El Duque - User Guide

By: `CS2113-T16-2` Since: `Aug 2019` Licence: `MIT`

1. Introduction

El Duque is a desktop task manager made for computing students. More importantly, El Duque is optimized for those who prefer to work with a Command Line Interface (CLI). If you can type fast, El Duque can organise your life better than anything else. Interested? Jump to the Section 2, "Quick Start" to get started. Enjoy!

2. Quick Start

1. Ensure you have Java 11 or above installed in your Computer.

2. Download the latest `El-Duque.jar` here.

3. Copy the file to the folder you want to use as the home folder for your personal Task Manager.

4. Open the command line and navigate to the selected folder and type in the following command.

```
java -jar El-Duque.jar
```

```
I see this is your first time running this program! I've taken the liberty to create a default task list for you!
Hello! I'm El Duque
What can I do for you?
list
ID | Filter |  Type  | Priority | Recurrence | Duration | Done? |     DateTime      | Description
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
01 |  self  |  Task  |   Low    |   Daily    |    1     |   N   |        NA        | go to the gym
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
02 | cs2113 |  Task  |   High   |    NA      |    N     |   N   | 29/10/2019 00:00 | Submit UG
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
03 |  home  |  Task  |   Low    |   Daily    |    2     |   N   |        NA        | sweep the floor
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
04 |  home  |  Task  |   Low    |    NA      |    N     |   N   |        NA        | wash bedsheet
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
05 |        |  Task  |   Low    |    NA      |    N     |   N   |        NA        | wash dishes
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
06 |        |  Task  |   Low    |    NA      |    N     |   N   |        NA        | cs2113 assignment
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
07 | cs2113 |  Task  |   High   |    NA      |    N     |   N   | 29/10/2019 00:00 | Submit DG
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
08 |  self  |  Task  |   Low    |   Daily    |    8     |   N   |        NA        | sleep
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
09 |        |  Task  |  Medium  |    NA      |    N     |   N   |        NA        | PPP submission
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
10 |        | Event  |  Medium  |    NA      |    N     |   N   | 29/10/2019 00:00 | Oral Exam
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
11 |        | Event  |   High   |    NA      |    N     |   N   | 10/02/2019 10:00 | CS2113 PE
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
12 |        |  Task  |   Low    |    NA      |    2     |   N   | 29/10/2019 00:00 | CG2271 Tutorial 10
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
13 |  cca   |  Task  |   Low    |    NA      |    N     |   N   | 10/11/2019 23:59 | cca camp proposal
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
14 |        |  Task  |  Medium  |   Weekly   |    N     |   N   |        NA        | st2334 tutorial
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
15 |  cca   |  Task  |   High   |    NA      |    N     |   N   |        NA        | Vehicle permit application
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
16 |  self  |  Task  |   High   |    NA      |    N     |   N   |        NA        | Internship application
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
17 |        |  Task  |  Medium  |    NA      |    N     |   N   |        NA        | Visa application
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
18 |  self  | Event  |   Low    |    NA      |    2     |   N   | 10/11/2019 23:59 | Ippt at Maju Camp
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
19 |  self  |  Task  |   High   |    NA      |    N     |   N   |        NA        | Defer reservice
-- | ------ | ------ | -------- | ---------- | -------- | ----- | ---------------- | -----------
20 |        |  Task  |   Low    |   Weekly   |    1     |   N   |        NA        | Kattis grind
```

> **NOTE** If you are running the program for the first time, El Duque will preload 20 tasks to get you started!

5. Type the command in the command box and press Enter to execute it.
   e.g. typing **help** and pressing Enter will show you a brief summary of commands and their format

6. Some example commands you can try:

   - `list` : lists all tasks

   - `task example` : adds a task with a description called example to the task manager

   - `delete 3` : deletes the 3rd task shown in the current list

   - `bye` : exits the app

7. Refer to Section 3, "Features" for details of each command.

3. Features

El Duque relies on a flexible command line syntax to parse in commands. Here is the general form of that said syntax.

```
[-FILTER] COMMAND [-MODIFICATIONS]
```

There are 3 main parts to the syntax and filter and modifications have to be prefixed with `-`. Filter and modifications are optional. They serve to add extra depth to the basic commands. You can think of filters as sublists. One very important point to note is that `INDEX` is relative to `FILTER`.

For example:
`edit 3 -description do laundry` can be doing exactly the same thing as `-housechores edit 1 -description do laundry` if the first task in housechores filter tasklist is the same as the third task in the overall tasklist.

Command Format

- Words in `UPPER_CASE` are the parameters to be supplied by the user e.g. in `task DESCRIPTION`, `DESCRIPTION` is a parameter which can be used as `task example`.

- Items in square brackets are optional e.g `DESCRIPTION [-p PRIORITY]` can be used as `task example -p high` or as `task example`.

- All optional modifications except FILTER can be in any order e.g. if the command specifies `task DESCRIPTION [-p PRIORITY] [-t DATETIME]`, `task DESCRIPTION [-t DATETIME] [-p PRIORITY]` is also acceptable.

3.1. Viewing help : `help`

```
help
```

3.2. Adding a task: `task`

Adds a task to the task manager.

```
[-FILTER] task DESCRIPTION [-p PRIORITY] [-t DATETIME] [-d DURATION] [-r RECURRENCE]
```

Modifications at the back are optional, feel free to use them if you wish.

```
task ST2334 Tutorial
```

If you wish to add a task to the CS2113 list.

```
-CS2113 task Project Meeting -p high -t tomorrow 1400 -r weekly
```

NOTE | Only one word filters are allowed.

3.3. Adding an event: event

Adds an event to the task manager. Events support the same functionalities of task but they have a compulsary date time modification.

```
event Student Life Fair -t tomorrow 1000
```

3.4. Modifications

3.4.1. Priority

Priority tagging for tasks have 3 levels. l , m and h to represent low, medium and high respectively.

```
-p h
```

NOTE | Feel free to type low, medium or high as well, El Duque understands.

3.4.2. Recurring Tasks

Recurring tasks are tasks that will mark itself as undone after the recurrence period. A daily task will be refreshed daily, saving you the trouble of having to recreate the task every day. Recurrence periods supported are daily and weekly.

```
-r daily
```

3.4.3. Duration

Duration assigned to task denotes the amount of time in hours you assigned for the task.

```
-d 5
```

3.4.4. Date and Time

Date and time input follows the following formats: ddMMyy or ddMMyy HHmm.

```
-t 071119 1000
```

El Duque also supports keywords `today`, `tomorrow` and `today/tomorrow HHmm` as well.

```
-t tomorrow 1200
```

| TIP | If a valid but non-existent date is passed in, El Duque will correct it to the closest existent date. e.g `-t 310219` will be corrected to 28-02-19. |

3.5. Listing all tasks : `list`

Shows a list of all tasks in the task manager.

```
list
```

3.5.1. Filtered List

Adding a `-FILTER` infront of the list command will show the respective filtered list.

```
-CS2113 list
```

3.5.2. Time-Based List

List based on certain time frames are also included: `day` and `week`.

```
list day
```

3.5.3. Uncompleted List

List based on completion of task: `undone`

```
list undone
```

3.5.4. Priority List

List based on priority: `priority`

```
list priority
```

3.6. Completing a task : done

Completes an existing task in the address book.

```
done INDEX
```

NOTE | Only valid indexes will work! If you try to perform the command using an index not shown in the current list, the task manager will ask you to type in a valid index!

This following command marks the first task in the list as complete.

```
done 1
```

Filters can be used with this command as well as shown.

```
-CS2113 done 2
```

Will mark the second task in the -CS2113 list as done.

3.7. Locating tasks by description: find

Finds task whose description contain any of the given keywords.

```
[-FILTER] find KEYWORD
```

The task manager supports partial matching in the search operation.

```
find bo
```

Will return tasks that have "bo" in their description.

3.8. Auto assign: autoassign

You can auto assign tasks that have yet to have filters using certain heuristics. This feature relies on filter name matching or having similar description words. For example, if cs filter has many tasks with description contating "programming", auto assigning a task with "programming" in the description will suggest cs as a filter. Having cs in its description will suggest cs as a filter.

It is possible that there can be more than one eligible filter to be recommended. For example, if a task has many similar words with tasks in cooking filter but has cs within its description, both cs and cooking are eligible. El Duque will first ask for all filter(s) that matched based on filter name. If user replies N for all of them, El Duque will then ask for all filter(s) that matched based on similar

description words.

```
autoassign 2
This task's description contains the filter "cs", would you like to assign to it?
(Y/N)
N
Alright then.
The closest filter is "cooking" based on similar words.
Would you like to assign task "prepare food for cs party" to filter "cooking"? (Y/N)
Y
```

NOTE After executing the `autoassign` command, even if the user chose not to assign a filter, it will still cause an addition to our stored commands to be undone. If the user executes `undo` after executing `autoassign`, regardless of whether a task was autoassigned, a message "The autoassign command has been undone" will be shown to the user.

3.9. Editing a task : `edit`

El Duque supports the editing of any parameters of a given task. The parameters to be edited must begin with `-KEYWORD` followed by the edit.

```
[-FILTER] edit INDEX -KEYWORD [MODIFICATION]
```

Editing the description and recurrence of the first task of the list is as follows.

```
edit 1 -desc gym -r daily
```

NOTE The order of the parameters does not matter!

Shown below are the list of keywords and the respective attributes that they edit.

- `-f` : filter
- `-des` / `-desc` / `-descript` : description
- `priority` : priority
- `t` : date and time
- `d` : duration of the task
- `r` : recurrence

3.10. Deleting a task : `delete`

Deletes the specified task from the task manager.

```
[-FILTER] delete INDEX
```

- Deletes the person at the specified INDEX.
- The index refers to the index number shown in the displayed task list.
- The index **must be a positive integer** 1, 2, 3, …

```
list
delete 2
```

Deletes the 2nd task in the task list.

```
-CS2113 list
-CS2113 delete 1
```

Deletes the 1st task in the list shown by the -CS2113 list command.

3.11. Undoing previous command : undo

Restores the task manager to the state before the previous *undoable* command was executed.

```
undo
```

NOTE	Undoable commands: those commands that modify the program's current state (task, event, edit, delete,clear,autoassign, and done).

```
delete 1
list
undo
```

The above undo will reverse the delete 1 command.

```
list
undo
```

This undo fails as there are no undoable commands executed previously.

```
delete 1
edit 2 -p high
undo
undo
```

The first undo will reverse the edit 2 -p high command.
The second undo will reverse the delete 1 command.

```
autoassign 2
This task's description contains the filter "cs", would you like to assign to it?
(Y/N)
N
Alright then.
There are no suitable filters to assign based on similar words.
undo
```

This undo will reverse the autoassign 2 command even though you chose to not assign the chosen filters to it. This is because the program recognises the autoassign command as being executed even though the state of the program has changed.

3.12. Clear: clear

Clears the entire task list.

```
clear
```

This command supports filters as well. Typing the following command will clear the cs2113 task list

```
-CS2113 clear
```

3.13. Pomodoro Timer: pomo

El Duque includes a pomodoro timer to complement the pomodoro workflow. This timer has 3 states, **work**, **short break** and **long break**. A work cycle is followed by a **short break**, every 4th break will be a **long break**.

Pomodoro supports the adding of task to a temporary pomodoro task list for you to keep track of tasks you wish to complete this pomodoro work session.

3.13.1. Starting a pomodoro: start

Starts the timer for the current cycle that it is in. First instance of this would be the work cycle.

```
pomo start
```

El Duque will subsequently update you on the time remaining for the current state every 5 minutes.

NOTE	Once the timer has ended, the next call of `pomo start` will be the next state. If the current state is **work**, the next call will be a **break**.

3.13.2. Stopping a pomodoro: `stop`

Ends the current pomodoro. The pomodoro will automatically update to the next state.+

```
pomo end
```

3.13.3. Viewing the status: `status`

Shows the remaining time and current status of the pomodoro.

```
pomo status
```

3.13.4. Resetting the pomodoro : `reset`

Resets the current state to the previous state. If current state is a **break**, it will be resetted to a **work** state.

```
pomo status
```

3.13.5. Restarts the pomodoro : `restart`

Restarts the entire pomodoro to the first **work** state.

```
pomo restart
```

3.13.6. Adding to pomodoro tasks: `add`

This pomodoro supports adding of tasks to the pomodoro to help you keep track of what you wish to complete in this cycle. Addition of task uses `add` followed by the `INDEX`.

The following command adds the first task of the task list into the pomodoro.

```
pomo add 1
```

3.13.7. Listing pomodoro tasks: `list`

Lists out all the tasks you have added to the pomodoro task list.

```
pomo list
```

3.13.8. Completing a task: `done`

After you have completed a pomodoro task, you can mark it as `done`. This will remove it from the pomodoro task list and update the task as completed in the main task list.

```
pomo done 1
```

3.13.9. Answers to brain teaser: `answer`

El Duque contains a bank of brain teasers that will be shown every time a break has started. We hope this gives you something to take your mind of work. You can use the `answer` keyword to reveal the answer to the brain teaser.

```
pomo answer
```

NOTE	Brain teaser bank is randomised, you might not get the same question the next break. Make sure to type `pomo answer` before you start your next break to view the answer to this current brain teaser!

3.14. Exiting the program : `bye`

Exits the program.

```
bye
```

3.15. Quality of life

To make life easier - the program helps to deal with minor errors in commands as well as provide shortcuts. Specifically, there are 2 checks that the parser will do on the command keyword (the first keyword; not filter/parameter/modification).

Firstly, it checks if the command keyword is off by exactly one character through substitution. If so, you are prompted for correction.

```
lisd
Did you mean list? (Y/N)
```

Secondly, it checks if the command keyword is a shortform. If there is a proper command that begins with the keyword, you will be prompted for correction.

```
l
Did you mean list? (Y/N)
```

Note that if more than one proper command that begins with the keyword, the priority is shown as such:

```
done > delete
event > edit
```

3.16. Saving the data

El Duque data are saved in the hard disk automatically after any command that changes the data. There is no need to save manually.

4. FAQ

Q: How do I transfer my data to another Computer?
A: Install the app in the other computer and overwrite the empty data file it creates with the file that contains the data of your previous El Duque folder.

5. Command Summary

- **Help** `help`
- **Task** `[-FILTER] task DESCRIPTION [-p PRIORITY] [-t DATETIME] [-d DURATION] [-r RECURRENCE]`
  e.g. `-cs task -p h -t tomorrow`
- **Event** `[-FILTER] event DESCRIPTION [-p PRIORITY] [-t DATETIME] [-d DURATION] [-r RECURRENCE]`
  e.g. `-cs event -d 3 -t tomorrow`
- **List** `[-FILTER] list [MODIFICATION]`
  e.g ` list undone`
- **Done** `[-FILTER] done INDEX`
  e.g `-cs done 2`
- **Find** `[-FILTER] find KEYWORD`
  e.g `-cs find tutorial`
- **Autoassign** `autoassign INDEX`
  e.g `autoassign 4`

- **Edit** `[-FILTER] edit [-desc DESCRIPTION] [-p PRIORITY] [-t DATETIME] [-d DURATION] [-r RECURRENCE] [-f FILTER]`
  e.g `-cs edit 2 -f cs2113 -p h`

- **Delete** `[-FILTER] delete INDEX`

- **Clear** `[-FILTER] clear`

- **Undo** `undo`

- **Pomodoro** `pomo [MODIFICATION]`
  e.g `pomo add 1`

- **Bye** `bye`