

Useful tricks to know when editing PDF files with VIM

1. Always start up using `vim -b!` (Without the `-b` Vim would try to do its clever tricks on a binary-including file the wrong way.) Make this a habit. Only binary editing mode (as triggered by `-b`) will correctly give byte-counting if needed.
2. Once a PDF is open, you can use the `:goto 3456` command to jump to byte offset 3456. Useful if you want to check `xref` entries.
3. Remember, how to open (in your default PDF viewer) the currently edited PDF file from within Vim: `:!open %` (OSX), `:!xdg-open %` (Linux), `:!start %` (Windows). (You know that `%` is a VIM shorthand variable for ‘currently opened file’, right?)
4. Define a custom status line which returns useful info about the current cursor position. Here is a suggestion:

```
:statusline=%F%m%r%h%w[L] [%{&ff}]%y [%p%%] [%04l,%04v] (%b) (%B) (%o)
```

- `%F` : currently open file name (with full path)
- `%m` : modified flag (`[+]` if modified)
- `%r` : readonly flag (`[RO]` if readonly)
- `%h` : helpfile flag (`[help]` if helpfile – maybe localized as `[Hilfe]`)
- `%w` : preview window flag (`[Preview]` if applicable)
- `%L` : total lines
- `%{&ff}` : file format (unix, dos,...)
- `%y` : file type as automatically recognized or manually set
- `%p%%` : relative position of cursor within file in percent
- `%06l` : current line position (column) of cursor, left padded with zeroes
- `%06v` : current line/row number of cursor, left padded with zeroes
- `%b` : ASCII value of the current character under cursor
- `%B` : HEX value of current character under cursor
- `%o` : file byte offset of cursor

Now a quick look on the status line shows the current file byte offset, line position, HEX value of character,...

5. How to jump to a specific byte offset (calculated from the start of the file):

```
:goto 37737
```

or

```
:go 37737
```

or simply (without the `:` to switch to command mode):

```
37737go
```

6. Looking at binary bytes? But want them displayed as Hex? Then try this:

```
:set display=uhex
```

Otherwise, the `ga` command displays the value of the character under the cursor.

`g CTRL+g` shows which byte offset you are at in the file.

