CSC209 Week 1 Notes

Hyungmo Gu

May 8, 2020

The Unix Command Line

- cat

    - *cat hello* : reads file

    ```
    1    >>> cat hello
    2    Hello World!
    3
    ```

    - *cat hello1 > hello2* : transfer contents from file *hello1* to *hello2*

    ```
    1    >>> cat hello > hello1
    2    >>> cat hello1
    3    Hello World!
    4
    ```

        * *hello2* created if doesn't exist

- rm

    - *-i* in *rm -i* : creates prompt
    - *-r* in *rm -r* : removes files recursively
    - *-f* in *rm -r -f* : supresses prompt (also removes files recursively).

    Don't use *-f* in *rm -r -f* unnecess necessary.

- alias

    - A shell alias is a shortcut to reference a command
    - avoids typing long commands

    ```
    1    >>> alias rm='rm -i'
    2    >>> rm  hello
    3    remove hello?
    4
    ```

    - *unalias* - remove alias

1

Software Tools Part 1 of 2

- Software Tool Principles

  1. Does one thing well
  2. Are small
  3. Interfaced cleanly
  4. Expect the output of every program to become the input of another
  5. Program inputs easy to generate or type

     i.e. [NO]: binary input tpye.
  6. Use regular expressions for all pattern matching

Software Tools Part 2 of 2

- Software Tools in Unix

  1. *grep*
     - outputs lines which match a pattern
     ```
     >>> grep '1' sample.txt
     1
     10
     ```

  2. *head, tail*
     - Returns the first or last n lines of file
     ```
     >>> head -10 sample.txt
     1
     2
     3
     4
     5
     ```

  3. *sort*
     - Sorts the input
     - *-n* does numeric sorting in file
     ```
     >>> sort -10 sample.txt
     1
     10
     2
     3
     4
     ```

```
7            5
8            6
9            7
10           8
11           9
12
```

4. *uniq*

   – Collapses adjacent identucal lines

```
1            >>> uniq sample2.txt
2        Hot
3        Cold
4        Hot
5        Cold
6        Warm
7
```

5. *sed*

   – Collapses adjacent identucal lines
   – *s* in $s/<FROM>/<TO>$ means subtitution

```
1            >>> cat sample3.txt
2        Hello
3        Hello
4        >>> sed 's/Hello/hi' sample3.txt
5        hi
6        hi
7
```

6. *diff*

   – Compares two files in a text-oriented way

```
1        diff -c sample4_1.txt sample4_2.txt
2        *** sample4_1.txt        2020-05-08 14:16:47.000000000
     +0900
3        --- sample4_2.txt        2020-05-08 14:16:59.000000000
     +0900
4        ***************
5        *** 1 ****
6        ! Hello beautiful world
7        \ No newline at end of file
8        --- 1 ----
9        ! Hello wonderful world.
10       \ No newline at end of file
11
```

7. *comm*

   – Compares two files in a text-oriented way

```
1        comm sample6_1.txt sample6_2.txt
2        001  Fred  Flintstone
3               002 80
4        002  Wilma  Flintstone
5               003 90
6        003  Betty  Rubble
7               004 75
8        004  Barney  Rubble
9               005 0
10       005  Pebbles  Flintstone
11       006  Bam-Bam  Rubble       007 68%
12
```

8. *join*

   − Performs database join of two files

```
1        >>> join sample6_1.txt sample6_2.txt
2        002  Wilma  Flintstone 80
3        003  Betty  Rubble 90
4        004  Barney  Rubble 75
5        005  Pebbles  Flintstone 0
6
```