Tutorial: Plotting and Colors

(this was included in the last tutorial handed out, but was not assigned in class)

We're going to try to understand a little more about IDL's colors. In the process, we're going to do quite a bit of program editing work.

In order to cut down a bit on typing interactively at the command prompt, I'm providing you with some functioning code.

Get the code from git

Change directories to your `ASTR2600_materials` directory and `git pull`. You should have new files in the `examples/` directory: `color_ref.pro` and `eyeball.pro` (unless you got them last class). Copy these into your `tutorials` directory.

Use the new code

Open `color_ref.pro` in `gvim` and keep it open.

Open IDL. At the IDL prompt, run `color_ref.pro` as a *procedure*.
`.r color_ref.pro`

You have a color table! But... what does it mean? Looking at `color_ref.pro` probably doesn't tell you much. There are no axis labels! There are no comments! OH NO!

Your job is to correct this! You need to figure out what each step of the code does and write a description of what it does as comments. This is intended as practice in commenting. You are also practicing reading code, just like in `whuduzitdo`'s. When people share code, they may have a different idea of commenting than you - you still need to be able to figure out what the code does, even if they do a poor job of explaining it.

If you don't know what a part of the code does, here are some hints on how you can figure it out:

- If a statement can be run in the interactive mode, try copying & pasting it onto the command line to see what it does (but code blocks with `begin...end` cannot be examined this way)
- Use the IDL "online help" to look up functions and parameters you don't know (especially `plot` and `color_convert`).
- Once you've read the help for `color_convert`, look at the linked article on `HSV` - you should probably include a comment in the code stating what the HSV values mean
- Try to `print` a variable if you don't know what it will be

ASSIGNMENT: Correct `color_ref.pro` to be more useful by adding axis labels. Make sure it is well-commented.

Add a second and third window where instead of varying the first parameter in `color_convert`, you vary the second and third parameters. This will show you what the different inputs into `color_convert` do, and give you a nice tool for looking up colors in the future. Commit & push your changes when you're done.

REMINDERS ABOUT VIM: If you're in 'command mode' (i.e., there is nothing in the bottom-left corner like `--INSERT--` or `--VISUAL--`), `i` gets you into insert mode. `a` does exactly the same thing, except AFTER the current character instead of BEFORE it. You can use `o` to add a new line AFTER the current line or `O` to add a new line above the current line. In the default color scheme, comments will show up as blue.

In command mode, use `:w` to save. Use `esc` to get to command mode. Use the vim graphical cheatsheet too!

Playing with Plots and Colors

Next, run the `eyeball.pro` script (note that this is a script, not a program, so you have to `@` it instead of `.r`'ing it). Also open it in a text editor.

Add comments to `eyeball.pro` explaining what the unexplained lines (which we identified in class) do.

This is more practice with commenting and comprehending code. However, it's also intended to be a 'just-for-fun' exercise to get you accustomed to working with the plotter. Coding in general, but especially plotting, tends to be an 'iterative process': you try something, see if it works, then fix it if it doesn't.

If there's anything you dislike about the so-called eyes, see if you can fix them. Possible fixes:

1. Convince IDL to draw within the lines
2. Make the eyes elliptical
3. Add some other features frequently seen in eyes (e.g. whites)

Finally, add a mouth! This can probably be done with a parabola. You will also probably need to change the plot limits.

Do this all within your personal `eyeball.pro`. Again, save and commit your changes, and when you're done, push your changes and create a pull request.

Reminder: a parabola is an even power of x, e.g. x^2.

ASSIGNMENT: Comment, fix up (as much as you'd like, but at least a little), and add a mouth to `eyeball.pro`.