

03

## Atalhos globais na aplicação

### Transcrição

Além dos atalhos que conseguimos colocar nos Menus das aplicações também existem outros tipos de atalhos, os atalhos Globais. A diferença básica entre um atalho do Menu e um atalho global é que os atalhos do Menu só funcionam se a aplicação estiver em foco, ou seja se dermos um `Alt+Tab` para trocar de aplicação, o atalho deixa de funcionar.

Se queremos poder utilizar alguma combinação de teclas como atalho para a nossa aplicação e que ela seja ouvida mesmo se a tela estiver em foco ou não, devemos utilizar o submódulo do Electron de `Global Shortcut`.

### Criando atalhos Globais.

O primeiro passo como em todas as vezes que utilizamos um novo submódulo é importá-lo no topo de nosso `main.js`, afinal este é um submódulo do processo principal:

```
const { app, BrowserWindow, ipcMain, Tray, Menu, globalShortcut } = require('electron');
```

Para registrar um novo atalho global, utilizamos a função `register()`, que aceita como parâmetro o `accelerator` que vai acionar aquele atalho e uma função que vai ser executado quando ouvir aquela combinação de teclas específica.

Para testar este submódulo, vamos fazer com que o atalho `CmdOrCtrl+Shift+S` acione ou pare o nosso timer, mesmo que aplicação não esteja em foco. Para isto, vamos registrar este atalho na função principal da aplicação e fazer com que seja enviado a janela da aplicação um evento que informa que devemos parar ou iniciar o timer:

```
//main.js
const { app, BrowserWindow, ipcMain, Tray, Menu, globalShortcut } = require('electron');
app.on('ready', () => {
  ...
  globalShortcut.register('CmdOrCtrl+Shift+S', () => {
    mainWindow.send('atalho-iniciar-parar');
  });
  ...
});
```

Para testar se o evento está sendo mesmo recebido quando pressionamos a combinação de teclas informada, podemos exibir uma pequena mensagem no Dev Tools da janela do App quando ele receber o evento de `atalho-iniciar-parar`:

```
//renderer.js
ipcRenderer.on('atalho-iniciar-parar', () => {
  console.log('Atalho no renderer process');
});
```

Fazendo o teste, toda vez que pressionarmos o atalho `CmdOrCtrl+Shift+S` a mensagem deve ser exibida no console do Dev Tools, mesmo que a nossa aplicação não esteja em foco.

Agora para que o Timer inicie e pare toda vez que escutarmos este atalho, basta simularmos um evento de click do mouse no botão de play/stop. Faremos isto com as API de Eventos do Javascript:

```
//renderer.js

ipcRenderer.on('atalho-iniciar-parar', () => {
  console.log('Atalho no renderer process');
  let click = new MouseEvent('click');
  botaoPlay.dispatchEvent(click);
});
```

1 Agora faça o teste e veja que toda vez que o atalho é executado, o nosso timer se inicia ou se interrompe. Atalho global registrado com sucesso :) !