

Sumario

1. Descripción.....	2
2. Instalación.....	2
3. Uso.....	2
4. Configuración.....	2
5. Mapeos.....	6
6. Documentación.....	7
7. Pruebas.....	7
8. Qué hacer (faltantes).....	7

1. Descripción

`Bullets.vim` es un complemento de Vim para listas de viñetas automatizadas.

2. Instalación

Con `VimPlug`:

```
Plug 'dkarter/bullets.vim'
```

Luego obtenga su archivo de paquete y ejecute `:PlugInstall`.

3. Uso

En Markdown o en un archivo de texto, comience una lista con viñetas usando `- o *`. Presione *Enter* para ir a la siguiente línea, se creará un nuevo elemento de lista.

4. Configuración

Puede elegir en qué tipos de archivos funcionará este complemento:

```
" Bullets.vim
let g:bullets_enabled_file_types = [
  \ 'markdown',
  \ 'text',
  \ 'gitcommit',
  \ 'scratch'
  \]
```

Puede deshabilitar este complemento para búferes vacíos (sin tipo de archivo):

```
let g:bullets_enable_in_empty_buffers = 0 " default = 1
```

Activar/desactivar asignaciones de teclas predeterminadas:

```
let g:bullets_set_mappings = 0 " default = 1
```

Agregue una clave de líder antes de las asignaciones predeterminadas:

```
let g:bullets_mapping_leader = '<M-b>' " default = ''
```

Habilita/deshabilita la eliminación de la última viñeta vacía al presionar `<cr>` (modo de inserción) u `o` (modo normal):

```
let g:bullets_delete_last_bullet_if_empty = 0 " default = 1
```

Espaciado de línea entre viñetas (1 = sin líneas en blanco, 2 = una línea en blanco, etc.):

```
let g:bullets_line_spacing = 2 " default = 1
```

No agregue relleno adicional entre la viñeta y el texto cuando las viñetas tienen varios caracteres:

```
let g:bullets_pad_right = 1 " default = 1
```

```
" I. text
```

```

" II. text
" III. text
" IV. text
" V. text
"   ^ extra spaces to align the text with the longest bullet

let g:bullets_pad_right = 0
" I. text
" II. text
" III. text
" IV. text
"   ^ no extra space between bullet and text
Número máximo de caracteres alfabéticos para usar en viñetas:
let g:bullets_max_alpha_characters = 2 " default = 2
" ...
" y. text
" z. text
" aa. text
" ab. text

let g:bullets_max_alpha_characters = 1
" ...
" y. text
" z. text
" text
Esquema de viñetas en niveles anidados:
let g:bullets_outline_levels = ['ROM', 'ABC', 'num', 'abc', 'rom', 'std-',
'std*', 'std+'] " default
" Ordered list containing the heirarchical bullet levels, starting from the
outer most level.
" Available bullet level options (cannot use the same marker more than once)
" ROM/rom = upper/lower case Roman numerals (e.g., I, II, III, IV)
" ABC/abc = upper/lower case alphabetic characters (e.g., A, B, C)
" std[-/*/+ ] = standard bullets using a hyphen (-), asterisk (*), or plus (+) as
the marker.
" chk = checkbox (- [ ])

let g:bullets_outline_levels = ['num', 'abc', 'std*']
" Example [keys pressed to get this bullet]:
" 1. first parent

```

```
" a. child bullet [ <cr><C-t> ]
"   - unordered bullet [ <cr><C-t> ]
" b. second child bullet [ <cr><C-d> ]
" 2. second parent [ <cr><C-d> ]
```

Habilite/deshabilite la reenumeración automática de la lista de viñetas ordenada actual al cambiar el nivel de sangría de las viñetas o al insertar una nueva viñeta:

```
let g:bullets_renumber_on_change = 1 " default = 1
" Example 1:
" 1. first existing bullet
"   a. second existing bullet [ hit <C-t> ]
" 2. third existing bullet [ this got renumbered 3 -> 2 when bullet 2 got demoted ]
"
" Example 2:
" 1. first existing bullet
" 2. second existing bullet [ use <cr>/o to add a new bullet below this ]
" 3. new bullet
" 4. third existing bullet [ this got renumbered 3 -> 2 when bullet 2 got demoted ]
```

```
let g:bullets_renumber_on_change = 0
" Example:
" 1. first existing bullet
"   a. second existing bullet [ hit <C-t> ]
" 3. third existing bullet [ no renumbering so this bullet remained `3` ]
"
" Example 2:
" 1. first existing bullet
" 2. second existing bullet [ use <cr>/o to add a new bullet below this ]
" 3. new bullet
" 3. third existing bullet [ no renumbering so this bullet remained `3` ]
```

Habilite/deshabilite la alternancia de casillas de verificación de padres e hijos para indicar "finalización" de las casillas de verificación secundarias:

```
let g:bullets_nested_checkboxes = 1 " default = 1
" Example:
" - [ ] first bullet
"   - [ ] child bullet [ type <leader>x ]
"     - [ ] sub-child
"   - [ ] child bullet
"
" Result:
```

```
" - [o] first bullet [ <- indicates partial completion of sub-tasks ]
"   - [X] child bullet
"     - [X] sub-child [ <- children get checked when parents get checked ]
"   - [ ] child bullet
```

Defina los marcadores de casillas de verificación que se utilizarán para indicar que:

- No están marcadas.
- Marcadas.
- "parcialmente" marcadas.

Cuando solo se definen dos caracteres de marcador, se desactivará el uso de marcadores de finalización parcial. Si se definen más de dos marcadores, cada carácter entre el primer y el último carácter se utilizará para indicar un porcentaje de las casillas de verificación secundarias que están marcadas. Cada marcador corresponde a $1/n$, donde n es el número de marcadores de finalización parcial. De forma predeterminada, hay tres marcadores de finalización parcial, `.`, `o` y `O`, correspondientes al 33%, 66% y hasta el final pero menos del 100%, respectivamente. Tenga en cuenta que los estados no marcados (`[]`) y marcados (`[x]` o `[X]`) que utilizan los marcadores predeterminados son siempre válidos, incluso si establece marcadores personalizados para no marcados y marcados.

```
let g:bullets_checkbox_markers = '.oOX'
```

" Example:

```
" - [o] parent bullet [ <- `o` indicates 66% - 99% of children are checked ]
"   - [ ] child bullet
"     - [.] child bullet [ <- partial completions don't count as complete ]
"       - [ ] sub-child bullet [ <- 1/4 of children checked so parent is `.` ]
"         - [ ] sub-child bullet
"           - [ ] sub-child bullet
"             - [X] sub-child bullet
"       - [X] child bullet
"     - [X] child bullet
"
```

" You can use fancy markers:

```
" let g:bullets_checkbox_markers = 'x◻◻✓'
```

```
" - [x] unchecked
" - [◻] partial
"   - [✓] checked
"     - [x] unchecked
"       - [x] unchecked
"         - [x] unchecked
```

Definir si alternar casillas de verificación parcialmente completas establece la casilla de verificación en marcada o sin marcar:

" Example 1:

```
let g:bullets_checkbox_partials_toggle = 1 " default = 1
```

```
" - [o] partially checked [ type <leader>x ]
```

```

" - [x] sub bullet
" - [ ] sub bullet
"
" Result:
" - [x] checked
" - [x] sub bullet
" - [x] sub bullet
"
" Example 2:
let g:bullets_checkbox_partials_toggle = 0
" - [o] partially checked [ type <leader>x ]
" - [x] sub bullet
" - [ ] sub bullet
"
" Result:
" - [ ] checked
" - [ ] sub bullet
" - [ ] sub bullet

```

5. Mapeos

- Insertar nueva viñeta en modo INSERT: `<cr>` (tecla de retorno).
- Igual que en caso de que desee desasignar en modo INSERT (la compatibilidad depende de su emulador de terminal): `<C-cr>`.
- Insertar nueva viñeta en modo NORMAL: `o`.
- Renumerar la selección visual actual: `gN`.
- Renumerar toda la lista de viñetas que contiene el cursor en modo NORMAL: `gN`.
- Alternar una casilla de verificación en el modo NORMAL: `<líder> x`.
- Degradar una viñeta (sangrarla, disminuir el nivel de la viñeta y convertirla en un elemento secundario de la viñeta anterior):
 - Modo NORMAL: `>>`.
 - Modo INSERTAR: `<C-t>`.
 - Modo VISUAL: `>`.
- Promocionar una viñeta (desangrarla y aumentar el nivel de la viñeta):
 - Modo NORMAL: `<<`.
 - Modo INSERTAR: `<C-d>`.
 - Modo VISUAL: `>`.

Deshabilitar asignaciones predeterminadas:

```
let g:bullets_set_mappings = 0
```

Agregue una clave de líder antes de las asignaciones predeterminadas:

```
let g:bullets_mapping_leader = '<M-b>'  
" Set <M-b> to the leader before all default mappings:  
" Example: renumbering becomes `<M-b>gN` instead of just `gN`
```

Simplemente agreguelo en la parte de arriba de su archivo de configuración `.vimrc`.

6. Documentación

`h:bullets`

7. Pruebas

El conjunto de pruebas está escrito con vimrunner. Se sabe que se ejecuta en macOS con MacVim instalado y en travis. Su Vim debe tener `+clientserver` y tener su propia GUI o en una ventana virtual de X11.

En su mac, ejecute:

```
bundle install  
bundle exec rspec
```

En GNU/Linux:

```
bundle install  
xvfb-run bundle exec rspec
```

8. Qué hacer (faltantes)

- ✓ Eliminar la viñeta final en la línea anterior si el usuario presiona dos veces.
- ✓ Permitir sangría en el modo de inserción (`C-l`: sangría a la derecha, `C-h`: sangría a la izquierda).
- ✓ Alcance las combinaciones de teclas y las funciones para markdown y *tal* vez texto plano.
- ✓ Permitir viñeta automática de casilla de verificación estilo GFM.
- ✓ Prefijar atajos y permitir deshabilitarlos.
- ✓ Agregar lista numerada.
- ✓ Restablecer números (el usuario selecciona las viñetas numeradas del 3 al 5 y las copia en el medio del documento, luego las vuelve a seleccionar y las restablece a 1-3).
- ✓ Compruebe si el complemento se inicializó y no cargue si lo hizo.
- ✓ Permitir la devolución sin crear una viñeta (solo es posible en GuiVim desafortunadamente).
- ✓ Compruebe si el usuario está en EOL antes de agregar viñeta automática; es posible que solo quieran.
- ✓ Intente mantener el mismo ancho total de viñeta incluso cuando el ancho del número varía (relleno derecho).
- ✓ Detectar listas que tienen viñetas de varias líneas (no deben tener líneas vacías entre líneas).
- ✓ Agregar lista alfabética.
- ✓ Soporte para viñetas alfanuméricas inteligentes con sangría, p. 1. \t a. \t 1.

Vim – bullets

- ✓ Cambiar niveles de esquema anidados en modo visual.
- ✓ Admite la renumeración de alfabéticos, números romanos y listas anidadas.
- ✓ Actualizar la documentación para viñetas anidadas.
- ✓ Admite viñetas anidadas con hijo y finalización parcial.
- ✓ Soporte para viñetas numéricas anidadas, por ejemplo, 1. -> 1.1 -> 1.1.1, 1.1.2.
- ✓ Agregue la opción para convertir las líneas que no son viñetas en nuevas viñetas con `<C-t> />> />`