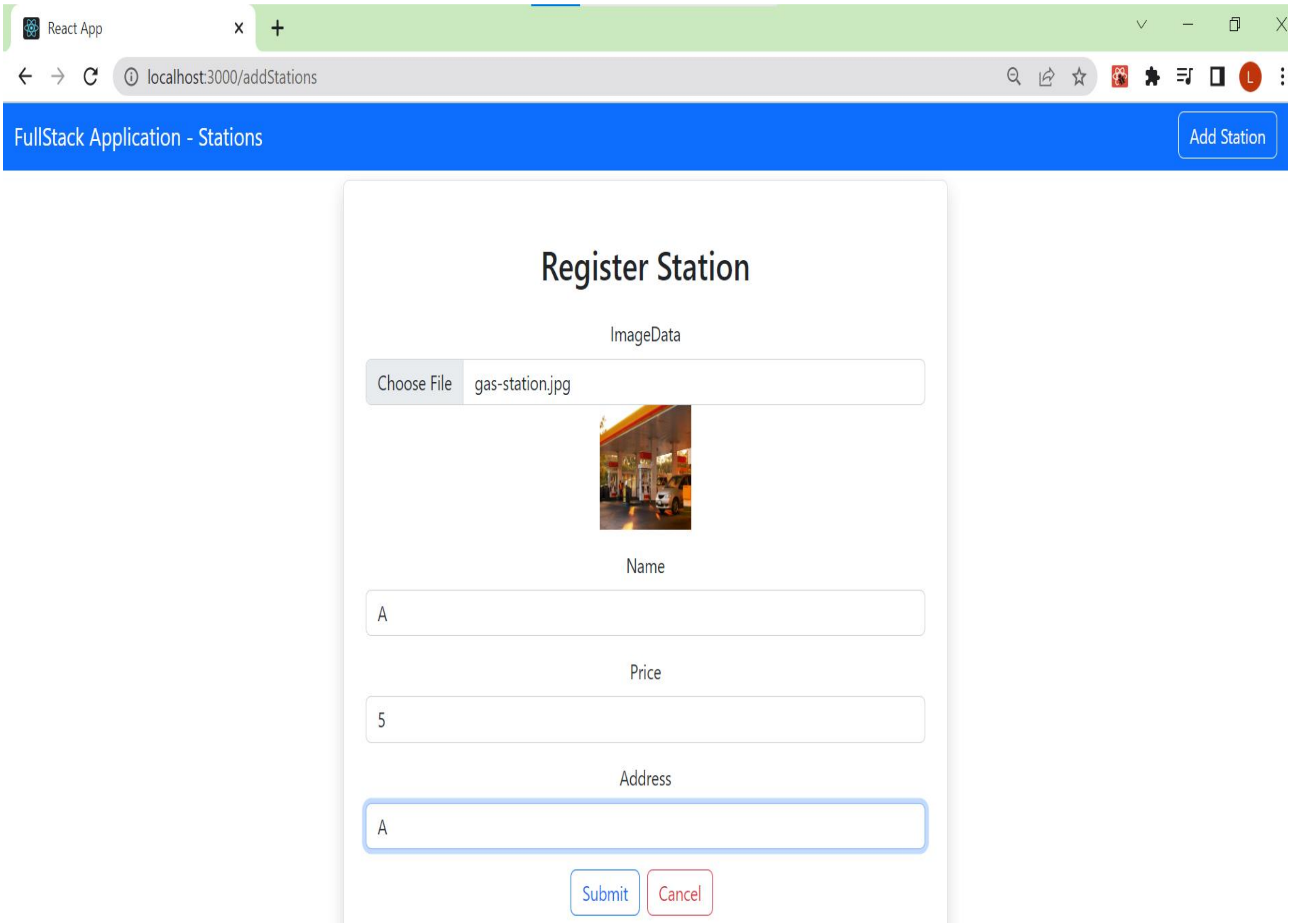
FullStack Application - Stations

S.No	Name	Image	Price	Address	Action
1	A		5	A	View Edit Delete
2	B		4	B	View Edit Delete
3	C		3	C	View Edit Delete
4	D		2	D	View Edit Delete
5	E		1	E	View Edit Delete

Note : Click on the column headers in order to sort according to that column. (ascending and descending order)

Stations – which contains AddStations.js,ViewStation.js,Editstation.js

AddStations.js



Register Station

ImageData

Choose File gas-station.jpg

Name

A

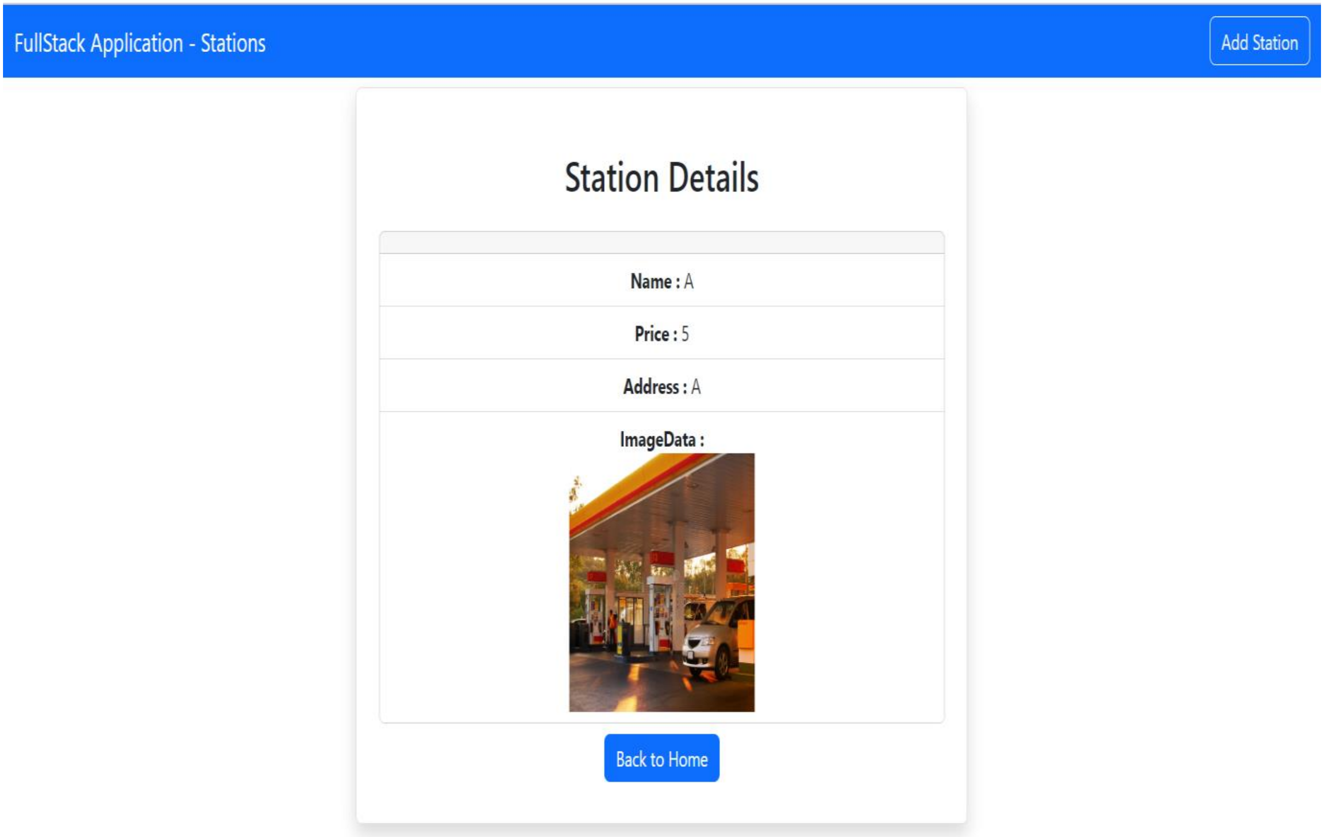
Price

5

Address

A

ViewStation.js:



Station Details

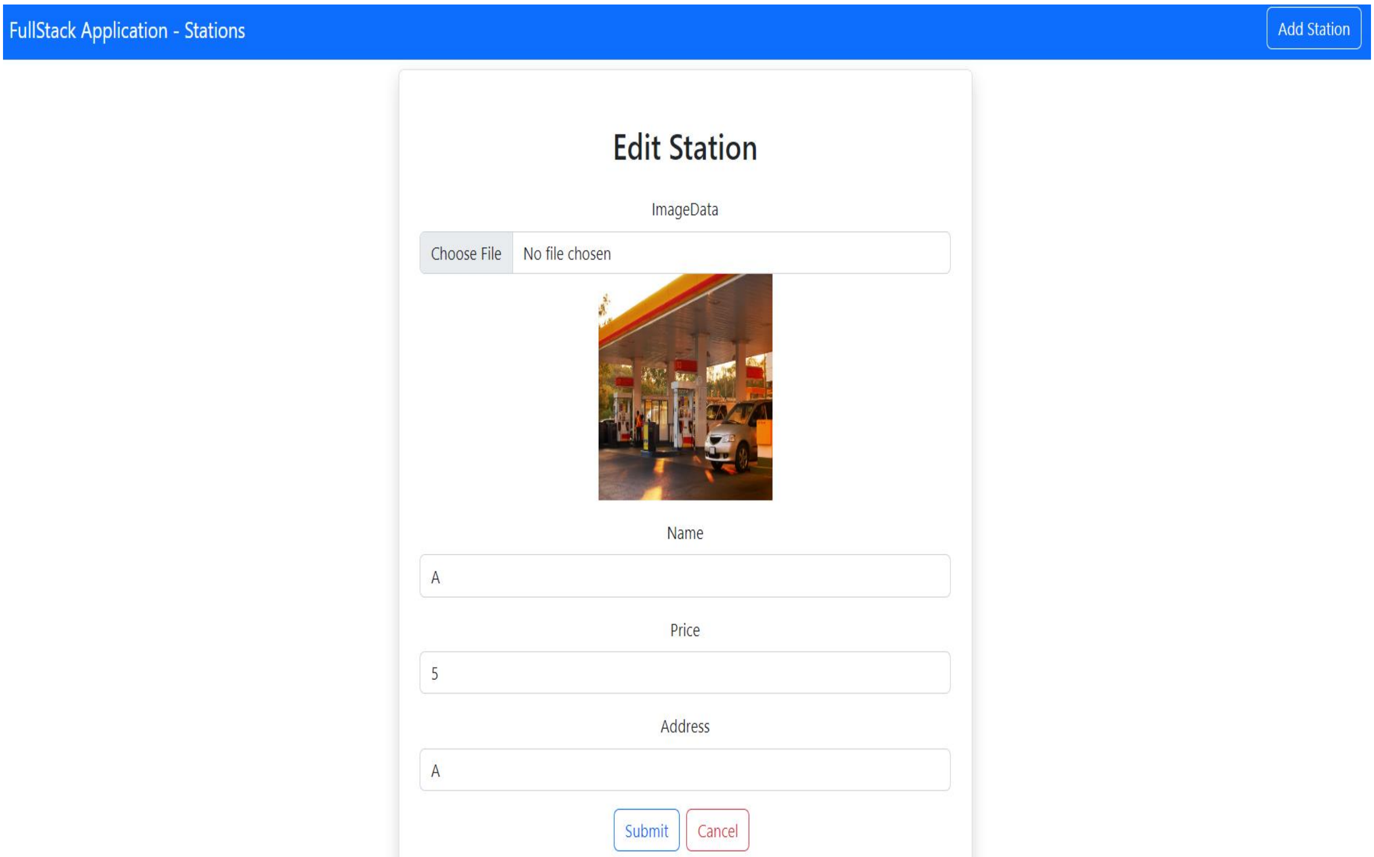
Name : A

Price : 5

Address : A

ImageData :

EditStation.js



Edit Station

ImageData

Choose File No file chosen

Name

A

Price

5

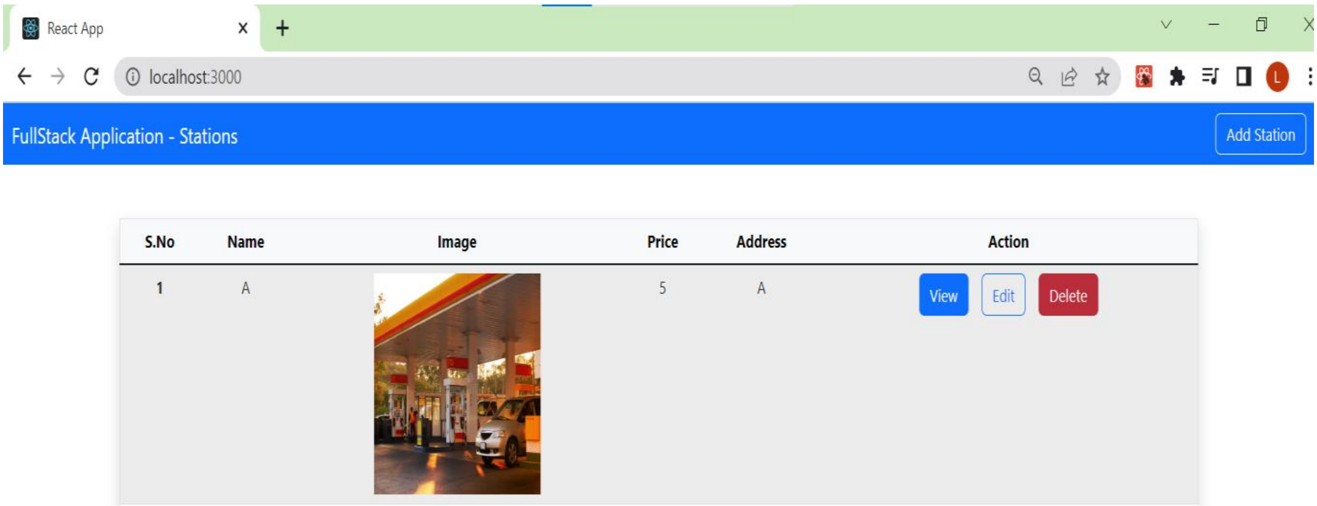
Address

A

DeleteStation.js :

By simply clicking the delete button the station record will be deleted from the database

And the remaining records will be loaded.

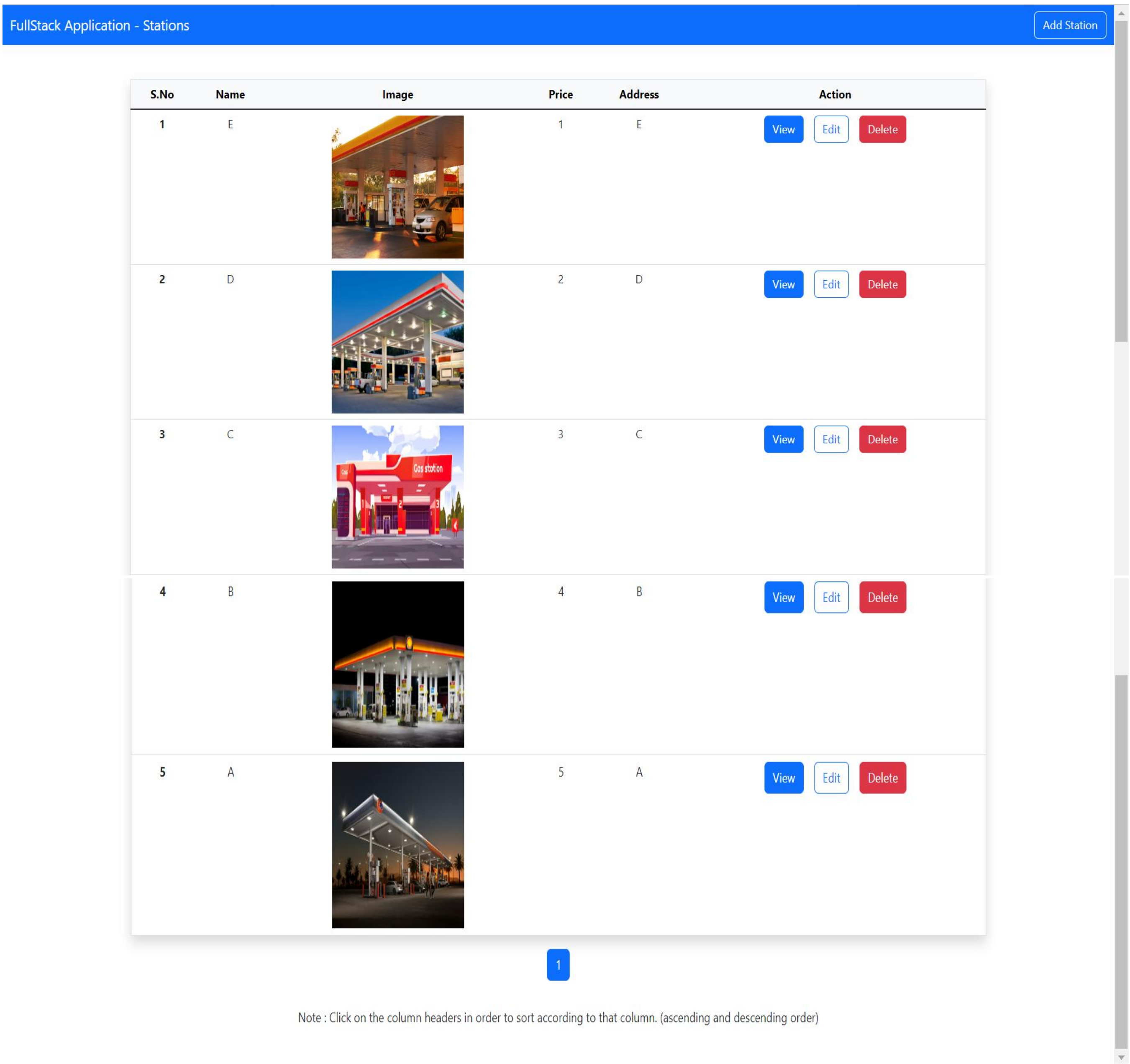


S.No	Name	Image	Price	Address	Action
1	A		5	A	View Edit Delete

Sorting according to column name:

Click on any column name (name,price,address) once you click on any column by default it sorts (ascending order) according to that column and when you click any column for the second time it sorts descending orderly and so on.

Example : At first clicked on price column and the result as follows:

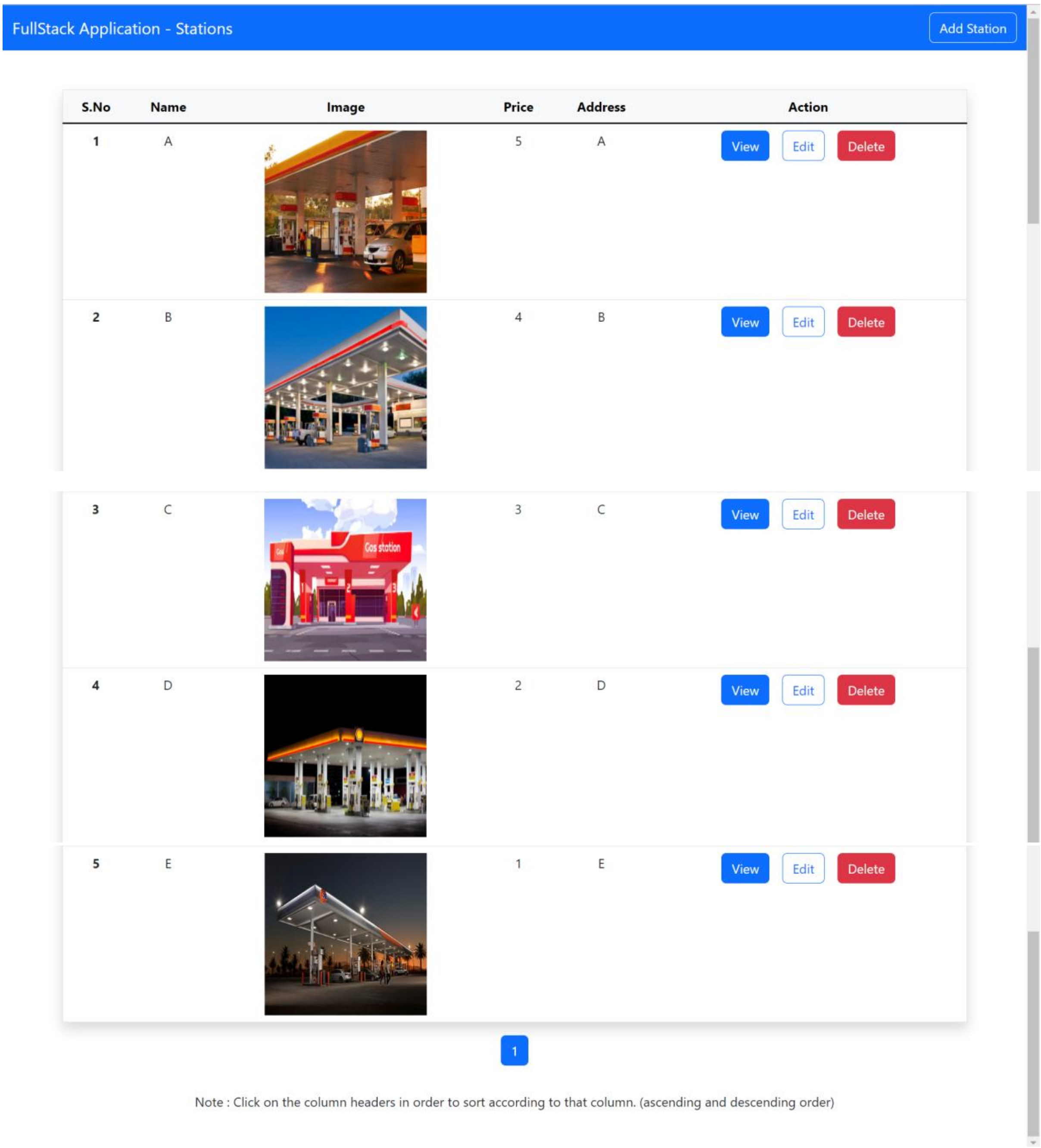


FullStack Application - Stations

S.No	Name	Image	Price	Address	Action
1	E		1	E	View Edit Delete
2	D		2	D	View Edit Delete
3	C		3	C	View Edit Delete
4	B		4	B	View Edit Delete
5	A		5	A	View Edit Delete

Note : Click on the column headers in order to sort according to that column. (ascending and descending order)

Again, clicked on price column and the result as follows:

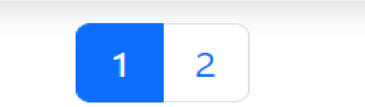


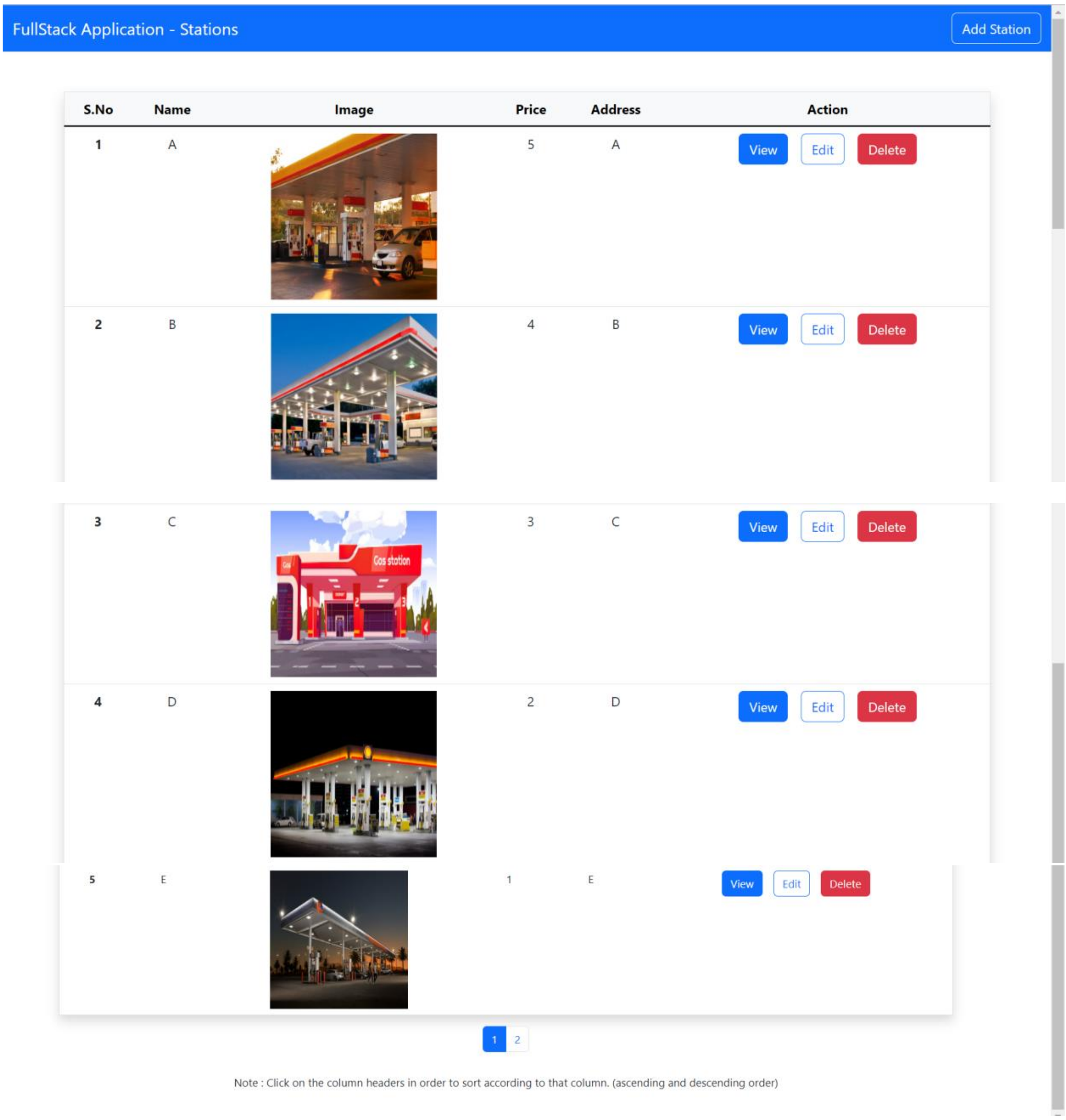
S.No	Name	Image	Price	Address	Action
1	A		5	A	View Edit Delete
2	B		4	B	View Edit Delete
3	C		3	C	View Edit Delete
4	D		2	D	View Edit Delete
5	E		1	E	View Edit Delete

Note : Click on the column headers in order to sort according to that column. (ascending and descending order)

Pagination :

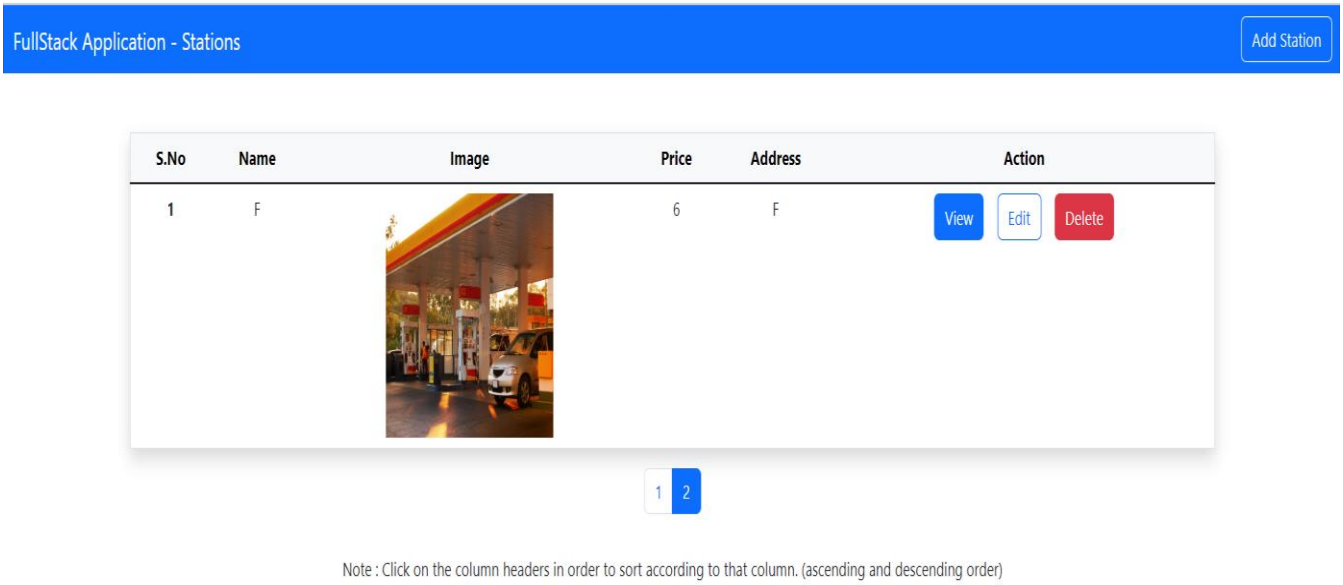
At this point my database contains 6 records and paginated the table as 5 records per page and the result as follows:





S.No	Name	Image	Price	Address	Action
1	A		5	A	View Edit Delete
2	B		4	B	View Edit Delete
3	C		3	C	View Edit Delete
4	D		2	D	View Edit Delete
5	E		1	E	View Edit Delete

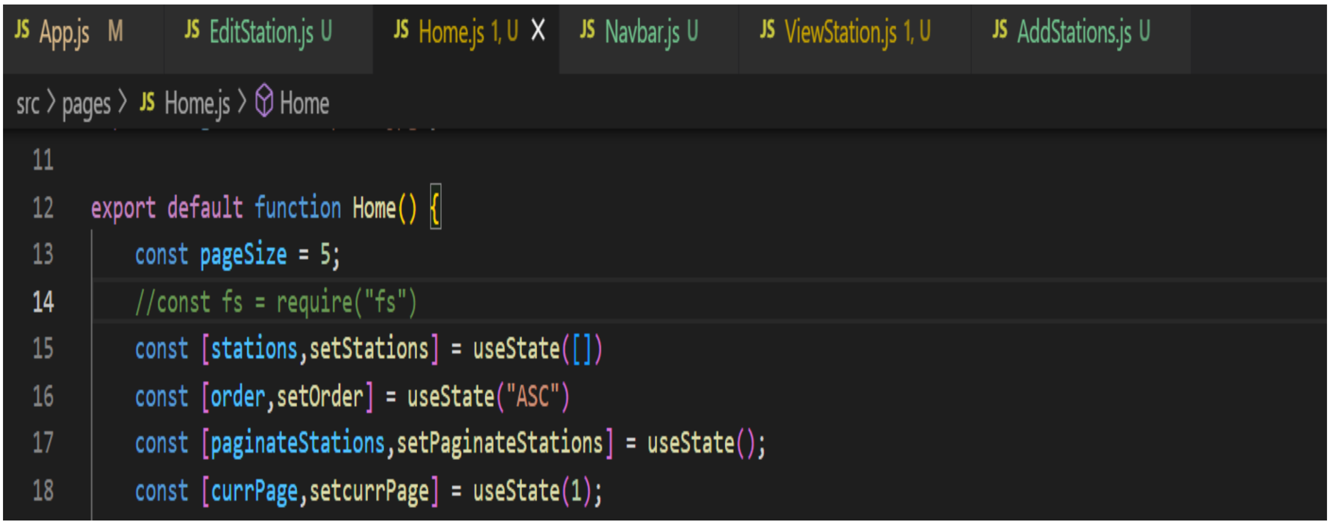
Note : Click on the column headers in order to sort according to that column. (ascending and descending order)



S.No	Name	Image	Price	Address	Action
1	F		6	F	View Edit Delete

Note : Click on the column headers in order to sort according to that column. (ascending and descending order)

And can change the page size using pageSize value:



```

export default function Home() {
    const pageSize = 5;
    //const fs = require("fs")
    const [stations,setStations] = useState([])
    const [order,setOrder] = useState("ASC")
    const [paginateStations,setPaginateStations] = useState();
    const [currPage,setcurrPage] = useState(1);
```