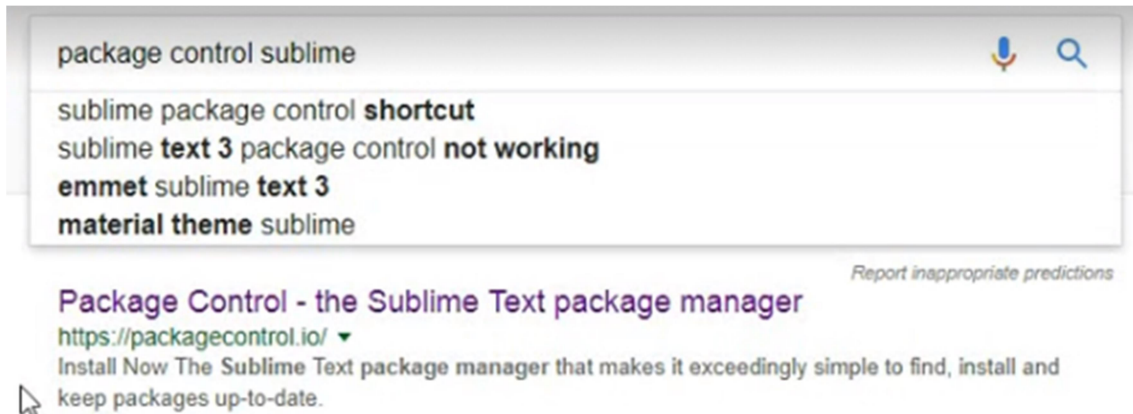


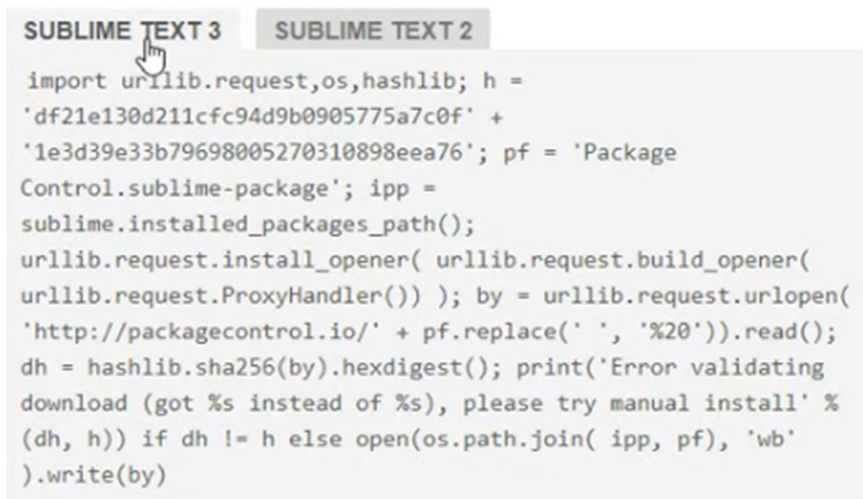
Instalar los Plugins en Sublime Text

Para poder instalar Plugins, que me ayudarán al programar, necesito primero instalar el control de paquetes.

Para ello:

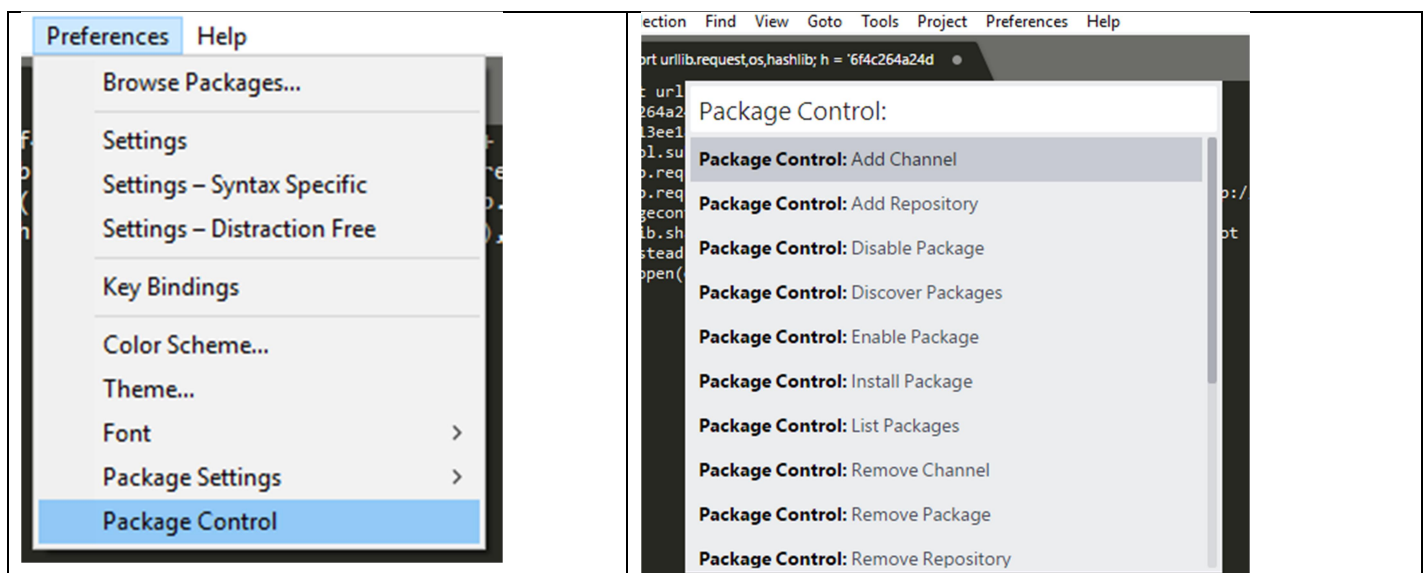


Copiamos el código que aparece para la instalación del paquete:



Vamos al Sublime text > View > Mostrar consola y lo pegamos y ejecutamos

Y ya debe aparecer en Preferencias > El controlador de Paquetes.



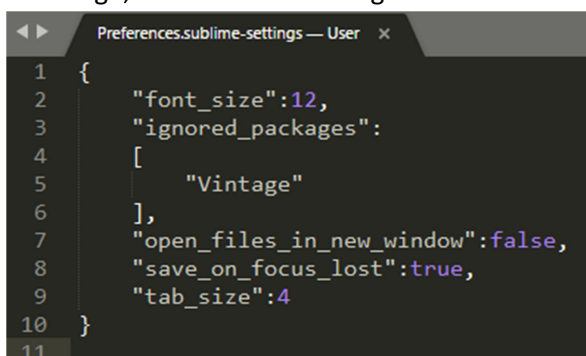
Abrimos el controlador de Paquetes > Install Packages>instalamos algunos Plugins:

1. **Plugin EMMET** atajos de Teclado usando el tabulador :
Como vamos a escribir un documento nuevo, para hacer pruebas, ponemos la vista View>Syntax HTML
!TAB → TAB es la tecla tabulador li*5 → Crea cinco
LoremTAB → TAB es la tecla tabulador
2. **AdvancedNewFile** → Para crear nuevos archivos: Ctrl+Alt+n
Aparece un cuadro, para poder poner el nombre del archivo a crear dentro del proyecto en el que estamos.
Si ponemos css/ej1.css creará la carpeta css y el archivo ej1.css
3. **AutoFileName** → Este plugin ayuda a Autocompletar el nombre del archivo que estamos vinculando.
!TAB → Creamos el nuevo Documento html
linkTAB → Crea una etiqueta link → Al posicicionarme en el href → Para seleccionar el archivo a vincular.
4. **BracketHighlighter** → Muestra aperturas y cierres de llaves, etiquetas y paréntesis.
5. **Commet-Snippets** → Permite crear comentarios básicos, de cabeza y pie, para css, html y javascript
Escribo en el documento comm-html-section TAB entonces aparecerá el comentario de bloque.
Si el documento es de css comm-section → Pero ojo, tengo que seleccionar comm-section y dar TAB

Otras Herramientas:

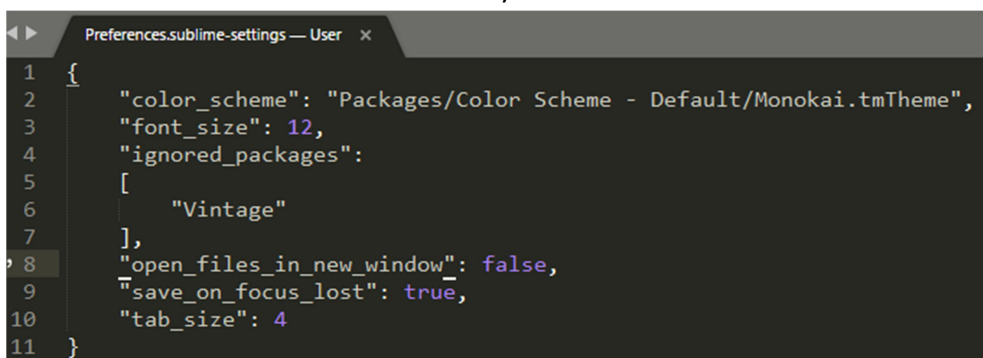
1. Configuración por Defecto

Preferences → Default , user (Por defecto, del usuario...) Puedo añadir más líneas en formato "json"
Ojo!! La ultima línea dentro de las llaves, va sin coma final. "save_on_focus_lost":true,
Se pondrá un circulito en el nombre del archivo para indicarme que debo guardar los cambios, con esta opción en el momento que vamos a otra pestaña, es decir, cuando hayamos perdido el foco de ese archivo de código, inmediatamente lo guarda.



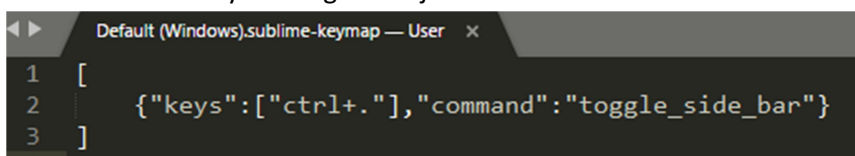
```
1 {
2   "font_size":12,
3   "ignored_packages":
4   [
5     "Vintage"
6   ],
7   "open_files_in_new_window":false,
8   "save_on_focus_lost":true,
9   "tab_size":4
10 }
11
```

Preferences → Color-Scheme → Monokay



```
1 {
2   "color_scheme": "Packages/Color Scheme - Default/Monokai.tmTheme",
3   "font_size": 12,
4   "ignored_packages":
5   [
6     "Vintage"
7   ],
8   "open_files_in_new_window": false,
9   "save_on_focus_lost": true,
10  "tab_size": 4
11 }
```

Preferences → key-binding → Atajo de teclado



```
1 [
2   {"keys":["ctrl+."], "command":"toggle_side_bar"}
3 ]
```

Me permite hacer desaparecer de la izquierda el panel lateral de navegación de archivos.

2. **Con Alt+Shift+2** → Se crearán dos columnas para poner poner por ejemplo el archivo css al lado.

Plugins para trabajar con JavaScript.

1. **JsFormat (Ctrl + Alt + f)** → Nos sirve para poner bonito el código que hagamos, organiza el código.
2. **Console Wrap (Ctrl + Shift + q)** → Nos ayuda a generar resultados en la consola de forma rápida.
3. **HTML Minifier (Ctrl + Shift + m)** → Comprime en una versión Mini, el código html, css o Javascript. Es muy útil para cuando ya estemos en producción, para subirlo al hosting, internet, para que las personas no tengan acceso de una manera clara y legible del código. Por eso lo comprimimos en una sola línea, quedando la versión .mini del código.
4. **TypeScript (sintaxis)** → Para que cuando escribamos el código en typescript, la sintaxis la muestre de una forma más agradable, coloreada.
5. **Angular2 HTML Syntax (sintaxis)** → Que permite ver sintaxis en html, cuando estemos trabajando con Angular. Aunque trabajemos con Angular4 el plugin sigue estando en la versión2, al hacer esta chuleta.

Nota:

JavaScript es un lenguaje de programación dinámico (cambiante), escalable (hacia el servidor), orientado a objetos.

JavaScript se utiliza sobre todo en el **Front-End** (Lado del Cliente, lo que el usuario ve)

Back –End (Lado del Servidor).

JavaScript permite interactuar con la página, teniendo más dinamismo.

Ejemplos:

- Una tienda Virtual → Los Slide-show (mostrando distintas imágenes, desplazándolas). Ofertas, publicidad...
- Algunos módulos al pasar el ratón por encima va generando efectos.
- Interacción de letras, efectos, sombras al pasar el cursor por encima
- Un catálogo de ropa, al pasar por encima de la ropa se mueve.
- Una página construida a nivel horizontal (mario Bross) va andando y creando su mundo.
- Con la interacción del scroll van apareciendo el efecto parallax, al compás que muevo hacia abajo el scroll, va cambiando la profundidad de las imágenes.
- El uso de Javascript para mostrar novelas gráficas, comics, viendo las viñetas a medida que avanzamos hacia abajo.