```
<CH.5>[Command Line Skills]
  'Most consumer OSes are designed to shield the user from the ins and outs of the CLI.'
  'The Linux community positively celebrates the CLI for its power and speed.'
<Shell>
  'The shell is the command line interpreter that translates commands entered by a user'
    'into actions performed by the operating system.'
  'Bash shell popular features:'
    >Scripting: 'place commands in a file and then interpret them all commands.'
    >Aliases: 'create short nicknames for longer commands.'
    >Variables: 'store information for the shell and user.'
  'the prompt indicates that commands can be run and'
    'also provides useful information to the user.'
  sysadmin@localhost:~$
    'sysadmin: user name',
    'localhost: system name',
    '~: home dir:/home/sysadmin'
<Commands>
  'a software program that, when executed on the CLI, performs an action on the computer.'
  'typical command format:'
    command [options] [argument]
-Arguments:
  'can be used to specify something for the command to act upon.'
  'the ls command accepts multiple arguments.'
-Options:
  'can be used with commands to expand or modify the way a command behaves.'
  'using the -l option in the ls command results in a long listing.'
  'can be combined such as in -lh (long-listing human-readable.)'
  'order of combined options isnt important'
  'single word: -h'
  'full-word: --human-readable'
```

```
-History:
  'commands are stored in a history list after execution.'
  history   →'view history list.'
  !n        →'execute nth command in history list.'
  history n →'outputs n previous commands.'
  !-n       →'execute nth from the bottom of history list.'
<Variables>
  'are given names and stored temporarily in memory.'
  'allow the user or the shell to store data.'
-Local:
  'Local (shell) variables exists only in the current shell and cannot affect'
    'other commands or applications.'
  'lowercase by convention and associated with with user-based tasks.'
  'input: variable=value'
  'output: echo $variable (display output in the terminal)'
-Environment:
  'or global variables, are available system-wide (all shells and performing tasks.)'
  'PATH, HOME, and HISTSIZE are some the global variables.'
  'env: outputs a list of the environment variables.'
    'ej: env | grep variable1'    'the pipe | character passes the output of'
                                  'the env command to the grep command.'
  'export: turns a local variable into an environment variable.'
  'unset: removes exported variables.'
-Path:
  'PATH: contains a list that defines which directories the shell looks in'
    'to find commands.'
<Command Types>
  'type: can be used to determine information about command type.'
-Internal:
  'also called built-in command, are built into the shell itself.'
    $type cd
    'cd is a shell builtin'
```

```
-External:
  'stored in files searched by the shell.'
  which command 'displays the full path to the command.'
    $which cal
    'cal is /usr/bin/cal'
  type -a command 'displays all locations of the command.'
-Aliases:
  alias name=command 'creates a new alias for the command.'
  'aliases created this way only persist while the shell is open.'
-Functions:
  'can be built using existing commands'
-Quoting:
  'information within quotes are treated in a specific way.'
  'Glob characters, also called wildcards, are symbols that have special'
    'meaning to the shell.'
  1>Double(""):
    'doesnt convert the glob patterns:' (*, ?, [])
    'still allow for command substitution:' $(command) or `command`
    'or variable substitution:' $[variable]
  2>Single(''):
    'prevent the shell from doing any interpreting of special characters.'
  3>Backquotes(``):
    'or backticks, are used to specify a command within a'
      'command (command substitution.)'
    $echo today is `date`
    'today is Wed Mar 24 01:55:15 PM AST 2021'
<Control Statements>
  'allow the use of multiple commands at once.'
  1>Semicolon:
    command1; command2; command3
    'used to run a command right after the other, regardless if'
      'the past command failed.'
```

```
2>Double Ampersand:
  command1 && command2
  'acts as a short-circuited logical `AND`'
3>Double Pipe:
  command1 || command2
  'acts as a short-circuited logical `OR`'
```