0.1   Introduction

The purpose of this document is to describe the standards for creating self-documenting makefiles (SDMF). As projects grow more complex and branch off into multiple makefiles, keeping track of variable-settings and targets becomes increasing complex.

This script is designed to collect the following information from any number of relevant makefiles you point it at:

- **Variables:** Defined through `VAR=x` syntax in the makefile preamble. Often references to directory paths (`bin/`, `incoming/`, etc)

- **Targets:** What's actually called from the command line, things like `PrepSubject`. These should be defined under `.PHONY`

- **Files:** Intermediate targets not called directly from the command line, something like `mprage/T1_brain.nii.gz`

0.2   Calling the script

Documenting makefiles is easy, simply call from the command line `document-makefile`, followed by the name(s) of one or more makefiles.

```
$ ./document-makefile ASL.mk
```

`document-makefile` requires the following scripts to be in the same directory:

- `makemakedoc.py` – A Python script that reads the makefiles and creates the output text files `variables.txt`, and `targets.txt`, `intermediates.txt` for later interpretation.

0.3   Makefile commenting syntax

Because we have three things to identify from the makefiles (variable, targets, and files), we introduce three comment "keywords" to facilitate information extraction. All makefile comments begin with `#`, so SDMF comments will also begin with `#`.

Semicolons (`;`) are **disallowed** in comments. They will be sanitized.

Comment	Description
`#`, `##`, ...	(Any number of only octothorpes.) Only octothorpes without any SDMF control characters will be ignored, for makefile-only comments, section headers and the like. E.g `### QA ###` will be completely ignored by the parser.

`#!`	Variable definition.
	The only comment to appear on the same line as its referent, this should be appended after the variable definition (to save vertical space)
	E.g.
	`PROJECT_DIR=/mnt/home/adrc/ADRC # top level of the`
	`project directory`

`#>`	File (intermediary target) descriptions.
	Any number of `#>` comments can be used before an intermediate target. I don't recommend using more than two or three. Newlines will not be preserved
	E.g
	`#> Register m0 to t1 using fnirt`
	`pcasl/pcasl_fnirt.nii.gz: pcasl/Pcasl_skstrip.nii.gz`
	`mprage/T1_brain.nii.gz`

`#?`	Target descriptions.
	One-line descriptions of targets to be called from the command line, e.g. `all`, `PrepSubject`.
	E.g.
	`#?  Make all the relevant PCASL registrations`
	`PCASL: pcasl/pcasl_fnirt.nii.gz pcasl/pcasl_M0_to_T1_-`
	`ANTs_Warped.nii.gz`