2. Two modes, insert and normal

1 Vim has two basic modes. One is `insert` mode, in which
  you write text as if in normal text editor.

2 Another is `normal` mode which provides you efficient
  ways to navigate and manipulate text.

3 At any time, you can see which mode you are in on the
  status bar which is located at the top of the editor.

4 To change between modes, use `Esc` for normal mode and `i`
  for insert mode

5 Let's try it out! First, change to insert mode.

6 Good, now you're in insert mode. Write something and
  change back to normal mode. Hello

7 Good. Let's move on to another section. █

3. Basic movement: h, j, k, and l

1 In contrast to regular text editor, you use keys `h`, `j`,
  `k`, and `l` instead of arrow keys to move the cursor.

2 Let's see how it works in practice!

4. Word movement: w, e, b

1 To navigate the text in terms of words, you can use keys
  `w`, `b`, and `e` (also W, B, E in real Vim).

2 `w` moves to the start of next word; `e` moves to the end of
  the word; and `b` moves to beginning of the `w`ord.

5. Number powered movement, e.g. 5w

mode: NORMAL   1, 24

```
1  Moving  within  the  text  is  not  limited  to  individual
   keys;  you  can  combine  movement  keys  with  a  number .  For
   example,  3 w  is  the  same  as  pressing  w  three  times.
```

6. Insert text repeatedly, e.g. 3iYes

mode: NORMAL   5, 47

```
1  You  can  insert  text  multiple  times.

2  For  example,  an  underline  of  a  header  might  consist  of
   30  - s.

3  ------------------------------

4  With  30i -  Esc ,  there's  no  need  to  press  -  30  times.

5  Let's  try  it  out:  insert  go  three  times.gogogo
```

7. Find a character, f and F

mode: NORMAL   2, 92

```
1  To  find  and  move  to  the  next  (or  previous)  occurrence  of
   a  character,  use  f  and  F ,  e.g.  fo  finds  next  o.

2  You  can  combine  f  with  a  number.  For  example,  you  can
   find  3rd  occurrence  of  'q'  with  3 f q ,  que?
```

8. Go to matching parentheses, %

mode: NORMAL   2, 9

```
1  In text that is structured with parentheses or brackets,
     ( or { or [, use % to jump to the matching parenthesis
     or bracket.

2  Here is (a sample) text to try that.
```

9. Go to start/end of line, 0 and $

mode: NORMAL   2, 1

```
1  To reach the beginning of a line, press 0.

2  For the end of a line, there's $
```
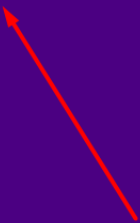
10. Find word under cursor, * and

mode: NORMAL   1, 29

```
1  Find the next occurrence of the word under cursor with
     *, and the previous with #.
```

11. Goto line, g and G

1 `gg` takes you to the beginning of the file; `G` to the end.

2 To jump directly to a specific line, give its `line` `number` along with `G`.

3 Now go to the beginning of this screen with `gg` and then back to end with `G`.

Linux Software Diagnostics (LSD) Shell Script Ver.1.0

```
Press Up/Down Keys to move
- e User name: hussain (Login name: hussain)
- e Home Directory: /home/hussain
- e Your O/s Type: linux-gnu
- e PATH: /usr/lib/git-core:/usr/local/sbin:/usr/local/bin:/usr/sb
- e Current directory: /home/hussain
- e Currently Logged: 1 user(s)
- e Available Shells:
- e ----------------------------------------------------------------
- e Computer CPU Information:
- e ----------------------------------------------------------------
processor       : 0
vendor_id       : GenuineIntel
cpu family      : 6
model           : 165
model name      : Intel(R) Core(TM) i7-10700 CPU @ 2.90GHz
stepping        : 5
cpu MHz                 : 2903.998
                                                              10%
                            < EXIT >
```