IntelliSense é um termo geral para vários recursos de edição de código, incluindo: conclusão de código, informações de parâmetro, informações rápidas e listas de membros. Os recursos do IntelliSense às vezes são chamados por outros nomes, como "conclusão de código", "assistente de conteúdo" e "dicas de código".

No Visual Studio Code IntelliSense é fornecido para JavaScript, TypeScript, JSON, HTML, CSS, SCSS e menos pronto para uso. O VS Code oferece suporte a conclusões baseadas em palavras para qualquer linguagem de programação, mas também pode ser configurado para ter um IntelliSense mais rico instalando uma extensão como por exemplo de idioma

Os recursos do VS Code IntelliSense são alimentados por um serviço de idioma. Um serviço de linguagem fornece conclusões de código inteligentes com base na semântica da linguagem e uma análise de seu código-fonte. Se um serviço de idiomas souber as conclusões possíveis, as sugestões do IntelliSense serão exibidas enquanto você digita. Se você continuar digitando caracteres, a lista de membros (variáveis, métodos, etc.) será filtrada para incluir apenas os membros que contêm seus caracteres digitados. Pressionar Tab ou Enter irá inserir o membro selecionado.

Você pode acionar o IntelliSense em qualquer janela do editor digitando Ctrl+Espaço ou digitando um caractere de acionamento (como o caractere de ponto ( . ) em JavaScript).

Caso seja necessário podemos desativar o IntelliSense enquanto digita. Consulte Personalizando o IntelliSense abaixo para saber como desabilitar ou personalizar os recursos do IntelliSense do VS Code.

Conforme fornecido pelo serviço de idiomas, você pode ver **informações rápidas** para cada método pressionando Ctrl+Espaço ou clicando no ícone de informações. A documentação que acompanha o método agora se expandirá ao lado. A documentação expandida permanecerá assim e será atualizada conforme você navega na lista. Você pode fechá-lo pressionando Ctrl+Espaço novamente ou clicando no ícone de fechamento.

A extensão VS Code Go

A extensão VS Code Go fornece suporte de linguagem avançada para a [linguagem de programação Go](https://golang.org/) .

A [depuração de anexação remota](https://github.com/golang/vscode-go/wiki/debugging#connecting-to-headless-delve-with-target-specified-at-server-start-up) agora está disponível por meio da implementação DAP nativa do Delve mais recente. Ele aprimora a depuração remota com eles [recursos de depuração](https://github.com/golang/vscode-go/wiki/debugging) que já estão em uso para depuração local. Agora é o padrão com a compilação Go Nightly da extensão e se tornará o padrão para as versões estáveis ​​em meados de 2022. Recomendamos alternar suas configurações de anexação remota launch.jsonpara usar "debugAdapter":"dlv-dap"agora para verificar se isso funciona para você. Registre [um novo problema](https://github.com/golang/vscode-go/issues/new/choose) se encontrar algum problema.

Assista à Debugging Treasure Hunt da GopherCon 2021 para uma versão divertida de uma demonstração de depuração com VS Code Go e Delve DAP.

**Começo rápido**

Quer você seja novo em Go ou um desenvolvedor Go experiente, esperamos que esta extensão atenda às suas necessidades e aprimore sua experiência de desenvolvimento.

1. Instale o [Go](https://golang.org/) 1.14 ou mais recente, caso ainda não o tenha feito.
2. Instale a extensão VS Code Go .
3. Abra qualquer diretório ou espaço de trabalho que contenha o código Go para ativar automaticamente a extensão. A [barra de status do Go](https://github.com/golang/vscode-go/wiki/ui) aparece no canto inferior esquerdo da janela e exibe sua versão do Go.
4. A extensão depende de , goe outras ferramentas opcionais. Se alguma das dependências estiver ausente, o aviso ⚠️ será exibido. Clique no aviso para baixar as dependências.goplsdlvAnalysis Tools Missing

Consulte a [documentação das ferramentas](https://github.com/golang/vscode-go/wiki/tools) para obter uma lista completa das ferramentas das quais a extensão depende.