**Scripting Languages**

**Assignment 1**

Write a bash file that will output the answer to the following questions

1. For this problem you will write a Shell script called todo.sh that allows a user to manage a todo list at the command line.

The todo.sh script can take numerous forms of command line arguments…

* ./todo.sh ***help***
* ./todo.sh ***add "task in quotes" [optional date string]***
* ./todo.sh ***del <number>***
* ./todo.sh ***list [optional date string]***

Any command line arguments that do not adhere to one of the aforementioned forms should be treated as an error.

**Help**

The help command should print out a synopsis of available commands along with a brief description as to what each command does.

**Add**

The add command requires at least 1 additional argument which is a quoted string. The quoting is necessary for your shell to hand off the entire string as a single argument to the script (otherwise, each word would be a separate argument). Note that when passing in a quoted string, the quotes automatically disappear from the input.

An add command should insert and save the specified task into the todo list data file (specified below). If that task already exists in the list (i.e. task description is the same), then an error should be presented to the user. When a task is added to the list, it should be assigned a unique “1-up” integer value (starting with 1 for the first task).

**List**

The list command should display the tasks in the todo list. The output should be neatly formatted (spacing) and should contain the task number, the task description & the due date if specified.

**Delete**

The del command should accept an additional argument which should be an integer greater than zero. The task with this number should then be located, removed and the data file updated accordingly.

There is the possibility for several errors here — the task may not exist, or the argument specified by the user may not be a positive integer. All of these cases should result in an appropriate error message.

**Data File**

The data for this problem **must be stored under a directory called “/todo” in the user's home directory**. Your script should gracefully handle the case when ~/todo/ doesn't yet exist. If the user tries to perform any command and it doesn't yet exist, you'll need to handle that gracefully and create directories/files as needed. The format of this file is left completely open for you to decide.

**You are not permitted to write to any location other than the ~/todo/ directory.** If the user wishes to blow out the entire todo list, they may simply remove the ~/todo/ directory and start over again from scratch.

Sample Input/Output

|  |  |
| --- | --- |
|  1 2 3 4 5 6 7 8 910111213141516171819202122232425262728293031323334353637 | $ ./todo.sh add "TPS Reports" “03-01-2015”ADDED #1 - TPS Reports – 03/01/2015$ ./todo.sh add "Go fishing this weekend " ADDED #2 - Go fishing this weekend $ ./todo.sh add "Meeting with friends" "03/08/2015"ADDED #3 - Meeting with friends – 03/08/2015$ ./todo.sh add "Help Samir with paper jam"ADDED #4 - Help Samir with paper jam$ ./todo.sh add "Update bank software for the 3000 switch" "9/30/2015"ADDED #5 - Update bank software for the 3000 switch - 09/30/2015$ ./todo.sh listUPCOMING ITEMS: 1: TPS Reports 03/01/2015 2: Go fishin' this weekend with Lawrence  3: Meeting with friends 03/08/2015 4: Help Samir with paper jam  5: Update bank software for the 3000 switch 09/30/2015 $ ./todo.sh del 4DELETED #4 - Help Samir with paper jam$ ./todo.sh listUPCOMING ITEMS: 1: TPS Reports 03/01/2015 2: Go fishing this weekend  3: Meeting with friends 03/08/2015 5: Update bank software for the 3000 switch 09/30/2015 $ ./todo.sh helpYour clever usage statement here |
|  |  |