A. The ‘genome model tools’ (gmt) command

```
ssmith@blade13-4-4:~ $ gmt
Sub-commands for gmt:
  smalt      Tools to run smalt or work with its output files.
  tophat     Tools to run TopHat or work with its output files.
  ccsahu2    Tools to run SSMHAU2 or work with its output files.
  bfast      Tools to run Bfast or work with its output files.
  tigra-sv   Tools to run tigra_sv or work with its output files.
  blat       Tools to run Blat or work with its output files.
  bwa        Tools to run BWA or work with its output files.
  crossmatch Tools to run Crossmatch or work with its output files.
  picard     Tools to run the Java toolkit Picard and work with SAM/BAM format files.
  novocraft  Tools to run Novocraft or work with its output files.
  bmap       Tools to run BSMAP or work with its output files.
  mosaiq     Tools to run Mosaiq or work with its output files.
```

B. Each top-level command provides access to a list of tools, or further sub-trees

```
bowtie      tools to work with the Bowtie aligner
bwa-sw      tools to work with Ssaha output
chimera-skyr tool to run chimera skyr detection
complete-genomics ... base class for commands which delegate to sub-commands
  ... the CopyCat copy number analysis tools
cufflinks   tools to run Cufflinks or work with its output files.
detect-variants ... A selection of variant detectors.
dgdist      ... Toolkit for DGI related process
ensemble    ... Tools to work with the local Ensemble API.
epitope-prediction ... Different pipeline steps for Immune Epitope Prediction for
  ... To trim adapter sequences
  ... tools for working with FASTA and QuaI files
fastq      ... tools for working with FASTQ files
fastqc     ... tools to run the Java toolkit FastQC and work with the
  ... output reports.
fastx       ... tools to run Fastx or work with its output files.
  ... the Galaxy web interface
gatk       ... tools to work with Gatk output
genome-torrent no description!!!: define `doc' in the class definition for
  Genome.Model::Tools::GenomeTorrent
gtf         ... tools to work with gtf format annotation files.
htseq       ... htsq tools (htseq-count and htsq-gp) work with
  ... gene/transcript hit-counts
lift-over   wrapper for the UCSC liftOver tool with support for
  ... additional input formats, maintaining additional columns
C. The ‘gmt fasta’ sub-tree contains script-like components for working with FASTA files

```
ssmith@blade12-1-1 ~ $ gmt fasta
Sub-commands for genome tools fasta:
apply-diff  --diff1? --input=  applies seq inserts and deletes from a diff file to a fasta file
chunk      --chunk-size=?  --... Divide fasta into chunk by chunk_size
concat     --input-files=?... Mixin that gives commands color option
deduplicator--fasta-file=?[].. remove duplicates from a file of reads
diff       [--debug] FILE1 ... use KDiff3 to show differences between fasta data files
dust       --dusted-file=?... Tools for working with FASTA and Qual files
filter-ids [--verbose] [--whitespace] filter sequences from a fasta file based on patterns applied to the IDs
orient     --fasta-file=?    --... Orients FASTA and Quality files by blastn given sense and anti-sense sequences
remove-a   --fasta-file=?    --... remove reads from file containing N
sanitize    --fasta-file=?    --... Cleans FASTA and Quality files
screen-vector--fasta-file=?... (freq = -fastq) screen for vector sequences
sort-by-name--input-fasta=?... --... Sorts a fasta by sequence name
```

D. Each tool has auto-generated help

```
ssmith@linux43 ~ $ gmt fasta filter-ids --h

USAGE
gmt fasta filter-ids [--verbose] [--whitelist-regex=] [--blacklist-regex=]
       INPUT-FILNAME OUTPUT-FILNAME

SYNOPSIS
gmt fasta filter-ids in.fa out.fa --whitelist '^(d+X,Y)$' --blacklist '6'

REQUIRED INPUTS
  INPUT-FILNAME
    the input file

OUTPUT-FILNAME
    the path to the file that will be created

OPTIONAL INPUTS
  whitelist-regex
    include only IDs that match this pattern
  blacklist-regex
    exclude any IDs that match this pattern

OPTIONAL PARAMS
  verbose
    more messages

DESCRIPTION
This tool filters a FASTA sequence file, removing entries based on the ID in the FASTA header.
If the "whitelist regex" (-w) option is supplied, only IDs that match this regular expression
will be included. If the "blacklist regex" (-b) option is supplied, only IDs that do NOT match
this regular expression will be included.

NOTE
If an ID matches both the black list and the white list, it is skipped.
```
E. The code for a GMT tool can be as simple as a short script

```perl
package Genome::Model::Tools::Fasta::FilterIds;
use strict;
use warnings;
use Genome;
use Bio::SeqIO;

class Genome::Model::Tools::Fasta::FilterIds {
    is => 'Command::V2',
    has_input => {
        input_filename => {
            is => 'filesystemPath', shell_args_position => 1,
            doc => 'the input file',
        },
        output_filename => {
            is => 'filesystemPath', shell_args_position => 2,
            doc => 'the path to the file that will be created',
        },
        whitelist_regex => {
            is => 'Text', is_optional => 1,
            doc => 'Include only IDs that match this pattern',
        },
        blacklist_regex => {
            is => 'Text', is_optional => 1,
            doc => 'Exclude any IDs that match this pattern',
        },
    },
    has_param => {
        verbose => {
            is => 'Boolean', is_optional => 1,
            doc => 'more messages',
        },
    },
    doc => 'Filter sequences from a fasta file based on patterns applied to the IDs',
}

sub execute {
    my $self = shift;
    my $input_filename = $self->input_filename;
    my $output_filename = $self->output_filename;
    my $verbose = $self->verbose;
    my $blacklist_regex = $self->blacklist_regex;
    my $whitelist_regex = $self->whitelist_regex;
    my $reader = Bio::SeqIO->new('-file' => '<'.$input_filename, '-format' => 'fasta');
    my $writer = Bio::SeqIO->new('-file' => '>'.$output_filename, '-format' => 'fasta');

    while (my $seq = $reader->next_seq) {
        my $id = $seq->id;
        if ($blacklist_regex and $id =~ $blacklist_regex) {
            $self->status_message("skipping $id because it matches the blacklist pattern");
            next;
        }
        elsif ($whitelist_regex and not $id =~ $whitelist_regex) {
            $self->status_message("skipping $id because it does not match the whitelist pattern");
            next;
        }
        elsif ($verbose) {
            $self->status_message("keeping $id");
        }
        $writer->write_seq($seq);
    }
    return 1;
}
```
F. Additional code can be added to the module to explicitly or dynamically generate other documentation

65 sub help_synopsis {
66    return "<EOS>
67    gmt fasta filter-ids in.fa out.fa --whitelist "^\((x|y)\)#" --blacklist "6"
68    <EOS
69    }
70    }
71
72 sub help_detail {
73    return "<EOS"
74    This tool filters a FASTA sequence file, removing entries based on the ID in the FASTA header.
75    If the "whitelist regex" (-w) option is supplied, only IDs that match this regular expression will be included.
76    If the "blacklist regex" (-b) option is supplied, only IDs that do NOT match this regular expression will be included.
77    <EOS
78    }
79
80 sub _additional_help_sections {
81    return {
82      "NOTE" =>
83      "If an ID matches both the black list and the white list, it is skipped."
84    },
85    }
86
87 sub _doc_manual_body {
88    # expect to return POD
89    my $help = shift->help_detail;
90    $help =~ s/\n\n/\n/g;
91    return $help;
92    }
93
94 sub _doc_authors {
95    return "<EOS"
96    Scott Smith
97    Edward Belter
98    <EOS
99    }
100
101 sub _doc_copyright_years { (2013) }
102
103 sub _doc_license {
104    my $self = shift;
105    my $by = $self->{_doc_copyright_years};
106    my $range;
107    if ($by == 1) { $range = "$by\[0\]"; }
108    elsif ($by > 1) { $range = "$by\[0\]...$by\[-1\]"; }
109    return "<EOS"
110    Copyright (C) $range Washington University in St. Louis.
111
112    It is released under the Lesser GNU Public License (LGPL) version 3. See the
113    associated LICENSE file in this distribution.
114    <EOS
115    }
116
117 sub _doc_credits {
118    return "'This software was created with funding from the National Human Genome Research Institute.';"
119    }
120
121 sub _doc_see_also {
122    return "<EOS"
123    B-Genome-(3)
124    <EOS
125    }
126
127