

AGgrid and Reactjs

- ag-Grid is the industry standard for React Enterprise Applications.
- Developers using ag-Grid are building applications that would not be possible if ag-Grid did not exist.

Reference:

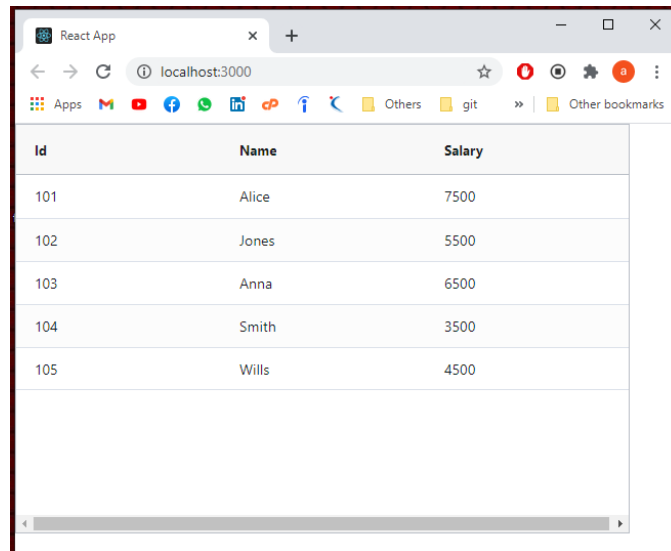
<https://www.ag-grid.com/react-grid/>

Installation AGgrid package.

```
npm install --save ag-grid-community ag-grid-react  
npm i ag-grid-enterprise
```

```
hello-app>npm install --save ag-grid-community ag-grid-react  
hello-app>npm i ag-grid-enterprise
```

Example: Basic Grid



A screenshot of a web browser window displaying a table with employee data. The browser's address bar shows 'localhost:3000'. The table has three columns: 'Id', 'Name', and 'Salary'. It contains five rows of data.

Id	Name	Salary
101	Alice	7500
102	Jones	5500
103	Anna	6500
104	Smith	3500
105	Wills	4500

index.js

```
import ReactDOM from "react-dom";
import "./index.css";

import React, { useState } from "react";
import { AgGridColumn, AgGridReact } from "ag-grid-react";

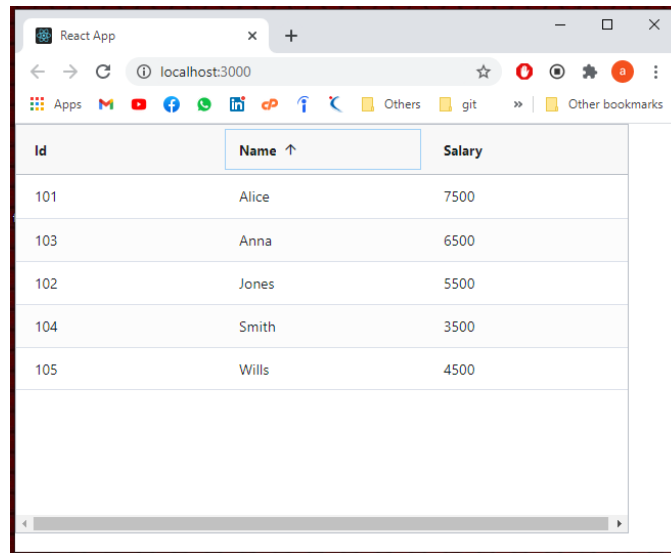
import "ag-grid-community/dist/styles/ag-grid.css";
import "ag-grid-community/dist/styles/ag-theme-alpine.css";

function Display() {
  const initialData = [
    { id: 101, name: "Alice", salary: 7500 },
    { id: 102, name: "Jones", salary: 5500 },
    { id: 103, name: "Anna", salary: 6500 },
    { id: 104, name: "Smith", salary: 3500 },
    { id: 105, name: "Wills", salary: 4500 },
  ];
  const [rowData, setRowData] = useState(initialData);

  return (
    <div className="ag-theme-alpine" style={{ height: 400, width: 600 }}>
      <AgGridReact rowData={rowData}>
        <AgGridColumn field="id" />
        <AgGridColumn field="name" />
        <AgGridColumn field="salary" />
      </AgGridReact>
    </div>
  );
}

ReactDOM.render(<Display />, document.getElementById("root"));
```

Example: Adding Sorting



Id	Name ↑	Salary
101	Alice	7500
103	Anna	6500
102	Jones	5500
104	Smith	3500
105	Wills	4500

index.js

```
import ReactDOM from "react-dom";
import "./index.css";

import React, { useState } from "react";
import { AgGridColumn, AgGridReact } from "ag-grid-react";

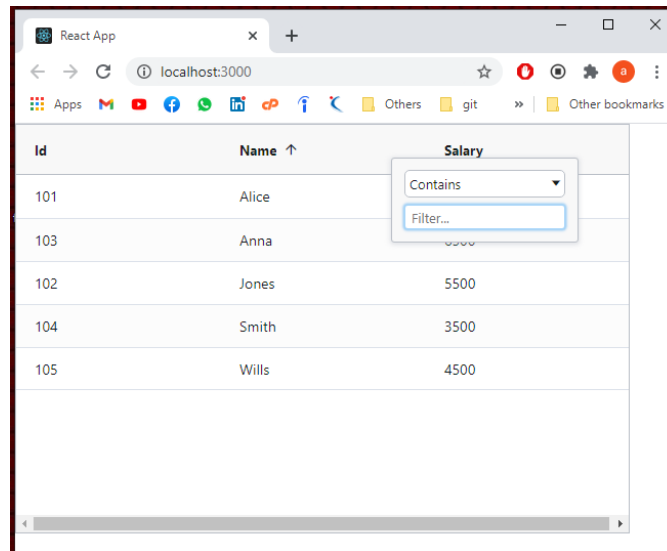
import "ag-grid-community/dist/styles/ag-grid.css";
import "ag-grid-community/dist/styles/ag-theme-alpine.css";

function Display() {
  const initialData = [
    { id: 101, name: "Alice", salary: 7500 },
    { id: 102, name: "Jones", salary: 5500 },
    { id: 103, name: "Anna", salary: 6500 },
    { id: 104, name: "Smith", salary: 3500 },
    { id: 105, name: "Wills", salary: 4500 },
  ];
  const [rowData, setRowData] = useState(initialData);

  return (
    <div className="ag-theme-alpine" style={{ height: 400, width: 600 }}>
      <AgGridReact rowData={rowData}>
        <AgGridColumn field="id" sortable={true} />
        <AgGridColumn field="name" sortable={true} />
        <AgGridColumn field="salary" sortable={true} />
      </AgGridReact>
    </div>
  );
}

ReactDOM.render(<Display />, document.getElementById("root"));
```

Example: Adding Filter



Id	Name ↑	Salary
101	Alice	7500
103	Anna	6500
102	Jones	5500
104	Smith	3500
105	Wills	4500

index.js

```
import ReactDOM from "react-dom";
import "./index.css";

import React, { useState } from "react";
import { AgGridColumn, AgGridReact } from "ag-grid-react";

import "ag-grid-community/dist/styles/ag-grid.css";
import "ag-grid-community/dist/styles/ag-theme-alpine.css";

function Display() {
  const initData = [
    { id: 101, name: "Alice", salary: 7500 },
    { id: 102, name: "Jones", salary: 5500 },
    { id: 103, name: "Anna", salary: 6500 },
    { id: 104, name: "Smith", salary: 3500 },
    { id: 105, name: "Wills", salary: 4500 },
  ];
  const [rowData, setRowData] = useState(initData);

  return (
    <div className="ag-theme-alpine" style={{ height: 400, width: 600 }}>
      <AgGridReact rowData={rowData}>
        <AgGridColumn field="id" sortable={true} filter={true} />
        <AgGridColumn field="name" sortable={true} filter={true} />
        <AgGridColumn field="salary" sortable={true} filter={true} />
      </AgGridReact>
    </div>
  );
}

ReactDOM.render(<Display />, document.getElementById("root"));
```

Example: Fetching data from backend server.

hello-app>json-server --watch db.json --port 4000

db.json

```
{
  "employees": [
    {
      "id": 1,
      "name": "Scott",
      "salary": 4000
    },
    {
      "id": 2,
      "name": "Jones",
      "salary": 7500
    }
  ]
}
```

index.js

```
import ReactDOM from "react-dom";
import "./index.css";

import React, { useState, useEffect } from "react";
import { AgGridColumn, AgGridReact } from "ag-grid-react";

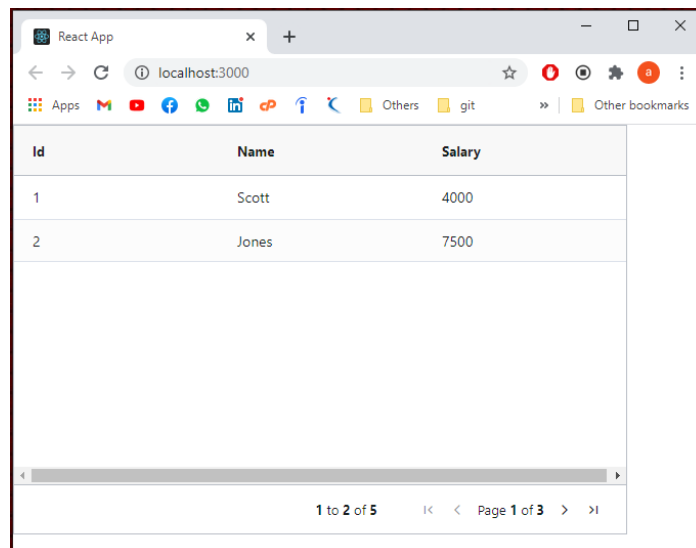
import "ag-grid-community/dist/styles/ag-grid.css";
import "ag-grid-community/dist/styles/ag-theme-alpine.css";

function Display() {
  const [rowData, setRowData] = useState([]);
  useEffect(() => {
    fetch("http://localhost:4000/employees")
      .then((result) => result.json())
      .then((rowData) => setRowData(rowData));
  }, []);

  return (
    <div className="ag-theme-alpine" style={{ height: 400, width: 600 }}>
      <AgGridReact rowData={rowData}>
        <AgGridColumn field="id" sortable={true} filter={true} />
        <AgGridColumn field="name" sortable={true} filter={true} />
        <AgGridColumn field="salary" sortable={true} filter={true} />
      </AgGridReact>
    </div>
  );
}

ReactDOM.render(<Display />, document.getElementById("root"));
```

Example: Adding Pagination



Id	Name	Salary
1	Scott	4000
2	Jones	7500

1 to 2 of 5 < > Page 1 of 3 > > >

hello-app>json-server --watch db.json --port 4000

db.json

```
{
  "employees": [
    {
      "id": 1,
      "name": "Scott",
      "salary": 4000
    },
    {
      "id": 2,
      "name": "Jones",
      "salary": 7500
    },
    {
      "id": 3,
      "name": "Allen",
      "salary": 5500
    },
    {
      "id": 4,
      "name": "Alice",
      "salary": 4000
    },
    {
      "id": 5,
      "name": "Smith",
      "salary": 7500
    }
  ]
}
```

index.js

```
import ReactDOM from "react-dom";
import "./index.css";

import React, { useState, useEffect } from "react";
import { AgGridColumn, AgGridReact } from "ag-grid-react";

import "ag-grid-community/dist/styles/ag-grid.css";
import "ag-grid-community/dist/styles/ag-theme-alpine.css";

function Display() {
  const [rowData, setRowData] = useState([]);
  useEffect(() => {
    fetch("http://localhost:4000/employees")
      .then((result) => result.json())
      .then((rowData) => setRowData(rowData));
  }, []);

  return (
    <div className="ag-theme-alpine" style={{ height: 400, width: 600 }}>
      <AgGridReact rowData={rowData} pagination={true} paginationPageSize="2">
        <AgGridColumn field="id" sortable={true} filter={true} />
        <AgGridColumn field="name" sortable={true} filter={true} />
        <AgGridColumn field="salary" sortable={true} filter={true} />
      </AgGridReact>
    </div>
  );
}

ReactDOM.render(<Display />, document.getElementById("root"));
```

Example: Separate columnDefs and defaultColDef.

```
import ReactDOM from "react-dom";
import "./index.css";

import React, { useState, useEffect } from "react";
import { AgGridColumn, AgGridReact } from "ag-grid-react";

import "ag-grid-community/dist/styles/ag-grid.css";
import "ag-grid-community/dist/styles/ag-theme-alpine.css";

function Display() {
  const columnDefs = [
    { headerName: "ID", field: "id" },
    { headerName: "NAME", field: "name" },
    { headerName: "SALARY", field: "salary" },
  ];

  const defaultColDef = {
    sortable: true,
    filter: true,
  };

  const [rowData, setRowData] = useState([]);
  useEffect(() => {
    fetch("http://localhost:4000/employees")
      .then((result) => result.json())
      .then((rowData) => setRowData(rowData));
  }, []);

  return (
    <div className="ag-theme-alpine" style={{ height: 400, width: 600 }}>
      <AgGridReact
        rowData={rowData}
        pagination={true}
        paginationPageSize="2"
        columnDefs={columnDefs}
        defaultColDef={defaultColDef}
      ></AgGridReact>
    </div>
  );
}

ReactDOM.render(<Display />, document.getElementById("root"));
```