El error que estás viendo, "Access to script has been blocked by CORS policy", indica un problema de política de mismo origen (CORS). Esta política de seguridad del navegador restringe cómo los scripts en una página web pueden interactuar con recursos en un dominio diferente al de la propia página.

La parte específica del error, "from origin 'null'", indica que estás intentando cargar un script desde un origen nulo, es decir, directamente desde el sistema de archivos local (`file://`). Los navegadores aplican políticas de CORS incluso cuando estás trabajando localmente con archivos HTML y JavaScript.

El navegador bloquea esta solicitud porque tu página web (que probablemente estás abriendo directamente desde el sistema de archivos local) está intentando cargar un script desde el sistema de archivos local que no es parte de la misma "origen" (mismo dominio, protocolo y puerto) que la página web.

Para solucionar este problema y poder cargar el archivo JavaScript correctamente, debes servir tu página web a través de un servidor web local. Esto implica configurar un servidor web local en tu computadora y acceder a tu página a través de `http://localhost` en lugar de `file:///`.

Aquí hay una breve guía para configurar un servidor web local:

1. \*\*Instala Node.js (si aún no lo has hecho):\*\*

Puedes descargar e instalar Node.js desde el sitio web oficial: [Node.js](https://nodejs.org/).

2. \*\*Crea un servidor web local con Node.js:\*\*

Puedes usar módulos como Express.js para configurar fácilmente un servidor web local en Node.js.

Aquí hay un ejemplo básico de cómo puedes configurar un servidor web local con Express.js:

```javascript

const express = require('express');

const app = express();

const PORT = 3000;

app.use(express.static(\_\_dirname));

app.listen(PORT, () => {

console.log(`Servidor web local escuchando en el puerto ${PORT}!`);

});

```

Guarda este código en un archivo, por ejemplo, `server.js`, y ejecútalo con Node.js desde la línea de comandos utilizando el comando `node server.js`.

3. \*\*Accede a tu página web a través del servidor local:\*\*

Una vez que hayas iniciado el servidor local, puedes acceder a tu página web en un navegador web escribiendo `http://localhost:3000` en la barra de direcciones.

Al servir tu página web a través de un servidor web local, tu página web y los recursos asociados se servirán desde un origen con el protocolo HTTP, lo que permitirá la carga de recursos de manera adecuada sin que ocurran errores de CORS.

**2da parte**

**El error que estás viendo indica que Node.js no puede encontrar el módulo 'express' que estás intentando requerir en tu archivo `server.js`. Este tipo de error ocurre cuando Node.js no puede localizar el paquete 'express' en tu proyecto.**

**Aquí hay algunas posibles razones y soluciones para este problema:**

**1. \*\*Express no está instalado:\*\***

**El error indica que Node.js no puede encontrar el módulo 'express'. Esto generalmente significa que Express no está instalado en tu proyecto. Para solucionar esto, necesitas instalar Express utilizando npm (Node Package Manager).**

**Abre una terminal en el directorio de tu proyecto y ejecuta el siguiente comando para instalar Express:**

**```**

**npm install express**

**```**

**Esto instalará Express y todas sus dependencias en tu proyecto.**

**2. \*\*Problemas de ruta:\*\***

**Asegúrate de que estás ejecutando el comando `npm install express` en el directorio correcto, es decir, en el directorio donde se encuentra tu archivo `server.js`. Si estás en el directorio correcto y aún así tienes problemas, puede ser necesario verificar si hay problemas con los permisos de escritura o si hay algún problema con el archivo `package.json` en tu proyecto.**

**3. \*\*Revisión de dependencias:\*\***

**Si estás trabajando en un proyecto que ya tiene un archivo `package.json`, asegúrate de que Express esté listado como una dependencia en ese archivo. Si no lo está, puedes agregarlo manualmente y luego ejecutar `npm install` para instalar todas las dependencias listadas en `package.json`.**

**Una vez que hayas instalado correctamente Express, puedes intentar ejecutar tu archivo `server.js` nuevamente. Esto debería resolver el problema y permitir que Node.js encuentre y cargue el módulo 'express' correctamente.**