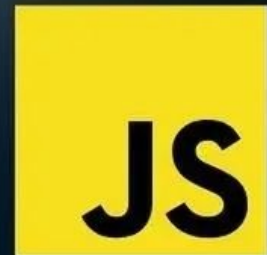




@pyplane_code

Battery API





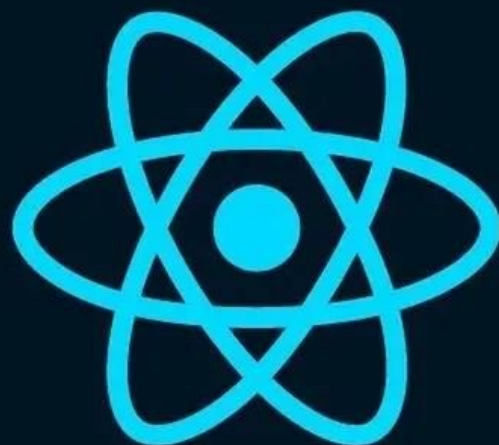
Introduction

Get battery information with just a few lines of code. You can check the battery status in your web app with this code:

```
navigator.getBattery()  
.then((battery) => {  
  console.log("Battery level: " + battery.level);  
  console.log("Charging? " + battery.charging);  
});
```

In this tutorial we'll do something a little bit more interesting using React

⏪ Swipe





Let's get started

Let's create battery status check, like the one in the image below. We'll be using React and tailwind css. To get started create a new react app using vite and add tailwind css

Battery Status

Battery level: 78%

Charging: No





Battery status jsx - 1

We will create a separate component called BatteryStatus and include it in our code. It uses the useState and useEffect hooks from React to manage the battery status state and listen for changes.

```
import { useEffect, useState } from 'react';

const BatteryStatus = () => {
  const [batteryLevel, setBatteryLevel] = useState(null);
  const [isCharging, setIsCharging] = useState(null);

  useEffect(() => {
    navigator.getBattery().then(battery => {
      const updateBatteryStatus = () => {
        setBatteryLevel(battery.level);
        setIsCharging(battery.charging);
      };

      updateBatteryStatus();
      battery.addEventListener('levelchange', updateBatteryStatus);
      battery.addEventListener('chargingchange', updateBatteryStatus);

      return () => {
        battery.removeEventListener('levelchange', updateBatteryStatus);
        battery.removeEventListener('chargingchange', updateBatteryStatus);
      };
    });
  }, []);
}
```



Battery status jsx - 2

The navigator.getBattery() method is called inside the useEffect hook to retrieve the battery information. The returned promise is used to update the battery status state variables (batteryLevel and isCharging). Event listeners are added to the battery object to update the state whenever there is a change in battery level or charging status. Rest of the code:

```
return (
  <div className='container mx-auto mt-12'>
    <div className='text-3xl font-semibold'>Battery Status</div>
    {batteryLevel !== null && (
      <>
        <div>Battery level: {batteryLevel * 100}%</div>
        <div>Charging: {isCharging ? 'Yes' : 'No'}</div>
        <div className="w-24 h-12 bg-gray-200 border-2
          border-gray-400 relative">
          <div className={`h-full flex justify-center items-center
            bg-green-400 w-[${batteryLevel * 100}%]`} >{batteryLevel *
            100}</div>
        </div>
      </>
    )}
  </div>
);

export default BatteryStatus;
```

Did You Find it Useful?



Alamin CodePapa
@CodePapa360

Follow for more

Like | Comment | Repost

