

Próximo:[Definições de Variáveis](#), Anterior:[Palavras-chave comuns](#), Acima:[Costumização](#) [Conteúdo]
[Índice](#)

15.2 Definindo Grupos de Personalização

Cada pacote Emacs Lisp deve ter um grupo de customização principal que contém todas as opções, faces e outros grupos do pacote. Se o pacote tiver um número pequeno de opções e faces, use apenas um grupo e coloque tudo nele. Quando houver mais de vinte opções e faces, você deve estruturá-los em subgrupos e colocar os subgrupos no grupo de personalização principal do pacote. Não há problema em colocar algumas das opções e faces no grupo principal do pacote ao lado dos subgrupos.

O grupo principal ou único do pacote deve ser membro de um ou mais dos grupos de personalização padrão. (Para exibir a lista completa deles, use M-x `customize`.) Escolha um ou mais deles (mas não muitos) e adicione seu grupo a cada um deles usando a palavra-`:group` chave.

A maneira de declarar novos grupos de personalização é com `defgroup`.

Macro: doc dos membros do grupo `defgroup` [*valor da palavra-chave*]...

Declare `group` como um grupo de personalização contendo `membros`. Não cite o `grupo` de símbolos . O argumento `doc` especifica a string de documentação para o grupo.

Os `membros` do argumento é uma lista que especifica um conjunto inicial de itens de customização para serem membros do grupo. No entanto, geralmente os `membros` são `nil`, e você especifica os membros do grupo usando a palavra-`:groupchave` ao definir esses membros.

Se você deseja especificar membros do grupo por meio de `membros` , cada elemento deve ter o formato . Aqui `nome` é um símbolo e `widget` é um tipo de widget para editar esse símbolo. Widgets úteis são para uma variável, para um rosto e para um grupo. (`name widget`)`custom-variable``custom-face``custom-group`

Ao introduzir um novo grupo no Emacs, use a palavra-`:version` chave no `defgroup`; então você não precisa usá-lo para os membros individuais do grupo.

Além das palavras-chave comuns (consulte [Palavras-chave comuns](#)), você também pode usar essa palavra-chave em `defgroup`:

`:prefix prefix`

Se o nome de um item no grupo começar com `prefix` , e a variável personalizável `custom-unlispify-remove-prefixes` for non-`nil`, a tag do item omitirá `prefix` . Um grupo pode ter qualquer número de prefixos.

As variáveis e subgrupos de um grupo são armazenados na `custom-group` propriedade do símbolo do grupo. Veja [Plists de Símbolos](#) . O valor dessa propriedade é uma lista de pares cujo caré o símbolo da variável ou do subgrupo e o `cdr` ou `custom-variable` ou `custom-group`.

Opção do usuário: `custom-unlispify-remove-prefixes`

Se esta variável for non-`nil`, os prefixos especificados pela palavra-`:prefixchave` de um grupo serão omitidos dos nomes de tags, sempre que o usuário customizar o grupo.

O valor padrão é `nil`, ou seja, o recurso de descarte de prefixo está desabilitado. Isso ocorre porque o descarte de prefixos geralmente leva a nomes confusos para opções e faces.

Próximo:[Definições de Variáveis](#), Anterior:[Palavras-chave comuns](#), Acima:[Costumização](#) [Conteúdo][

[Índice](#)