EV Charging Station

IT832: Blockchain Technologies and Applications- Decentralization and Smart Contracts

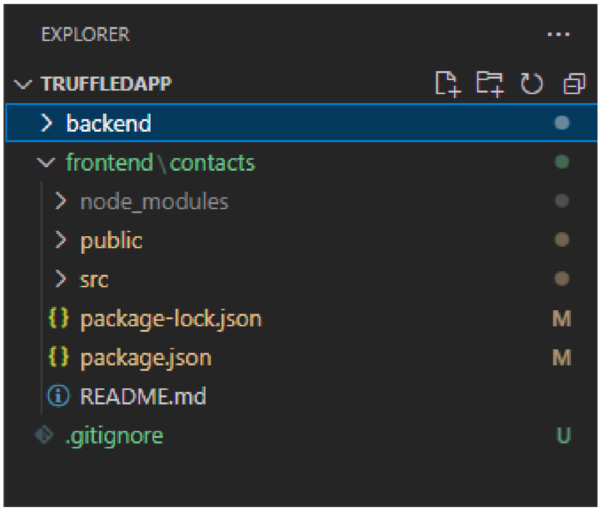
Assignment I

222IT008-Deepak Gaur

222IT007-Tushar Chaudhari

Frontend is developed using **React.js** framework along with **Schemantic UI framework**.

The code base structure looks like :



```
EXPLORER                         ...
v TRUFFLEDAPP
  > backend                      ●
  v frontend \ contacts          ●
    > node_modules               ●
    > public                     ●
    > src                        ●
    {} package-lock.json         M
    {} package.json              M
    ⓘ README.md
  .gitignore                     U
```

The entire project is uploaded on Github.

Repository link : https://github.com/tushar-chaudhari/Blockchain-Assignment-1

Steps to run Frontend :

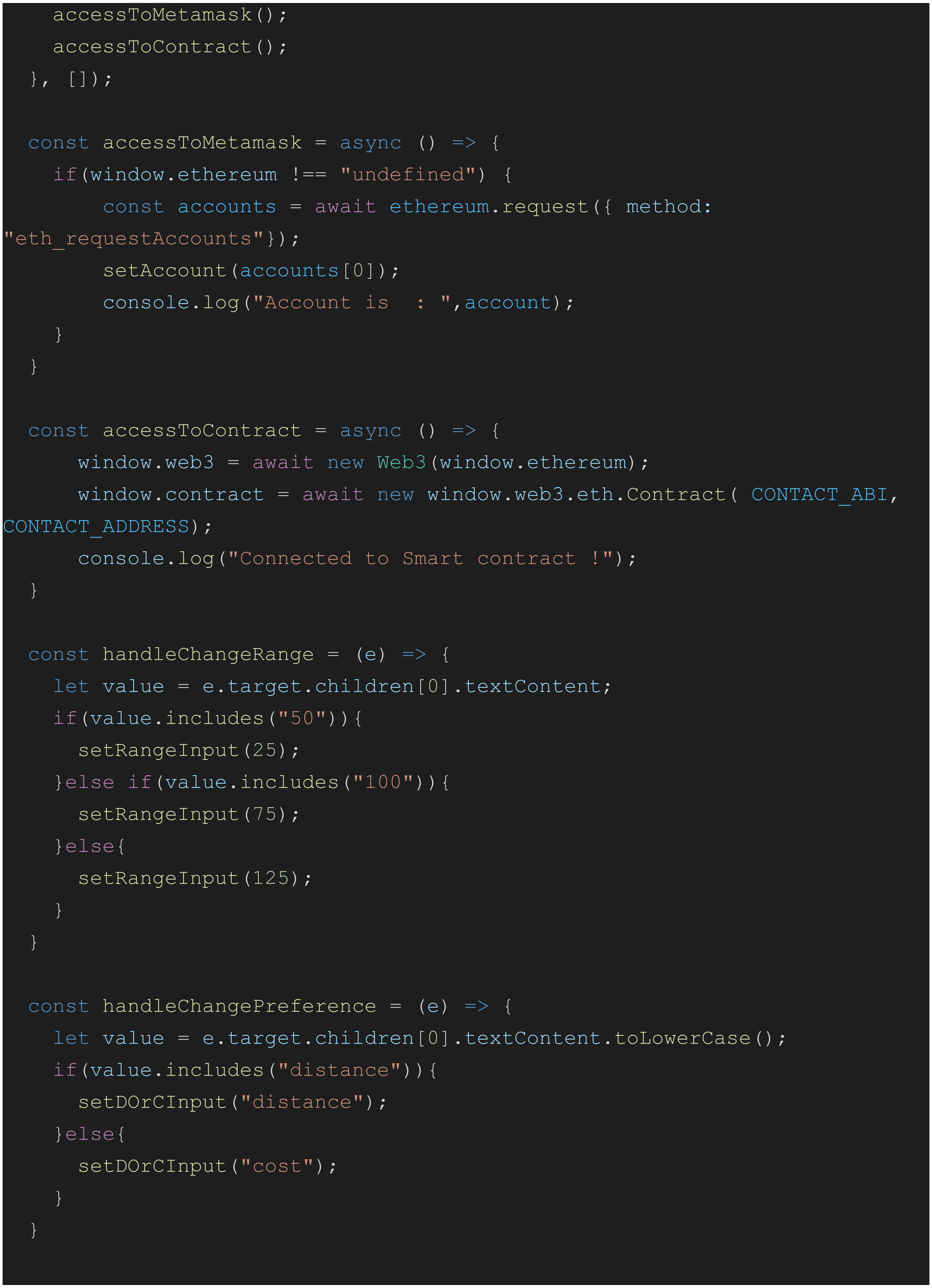
1. Go in frontend directory and open terminal/cmd.
2. Type **“npm start”** and click enter.It will start server and we can reach frontend on 3000(default port).
3. Run” https://http://localhost:3000/” to run the frontend

Some Important files are :

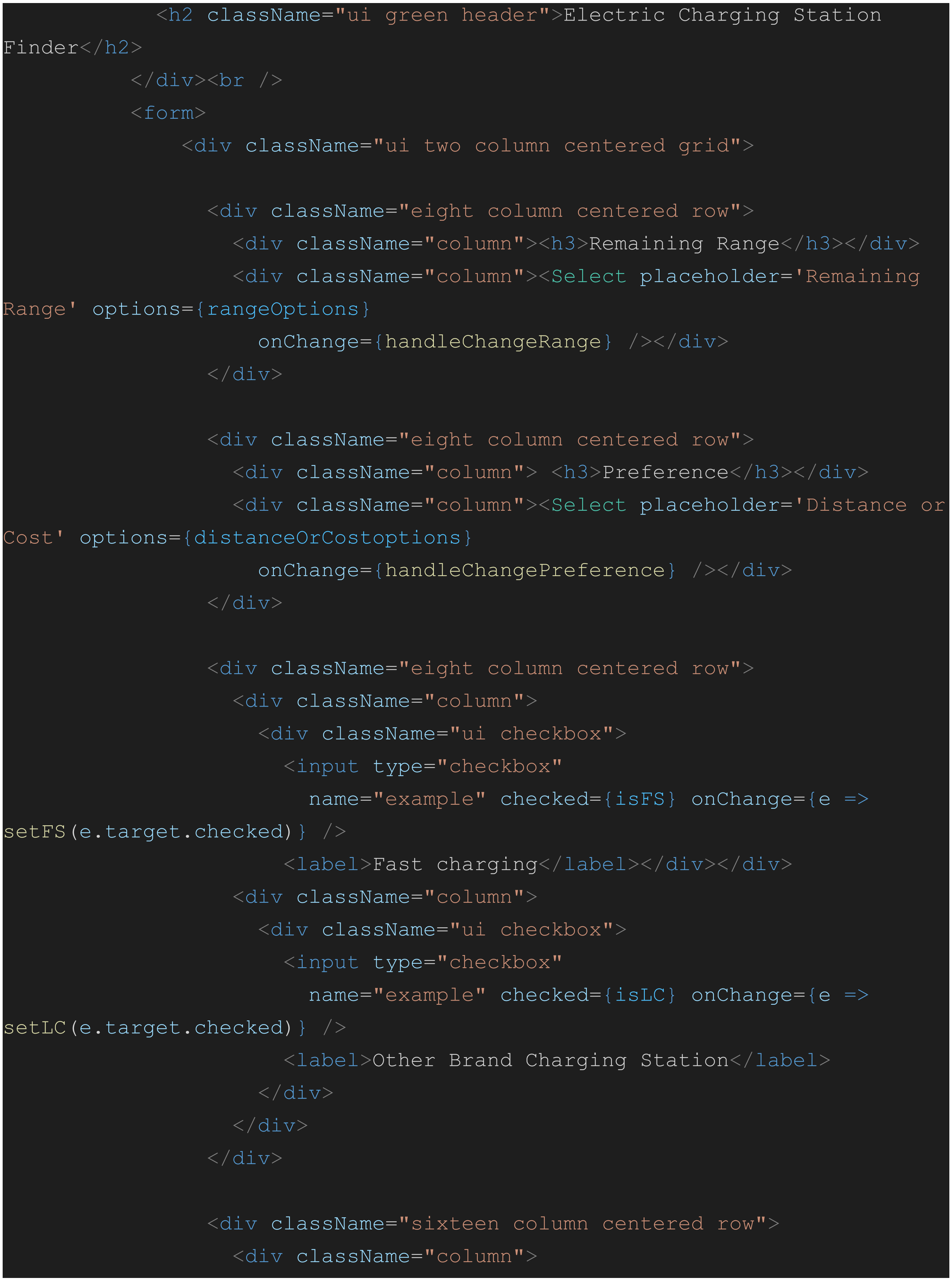
1. App.js
2. Index.css
3. configEVContract.js

⇒ App.js

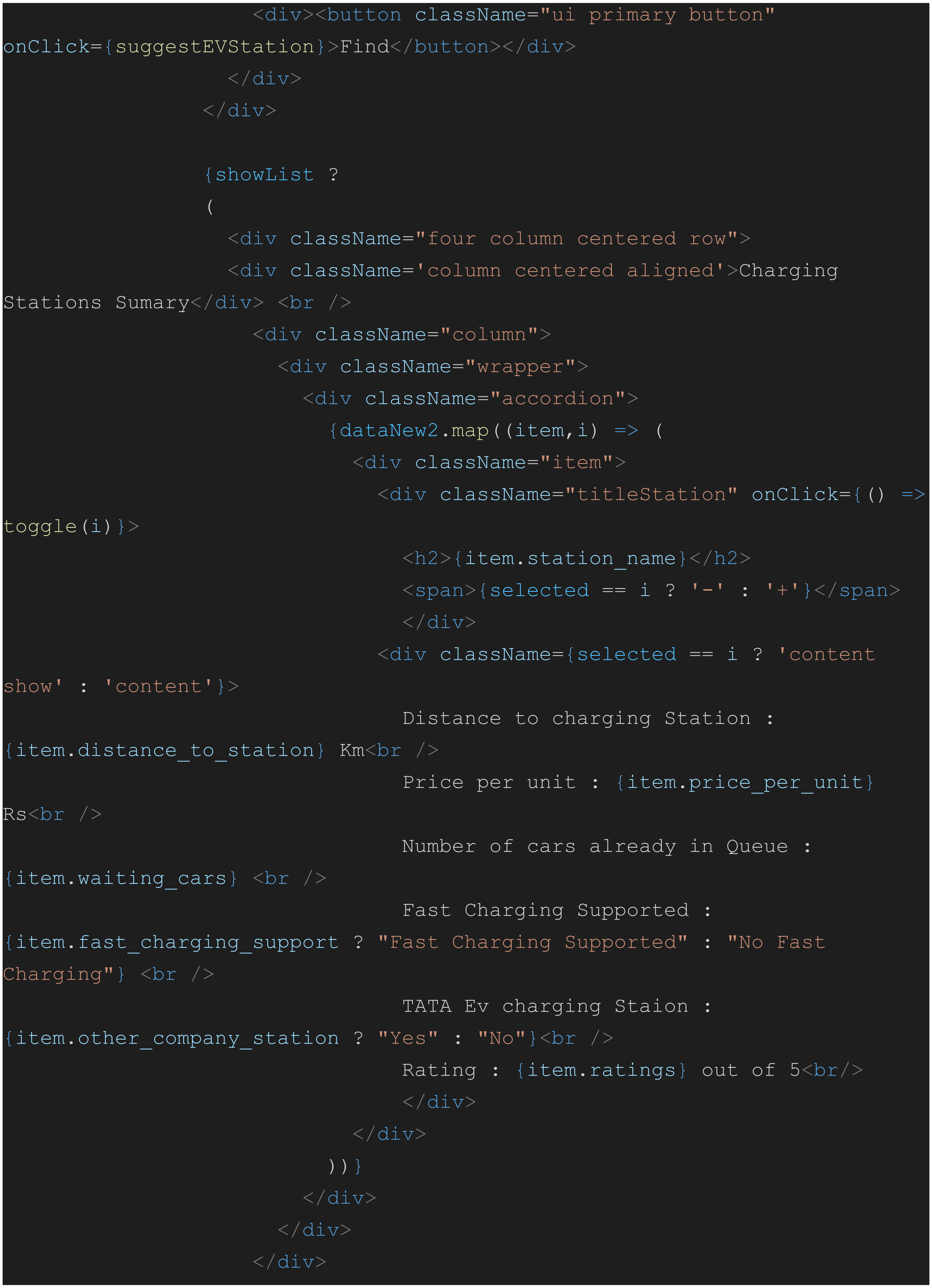
```
import { useEffect, useState } from 'react';
import Web3 from 'web3';
import { CONTACT_ABI, CONTACT_ADDRESS } from './configEVContract';
import 'semantic-ui-css/semantic.min.css'
import { Select } from 'semantic-ui-react'

function App() {
  const [dataNew2 , setDataNew2] = useState();
  const { ethereum } = window;
  const [account, setAccount] = useState();

  const [selected,setselected] = useState(null);
  const [rangeInput, setRangeInput]=useState();
  const [dorCInput, setDOrCInput]=useState();
  const [isFS, setFS] = useState(false);
  const [isLC, setLC] = useState(false);
  const [lengthOfAllotedStations,setLengthOfAllotedStations] = useState();
  const [showList,setShowList] = useState(false);

  const rangeOptions = [
    { key: 'af', value: 'af', text: '1-50' },
    { key: 'ax', value: 'ax', text: '51-100' },
    { key: 'al', value: 'al', text: '101-150' }
  ]

  const distanceOrCostoptions = [
    { key: 'af', value: 'af', text: 'Less Distance' },
    { key: 'ax', value: 'ax', text: 'Less Price per Unit' },
  ]

  const toggle = (i) => {
    if(selected == i){
      return setselected(null);
    }
    setselected(i);
  }

  useEffect(() => {
```



```
    accessToMetamask();
    accessToContract();
  }, []);

  const accessToMetamask = async () => {
    if(window.ethereum !== "undefined") {
        const accounts = await ethereum.request({ method:
"eth_requestAccounts"});
        setAccount(accounts[0]);
        console.log("Account is  : ",account);
    }
  }

  const accessToContract = async () => {
      window.web3 = await new Web3(window.ethereum);
      window.contract = await new window.web3.eth.Contract( CONTACT_ABI,
CONTACT_ADDRESS);
      console.log("Connected to Smart contract !");
  }

  const handleChangeRange = (e) => {
    let value = e.target.children[0].textContent;
    if(value.includes("50")){
      setRangeInput(25);
    }else if(value.includes("100")){
      setRangeInput(75);
    }else{
      setRangeInput(125);
    }
  }

  const handleChangePreference = (e) => {
    let value = e.target.children[0].textContent.toLowerCase();
    if(value.includes("distance")){
      setDOrCInput("distance");
    }else{
      setDOrCInput("cost");
    }
  }
  const suggestEVStation  = async (e) => {
    e.preventDefault();

    await window.contract.methods.suggestChargingStation(rangeInput,dorCInput,isFS,isLC).send({from : account});

    const data = await window.contract.methods.getCurrentRequestData().call();

    setLengthOfAllotedStations(data.allotedChargingStations.length);
    const dataLat = [];
    for (let i = 0; i < data.allotedChargingStations.length; i++) {

      let indexOfStation = data.allotedChargingStations[i];
      const data1 = await window.contract.methods.getChargingStationDetails(indexOfStation).call();
        const obj = {};

        if (!Object.keys(obj).length) {
            Object.assign(obj, { station_name: data1.station_name, distance_to_station: data1.remaining_distance,price_per_unit : data1.price_per_unit,waiting_cars: data1.count_of_waiting_cars,fast_charging_support : data1.fast_charging, other_company_station : data1.other_company_station, ratings :data1.rating});
        }
        dataLat.push(obj);
    }

    setDataNew2(dataLat);
    setFS(false);
    setLC(false);
    setShowList(true);
  }

  return (
    <div>
         <div className="ui container center aligned">
```



```
            <h2 className="ui green header">Electric Charging Station
Finder</h2>
          </div><br />
          <form>
              <div className="ui two column centered grid">

                <div className="eight column centered row">
                  <div className="column"><h3>Remaining Range</h3></div>
                  <div className="column"><Select placeholder='Remaining
Range' options={rangeOptions}
                    onChange={handleChangeRange} /></div>
                </div>

                <div className="eight column centered row">
                  <div className="column"> <h3>Preference</h3></div>
                  <div className="column"><Select placeholder='Distance or
Cost' options={distanceOrCostoptions}
                    onChange={handleChangePreference} /></div>
                </div>

                <div className="eight column centered row">
                  <div className="column">
                    <div className="ui checkbox">
                      <input type="checkbox"
                        name="example" checked={isFS} onChange={e =>
setFS(e.target.checked)} />
                      <label>Fast charging</label></div></div>
                  <div className="column">
                    <div className="ui checkbox">
                      <input type="checkbox"
                        name="example" checked={isLC} onChange={e =>
setLC(e.target.checked)} />
                      <label>Other Brand Charging Station</label>
                    </div>
                  </div>
                </div>

                <div className="sixteen column centered row">
                  <div className="column">
```



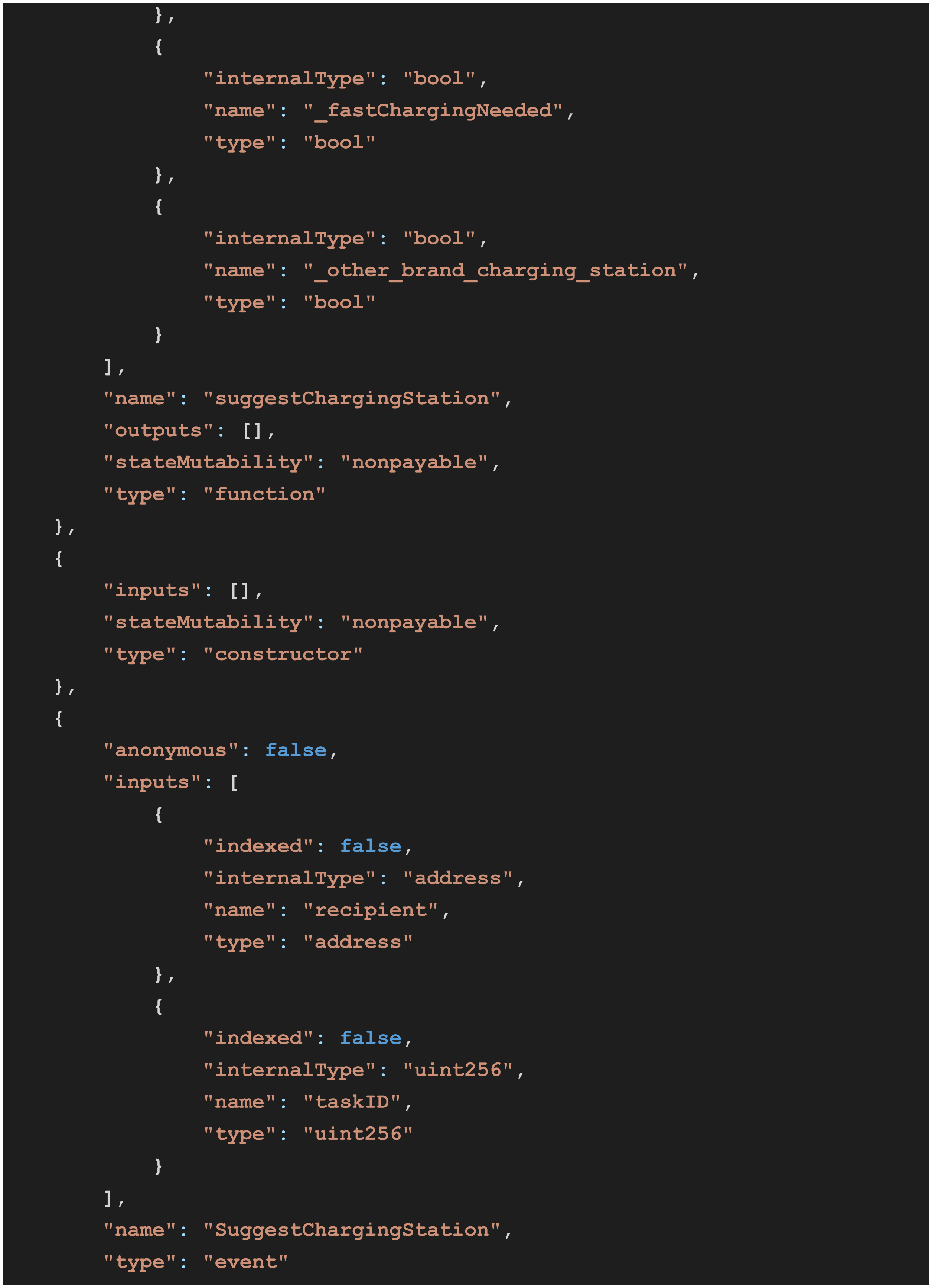
```
                    <div><button className="ui primary button"
onClick={suggestEVStation}>Find</button></div>
                  </div>
                </div>

                {showList ?
                (
                  <div className="four column centered row">
                  <div className='column centered aligned'>Charging
Stations Sumary</div> <br />
                    <div className="column">
                      <div className="wrapper">
                        <div className="accordion">
                          {dataNew2.map((item,i) => (
                            <div className="item">
                              <div className="titleStation" onClick={() =>
toggle(i)}>
                                <h2>{item.station_name}</h2>
                                <span>{selected == i ? '-' : '+'}</span>
                                </div>
                              <div className={selected == i ? 'content
show' : 'content'}>
                                Distance to charging Station :
{item.distance_to_station} Km<br />
                                Price per unit : {item.price_per_unit}
Rs<br />
                                Number of cars already in Queue :
{item.waiting_cars} <br />
                                Fast Charging Supported :
{item.fast_charging_support ? "Fast Charging Supported" : "No Fast
Charging"} <br />
                                TATA Ev charging Staion :
{item.other_company_station ? "Yes" : "No"}<br />
                                Rating : {item.ratings} out of 5<br/>
                                </div>
                            </div>
                          ))}
                        </div>
                      </div>
                    </div>
                  </div>
                )
                : null}
              </div>
            </form>
    </div>
  );
}

export default App;
```

⇒index.css

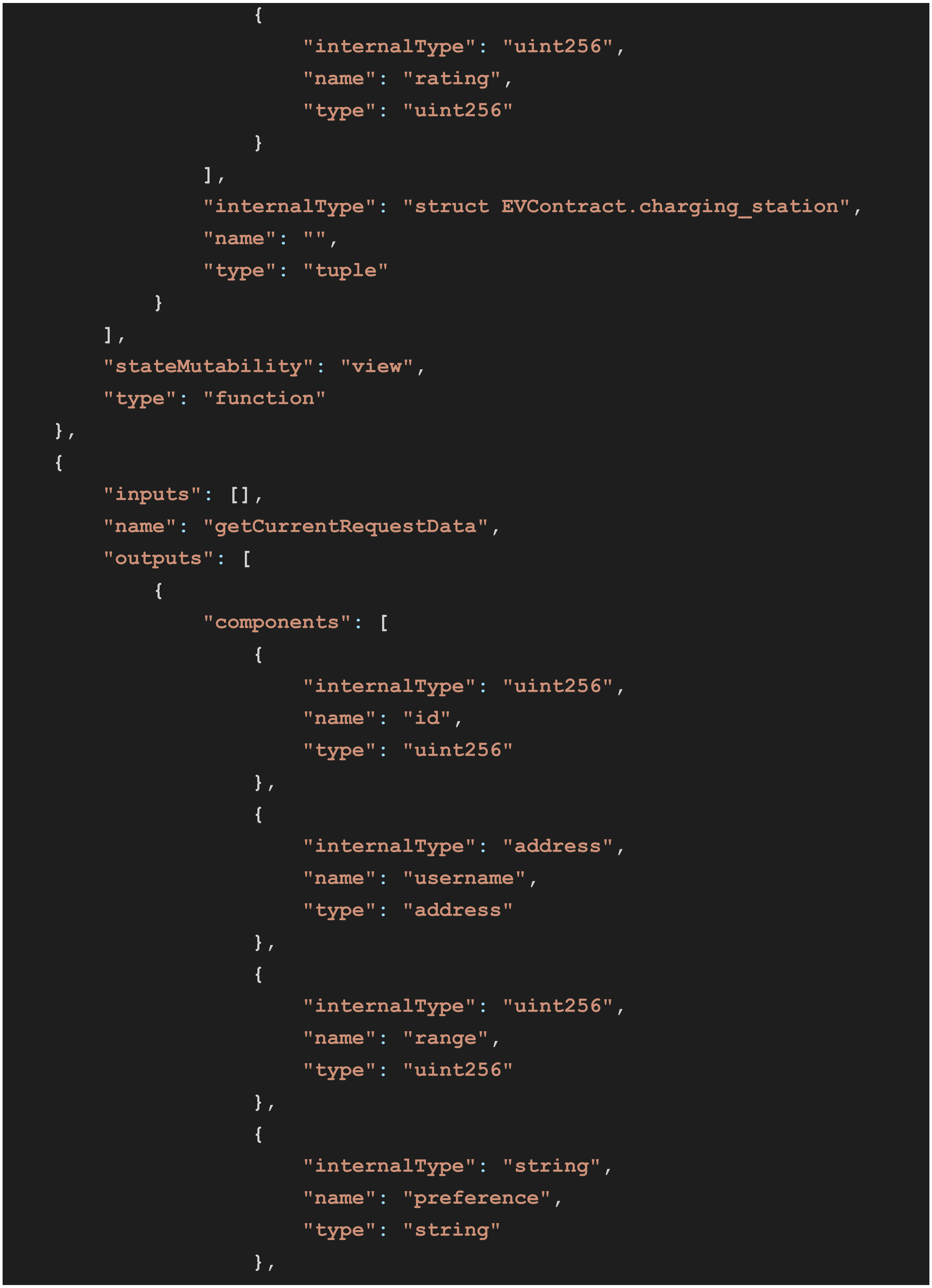
```
body {
  margin: 0;
  font-family: -apple-system, BlinkMacSystemFont, 'Segoe UI', 'Roboto',
'Oxygen',
    'Ubuntu', 'Cantarell', 'Fira Sans', 'Droid Sans', 'Helvetica Neue',
    sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
}

code {
  font-family: source-code-pro, Menlo, Monaco, Consolas, 'Courier New',
    monospace;
}
.item {
  background : #f0ebe1;
  margin-bottom: 5px;
  padding: 10px 20px;
}

.titleStation {
  color : #664e20;
  display: flex;
  justify-content : space-between;
  align-items: center;
  cursor: pointer;

}
.content {
  color : #8a715d;
  max-height: 0;
  overflow: hidden;
  transition: all 0.5s cubic-bezier(0,1,0,);
}
.content.show {
  height: auto;
  max-height: 9999px;
  transition: all 0.5s cubic-bezier(1,0,1,0);
}
```

⇒configEVContract.js

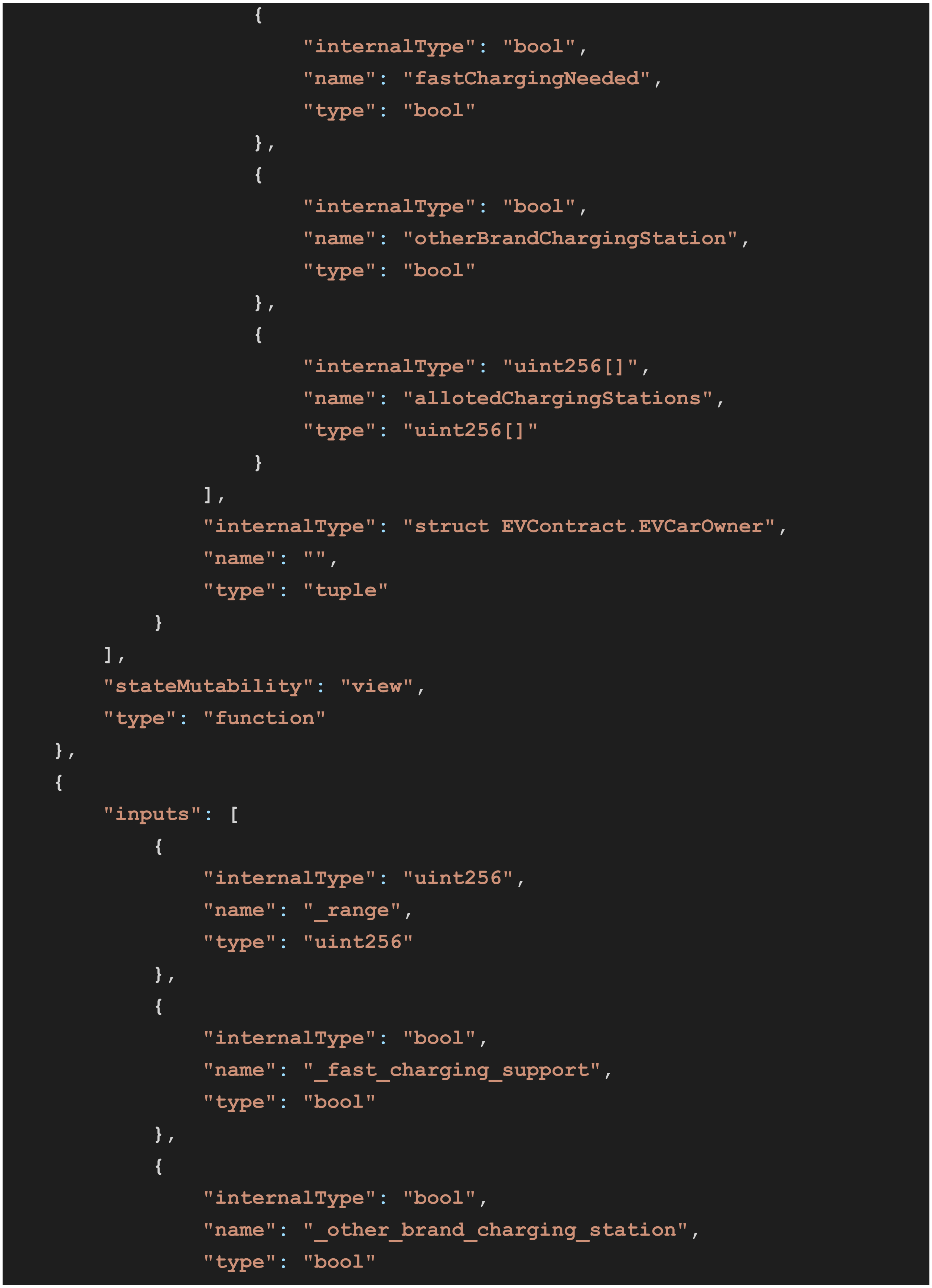
```
export const CONTACT_ADDRESS =
'0xe37a4eA16E9952066B181BEC60B4756309AeFc87';

export const CONTACT_ABI = [
    {
        "inputs": [
            {
                "internalType": "uint256",
                "name": "station_id",
                "type": "uint256"
            },
            {
                "internalType": "string",
                "name": "station_name",
                "type": "string"
            },
            {
                "internalType": "uint256",
                "name": "remaining_distance",
                "type": "uint256"
            },
            {
                "internalType": "uint256",
                "name": "price_per_unit",
                "type": "uint256"
            },
            {
                "internalType": "uint256",
                "name": "count_of_waiting_cars",
                "type": "uint256"
            },
            {
                "internalType": "bool",
                "name": "fast_charging",
                "type": "bool"
            },
            {
                "internalType": "bool",
                "name": "other_charging_station",
                "type": "bool"
            },
            {
                "internalType": "uint256",
                "name": "rating",
                "type": "uint256"
            }
        ],
        "name": "addChargingStation",
        "outputs": [],
        "stateMutability": "nonpayable",
        "type": "function"
    },
    {
        "inputs": [
            {
                "internalType": "uint256",
                "name": "_range",
                "type": "uint256"
            },
            {
                "internalType": "string",
                "name": "_preference",
                "type": "string"
```



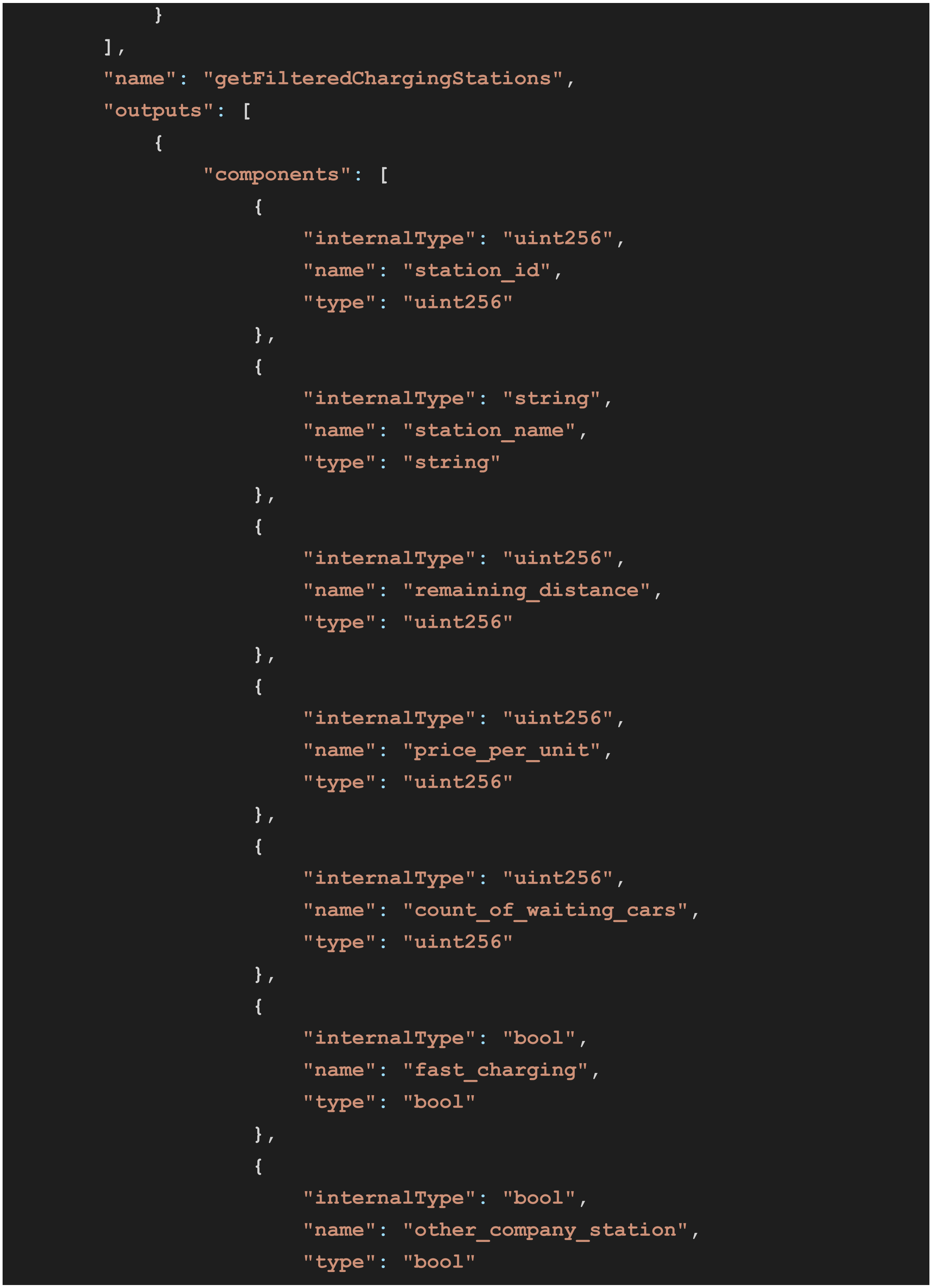
```
            },
            {
                "internalType": "bool",
                "name": "_fastChargingNeeded",
                "type": "bool"
            },
            {
                "internalType": "bool",
                "name": "_other_brand_charging_station",
                "type": "bool"
            }
        ],
        "name": "suggestChargingStation",
        "outputs": [],
        "stateMutability": "nonpayable",
        "type": "function"
    },
    {
        "inputs": [],
        "stateMutability": "nonpayable",
        "type": "constructor"
    },
    {
        "anonymous": false,
        "inputs": [
            {
                "indexed": false,
                "internalType": "address",
                "name": "recipient",
                "type": "address"
            },
            {
                "indexed": false,
                "internalType": "uint256",
                "name": "taskID",
                "type": "uint256"
            }
        ],
        "name": "SuggestChargingStation",
        "type": "event"
    },
    {
        "inputs": [],
        "name": "count",
        "outputs": [
            {
                "internalType": "uint256",
                "name": "",
                "type": "uint256"
            }
        ],
        "stateMutability": "view",
        "type": "function"
    },
    {
        "inputs": [
            {
                "internalType": "uint256",
                "name": "",
                "type": "uint256"
            }
        ],
        "name": "evOwnerRequests",
        "outputs": [
            {
                "internalType": "address",
                "name": "",
                "type": "address"
            }
        ],
        "stateMutability": "view",
        "type": "function"
    },
    {
        "inputs": [
            {
                "internalType": "uint256",
                "name": "_id",
                "type": "uint256"
            }
        ],
        "name": "getChargingStationDetails",
        "outputs": [
            {
                "components": [
                    {
                        "internalType": "uint256",
                        "name": "station_id",
                        "type": "uint256"
                    },
                    {
                        "internalType": "string",
                        "name": "station_name",
                        "type": "string"
                    },
                    {
                        "internalType": "uint256",
                        "name": "remaining_distance",
                        "type": "uint256"
                    },
                    {
                        "internalType": "uint256",
                        "name": "price_per_unit",
                        "type": "uint256"
                    },
                    {
                        "internalType": "uint256",
                        "name": "count_of_waiting_cars",
                        "type": "uint256"
                    },
                    {
                        "internalType": "bool",
                        "name": "fast_charging",
                        "type": "bool"
                    },
                    {
                        "internalType": "bool",
                        "name": "other_company_station",
                        "type": "bool"
                    },
```



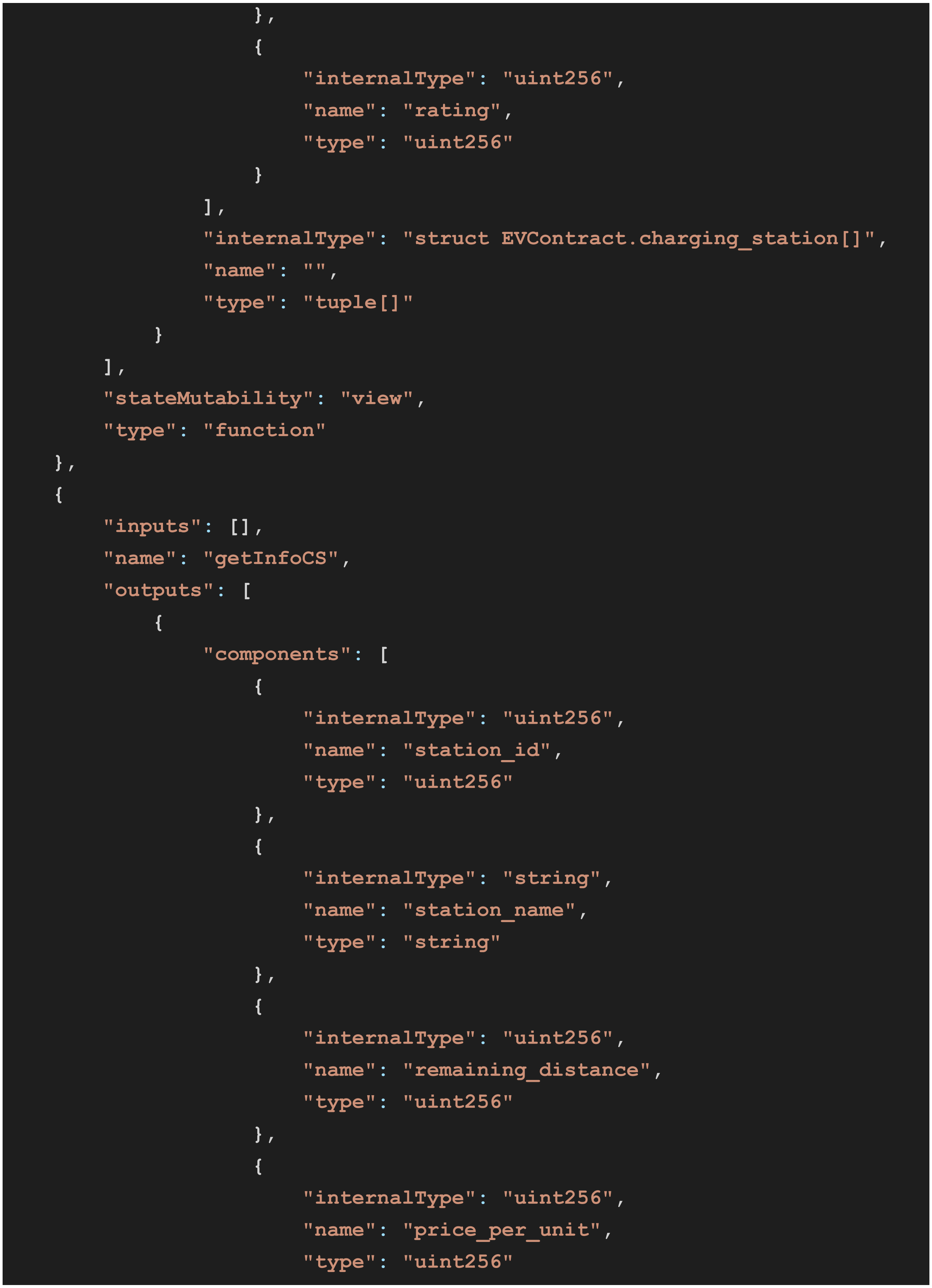
```
                    {
                        "internalType": "uint256",
                        "name": "rating",
                        "type": "uint256"
                    }
                ],
                "internalType": "struct EVContract.charging_station",
                "name": "",
                "type": "tuple"
            }
        ],
        "stateMutability": "view",
        "type": "function"
    },
    {
        "inputs": [],
        "name": "getCurrentRequestData",
        "outputs": [
            {
                "components": [
                    {
                        "internalType": "uint256",
                        "name": "id",
                        "type": "uint256"
                    },
                    {
                        "internalType": "address",
                        "name": "username",
                        "type": "address"
                    },
                    {
                        "internalType": "uint256",
                        "name": "range",
                        "type": "uint256"
                    },
                    {
                        "internalType": "string",
                        "name": "preference",
                        "type": "string"
                    },
```



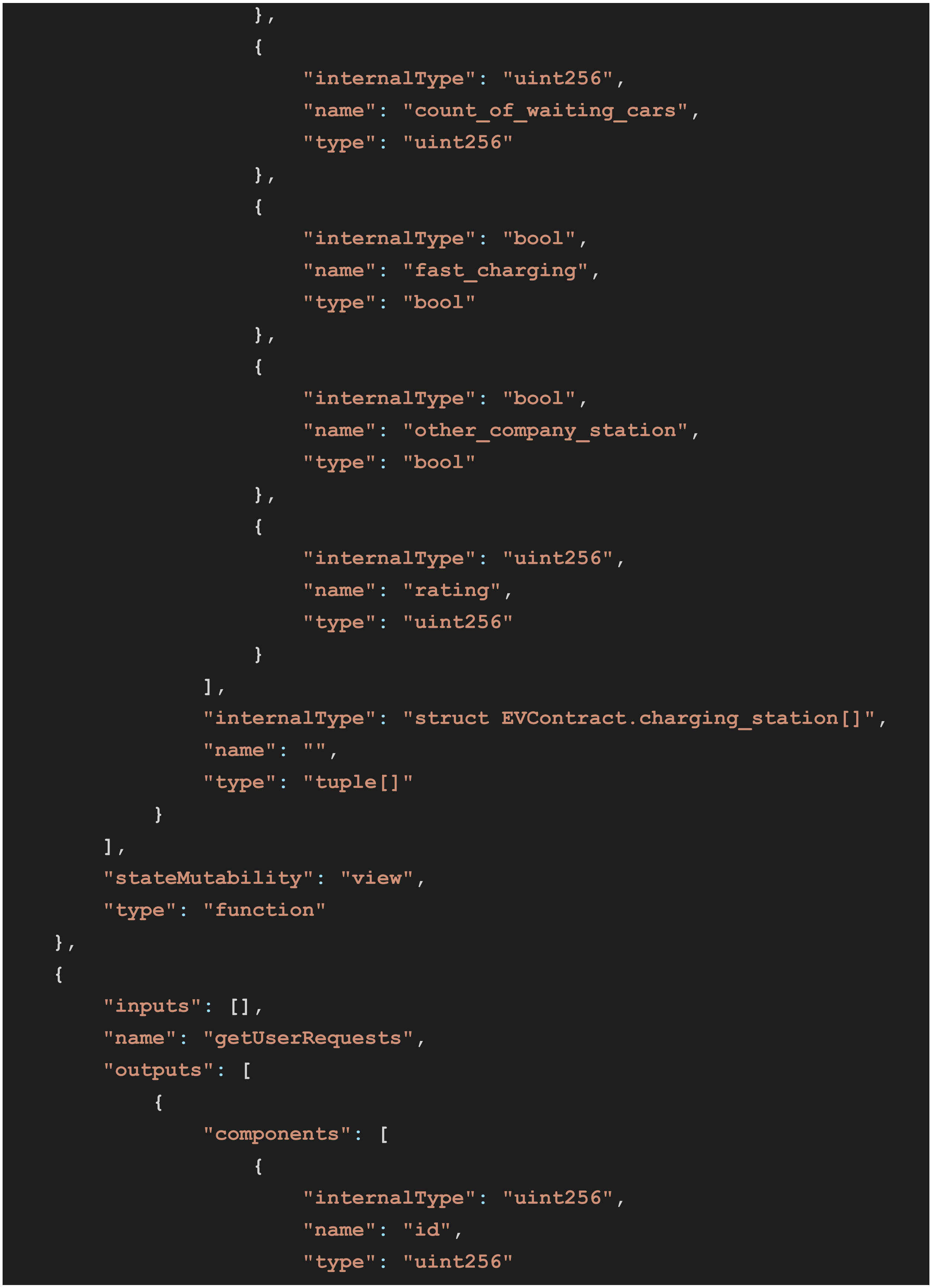
```
                {
                    "internalType": "bool",
                    "name": "fastChargingNeeded",
                    "type": "bool"
                },
                {
                    "internalType": "bool",
                    "name": "otherBrandChargingStation",
                    "type": "bool"
                },
                {
                    "internalType": "uint256[]",
                    "name": "allotedChargingStations",
                    "type": "uint256[]"
                }
            ],
            "internalType": "struct EVContract.EVCarOwner",
            "name": "",
            "type": "tuple"
        }
    ],
    "stateMutability": "view",
    "type": "function"
},
{
    "inputs": [
        {
            "internalType": "uint256",
            "name": "_range",
            "type": "uint256"
        },
        {
            "internalType": "bool",
            "name": "_fast_charging_support",
            "type": "bool"
        },
        {
            "internalType": "bool",
            "name": "_other_brand_charging_station",
            "type": "bool"
```



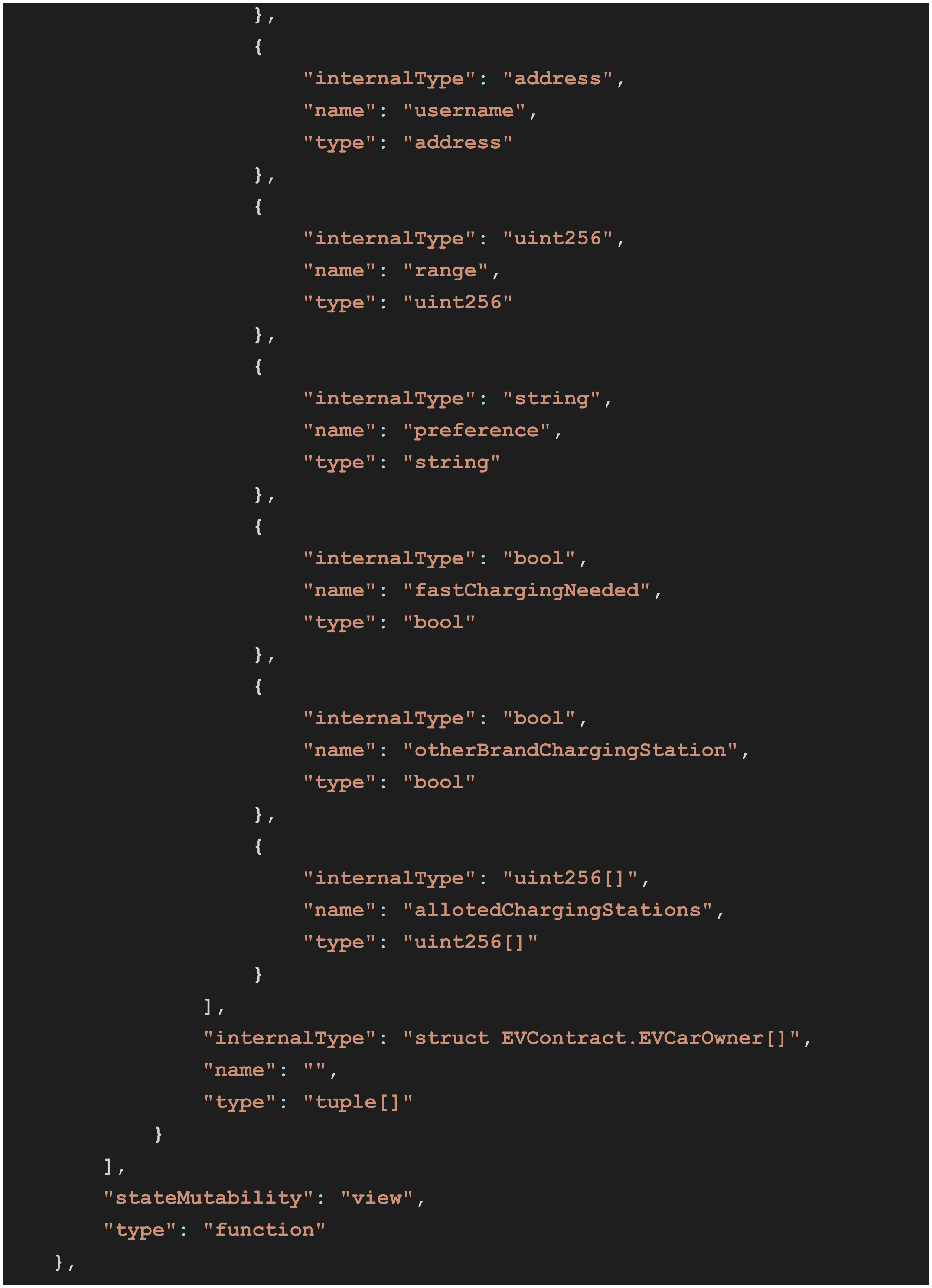
```
            }
        ],
        "name": "getFilteredChargingStations",
        "outputs": [
            {
                "components": [
                    {
                        "internalType": "uint256",
                        "name": "station_id",
                        "type": "uint256"
                    },
                    {
                        "internalType": "string",
                        "name": "station_name",
                        "type": "string"
                    },
                    {
                        "internalType": "uint256",
                        "name": "remaining_distance",
                        "type": "uint256"
                    },
                    {
                        "internalType": "uint256",
                        "name": "price_per_unit",
                        "type": "uint256"
                    },
                    {
                        "internalType": "uint256",
                        "name": "count_of_waiting_cars",
                        "type": "uint256"
                    },
                    {
                        "internalType": "bool",
                        "name": "fast_charging",
                        "type": "bool"
                    },
                    {
                        "internalType": "bool",
                        "name": "other_company_station",
                        "type": "bool"
```



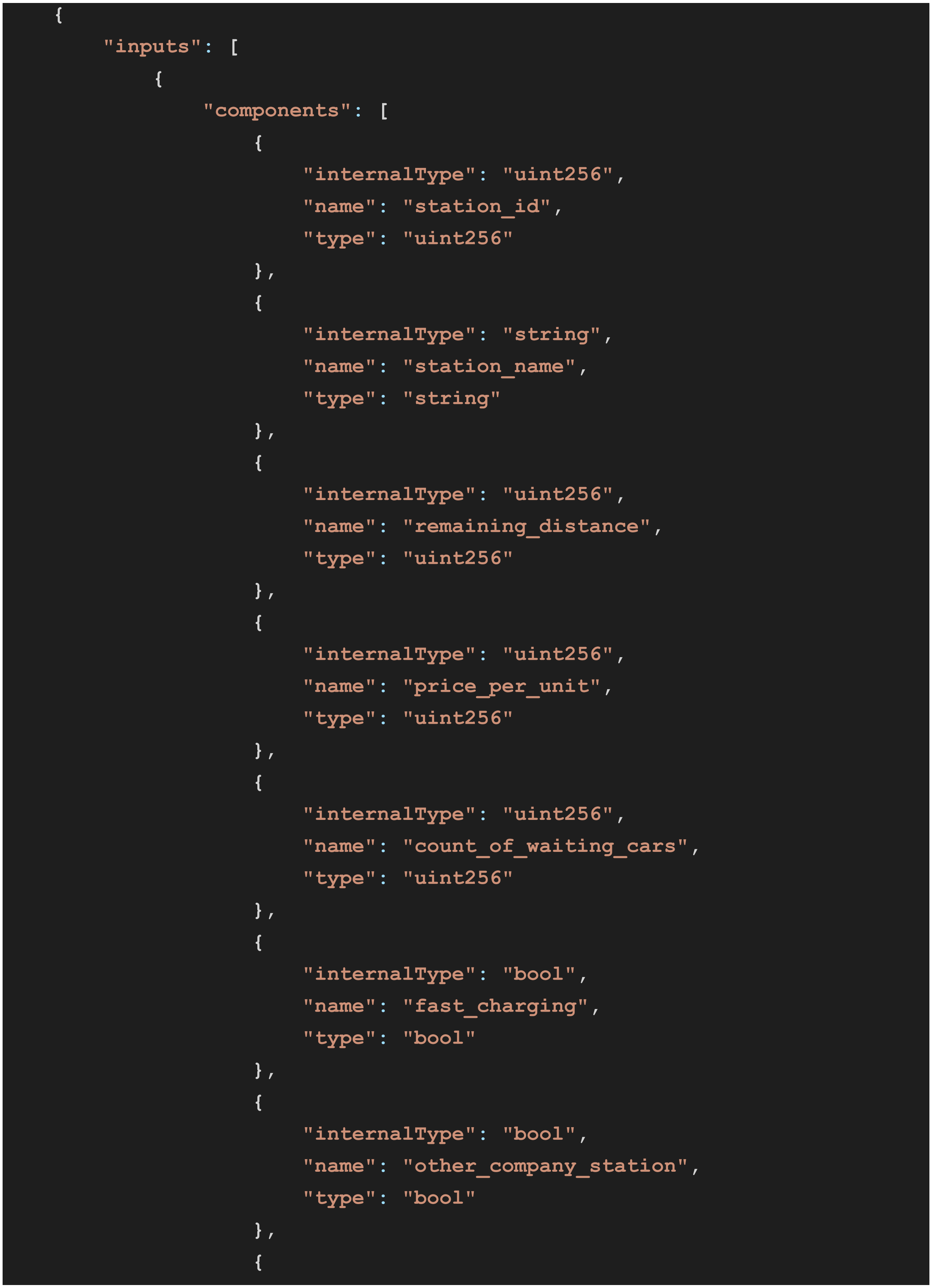
```
                    },
                    {
                        "internalType": "uint256",
                        "name": "rating",
                        "type": "uint256"
                    }
                ],
                "internalType": "struct EVContract.charging_station[]",
                "name": "",
                "type": "tuple[]"
            }
        ],
        "stateMutability": "view",
        "type": "function"
    },
    {
        "inputs": [],
        "name": "getInfoCS",
        "outputs": [
            {
                "components": [
                    {
                        "internalType": "uint256",
                        "name": "station_id",
                        "type": "uint256"
                    },
                    {
                        "internalType": "string",
                        "name": "station_name",
                        "type": "string"
                    },
                    {
                        "internalType": "uint256",
                        "name": "remaining_distance",
                        "type": "uint256"
                    },
                    {
                        "internalType": "uint256",
                        "name": "price_per_unit",
                        "type": "uint256"
```



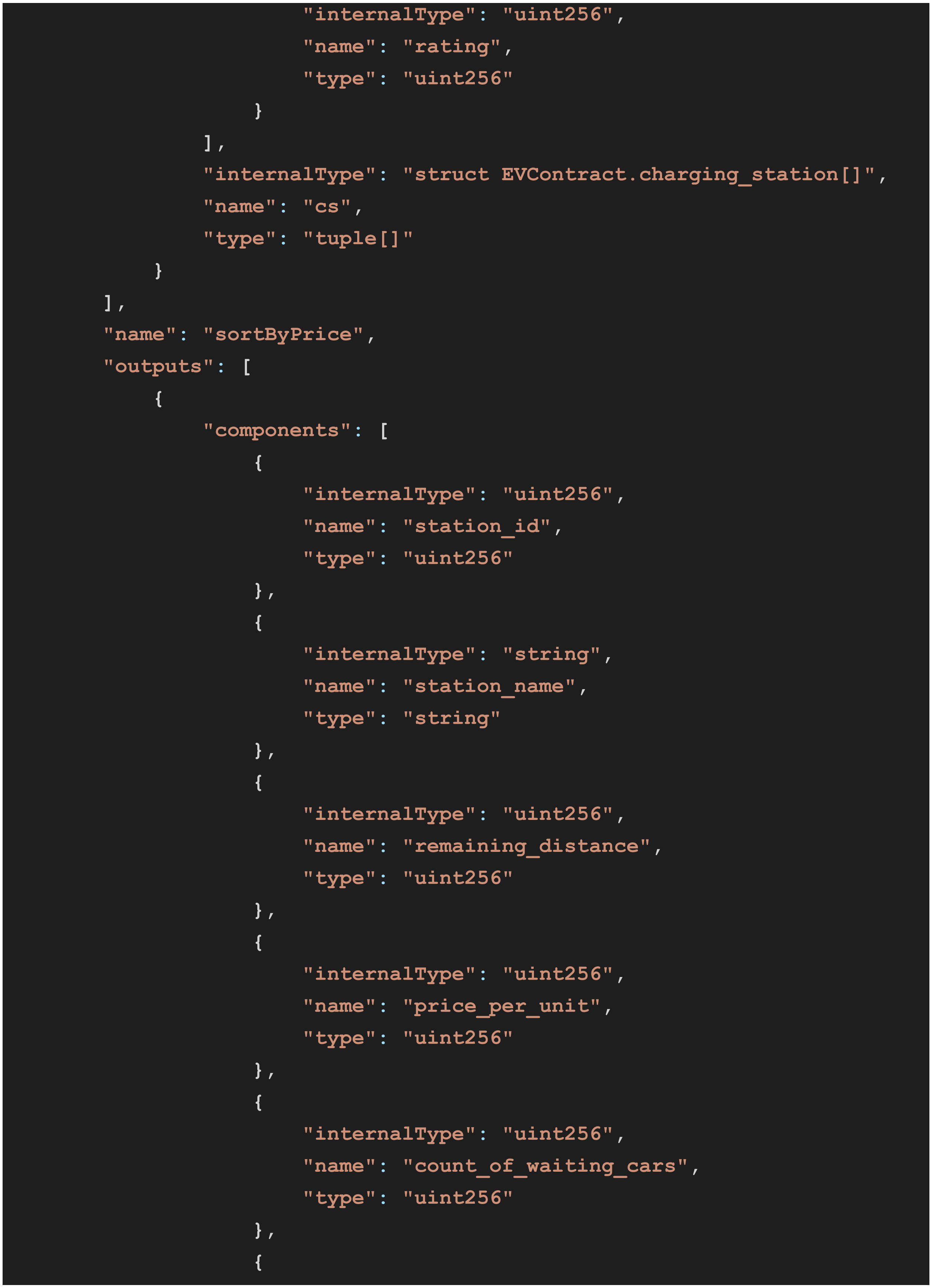
```
                    },
                    {
                        "internalType": "uint256",
                        "name": "count_of_waiting_cars",
                        "type": "uint256"
                    },
                    {
                        "internalType": "bool",
                        "name": "fast_charging",
                        "type": "bool"
                    },
                    {
                        "internalType": "bool",
                        "name": "other_company_station",
                        "type": "bool"
                    },
                    {
                        "internalType": "uint256",
                        "name": "rating",
                        "type": "uint256"
                    }
                ],
                "internalType": "struct EVContract.charging_station[]",
                "name": "",
                "type": "tuple[]"
            }
        ],
        "stateMutability": "view",
        "type": "function"
    },
    {
        "inputs": [],
        "name": "getUserRequests",
        "outputs": [
            {
                "components": [
                    {
                        "internalType": "uint256",
                        "name": "id",
                        "type": "uint256"
```



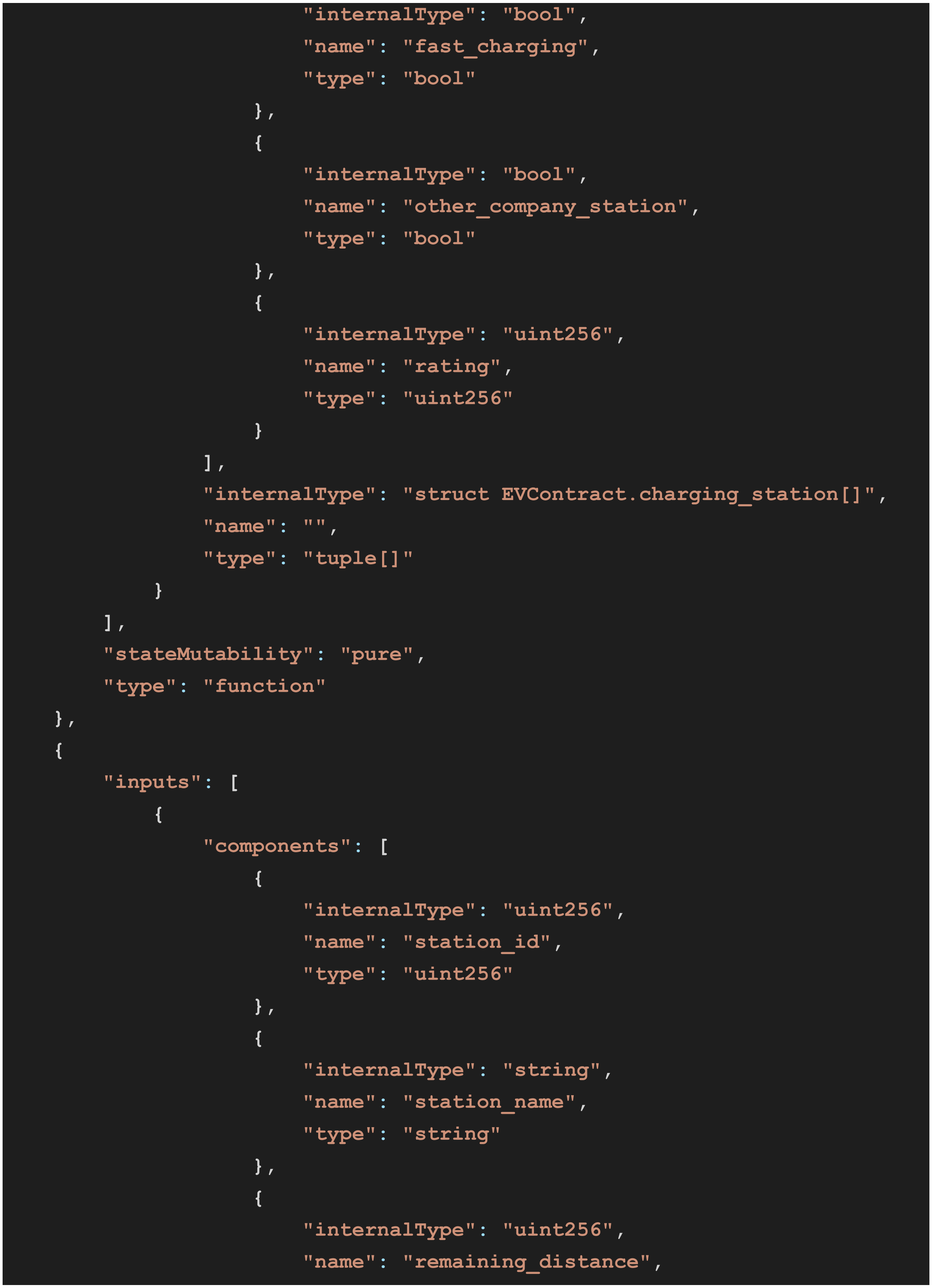
```
                    },
                    {
                        "internalType": "address",
                        "name": "username",
                        "type": "address"
                    },
                    {
                        "internalType": "uint256",
                        "name": "range",
                        "type": "uint256"
                    },
                    {
                        "internalType": "string",
                        "name": "preference",
                        "type": "string"
                    },
                    {
                        "internalType": "bool",
                        "name": "fastChargingNeeded",
                        "type": "bool"
                    },
                    {
                        "internalType": "bool",
                        "name": "otherBrandChargingStation",
                        "type": "bool"
                    },
                    {
                        "internalType": "uint256[]",
                        "name": "allotedChargingStations",
                        "type": "uint256[]"
                    }
                ],
                "internalType": "struct EVContract.EVCarOwner[]",
                "name": "",
                "type": "tuple[]"
            }
        ],
        "stateMutability": "view",
        "type": "function"
    },
```



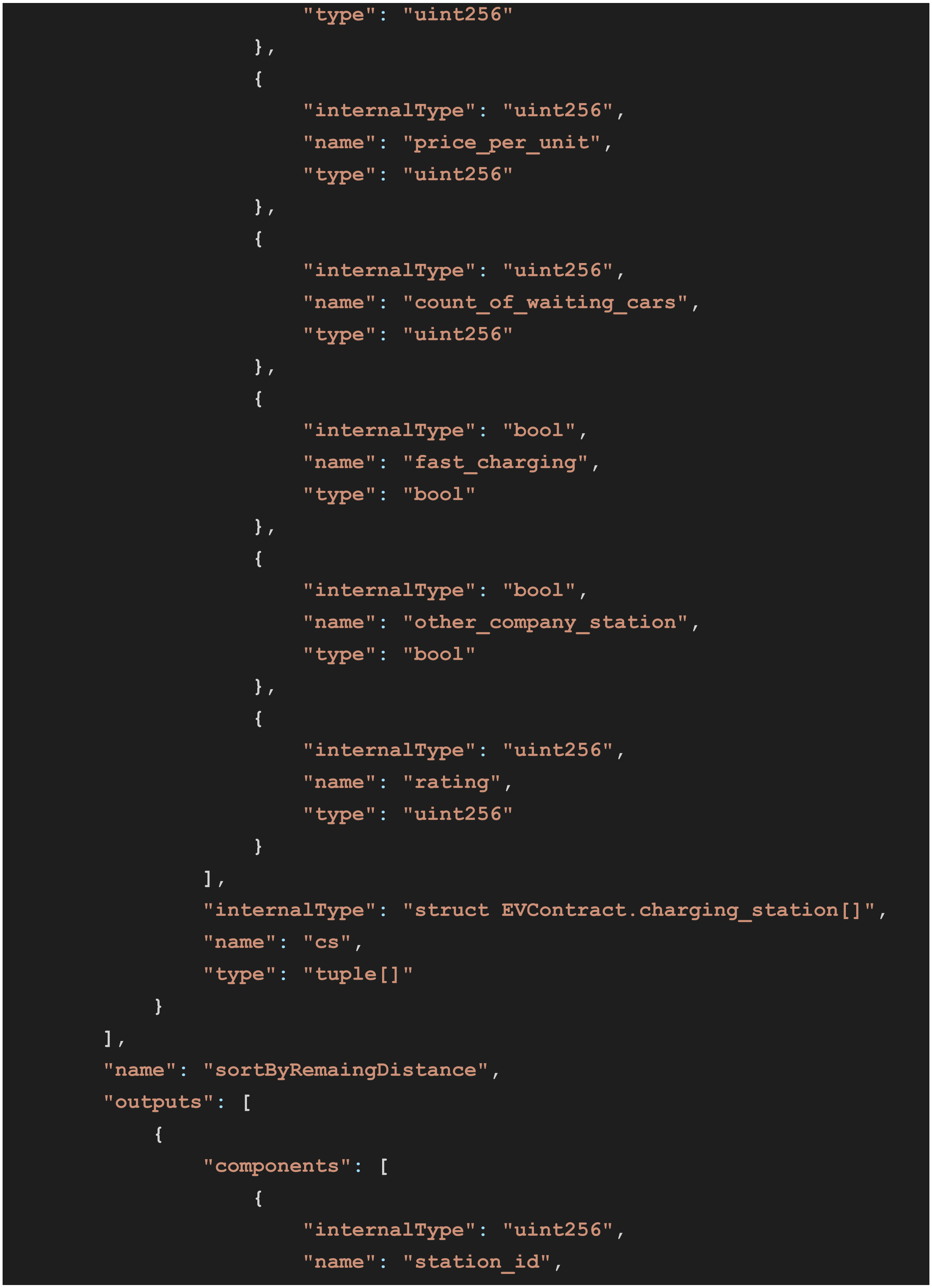
```
    {
        "inputs": [
            {
                "components": [
                    {
                        "internalType": "uint256",
                        "name": "station_id",
                        "type": "uint256"
                    },
                    {
                        "internalType": "string",
                        "name": "station_name",
                        "type": "string"
                    },
                    {
                        "internalType": "uint256",
                        "name": "remaining_distance",
                        "type": "uint256"
                    },
                    {
                        "internalType": "uint256",
                        "name": "price_per_unit",
                        "type": "uint256"
                    },
                    {
                        "internalType": "uint256",
                        "name": "count_of_waiting_cars",
                        "type": "uint256"
                    },
                    {
                        "internalType": "bool",
                        "name": "fast_charging",
                        "type": "bool"
                    },
                    {
                        "internalType": "bool",
                        "name": "other_company_station",
                        "type": "bool"
                    },
                    {
```



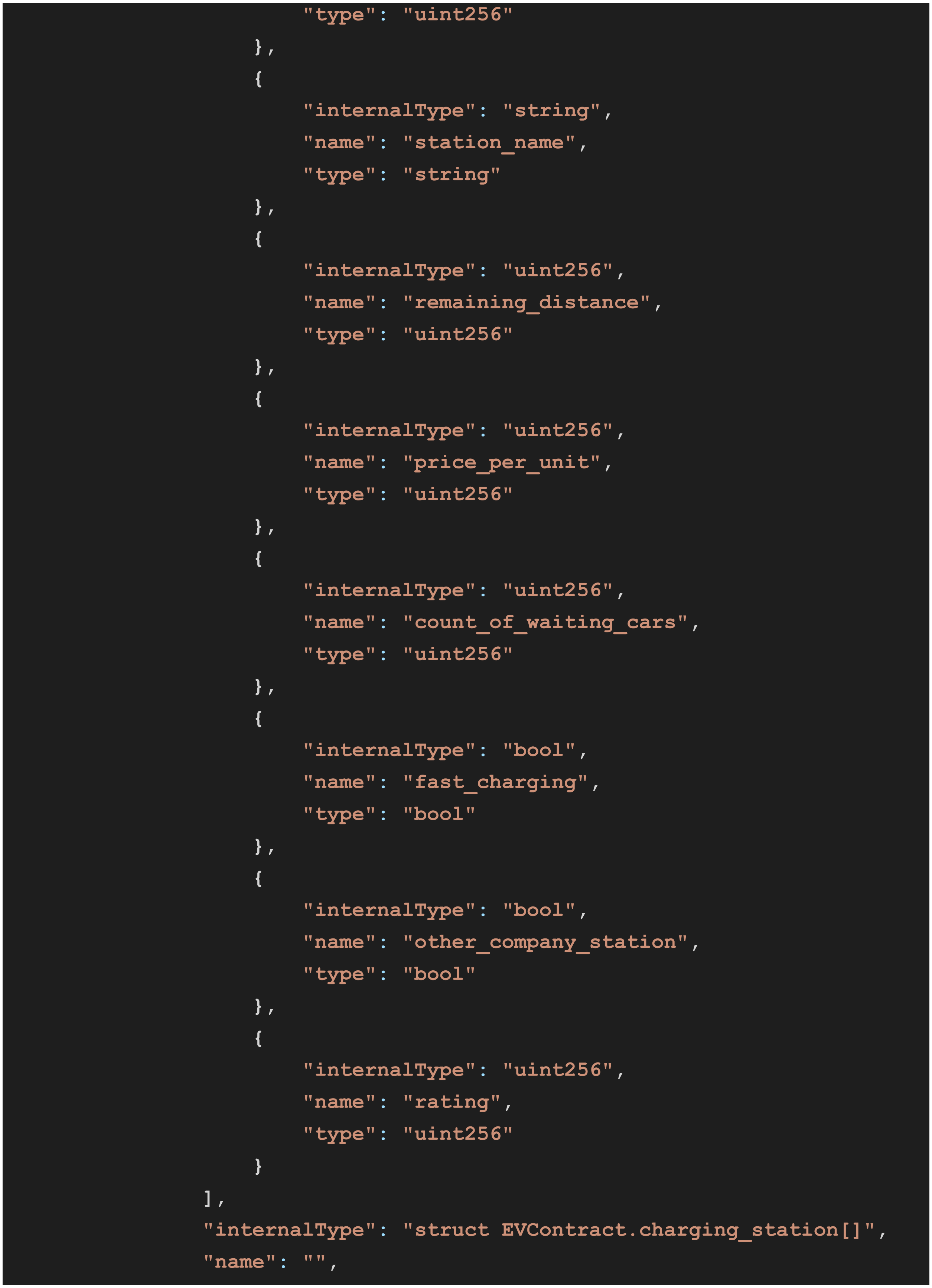
```
                        "internalType": "uint256",
                        "name": "rating",
                        "type": "uint256"
                    }
                ],
                "internalType": "struct EVContract.charging_station[]",
                "name": "cs",
                "type": "tuple[]"
            }
        ],
        "name": "sortByPrice",
        "outputs": [
            {
                "components": [
                    {
                        "internalType": "uint256",
                        "name": "station_id",
                        "type": "uint256"
                    },
                    {
                        "internalType": "string",
                        "name": "station_name",
                        "type": "string"
                    },
                    {
                        "internalType": "uint256",
                        "name": "remaining_distance",
                        "type": "uint256"
                    },
                    {
                        "internalType": "uint256",
                        "name": "price_per_unit",
                        "type": "uint256"
                    },
                    {
                        "internalType": "uint256",
                        "name": "count_of_waiting_cars",
                        "type": "uint256"
                    },
                    {
```



```
                        "internalType": "bool",
                        "name": "fast_charging",
                        "type": "bool"
                    },
                    {
                        "internalType": "bool",
                        "name": "other_company_station",
                        "type": "bool"
                    },
                    {
                        "internalType": "uint256",
                        "name": "rating",
                        "type": "uint256"
                    }
                ],
                "internalType": "struct EVContract.charging_station[]",
                "name": "",
                "type": "tuple[]"
            }
        ],
        "stateMutability": "pure",
        "type": "function"
    },
    {
        "inputs": [
            {
                "components": [
                    {
                        "internalType": "uint256",
                        "name": "station_id",
                        "type": "uint256"
                    },
                    {
                        "internalType": "string",
                        "name": "station_name",
                        "type": "string"
                    },
                    {
                        "internalType": "uint256",
                        "name": "remaining_distance",
```



```
                        "type": "uint256"
                    },
                    {
                        "internalType": "uint256",
                        "name": "price_per_unit",
                        "type": "uint256"
                    },
                    {
                        "internalType": "uint256",
                        "name": "count_of_waiting_cars",
                        "type": "uint256"
                    },
                    {
                        "internalType": "bool",
                        "name": "fast_charging",
                        "type": "bool"
                    },
                    {
                        "internalType": "bool",
                        "name": "other_company_station",
                        "type": "bool"
                    },
                    {
                        "internalType": "uint256",
                        "name": "rating",
                        "type": "uint256"
                    }
                ],
                "internalType": "struct EVContract.charging_station[]",
                "name": "cs",
                "type": "tuple[]"
            }
        ],
        "name": "sortByRemaingDistance",
        "outputs": [
            {
                "components": [
                    {
                        "internalType": "uint256",
                        "name": "station_id",
```



```
                        "type": "uint256"
                    },
                    {
                        "internalType": "string",
                        "name": "station_name",
                        "type": "string"
                    },
                    {
                        "internalType": "uint256",
                        "name": "remaining_distance",
                        "type": "uint256"
                    },
                    {
                        "internalType": "uint256",
                        "name": "price_per_unit",
                        "type": "uint256"
                    },
                    {
                        "internalType": "uint256",
                        "name": "count_of_waiting_cars",
                        "type": "uint256"
                    },
                    {
                        "internalType": "bool",
                        "name": "fast_charging",
                        "type": "bool"
                    },
                    {
                        "internalType": "bool",
                        "name": "other_company_station",
                        "type": "bool"
                    },
                    {
                        "internalType": "uint256",
                        "name": "rating",
                        "type": "uint256"
                    }
                ],
                "internalType": "struct EVContract.charging_station[]",
                "name": "",
                "type": "tuple[]"
            }
        ],
        "stateMutability": "pure",
        "type": "function"
    }
]
```