

Simulate key press by user with KeyEvent in Android

Each key press is described by a sequence of key events. A key press starts with a key event with ACTION_DOWN. If the key is held sufficiently long that it repeats, then the initial down is followed additional key events with ACTION_DOWN and a non-zero value for getRepeatCount(). The last key event is aACTION_UP for the key up. If the key press is canceled, the key up event will have the FLAG_CANCELED flag set.

Key events are generally accompanied by a key code (getKeyCode()), scan code (getScanCode()) and meta state (getMetaState()). Key code constants are defined in this class. Scan code constants are raw device-specific codes obtained from the OS and so are not generally meaningful to applications unless interpreted using the KeyCharacterMap. Meta states describe the pressed state of key modifiers such as META_SHIFT_ON or META_ALT_ON.

Key codes typically correspond one-to-one with individual keys on an input device. Many keys and key combinations serve quite different functions on different input devices so care must be taken when interpreting them. Always use the KeyCharacterMap associated with the input device when mapping keys to characters. Be aware that there may be multiple key input devices active at the same time and each will have its own key character map.

Key codes (numbers)

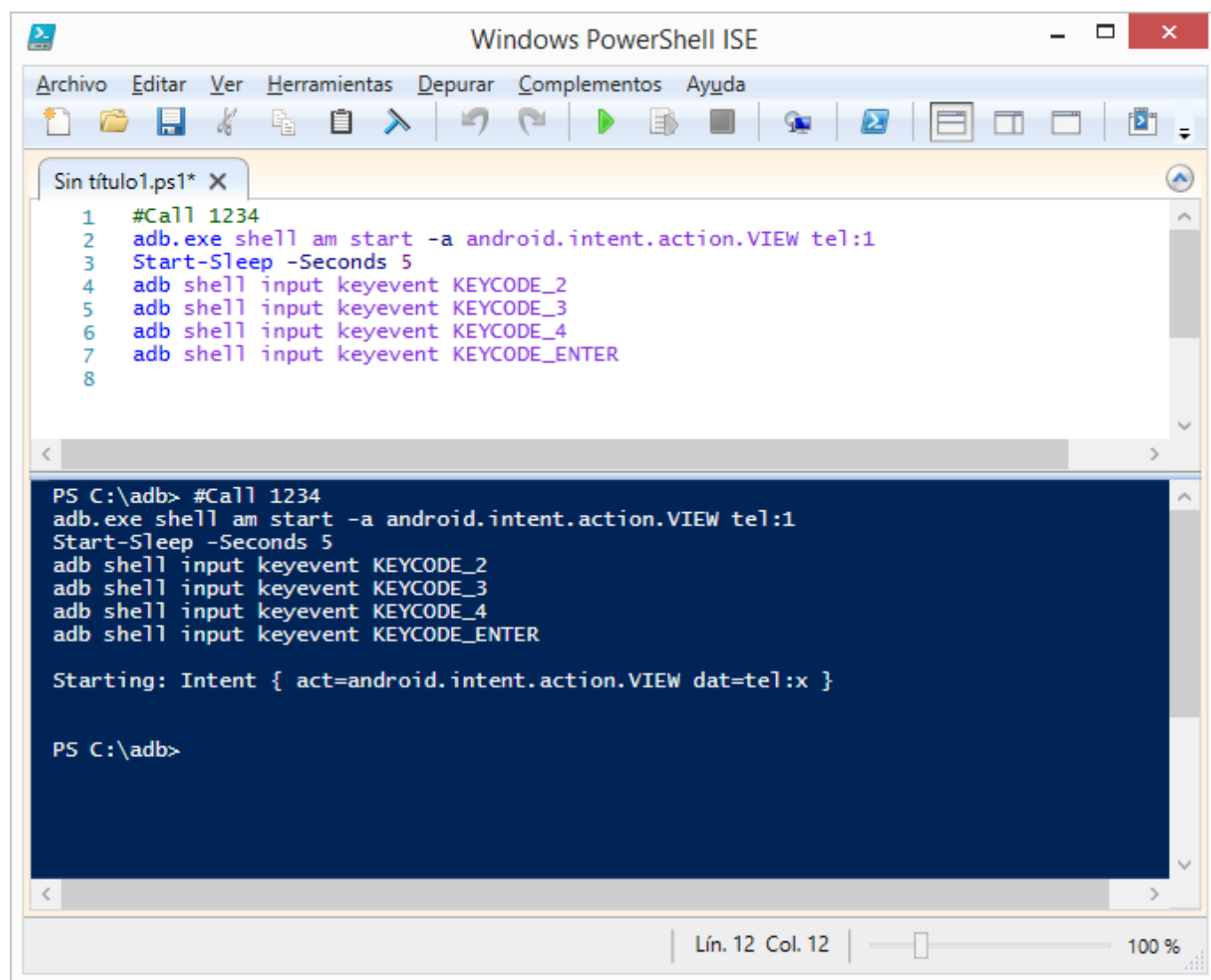
[crayon-6a9d6762b31a2224989626/]

More key codes

<https://developer.android.com/intl/es/reference/android/view/KeyEvent.html>

Example

[crayon-6a9d6762b31ab467191539/]



The screenshot shows the Windows PowerShell ISE interface. The top menu bar includes 'Archivo', 'Editar', 'Ver', 'Herramientas', 'Depurar', 'Complementos', and 'Ayuda'. Below the menu is a toolbar with various icons. The main editor area has a tab labeled 'Sin título1.ps1*'. The script content is as follows:

```
1 #Call 1234
2 adb.exe shell am start -a android.intent.action.VIEW tel:1
3 Start-Sleep -Seconds 5
4 adb shell input keyevent KEYCODE_2
5 adb shell input keyevent KEYCODE_3
6 adb shell input keyevent KEYCODE_4
7 adb shell input keyevent KEYCODE_ENTER
8
```

The console output at the bottom shows the execution of the script:

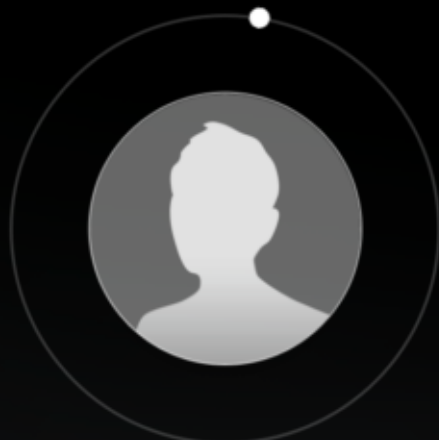
```
PS C:\adb> #Call 1234
adb.exe shell am start -a android.intent.action.VIEW tel:1
Start-Sleep -Seconds 5
adb shell input keyevent KEYCODE_2
adb shell input keyevent KEYCODE_3
adb shell input keyevent KEYCODE_4
adb shell input keyevent KEYCODE_ENTER

Starting: Intent { act=android.intent.action.VIEW dat=tel:x }

PS C:\adb>
```

The status bar at the bottom indicates 'Lín. 12 Col. 12' and a zoom level of '100 %'.

LLAMANDO



1234

Desconocido



Calendario



Retener



Bloc de notas



Añadir llamada



Silenciar



Contactos

