**Contents**

1: About ........................................................................................................................ 2

2: Quick Start ................................................................................................................ 4

3: Features .................................................................................................................... 6

4: Frequently Asked Questions ................................................................................... 8

5: Command Summary .............................................................................................. 10

USER GUIDE

**1: About**

Do-*er* List is a beginner-friendly desktop program that aids completion of daily tasks. It has a simple and intuitive display which allows you to set up your schedules and to-do lists easily. It sets itself apart from existing similar products in the market through simplicity, by having no extra functions beyond what is necessary for a schedule manager, providing you with a no-frills scheduling experience. Do-*er* List will help you to organize your work and schedules more easily and efficiently.

[Placeholder text to be replaced]

**2: Quick Start**

1. Ensure you have Java version `1.8.0\_60` or later installed in your Computer.

Having any Java 8 version is not enough.

This app will not work with earlier versions of Java 8.

2. Download the latest doerlist.jar from the [releases](https://github.com/TeamPGP/main/releases) tab.

3. Copy the file to the folder you want to use as the home folder for the Do-\*er\*List.

4. Double-click the file to start the app. The GUI should appear in a few seconds.

5. Type the command in the command box and press Enter to execute it.

6. Some example commands you can try:

* add -t Do post-lecture quiz <today->tomorrow> -c CS2103: adds a task called [Do post-lecture quiz] to the Do-*er* List that starts today and ends tomorrow under the category CS2103.
* list CS2103: lists all tasks in CS2103
* delete 1: deletes the 1st task shown in the current list
* exit: exits the application

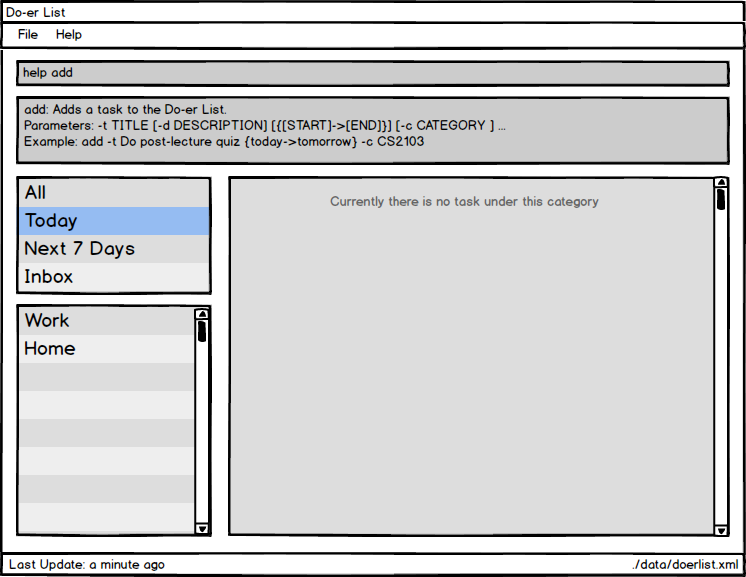
**Features**

**Command Format**

* Words in UPPER\_CASE are the parameters.
* Items in SQUARE\_BRACKETS are optional.
* Items with ... after them can have multiple instances.
* If a command has multiple parameters, the order of parameters doesn't matter.

**Viewing help : help**

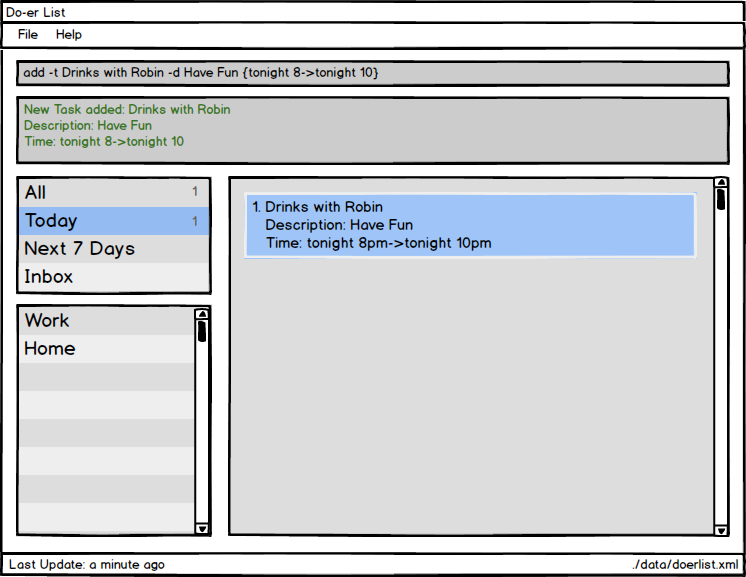
Format: help or help [COMMAND\_NAME]

[](https://github.com/TeamPGP/main/blob/master/docs/images/UI_Guide/help_command.png)

If the COMMAND\_NAME is supplied, it will display the instructions of using that command.  
Help is also shown if you enter an incorrect command e.g. abcd

**Adding a task / an event: add**

Adds a task/event to the Do-*er*List  
Format: add -t TITLE [-d DESCRIPTION] [{[START]->[END]}] [-c CATEGORY] ...

[](https://github.com/TeamPGP/main/blob/master/docs/images/UI_Guide/add_command.png)

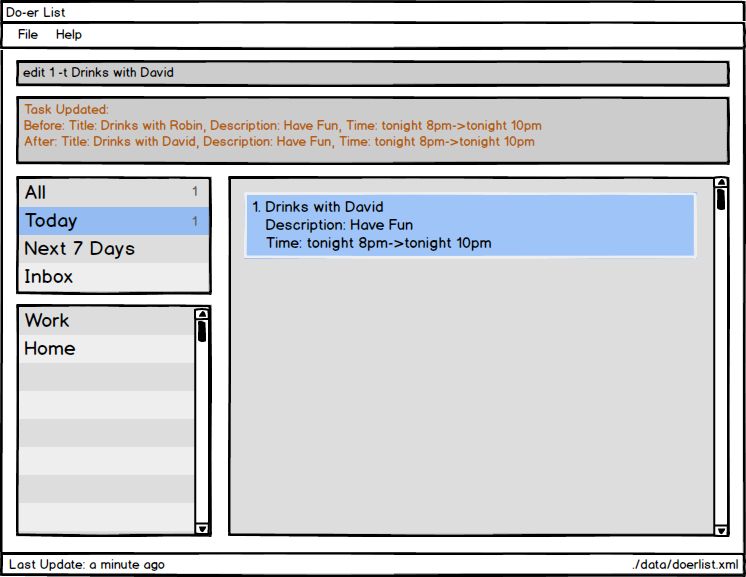
* Task can have any number of categories (including 0)
* The START or END parameter can be in natural language (next X hours, today, tomorrow, next X days, next week, next month) or in standard format “2016-10-3 10:00”
* If the START date is missing, the Do*er*-List set it to today by default.
* If the there is not START->END parameters or END date is missing, the Do*er*-list will create task without start date and end date and move it to Inbox build-in category.

Examples:

* add -t Do post-lecture quiz {today->tomorrow} -c CS2103
* add -t Do CA1 -d Oral Presentation {->next 2 days} -c CS2101
* add {2016-10-4 10:00->2016-10-4 12:00} -t Take lecture -c CS2102

**Editing an event : edit**

Edit an existing task / event in the Do-*er*List  
Format: edit INDEX [-n TITLE] [-d DESCRIPTION] [{[START]->[END]}] [-c CATEGORY] ...

[](https://github.com/TeamPGP/main/blob/master/docs/images/UI_Guide/edit_command.png)

Edit an existing task by calling its index. The event's title, description, start date, end date and category can be edited.

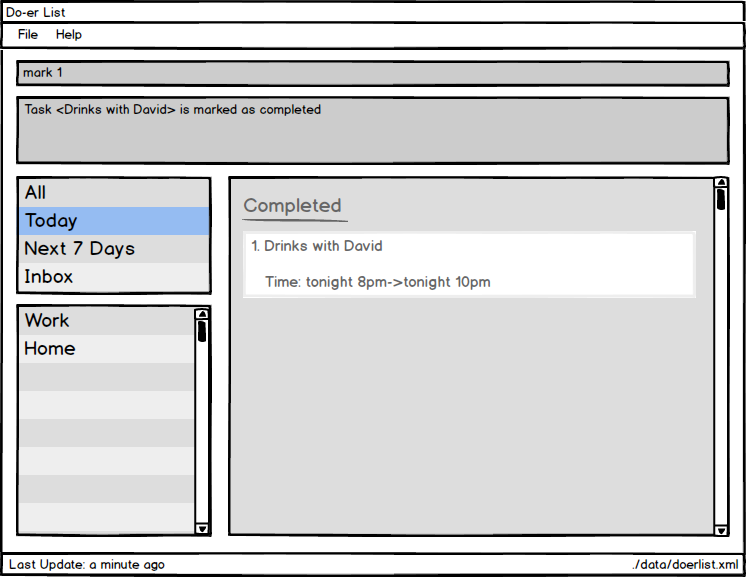
Examples:

* edit 1 -t Do ST2334 quiz -c ST2334
* edit 2 {->next 5 days}

Attributes that aren't supplied will not be updated

**Mark task as done : mark**

Marks a certain task as done in the Do*er*-list.  
Format: mark TASK\_NUMBER

[](https://github.com/TeamPGP/main/blob/master/docs/images/UI_Guide/mark_command.png)

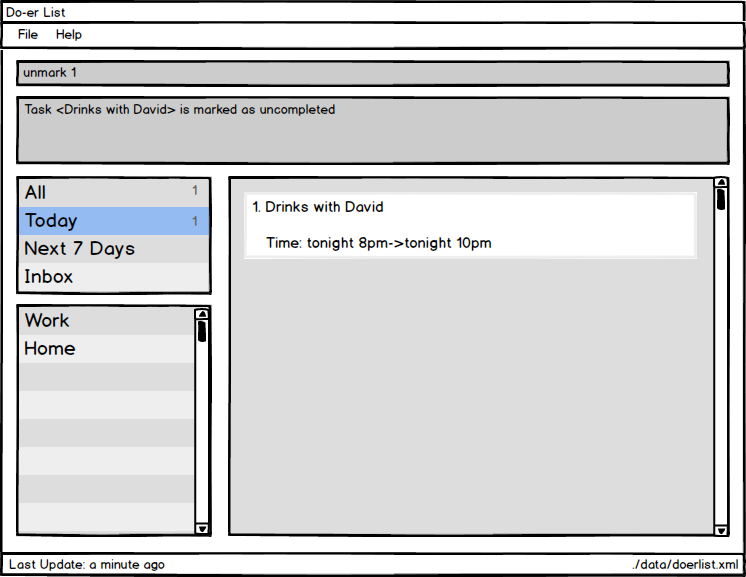
Mark task TASK\_NUMBER as done. The task must exist in the Do*er*-list.

Examples:

* mark 5   
  Returns task number 5 as done.

**Unmark task as done : unmark**

Marks a certain task as undone in the Do*er*-list.  
Format: unmark TASK\_NUMBER

[](https://github.com/TeamPGP/main/blob/master/docs/images/UI_Guide/unmark_command.png)

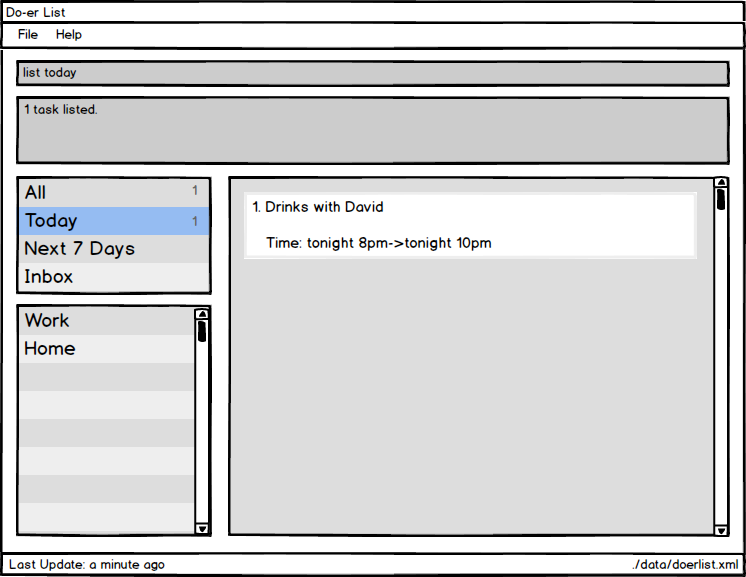
Mark task TASK\_NUMBER as undone. The task must exist in the Do*er*-list.

Examples:

* unmark 5   
  Returns task number 5 as undone.

**Listing tasks in a certain category : list**

Shows a list of all tasks in the Do*er*-list under the specific category.  
Format: list [CATEGORY]

[](https://github.com/TeamPGP/main/blob/master/docs/images/UI_Guide/list_command.png)

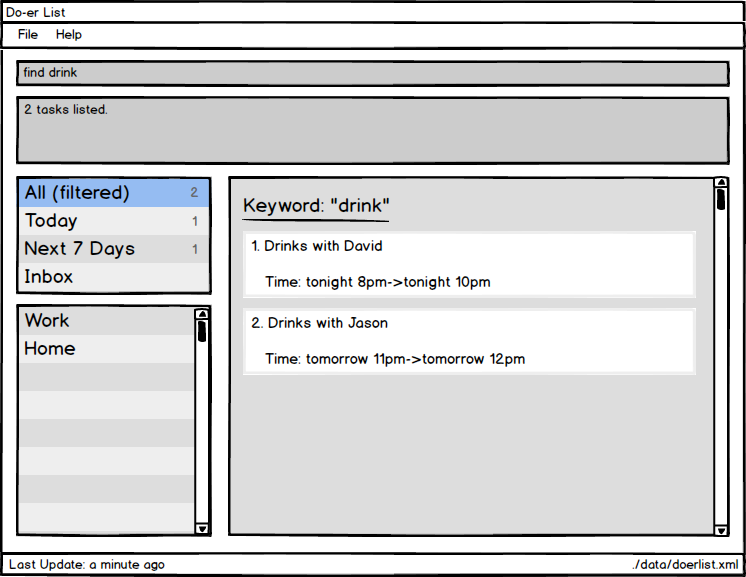
If the CATEGORY parameter is not supplied, then list all tasks.

Examples:

* list
* list CS2101

**Finding all tasks / events containing any keyword in their name: find**

Finds tasks / events whose names contain any of the given keywords.  
Format: find KEYWORD [MORE\_KEYWORDS]

[](https://github.com/TeamPGP/main/blob/master/docs/images/UI_Guide/find_command.png)

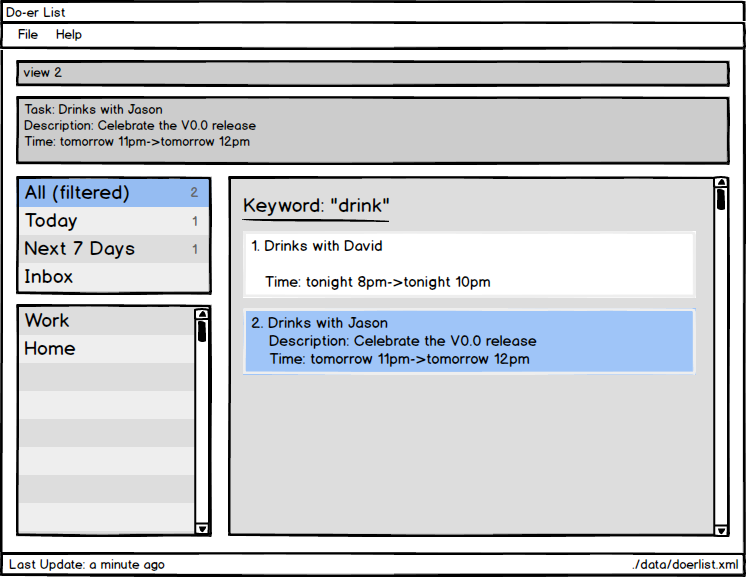
* The search is not case sensitive. e.g lecture will match LecTure
* The order of the keywords does not matter. e.g. go to lecture will match Lecture go to
* Title and Description is searched.
* All data in the Do-*er*List matching at least one keyword will be returned (i.e. OR search). e.g. lecture will matchhave lecture

Examples:

* find david  
  Returns Drinks with David

**View a task : view**

Views the task identified by the index number used in the last task listing.  
Format: view INDEX

[](https://github.com/TeamPGP/main/blob/master/docs/images/UI_Guide/view_command.png)

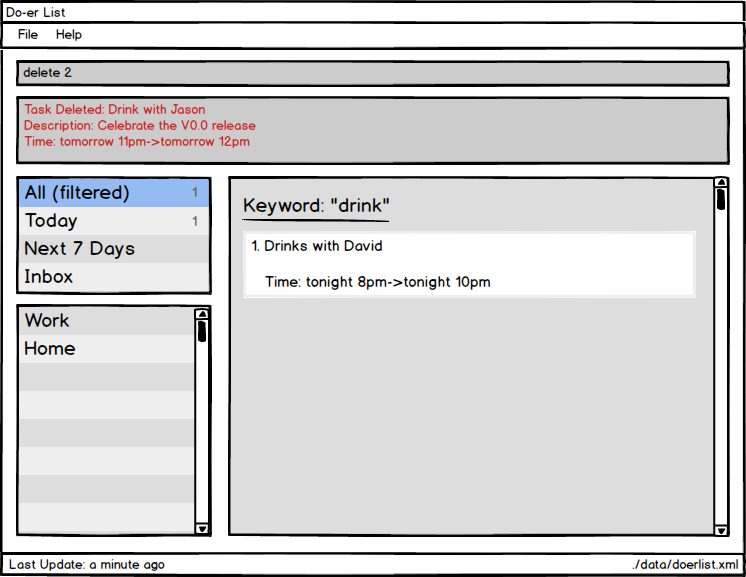
* Views the details of the task at the specified INDEX.
* The index refers to the index number shown in the most recent listing.  
  The index **must be a positive integer** 1, 2, 3, ...

Examples:

* list  
  view 2  
  Views the 2nd task in the Do*er*-list.
* find David   
  view 1  
  Views the 1st task in the results of the find command.

**Deleting a task / an event : delete**

Deletes the specified task / event from the Do-*er*List. Irreversible.  
Format: delete INDEX

[](https://github.com/TeamPGP/main/blob/master/docs/images/UI_Guide/delete_command.png)

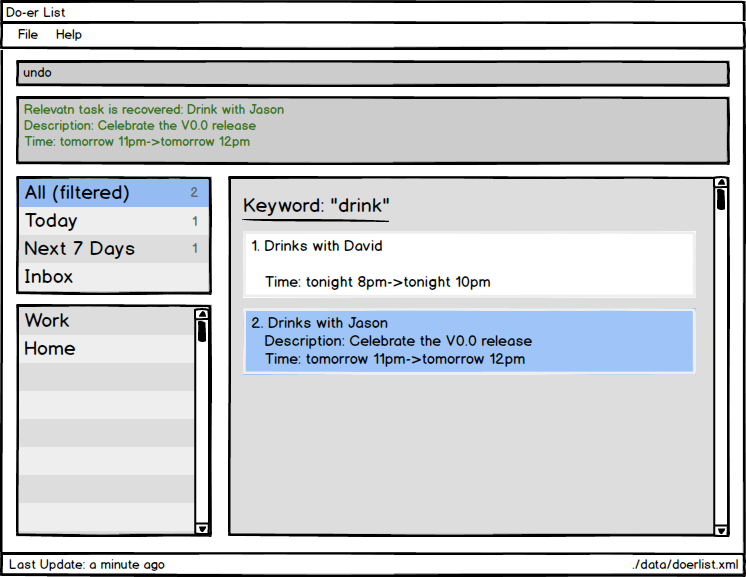
* Deletes the task / event at the specified INDEX.
* The index refers to the index number shown in the most recent listing.  
  The index **must be a positive integer** 1, 2, 3, ...

Examples:

* list  
  delete 2  
  Deletes the 2nd task / event in the Do-*er*List.
* find David  
  delete 1  
  Deletes the 1st task / event in the results of the find command.

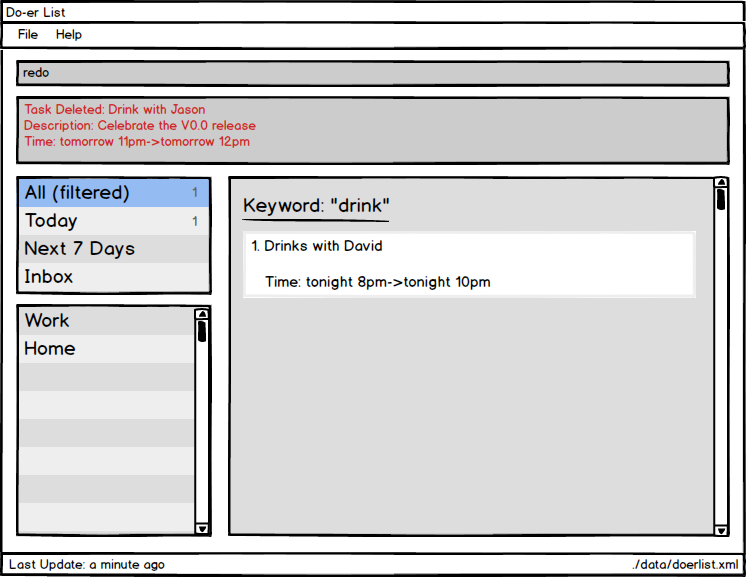
**Undo the most recent operation : undo**

Undo the most recent operation which modify the data in the Do*er*-list  
Format: undo

[](https://github.com/TeamPGP/main/blob/master/docs/images/UI_Guide/undo_command.png)

**Redo the most recent operation : redo**

Redo the most recent undo  
Format: redo

[](https://github.com/TeamPGP/main/blob/master/docs/images/UI_Guide/redo_command.png)

**Find all tasks due : taskdue**

Finds all tasks due on and before the date specified in the Do*er*-list.  
Format: taskdue END\_DATE

[](https://github.com/TeamPGP/main/blob/master/docs/images/UI_Guide/taskdue_command.png)

Finds all tasks due on and before END\_DATE.  
The date can be in natural language (E.g. next X hours, today, tomorrow, next X days, next week, next month) or in standard format (E.g. 2016-10-3 10:00)

Examples:

* taskdue tomorrow

**Exiting the program : exit**

Exits the program.  
Format: exit

**Saving the data**

The Do*er*-list data are saved in the hard disk automatically after any command that changes the data.  
There is no need to save manually.