Preload

Defina o que a página pode pedir ao Electron

Quando o preload executa

O preload é um script associado à janela que executa antes dos scripts da página. Com contextIsolation ativo, ele roda em um contexto JavaScript separado e usa contextBridge para publicar uma interface que a página pode chamar.

Uma operação por necessidade

Para preencher a agenda, a interface precisa consultar suas reservas. Exponha listarReservas, que usa um canal fixo. O processo principal decide quais dados devolver e valida a origem do pedido e as permissões aplicáveis.

Configuração da janela

Em webPreferences, defina preload com o caminho absoluto para preload.cjs, contextIsolation e sandbox como true e nodeIntegration como false. O preload com sandbox dispõe de um conjunto limitado de módulos, incluindo os necessários no exemplo.

Preload

Exponha uma consulta de reservas

preload.cjs

```
const { contextBridge, ipcRenderer } = require("electron");

contextBridge.exposeInMainWorld("agenda", {
  listarReservas: () => ipcRenderer.invoke("agenda:listar"),
});
```

Registro no principal

Registre agenda:listar com ipcMain.handle antes de carregar a página. Para praticar, devolva uma lista fixa de reservas fictícias após verificar que o pedido veio do frame principal da janela de agenda e da página esperada. Em um app autenticado, verifique também o acesso aos dados.

Chamada no renderer

No script da página, aguarde window.agenda.listarReservas() e use o resultado para atualizar a lista. Se a promessa for rejeitada, mostre uma mensagem de falha e permita uma nova tentativa. Retorne objetos simples com os campos de exibição.

Preload

Exercício resolvido

Cenário

A janela precisa listar horários. A primeira proposta expõe uma função que aceita um nome de canal escolhido pelo renderer. Como limitar a interface à operação de consulta?

Resolução

Publique apenas agenda.listarReservas e fixe agenda:listar no preload. Registre o handler correspondente no principal. Uma chamada com argumentos extras no renderer continua usando o canal fixo e sem argumentos. A verificação de origem e de acesso aos dados fica no handler.

Verificação

Em um teste com dublês de contextBridge e ipcRenderer, capture as funções publicadas e confirme que existe apenas listarReservas. Chame a função e confira que o único pedido capturado usa agenda:listar. Na janela real, confira que o resultado fictício aparece na lista.

Referências primárias

Electron — Scripts preload

Electron — contextBridge

Preload

Mantenha a API pública da janela pequena

Exponha funções específicas

Publique uma função listarReservas com canal fixo. Assim, o contrato do renderer expressa a consulta de que a tela precisa. Valide a origem e as permissões no handler do processo principal.

Exercício adicional

A interface só precisa consultar reservas. Qual API deve receber?

Resolução

Exponha listarReservas sem um parâmetro que selecione o canal. Teste a função com um dublê e confirme que ela chama apenas o canal de consulta esperado.

Ao assistir ao vídeo

Observe como o preload faz a ponte entre a interface e as operações do Electron.

Vídeo complementar

Electron & Context Bridge - IPC Deep Dive (em inglês)