A PROJECT REPORT OF

React.JS Development ON

Digital Grade Book App

LOVELY PROFESSIONAL UNIVERSITY

JALANDHAR, PUNJAB

(2019-2023)

Submitted to: -

Mr. Mir Junaid Rasool

Signature : Junaid Rasool

Submitted by:-

N. Yogendra Reddy

Reg no : 11909002

Signature : Yogendra

INDEX

ACKNOWLEDGEMENT

The completion of this digital grade book app project would not have been possible without the support, guidance, and contributions of many individuals and organizations.

Firstly, I would like to express my sincere gratitude to Lovely Professional University for providing me with the opportunity to undertake this project as a part of my EPAM training in front-end web development. The learning experience and resources provided by the university have been invaluable in helping me to develop my skills and knowledge.

I would also like to extend my thanks to my mentor, MS. Sandeep Kaur, for her expert guidance, encouragement, and feedback throughout the project. Their insights and support were instrumental in helping me to develop and refine the digital grade book app, and I am deeply grateful for her time and dedication.

Additionally, I would like to thank my peers and colleagues for their support and input during the development of the app. Their feedback and suggestions helped me to refine and improve the app's functionality and usability.

Finally, I would like to acknowledge the contribution of the wider academic community and educational institutions who have provided the inspiration and motivation for this project. By working together to develop innovative solutions to educational challenges, we can enhance the learning experience for students and teachers alike.

In conclusion, I am grateful for the support and collaboration of everyone who has contributed to this project. The completion of this digital grade book app represents a significant step towards improving the efficiency, accuracy, and effectiveness of the grade book system, and I am proud to have been a part of it..

CERTIFICATION:

udemy

Certificate no: UC-58b9fa37-d2e6-4b1a-8185-5d1a4faaca7c
Certificate url: ude.my/UC-58b9fa37-d2e6-4b1a-8185-5d1a4faaca7c
Reference Number: 0004

CERTIFICATE OF COMPLETION

Javascript Tutorial and Projects Course (2023)

Instructors **John Smilga**

Nallapa Reddy Yogendra Reddy

Date **April 24, 2023**
Length **46.5 total hours**

INTRODUCTION:-

"The traditional paper-based grade book system used by teachers and academic administrators to track student performance and progress can be time-consuming, error-prone, and inefficient. With the increasing demand for digital solutions in education, there is a growing need for a more modern and efficient way to manage grades and student performance data. To meet this need, we have developed a digital grade book app using React, a popular JavaScript library for building user interfaces.

The digital grade book app provides a user-friendly interface for teachers and administrators to easily input and access student performance data in real time. With the app's features and functionalities, users can manage student profiles, view and record grades, and monitor attendance. By using React for the front-end development of the app, we can leverage its advantages such as ease of use, modularity, and scalability to create a robust and flexible solution.

This project aims to improve the efficiency and accuracy of the gradebook system, reduce administrative burden, and enhance the learning experience for students. The app is designed to be adaptable to various educational settings, from K-12 to higher education institutions. Overall, this digital gradebook app project represents a significant step towards improving the quality and effectiveness of education through the use of modern technology."

OBJECTIVE AND GOALS OF THE PROJECT: -

The objectives and goals of your digital grade book app project may include:

To develop a user-friendly and efficient digital solution to replace the traditional paper-based grade book system.

To improve the accuracy and efficiency of tracking student performance and progress.

To reduce the administrative burden on teachers and academic administrators by streamlining the grade book management process.

To enhance the learning experience for students by providing timely and accurate feedback on their performance.

To provide a flexible and adaptable solution that can be customized to meet the needs of different educational settings.

To leverage modern technology such as React to create a robust and scalable solution.

To provide built-in reporting and analysis tools to help teachers and administrators make data-driven decisions.

To enhance communication and collaboration between teachers, administrators, and students.

Overall, the main objectives and goals of your digital gradebook app project should be to improve the efficiency, accuracy, and effectiveness of the gradebook system, and to enhance the learning experience for students.

SOFTWARE REQUIREMENTS

- Operating system: - Run in Every OS
- Front End: - React.JS
- Back End: - Not Used.
- IDE used: - Visual Studio Code
- Runs project: - NPM (Node Package Manager) package manager for the Node JavaScript platform

The benefits of your digital gradebook app project may include:

Improved efficiency: The app can streamline the gradebook management process and reduce the time and effort required to manage and manipulate gradebook data.

Improved accuracy: The app can reduce the likelihood of errors in gradebook data, such as miscalculations or data entry errors.

Enhanced accessibility: The app can allow teachers, students, and administrators to access gradebook data anytime, anywhere, and on any device.

Improved communication: The app can enhance communication and collaboration between teachers, students, and administrators by providing a centralized platform to manage gradebook data.

Enhanced learning experience: The app can provide timely and accurate feedback on student performance, which can help students identify areas for improvement and enhance their learning experience.

Data-driven decision making: The app can provide built-in reporting and analysis tools to help teachers and administrators make data-driven decisions about student performance and progress.

Customization: The app can be customized to meet the specific needs of different educational settings, such as different grading scales or assessment types.

Overall, the benefits of your digital gradebook app project can lead to an improved, more efficient, and effective gradebook management system that can enhance the learning experience for students and reduce the administrative burden on teachers and academic administrators.

Design and Architecture:

The design of the grade book application was based on a modular approach, where each component had a specific function and responsibility. The header component was responsible for displaying information about the teacher, exam, and course. The table component was responsible for displaying the students' exam grades and included a filter feature that allowed the user to filter the students by their pass or fail status. The statistics component was responsible for calculating and displaying the minimum, maximum, and average grades for each exam. The footer component was responsible for displaying the overall average grade for all the students.

The architecture of the application was based on the React framework, which uses a component-based approach to web development. The application was built using JavaScript, HTML, and CSS.

Features Implemented:

The grade book application has the following features:

The user can see the details of the teacher, exam, and course in the header component.

The user can watch, the student records in the table component.

The table component calculates the average grade for each student and displays it in a separate column.

The user can filter the students by their pass or fail status in the table component.

The statistics component calculates and displays the minimum, maximum, and average grade for each exam.

The footer component shows my very own unique signature.

Implementation:

The implementation of the grade book application was divided into four main components: header, table, statistics, and footer. Each component was implemented using React and consisted of several sub-components that performed specific functions.

The header component was implemented using a form that allowed the user to input the details of the teacher, exam, and course. The form was created using React's controlled component approach, which ensured that the data was always synchronized between the form and the component's state.

The table component was implemented using a table that displayed the students' exam grades. The table was created using React's JSX syntax and included edit and delete buttons that allowed the user to modify or remove student records. Additionally, the table included three buttons: "All", "Passed", and "Failed". When the user clicked one of these buttons, the table displayed the corresponding student records. The filter functionality was implemented using JavaScript's filter method.

The statistics component was implemented using a table that displayed the minimum, maximum, and average grade for each exam. The minimum and maximum grades were calculated using JavaScript's Math object, while the average grade was calculated using an algorithm that iterated through the students' exam grades and calculated the sum of all grades divided by the total number of students.

The footer component displays the unique signature.

Challenges Faced:

During the development of the application, I faced several challenges, including:

Managing the state of the application and passing data between components.

Implementing the filter functionality for the table component.

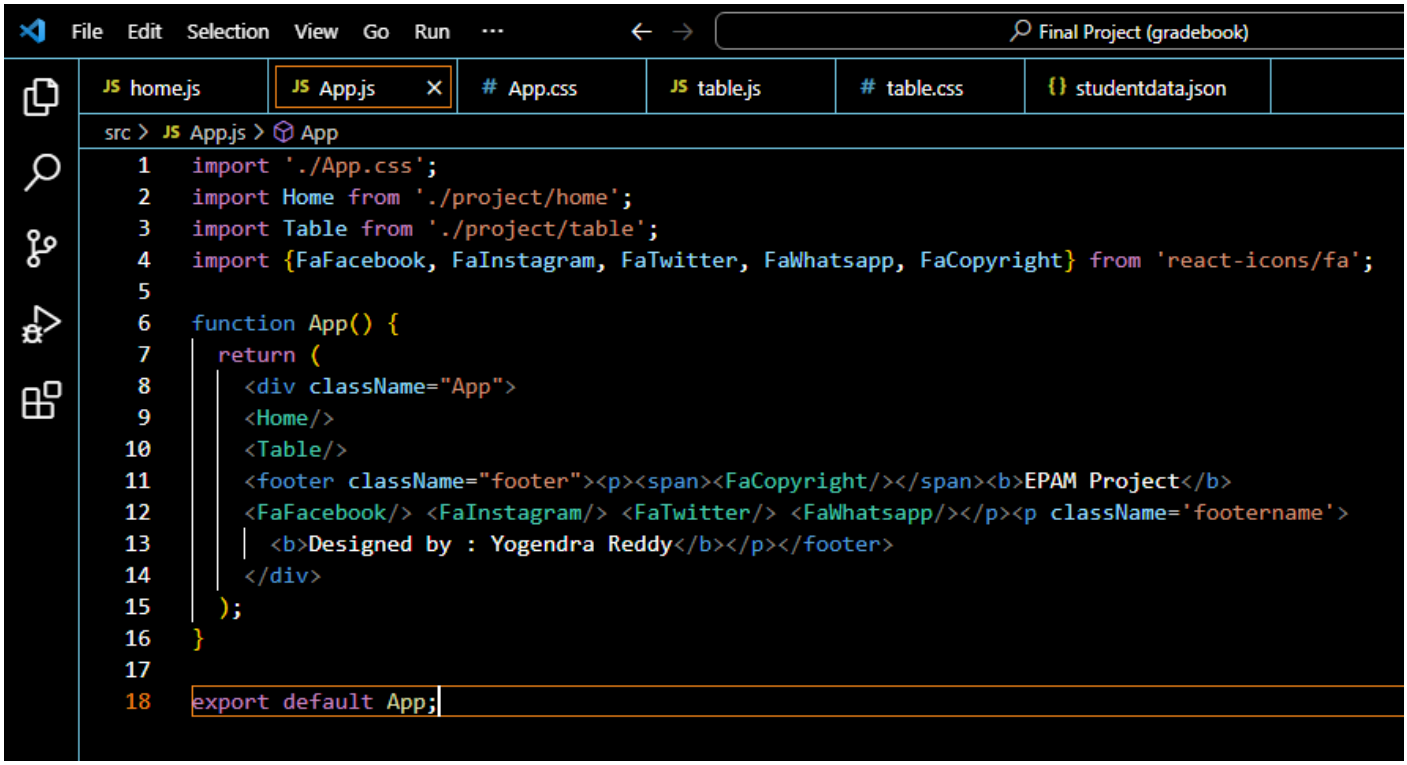
Formatting the table component to display the data in a readable and organized manner.

Implementing the statistics component to calculate and display the minimum, maximum, and average grades for each exam.

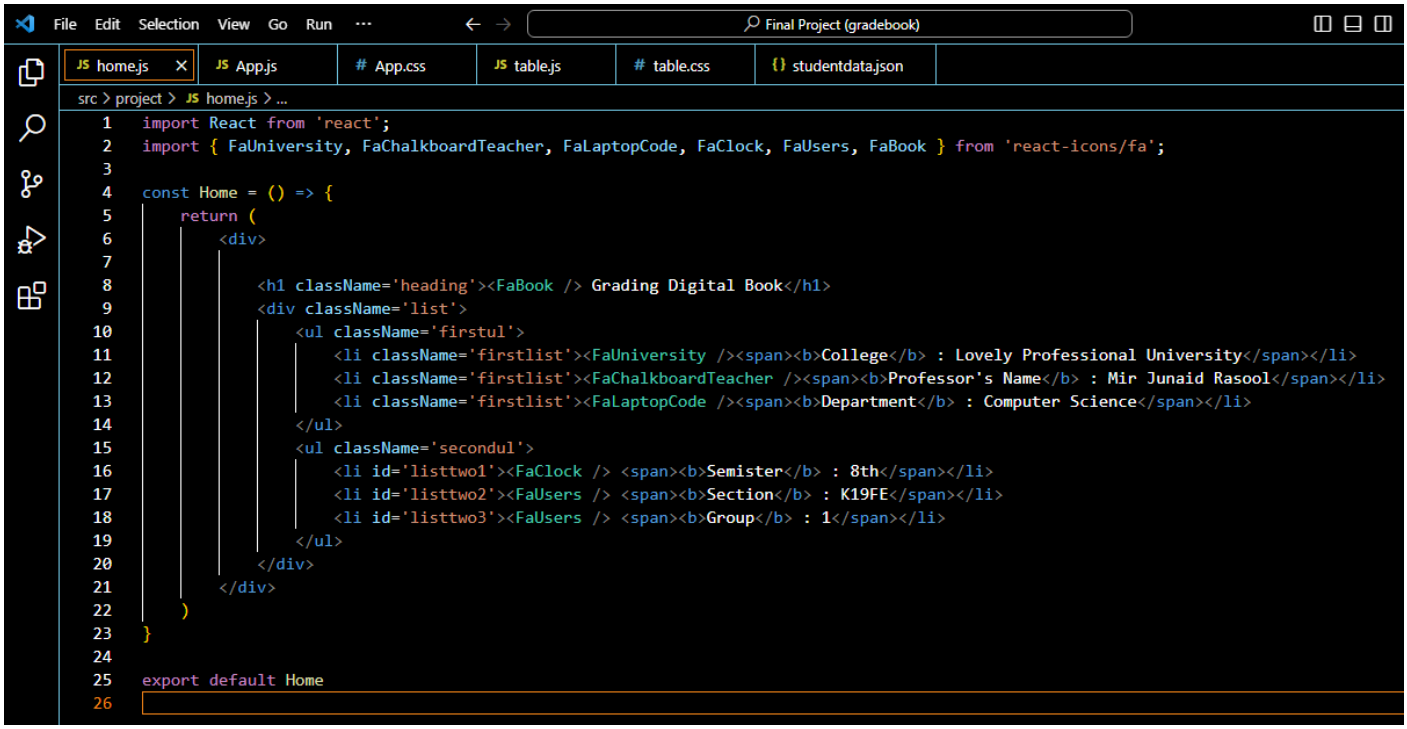
To overcome these challenges, I utilized various resources such as online tutorials, documentation, and community forums. Additionally, I employed debugging techniques and conducted thorough testing to ensure that the application was functioning correctly.

Coding:

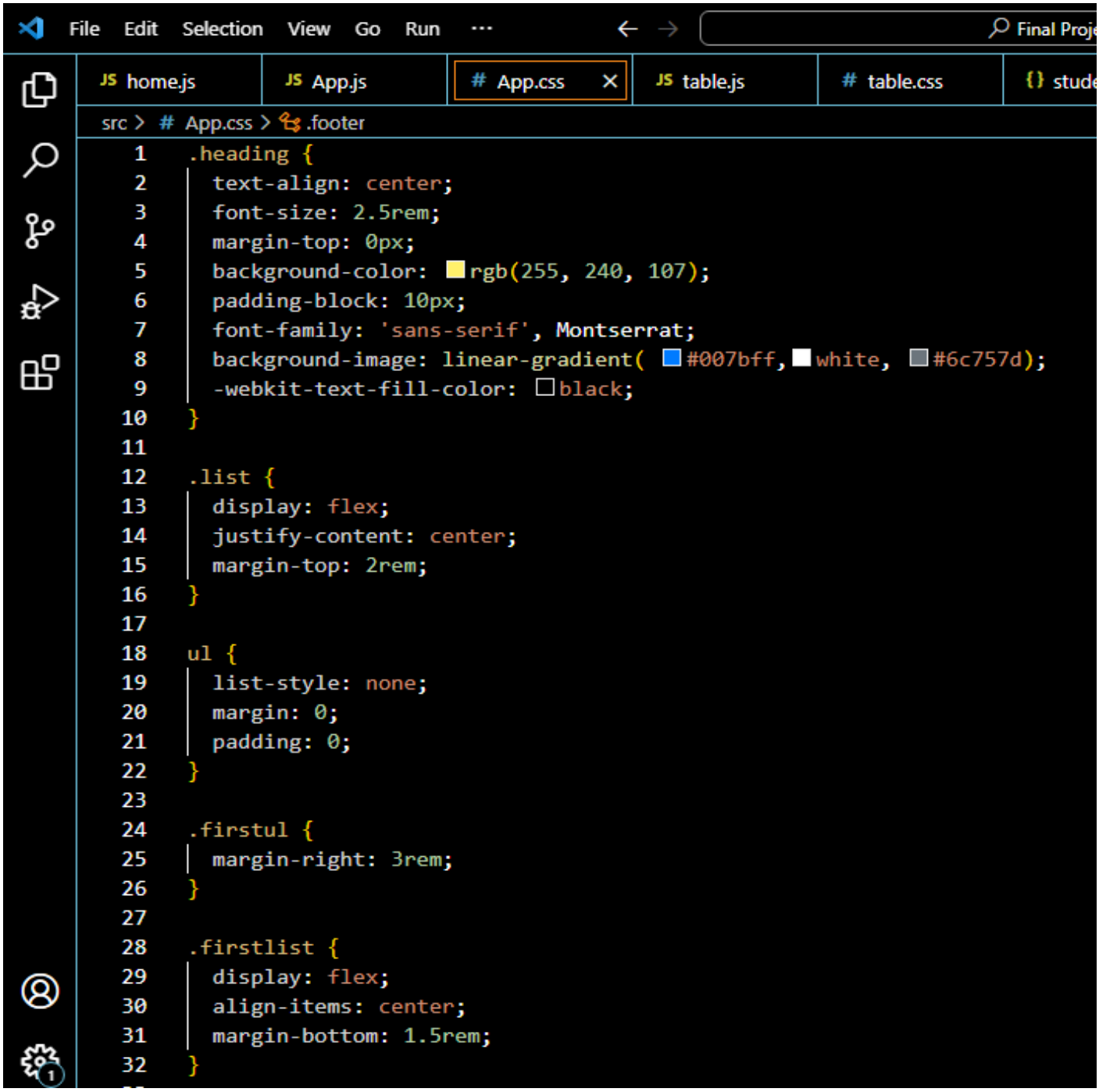
APP.JS



HOME.JS


File Edit Selection View Go Run ···
Final Project (gradebook)
home.js
App.js
App.css
table.js
table.css
studentdata.json
src > project > home.js > ...
1 import React from 'react';
2 import { FaUniversity, FaChalkboardTeacher, FaLaptopCode, FaClock, FaUsers, FaBook } from 'react-icons/fa';
3
4 const Home = () => {
5 return (
6
7
8 <h1 className='heading'><FaBook /> Grading Digital Book</h1>
9
10 <ul className='firstul'>
11 <li className='firstlist'><FaUniversity />College : Lovely Professional University
12 <li className='firstlist'><FaChalkboardTeacher />Professor's Name : Mir Junaid Rasool
13 <li className='firstlist'><FaLaptopCode />Department : Computer Science
14
15 <ul className='secondul'>
16 <li id='listtwo1'><FaClock /> Semister : 8th
17 <li id='listtwo2'><FaUsers /> Section : K19FE
18 <li id='listtwo3'><FaUsers /> Group : 1
19
20
21
22)
23 }
24
25 export default Home
26

APP.CSS

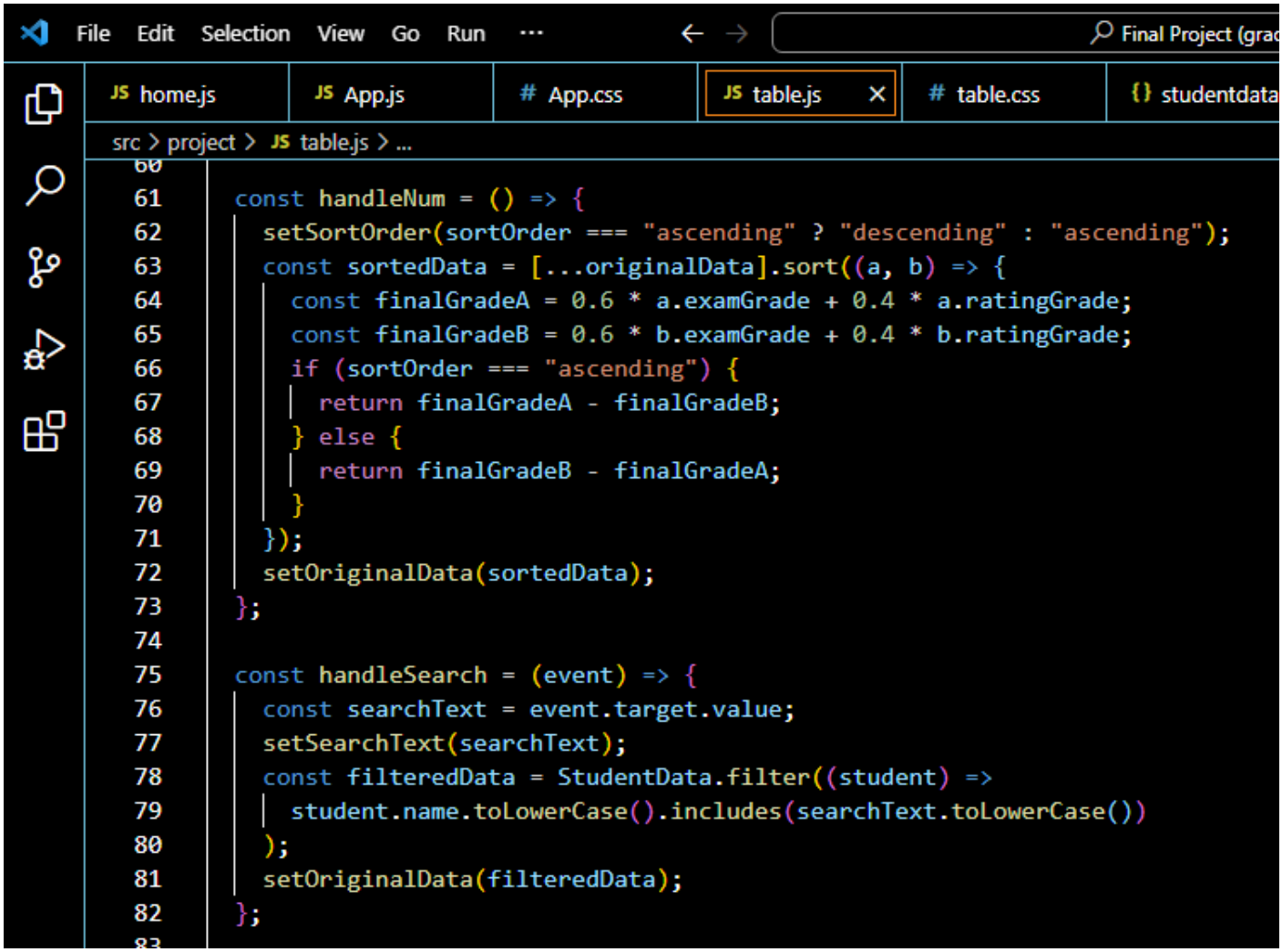

File Edit Selection View Go Run ···
Final Proje
home.js
App.js
App.css
table.js
table.css
stude
src > # App.css > .footer
.heading {
text-align: center;
font-size: 2.5rem;
margin-top: 0px;
background-color: rgb(255, 240, 107);
padding-block: 10px;
font-family: 'sans-serif', Montserrat;
background-image: linear-gradient(#007bff, white, #6c757d);
-webkit-text-fill-color: black;
}
.list {
display: flex;
justify-content: center;
margin-top: 2rem;
}
ul {
list-style: none;
margin: 0;
padding: 0;
}
.firstul {
margin-right: 3rem;
}
.firstlist {
display: flex;
align-items: center;
margin-bottom: 1.5rem;
}


File Edit Selection View Go Run ···
Final Pr
home.js
App.js
App.css
table.js
table.css
src > # App.css > .footer
}
.firstlist {
display: flex;
align-items: center;
margin-bottom: 1.5rem;
}
#listtwo1,
#listtwo2,
#listtwo3 {
display: flex;
align-items: center;
margin-bottom: 1.5rem;
}
span {
margin-left: 5px;
font-weight: bold;
}
.footer{
background-image: linear-gradient(#6c757d, white, yellow);
padding-bottom: 10px;
padding-top: 1px;
font-size: 15px;
color: black;
margin-bottom: 0px;
position: relative;
bottom: 0px;
text-align: center;
}

MAIN TABLE COMPONENT:-


```
1   import React, { useState } from "react";
2   import StudentData from './studentdata.json';
3   import './table.css';
4
5   const Table = () => {
6     const [selectedStudent, setSelectedStudent] = useState(null);
7     const [filter, setFilter] = useState("All");
8     const [sortOrder, setSortOrder] = useState("ascending");
9     const [originalData, setOriginalData] = useState(StudentData);
10    const [searchText, setSearchText] = useState("");
11    const [selectedRow, setSelectedRow] = useState(null);
12    const [showStats, setShowStats] = useState(false);
13    const grades = StudentData.map((student) => student.examGrade);
14    const maxGrade = Math.max(...grades);
15    const minGrade = Math.min(...grades);
16    const avgGrade = grades.reduce((acc, grade) => acc + grade, 0) / grades.length;
17
18    let gradeRange3to4 = 0;
19  StudentData.forEach((student) => {
20    const finalGrade = 0.6 * student.examGrade + 0.4 * student.ratingGrade;
21    if (finalGrade >= 3 && finalGrade < 4) {
22      gradeRange3to4++;
23    }
24  });
25
26  let gradeRange4to5 = 0;
27  StudentData.forEach((student) => {
28    const finalGrade = 0.6 * student.examGrade + 0.4 * student.ratingGrade;
29    if (finalGrade >= 4 && finalGrade <= 5) {
30      gradeRange4to5++;
31    }
32  });
```

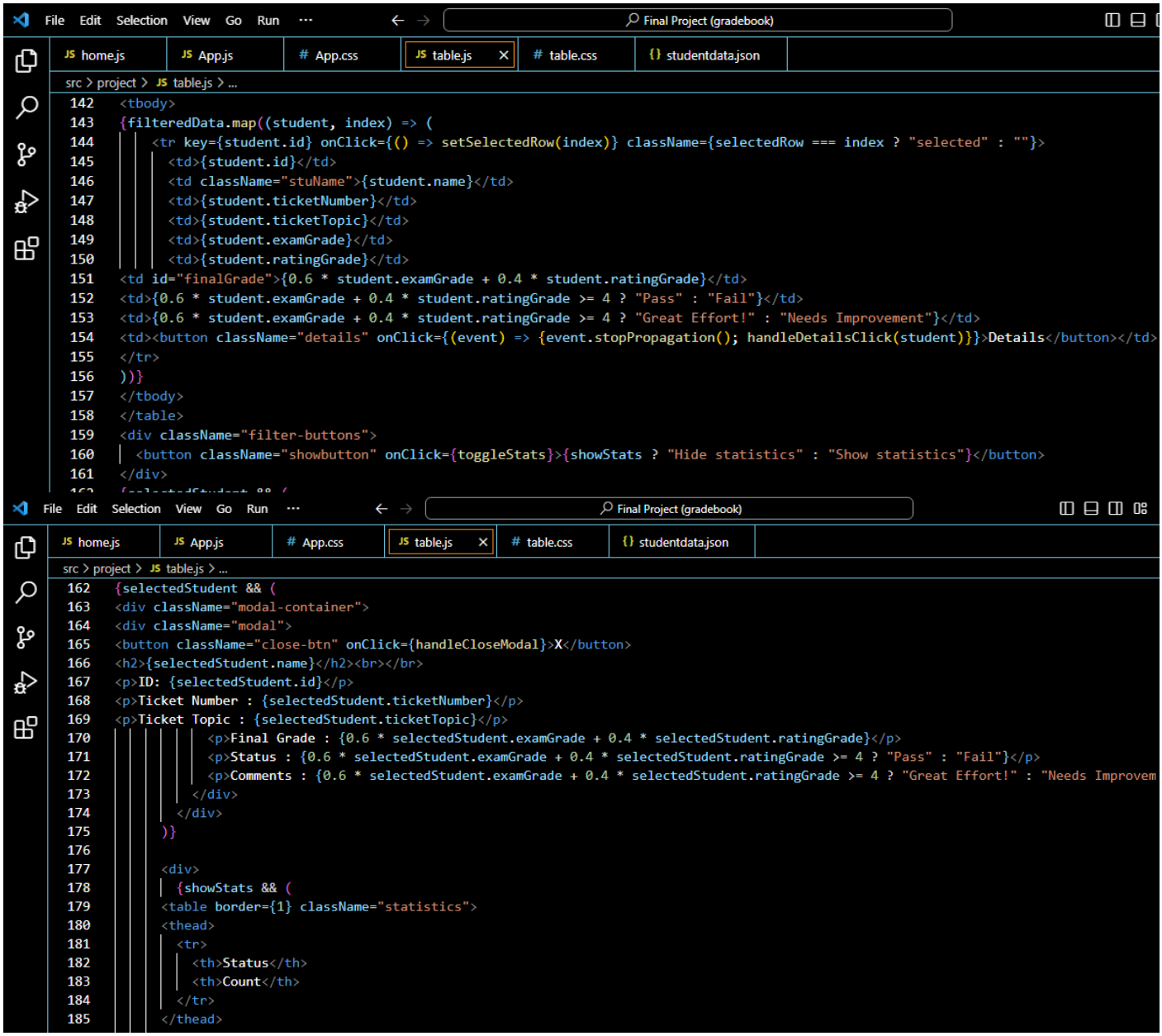

File Edit Selection View Go Run ···
Final Project (gr
home.js
App.js
App.css
table.js
table.css
studentda
src > project > table.js > ...
34 const gradeCounts = {
35 "3-4": gradeRange3to4,
36 "4-5": gradeRange4to5,
37 };
38
39 const handleCloseModal = () => {
40 setSelectedStudent(null);
41 };
42
43 const handleFilter = (filter) => {
44 setFilter(filter);
45 setSortOrder("ascending");
46 setOriginalData(StudentData);
47 };
48
49 const handleSort = () => {
50 setSortOrder(sortOrder === "ascending" ? "descending" : "ascending");
51 const sortedData = [...originalData].sort((a, b) => {
52 const nameA = a.name.toLowerCase();
53 const nameB = b.name.toLowerCase();
54 if (nameA < nameB) return sortOrder === "ascending" ? -1 : 1;
55 if (nameA > nameB) return sortOrder === "ascending" ? 1 : -1;
56 return 0;
57 });
58 setOriginalData(sortedData);
59 };
60

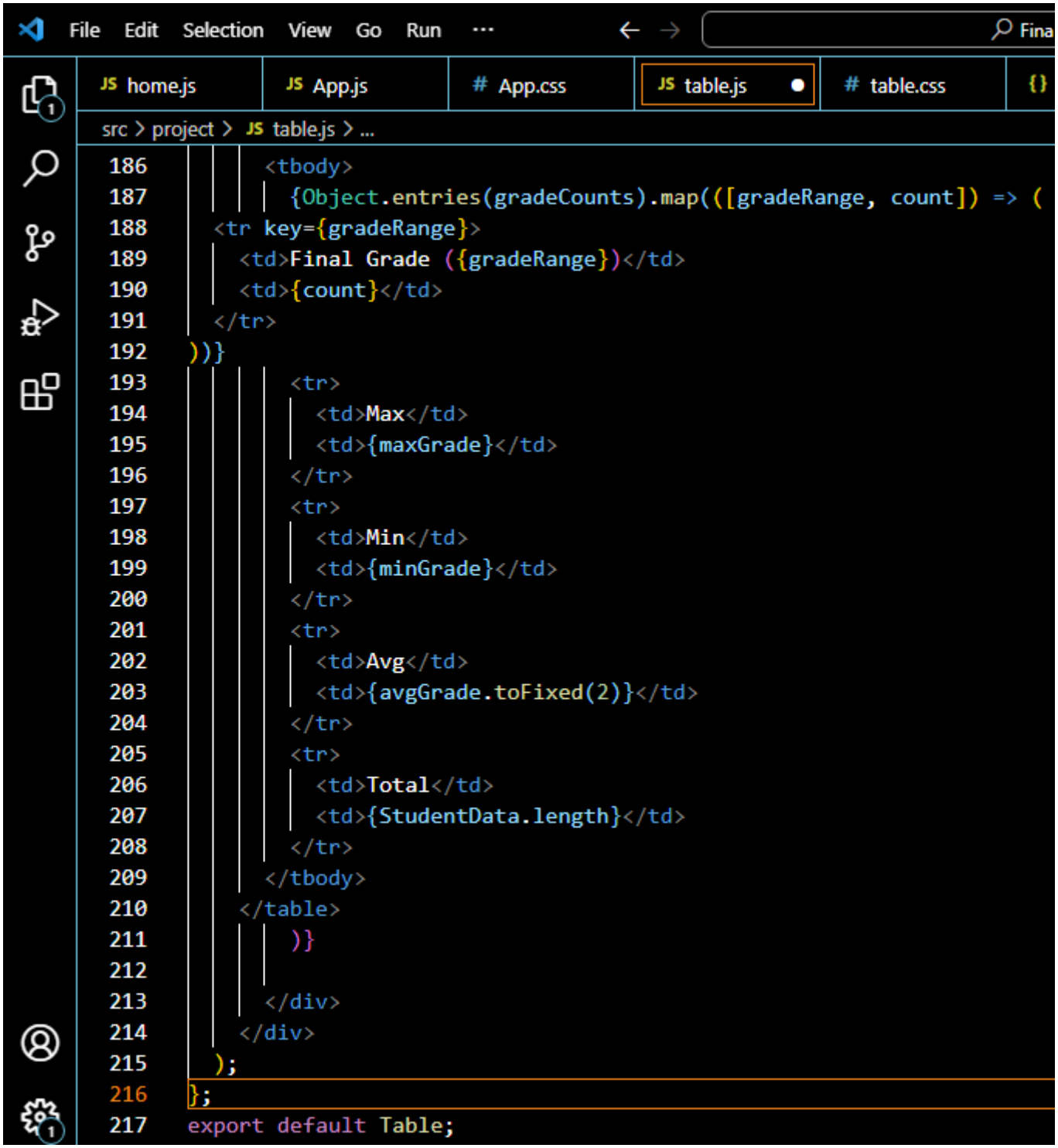

File Edit Selection View Go Run ···
Final Project (grad
home.js
App.js
App.css
table.js
table.css
studentdata
src > project > table.js > ...
60
61 const handleNum = () => {
62 setSortOrder(sortOrder === "ascending" ? "descending" : "ascending");
63 const sortedData = [...originalData].sort((a, b) => {
64 const finalGradeA = 0.6 * a.examGrade + 0.4 * a.ratingGrade;
65 const finalGradeB = 0.6 * b.examGrade + 0.4 * b.ratingGrade;
66 if (sortOrder === "ascending") {
67 return finalGradeA - finalGradeB;
68 } else {
69 return finalGradeB - finalGradeA;
70 }
71 });
72 setOriginalData(sortedData);
73 };
74
75 const handleSearch = (event) => {
76 const searchText = event.target.value;
77 setSearchText(searchText);
78 const filteredData = StudentData.filter((student) =>
79 student.name.toLowerCase().includes(searchText.toLowerCase())
80);
81 setOriginalData(filteredData);
82 };
83


File Edit Selection View Go Run ···
Final Project (gradebook)
home.js
App.js
App.css
table.js
table.css
studentdata.json
src > project > table.js > ...
const filteredData = originalData.filter((student) => {
const finalGrade = 0.6 * student.examGrade + 0.4 * student.ratingGrade;
switch (filter) {
case "All":
return true;
case "Passed":
return finalGrade >= 4;
case "Failed":
return finalGrade < 4;
default:
return true;
}
});
const downloadFile = () => {
const blob = new Blob([JSON.stringify(StudentData)], {type: 'application/json'});
const url = URL.createObjectURL(blob);
const a = document.createElement('a');
a.href = url;
a.download = 'studentdata.json';
a.click();
};
const handleDetailsClick = (student, index) => {
setSelectedStudent(student);
setSelectedRow(null);
};
const toggleStats = () => {
setShowStats(!showStats);
};
File Edit Selection View Go Run ···
Final Project (gradebook)
home.js
App.js
App.css
table.js
table.css
studentdata.json
src > project > table.js > ...
return (

<button className="sort" onClick={() => handleFilter("All")}>All</button>
<button className="sort" onClick={() => handleFilter("Passed")}>Passed</button>
<button className="sort" onClick={() => handleFilter("Failed")}>Failed</button>
<button className="sort" onClick={handleSort}>{sortOrder === "ascending" ? "A - Z" : "Z - A"}</button>
<button className="sort" onClick={handleNum}>{sortOrder === "ascending" ? "1 - 10" : "10 - 1"}</button>
<input className="search" type="text" placeholder="Search by name..." value={searchText} onChange={handleSearch} />
<button className="download" onClick={downloadFile}>Download</button>

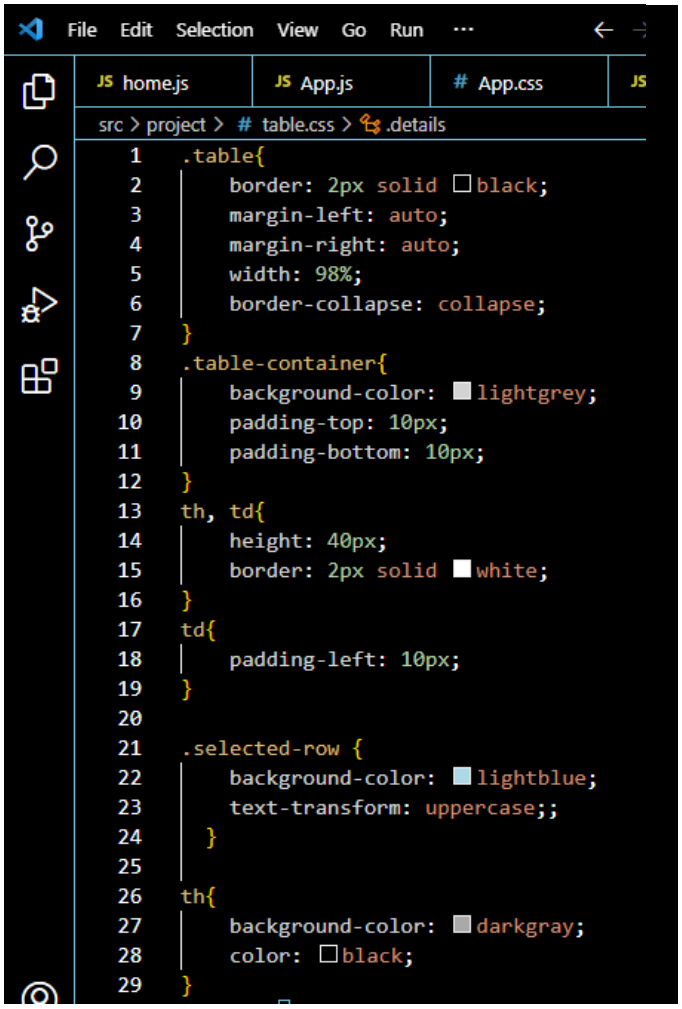
<table border={1} className='table'>
<thead>
<tr>
<th>Id</th>
<th>Name</th>
<th>Ticket Number</th>
<th>Ticket Topic</th>
<th>Exam Grade</th>
<th>Rating Grade</th>
<th>Final Grade</th>
<th>Status</th>
<th>Comments</th>
<th>Info</th>
</tr>
</thead>


File Edit Selection View Go Run ···
Final Project (gradebook)
home.js
App.js
App.css
table.js
table.css
studentdata.json
src > project > table.js > ...
142 <tbody>
143 {filteredData.map((student, index) => (
144 <tr key={student.id} onClick={() => setSelectedRow(index)} className={selectedRow === index ? "selected" : ""}>
145 <td>{student.id}</td>
146 <td className="stuName">{student.name}</td>
147 <td>{student.ticketNumber}</td>
148 <td>{student.ticketTopic}</td>
149 <td>{student.examGrade}</td>
150 <td>{student.ratingGrade}</td>
151 <td id="finalGrade">{0.6 * student.examGrade + 0.4 * student.ratingGrade}</td>
152 <td>{0.6 * student.examGrade + 0.4 * student.ratingGrade >= 4 ? "Pass" : "Fail"}</td>
153 <td>{0.6 * student.examGrade + 0.4 * student.ratingGrade >= 4 ? "Great Effort!" : "Needs Improvement"}</td>
154 <td><button className="details" onClick={(event) => {event.stopPropagation(); handleDetailsClick(student)}}>Details</button></td>
155 </tr>
156))}
157 </tbody>
158 </table>
159
160 <button className="showbutton" onClick={toggleStats}>{showStats ? "Hide statistics" : "Show statistics"}</button>
161
File Edit Selection View Go Run ···
Final Project (gradebook)
home.js
App.js
App.css
table.js
table.css
studentdata.json
src > project > table.js > ...
162 {selectedStudent && (
163
164
165 <button className="close-btn" onClick={handleCloseModal}>X</button>
166 <h2>{selectedStudent.name}</h2>
</br>
167 ID: {selectedStudent.id}
168 Ticket Number : {selectedStudent.ticketNumber}
169 Ticket Topic : {selectedStudent.ticketTopic}
170 Final Grade : {0.6 * selectedStudent.examGrade + 0.4 * selectedStudent.ratingGrade}
171 Status : {0.6 * selectedStudent.examGrade + 0.4 * selectedStudent.ratingGrade >= 4 ? "Pass" : "Fail"}
172 Comments : {0.6 * selectedStudent.examGrade + 0.4 * selectedStudent.ratingGrade >= 4 ? "Great Effort!" : "Needs Improvem
173
174
175)}
176
177
178 {showStats && (
179 <table border={1} className="statistics">
180 <thead>
181 <tr>
182 <th>Status</th>
183 <th>Count</th>
184 </tr>
185 </thead>

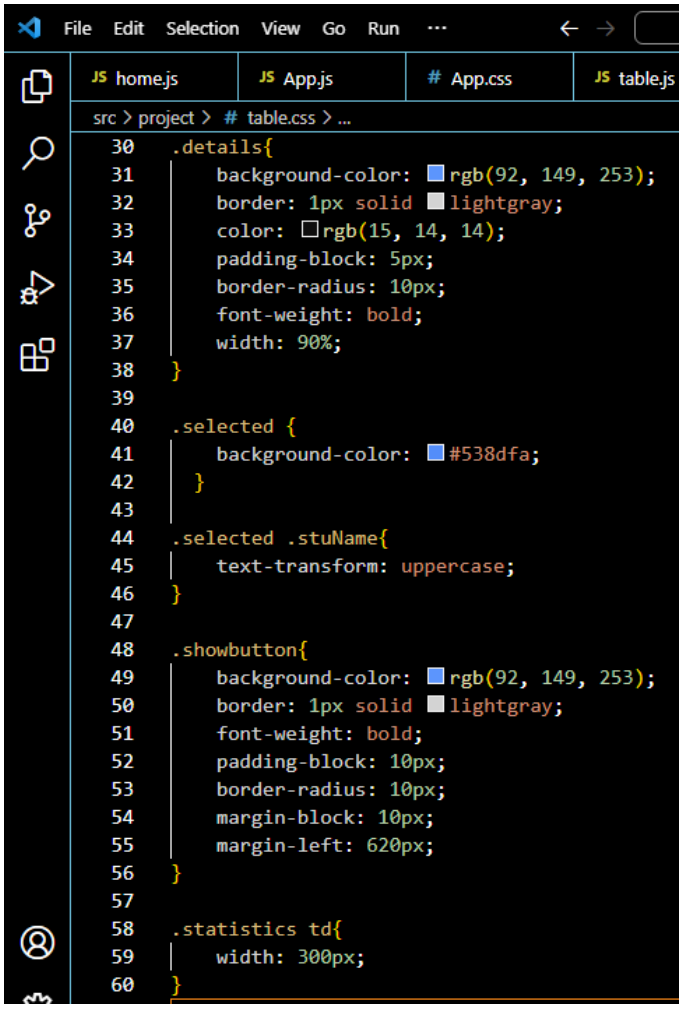

File Edit Selection View Go Run ···
home.js
App.js
App.css
table.js
table.css
src > project > table.js > ...
<tbody>
{Object.entries(gradeCounts).map(([gradeRange, count]) => (
<tr key={gradeRange}>
<td>Final Grade ({gradeRange})</td>
<td>{count}</td>
</tr>
))}
<tr>
<td>Max</td>
<td>{maxGrade}</td>
</tr>
<tr>
<td>Min</td>
<td>{minGrade}</td>
</tr>
<tr>
<td>Avg</td>
<td>{avgGrade.toFixed(2)}</td>
</tr>
<tr>
<td>Total</td>
<td>{StudentData.length}</td>
</tr>
</tbody>
</table>
)}

);
};
export default Table;

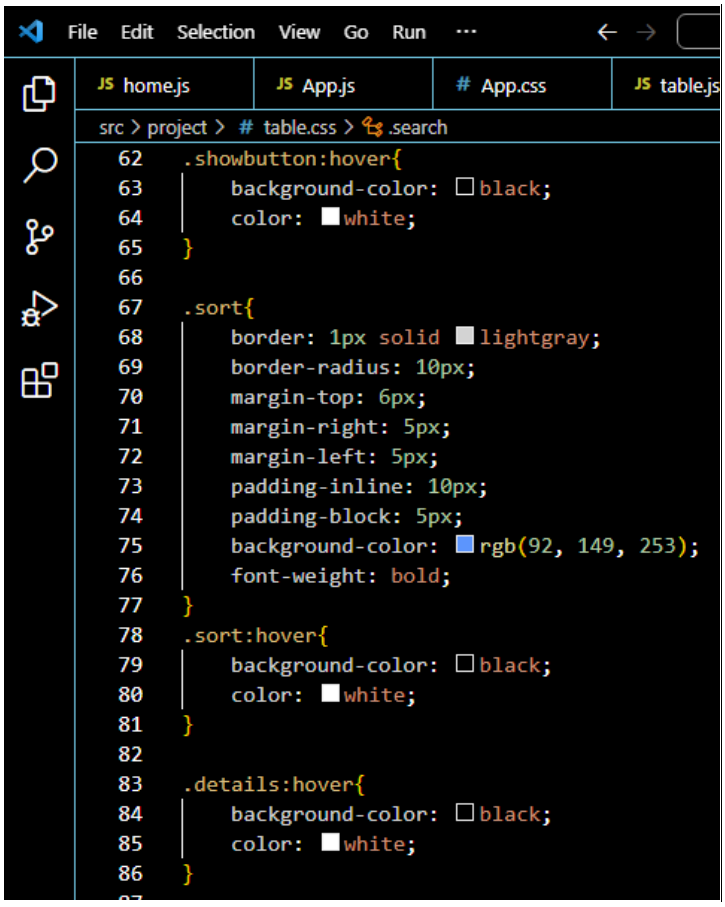
TABLE.CSS



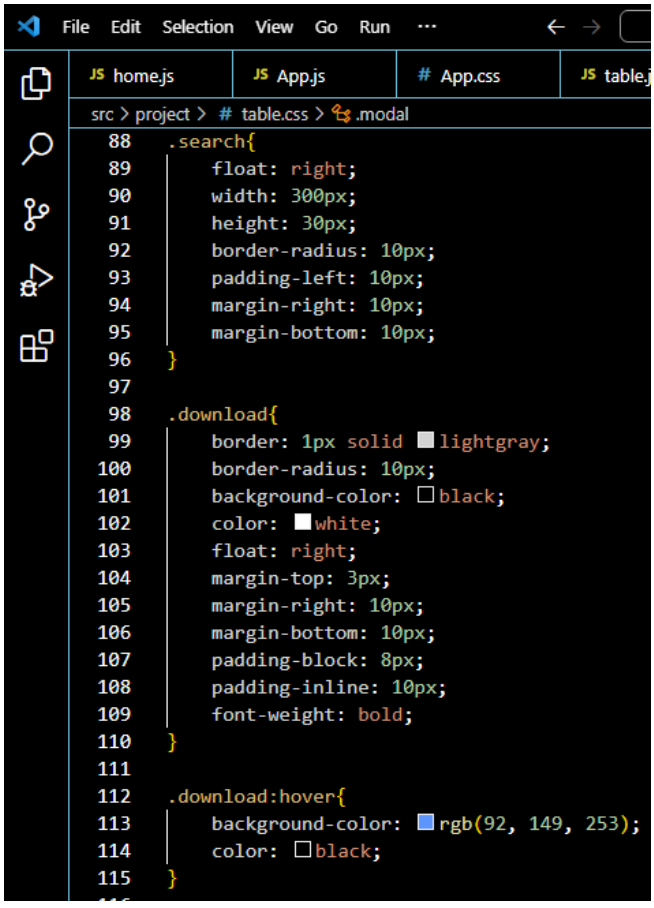
```
.table{
    border: 2px solid black;
    margin-left: auto;
    margin-right: auto;
    width: 98%;
    border-collapse: collapse;
}
.table-container{
    background-color: lightgrey;
    padding-top: 10px;
    padding-bottom: 10px;
}
th, td{
    height: 40px;
    border: 2px solid white;
}
td{
    padding-left: 10px;
}

.selected-row {
    background-color: lightblue;
    text-transform: uppercase;;
  }

th{
    background-color: darkgray;
    color: black;
}
```



```
.details{
    background-color: rgb(92, 149, 253);
    border: 1px solid lightgray;
    color: rgb(15, 14, 14);
    padding-block: 5px;
    border-radius: 10px;
    font-weight: bold;
    width: 90%;
}

.selected {
    background-color: #538dfa;
  }

.selected .stuName{
    text-transform: uppercase;
}

.showbutton{
    background-color: rgb(92, 149, 253);
    border: 1px solid lightgray;
    font-weight: bold;
    padding-block: 10px;
    border-radius: 10px;
    margin-block: 10px;
    margin-left: 620px;
}

.statistics td{
    width: 300px;
}
```



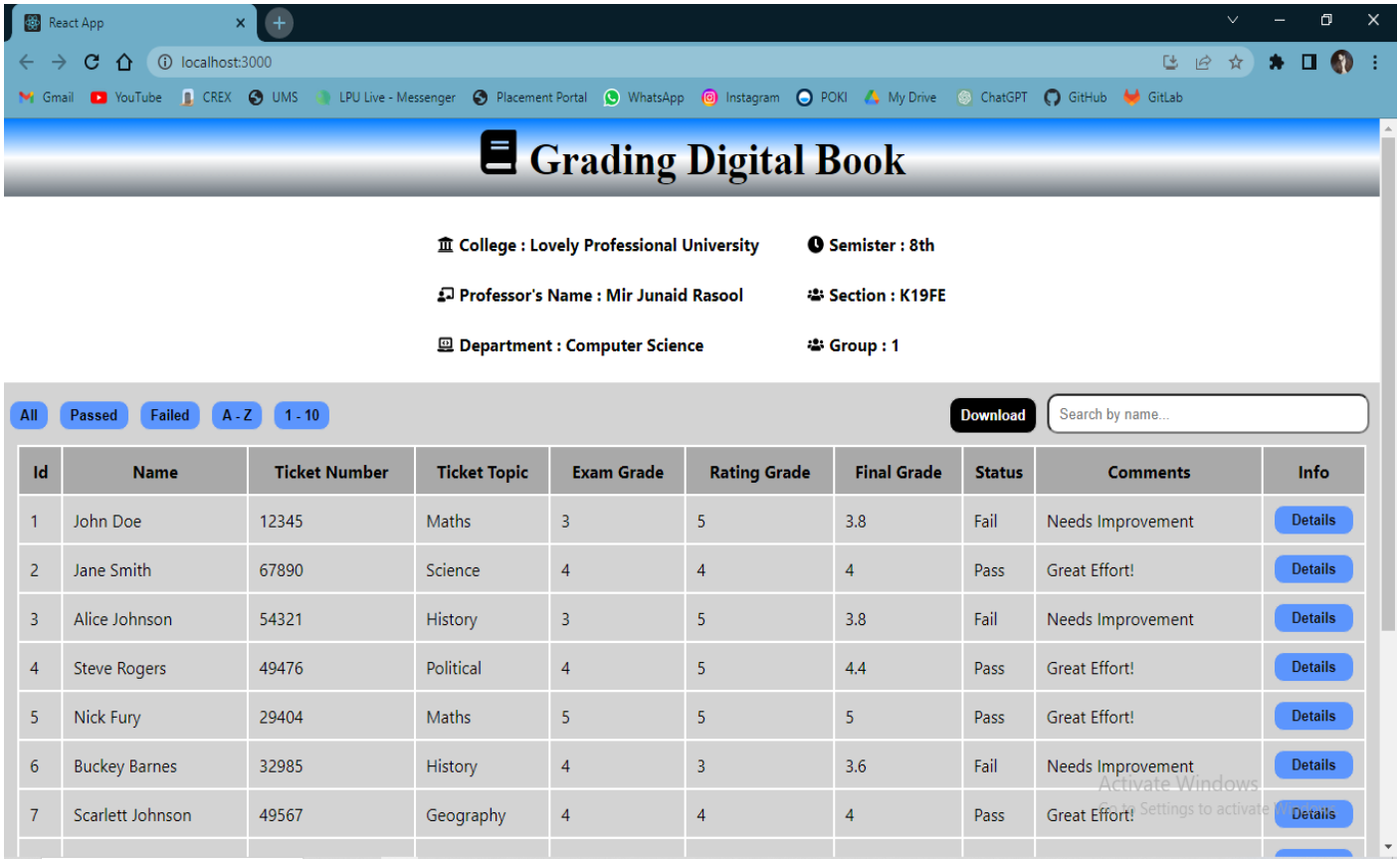
```
.showbutton:hover{
    background-color: black;
    color: white;
}

.sort{
    border: 1px solid lightgray;
    border-radius: 10px;
    margin-top: 6px;
    margin-right: 5px;
    margin-left: 5px;
    padding-inline: 10px;
    padding-block: 5px;
    background-color: rgb(92, 149, 253);
    font-weight: bold;
}
.sort:hover{
    background-color: black;
    color: white;
}

.details:hover{
    background-color: black;
    color: white;
}
```



```
.search{
    float: right;
    width: 300px;
    height: 30px;
    border-radius: 10px;
    padding-left: 10px;
    margin-right: 10px;
    margin-bottom: 10px;
}

.download{
    border: 1px solid lightgray;
    border-radius: 10px;
    background-color: black;
    color: white;
    float: right;
    margin-top: 3px;
    margin-right: 10px;
    margin-bottom: 10px;
    padding-block: 8px;
    padding-inline: 10px;
    font-weight: bold;
}

.download:hover{
    background-color: rgb(92, 149, 253);
    color: black;
}
```

```
File  Edit  Selection  View  Go  Run  ···

JS home.js | JS App.js | # App.css

src > project > # table.css > ...

117  .modal{
118      border-radius: 10px;
119      padding: 20px;
120      background-color: black;
121      color: white;
122      position: absolute;
123      left: 42%;
124      top: 200px;
125  }
126  .close-btn{
127      float: right;
128      padding: 5px;
129      border-radius: 10px;
130      font-size: xx-small;
131  }
```

FINAL OUTPUT: -



Id	Name	Ticket Number	Ticket Topic	Exam Grade	Rating Grade	Final Grade	Status	Comments	Info
1	John Doe	12345	Maths	3	5	3.8	Fail	Needs Improvement	Details
2	Jane Smith	67890	Science	4	4	4	Pass	Great Effort!	Details
3	Alice Johnson	54321	History	3	5	3.8	Fail	Needs Improvement	Details
4	Steve Rogers	49476	Political	4	5	4.4	Pass	Great Effort!	Details
5	Nick Fury	29404	Maths	5	5	5	Pass	Great Effort!	Details
6	Buckey Barnes	32985	History	4	3	3.6	Fail	Needs Improvement	Details
7	Scarlett Johnson	49567	Geography	4	4	4	Pass	Great Effort!	Details
8	Jake Sully	93257	Geography	5	3	4.2	Pass	Great Effort!	Details
9	Rachel Green	50683	Fashion	4	4	4	Pass	Great Effort!	Details
10	Joey Tribiyani	80765	Acting	4	3	3.6	Fail	Needs Improvement	Details

Show statistics

EPAM Project

Designed by : Yogendra Reddy

Id	Name	Ticket Number	Ticket Topic	Exam Grade	Rating Grade	Final Grade	Status	Comments	Info
7	Scarlett Johnson	49567	Geography	4	4	4	Pass	Great Effort!	Details
8	Jake Sully	93257	Geography	5	3	4.2	Pass	Great Effort!	Details
9	Rachel Green	50683	Fashion	4	4	4	Pass	Great Effort!	Details
10	Joey Tribiyani	80765	Acting	4	3	3.6	Fail	Needs Improvement	Details

Hide statistics

Status	Count
Final Grade (3-4)	4
Final Grade (4-5)	6
Max	5
Min	3
Avg	4.00
Total	10

EPAM Project

Designed by : Yogendra Reddy

CONCLUSION:-

In conclusion, the development of a digital gradebook app using React has the potential to revolutionize the gradebook management process in educational institutions. By providing a centralized platform for teachers, students, and administrators to manage gradebook data, the app can streamline the gradebook management process, reduce errors, and enhance communication and collaboration among stakeholders.

The app can also enhance the learning experience for students by providing timely and accurate feedback on their performance, helping them identify areas for improvement, and allowing them to track their progress over time. Moreover, the app's built-in reporting and analysis tools can help teachers and administrators make data-driven decisions about student performance and progress, leading to a more effective educational system.

In addition to the benefits outlined above, the development of a digital gradebook app can also provide opportunities for customization and integration with other educational systems, further enhancing its utility and impact.

Overall, the development of a digital gradebook app using React can lead to an improved, more efficient, and effective gradebook management system that can benefit both teachers and students in educational institutions.

FUTURE SCOPE

The development of a digital gradebook app using React has great potential for future scope and expansion. The following are some of the future scopes for this project:

Integration with other educational systems: The app can be integrated with other educational systems such as learning management systems (LMS) and student information systems (SIS) to provide a seamless experience for teachers and students.

Enhancing security and privacy: The app can be enhanced with security and privacy features to ensure the protection of sensitive grade book data.

Artificial Intelligence (AI) integration: The app can be integrated with AI technology to provide advanced analysis of grade book data and predict student performance based on historical data.

Customization options: The app can be further customized to meet the specific needs of different educational institutions, such as the ability to customize grading scales or assessment types.

Mobile optimization: The app can be optimized for mobile devices, making it more accessible and convenient for teachers and students to access gradebook data on the go.

Cloud-based storage: The app can be moved to cloud-based storage to ensure easy access to gradebook data and enable real-time updates.

Gamification elements: The app can include gamification elements to make the gradebook management process more engaging and fun for students.

In conclusion, the future scope of the digital grade book app is promising, and there are numerous opportunities for expansion and enhancement to further improve the educational experience for teachers and students. With continued development and integration of advanced technologies, the digital grade book app can become an essential tool in the modern education system.

REFERENCES:-

WWW.YOUTUBE.COM

WWW.OPENAI.COM

----------THANK YOU ----------