Do-er List

File   Help

Enter your command here

Welcome to Do-er List!
Need help? Try type `help` to see the usage of different command!

All
Today
Next 7 Days
Inbox

Work
Home

Currently there is no task under this category

Last Update: a minute ago   ./data/doerlist.xml

USER GUIDE

TEAM PGP

ABSTRACT

Do-er List is a user-friendly desktop application that helps you in planning and completing your daily tasks. Be it from a large corporate event to a recurring task such as washing your laundry every now and then, Do-er List is here to solve these issues for you!

CS2101/CS2103T

Software Engineering & Effective Communication for Computing Professionals

Contents

USER GUIDE

1: About

Living in the modern and fast-paced world, we are constantly overwhelmed with errands every day. Many face the problem of using traditional methods of time management which are irrelevant and ineffective. That is where Do-*er* List steps in.

Do-*er* List is a task manager that is designed for students and office workers. It is a beginner-friendly desktop program that aids you in the planning and completion of your daily tasks. It does not matter if you are planning a big birthday surprise event or a recurring task of handling the laundry every now and then, Do-*er* List is here to resolve your problems.

This user guide aims to allow any user to seamlessly use our product as we intend to. Just follow the instructions as stated and you will get the results you desire.

Eager and excited? Then let's proceed!

2: Overview

Below is an overview of the various terms and components Do-*er* List's minimalistic and easy-to-use interface.
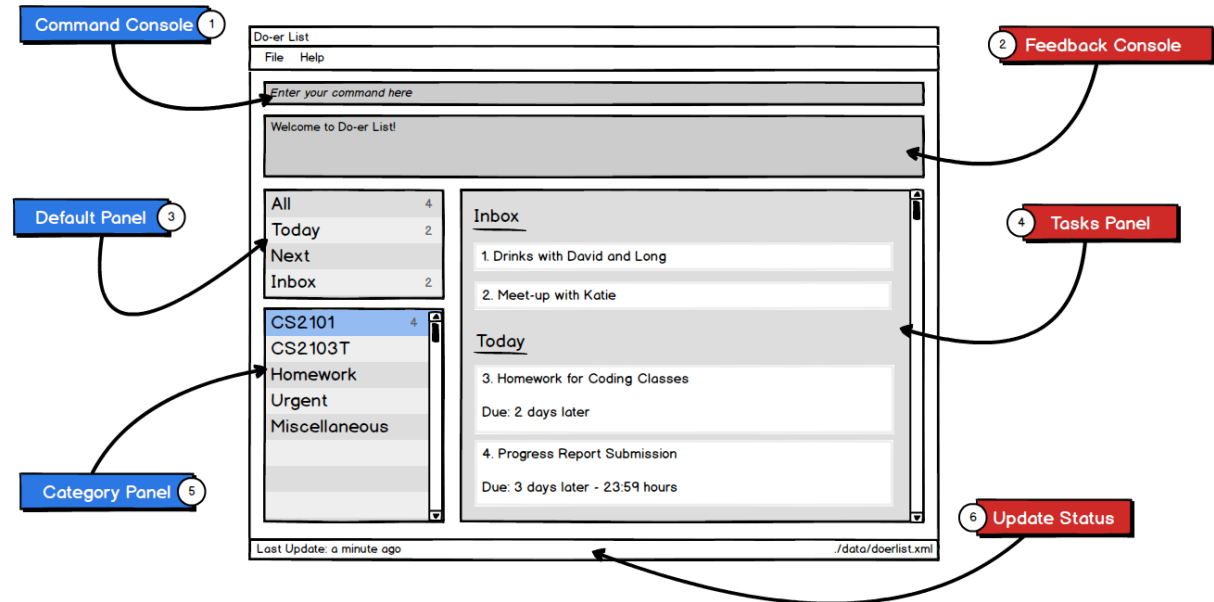

Figure 1: Graphic User Interface

Labels	Description
1	Type your commands in the **Command Console** to execute the desired commands.
2	**Feedback Console** shows if your command is properly executed.
3	View your default categories in the **Default Panel**.
4	**Tasks Panel** displays all the tasks in a panel.
5	**Category Panel** shows all the custom categories that you have created.
6	View your last update and file storage in **Update Status.**

3: Getting Started

Do-*er* List makes the process of adding, editing or deleting tasks a seamless process. Long gone are the days when you must type in long and complicated commands or endlessly clicking buttons to get what you desire.

All commands have this standard format:

```
>> Command required_fields [optional_fields] ...
```

All commands start with a command word, followed by fields that are replaced by your inputs. Fields in the square bracket "[" and "]" are optional, you have a choice of not including them when entering your command.
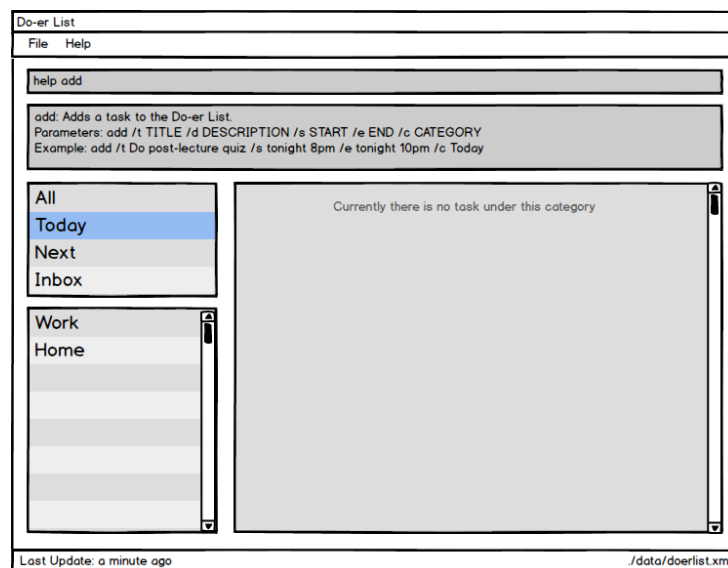

Figure 2: An example of a Graphic User Interface

3.1.1: Launch

Start your day right by launching Do-*er* List to view what you need to do for the day! A simple double-click on the app icon will open the elegant and beautiful user interface in a few seconds.

From there, you can type the command in the user box and press ENTER to execute it. The results will shortly appear.
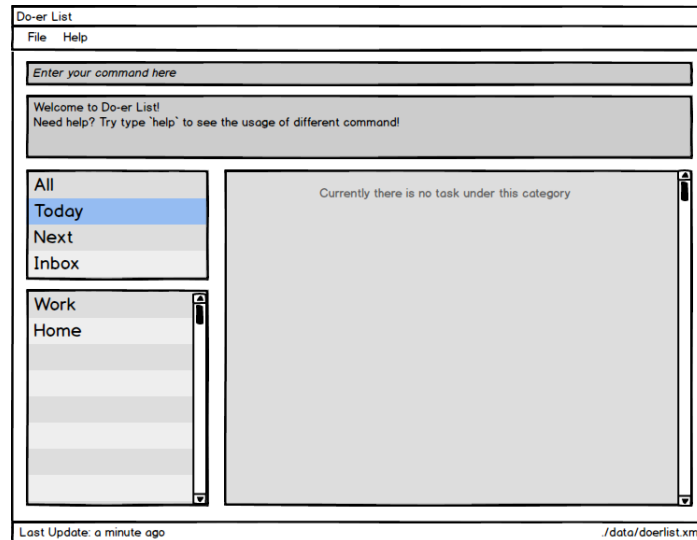

Figure 3: Our Graphic User Interface

3.1.2: Your friendly guide

If you forget how to use Do-*er* List, simply enter this command in the command console.

```
>> help
```

This will display all the available commands and the way to use them properly.

If you forget a certain command, you can always specify it like this. For this example, we will be using the "add" command.

```
>> help add
```

The proper usage format of the "add" command will be shown. This is applicable to all commands.

If you carelessly typed a command that is not properly formatted, its proper usage format will also show up.
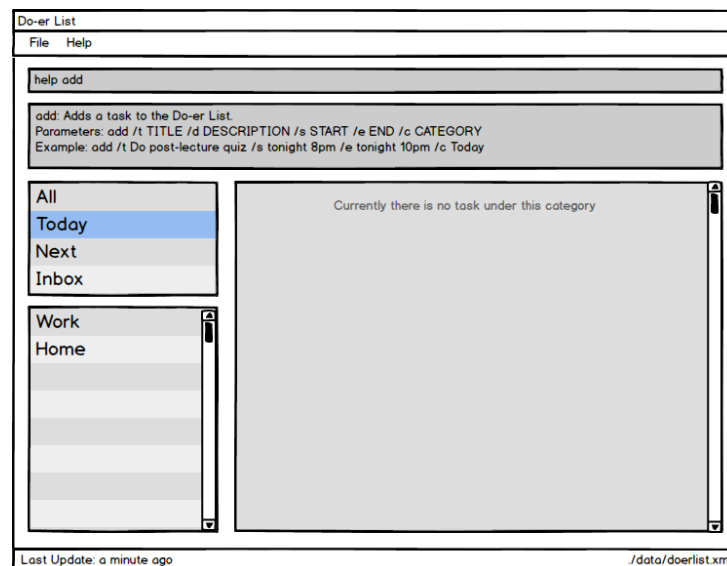

Figure 3: The Graphic User Interface when the "help add" command is used

3.1.3: Adding Items into Do-*er* List

This morning your friend, Robin, asked you out for a drinking session at night and you would like to mark it down in your Do-*er* List. You can do that by typing the following in command console:

```
>> add /t Drinks with Robin /d Have Fun /s tonight /e tonight 10pm /c
Today
```
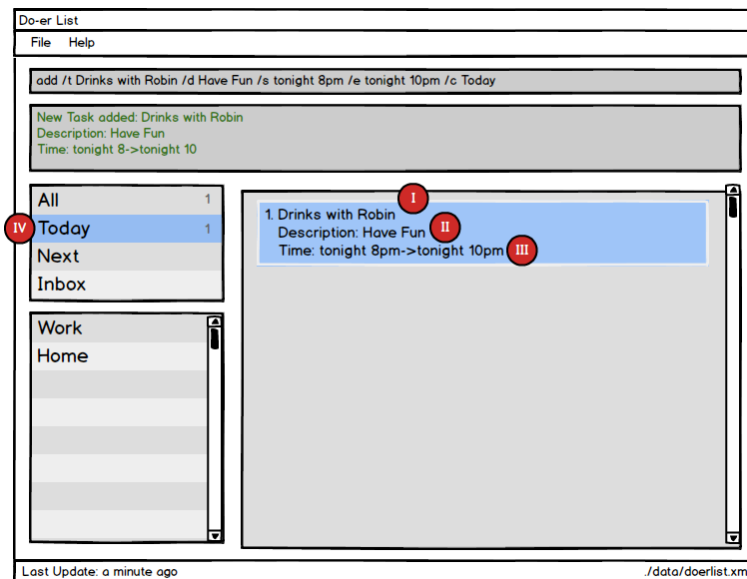
The event will then be added to the Do-*er* List.

Figure 4: Using the "add" command

As you can see from the above figure:

    I.   "Drinks with Robin" is the **title**.

    II.   "Have Fun" is the **description**.

    III.   "tonight 8pm -> tonight 10pm" is the **time**.

    IV.   The category is placed in **Today**.

General Usage:

```
>> add /t TITLE /d DESCRIPTION /s START /e END /c CATEGORY
```

*Note: This just a general summary of how the command is used and may not be used in the following way. On how to use it, please look at some of our samples below.

Understanding the `"add"` command:
- `[/t TITLE]`
  - Creates a title for your task or event.
  - The only compulsory section when using the "add" command, the rest are all optional.
- `[/d DESCRIPTION]`
  - To elaborate more on the following task or event.

- **[/s *START*] & [/e *END*]**
    - ○ Your dates should be in the following format: "`yyyy-mm-dd`" and "`hh:mm`".
        - ▪ The time specified should be in a 24-hour clock format.
    - ○ Supports several layman words such as, regardless of letter-case:
        - ▪ next X hours / days / weeks / months
            - • (*"X" can be any number:1, 2, 3...*)
        - ▪ today
        - ▪ tomorrow
        - ▪ next week / month

- **[/c *CATEGORY*]**
    - ○ A task can have one or more categories.
    - ○ You can choose to add more categories by repeating `/c CATEGORY` at the end of your command input.

Some samples on how to use the **"add"** command:

```
>> add /t Weekly Laundry /s 2016-11-23 21:00 /c Chores
```

```
>> add /t Daily Exercise and Workout!
```

```
>> add /t Call Mum in Hanoi /d Limit chat timing for overseas charges
/s tomorrow 8pm /e tomorrow 10pm /c Optional
```

3.1.4: Editing Items on Do-*er* List

Robin could not make it tonight due to personal reasons and had asked David to attend the drinking session with you instead. As such, you need to edit your schedule to ensure that the changes are recorded. You can type the following in the command console:

```
>> edit 1 /t Drinks with David
```

The task title changes from "Drinks with Robin" to "Drinks with David".

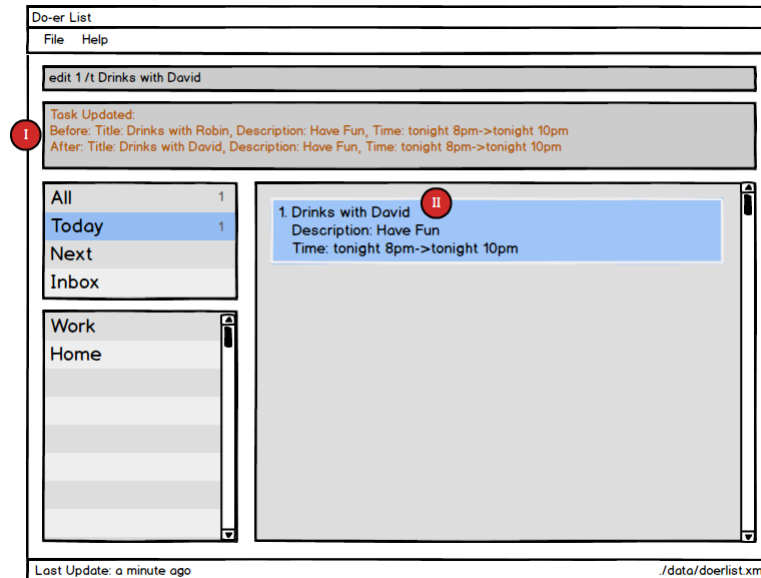

Figure 5: Using the "edit" command

In reference to Figure 5:

    I.   The feedback console shows the previous entry as well as your edited entry together to allow you to compare the changes between the two entries.

    II.   The Tasks Panel reflects the changes in the entry you have edited.

    III.

General Usage:

```
>> edit INDEX /t TITLE /d DESCRIPTION /s START /e END /c CATEGORY
```

*Note: This just a general summary of how the command is used and may not be used in the following way. On how to use it, please look at some of our samples below.

Understanding the **"edit"** command:
- **[/t *TITLE*]**
    - Edits title for your task or event.
- **[/d *DESCRIPTION*]**
    - To edit the elaboration on your task or event.
- **[/s *START*] & [/e *END*]**
    - Editing of the dates should be in the following format: "`yyyy-mm-dd`" and "`hh:mm`".
        - The time specified should be in a 24-hour clock format

- Supports several layman words such as, regardless of letter-case:
    - next X hours / days / weeks / months
        - *("X" can be any number:1, 2, 3…)*
    - today
    - tomorrow
    - next week / month
- **[/c *CATEGORY*]**
    - Editing a task's category.
    - You can choose to add more categories by repeating `/c CATEGORY` at the end of your command input.

Some samples on how to use the **"edit"** command:

```
>> edit 2 /t Daily Laundry /c Chores /c Daily
```

```
>> edit 3 /c Do Homework
```

```
>> edit 3 /s tomorrow 23 00
```

3.1.5: Marking Out Tasks on Do-*er* List

You have just come back home after the drinking session with David. As the task is already over, you want to mark it as "done". You can do this by entering the following command:

```
>> mark 1
```

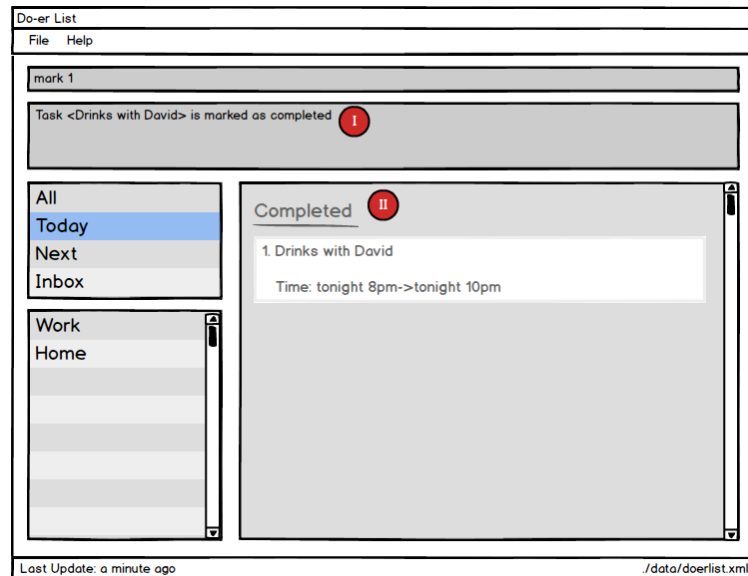And the indicated task will be marked as completed.

Figure 6: Using the "mark" command

In reference to Figure 6:

1. The feedback console displays that your entry has been successfully marked.

2. The Tasks Panel displays a new category "Completed" and greys out the entry that has been marked.

General Usage:

```
>> mark INDEX
```

Understanding the **"mark"** command:
- **[*INDEX*]**
  - Selects the given index that is displayed.
  - Do be mindful that viewing different categories might change the indexes of the same entries displayed.

A sample on how to use the **"mark"** command:

```
>> mark 3
```

3.1.6: Unmarking a Completed Task

You had done your tutorials and marked it as completed. However, after going through your tutorials, you realised that you had missed out quite a few questions. You decide to unmark your task, which can be done by entering the following command:

```
>> unmark 1
```

And the indicated completed task will be moved out of the "Complete" category.
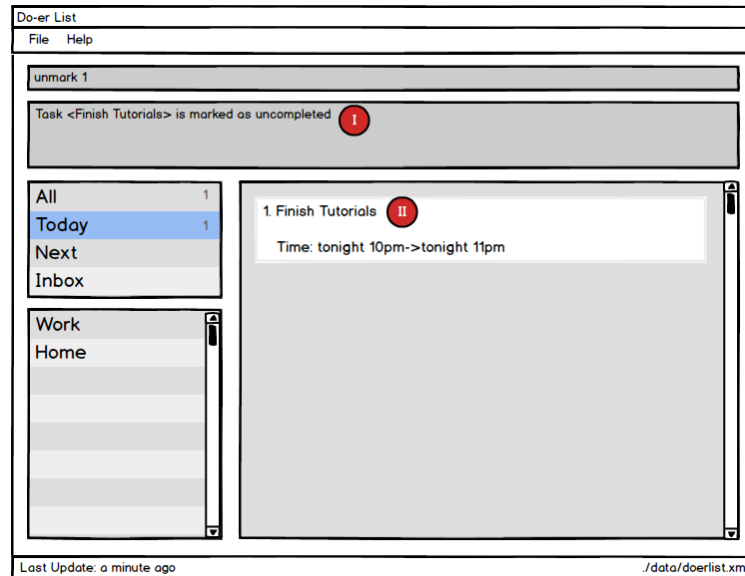

Figure 7: Using the "unmark" command

In reference to the Figure 7:

I. The feedback console displays that your completed task has been changed into uncompleted.

II. The Tasks Panel removes the specified task from "**Completed**" category into the "**Today**" category where it once was.

General Usage:

```
>> unmark INDEX
```

Understanding the **"unmark"** command:
- **[INDEX]**
  - Selects the given index that is displayed.
  - Be forewarned that there must be at least a task in the "Complete" section for the **"unmark"** command to work.
    - Overstating the **[INDEX]** will also not work on this program.
      - E.g: **"unmark 5"** on a list with only 3 completed tasks.

A sample on how to use the **"unmark"** command:

```
>> unmark 5
```

3.1.7: List Tasks

You want to check the tasks under a specific category or date. For example:

```
>> list today
```

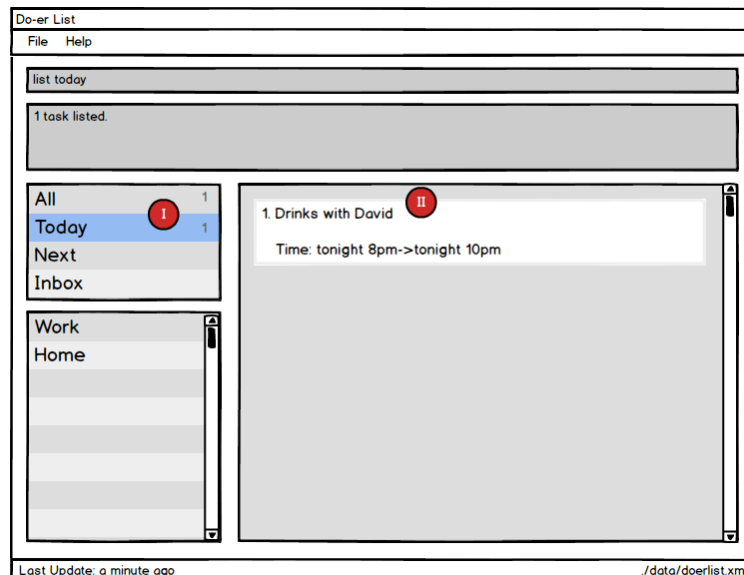Do-*er* List will display all the tasks needed to be done by today.

Figure 8: Using the "list" command

In reference to the figure 8:

I. The "**Today**" category is selected.

II. Tasks that are going to happen "**Today**" are displayed.\

General Usage:

```
>> list [CATEGORY_NAME]
>> list [DATE]
```

Understanding the **"list"** command:
- **[CATEGORY_NAME]**
  - o Tasks under the **[CATEGORY_NAME]** will be displayed. You can refer to the instructions of **"add"** command to see how you can add categories with your favourite names.
  - o Do-*er* List will alert to you if it cannot find category with **[CATEGORY_NAME]**.
  - o The case of **[CATEGORY_NAME]** does not matter.
    - ▪ E.g: **"list WORK"** and **"list Work"** will give you the same results.
  - o Some built-in categories is available for you to use.
    - ▪ **All**, **Today**, **Next**, **Inbox** and **Complete** are the built-in categories.

▪ E.g: **"list All"** will display all the tasks in the Do-*er* List.

Continued on the next page.

- **[*DATE*]**
  - o Tasks that are going to happen in the **[*DATE*]** will be displayed.
  - o The **[*DATE*]** argument can be either in standard format "**yyyy-mm-dd**" or in natural language to express date. For list of natural language supported in Do-*er* List, please refer to the instructions of "**add**" command for more details.
- If you don't supply **[*CATEGORY_NAME*]** or **[*DATE*]** in your command, all tasks will be shown. (This is the same as using **"list All"**)

A sample on how to use the **"list"** command:

```
>> list
```

```
>> list Work
```

```
>> list 2016-10-13
```

```
>> list Tomorrow
```

3.1.8: Find Tasks

You remembered that you have a drinking session with someone but you could not recall his name. You decide to find that task via the application. You entered:

```
>> find drinks
```

Do-*er* List will display the search results and you find that you have drinking sessions with David or Jason.
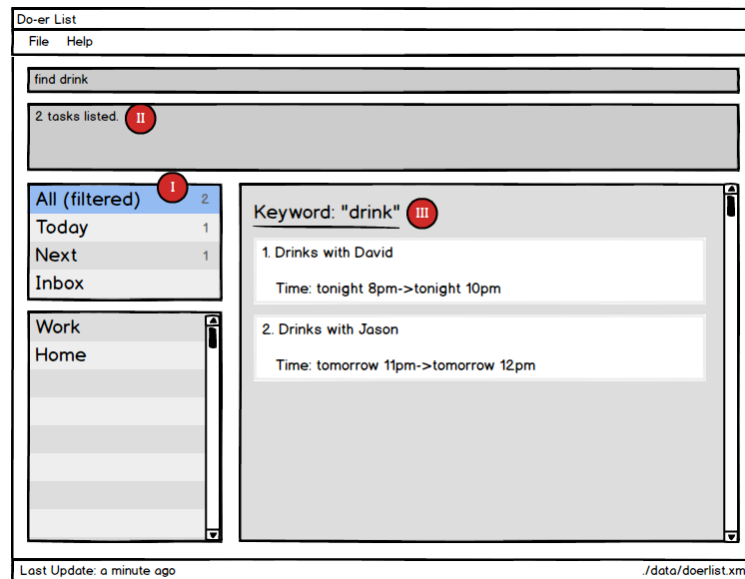

Figure 9: Using the "find" command

In reference to the above figure:

I.   The "`All`" category name has been updated to "`All(filtered)`" due to the search.

II.   Summary of the searched results is displayed.

III.   Tasks containing `[KEYWORD]` are shown.

General Usage:

```
>> find KEYWORD [MORE_KEYWORD]
```

*Note: This just a general summary of how the command is used and may not be used in the following way. On how to use it, please take a look at some of our samples below.

Continued on the next page.
Understanding the "`find`" command:
- `[KEYWORD]`

- o   Do-*er* List will examine task's title and description. If one of the fields contains **[*KEYWORD*]**, the task will be included in the results.
- o   The search is not case sensitive.
  - ▪   E.g: "**list LECTURE**" and "**list lecture**" will give you the same result
- **[*MORE_KEYWORD*]**
  - o   You can supply more **[*KEYWORD*]** in the "**find**" command, Do-*er* List will search the **[*KEYWORD*]** separately and combine the results.

A sample on how to use the "**find**" command:

```
>> find lecture
```

```
>> find math programming
```

3.1.9: View Tasks

You added some descriptions when you created the task "Drinks with Jason". Since you are going for a drinking session with Jason tomorrow, you want to know the details of the task. You type the following into the command console:

```
>> view 2
```

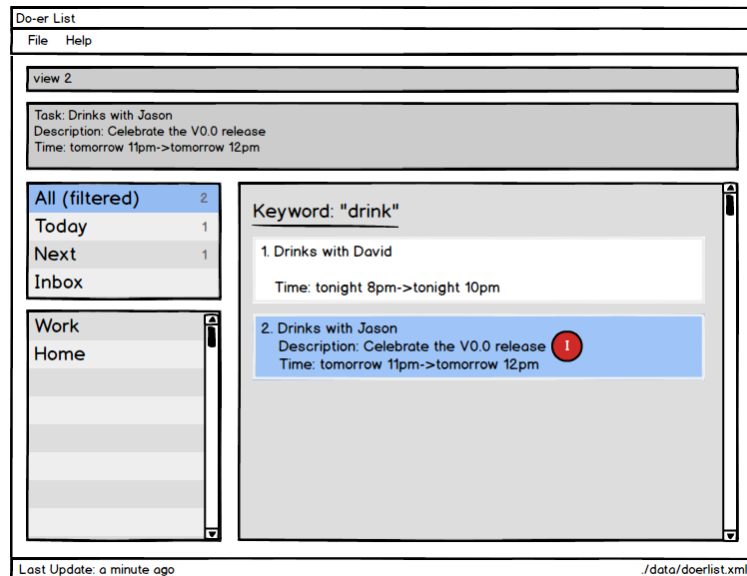Do-*er* List will display the second task in the **last selected** list.

Figure 10: Using the "view" command

In reference to Figure 10:

I.   The description of a certain task is shown.

General Usage:

```
>> view INDEX
```

Understanding the "`view`" command:
- **[INDEX]**
    - Selects the given index that is displayed. Description of the task will be shown.

A sample on how to use the "`view`" command:

```
>> view 1
```

3.1.10: Delete Tasks

You just heard from Jason that the drinking session is cancelled and you want to delete the task as you do not need to keep track of it. You enter the following in the command console:

```
>> delete 2
```

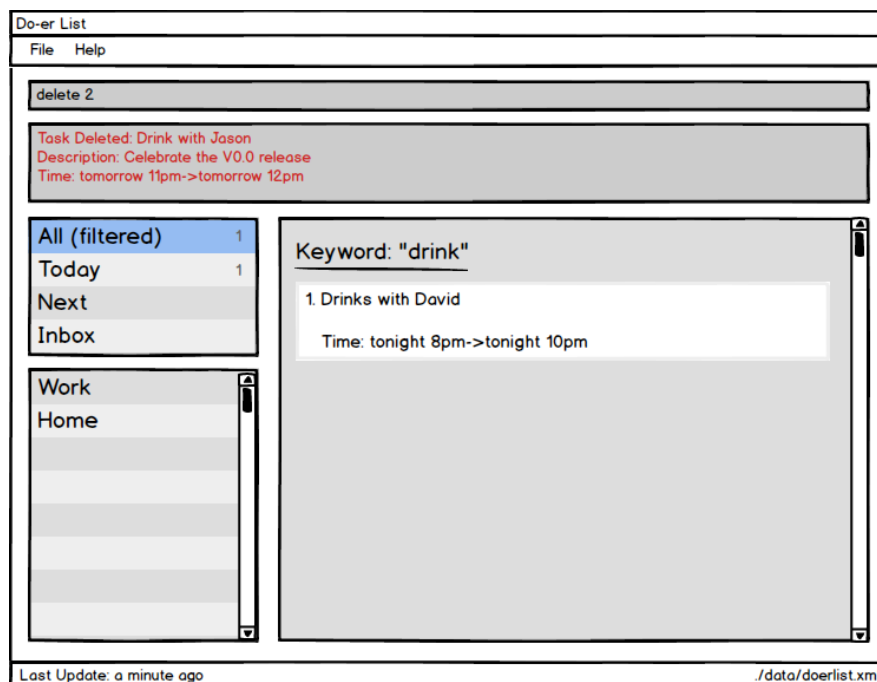Do-*er* List will delete the second task in the **last selected** list.

Figure 11: Using the "delete" command

In reference to Figure 11:

I. The summary of the deleted task is shown in red.

General Usage:

```
>> delete INDEX
```

Understanding the "delete" command:
- **[INDEX]**
  - Delete the task with given **[INDEX]** as shown in Do-*er* List.

A sample on how to use the "delete" command:

```
>> delete 2
```

3.1.11: Undo Operation

Jason suddenly tells you he can make it tonight. As such, you need to recover the task you had just deleted. You can use the "undo" command to undo the most recent operation that caused a change in data.

```
>> undo
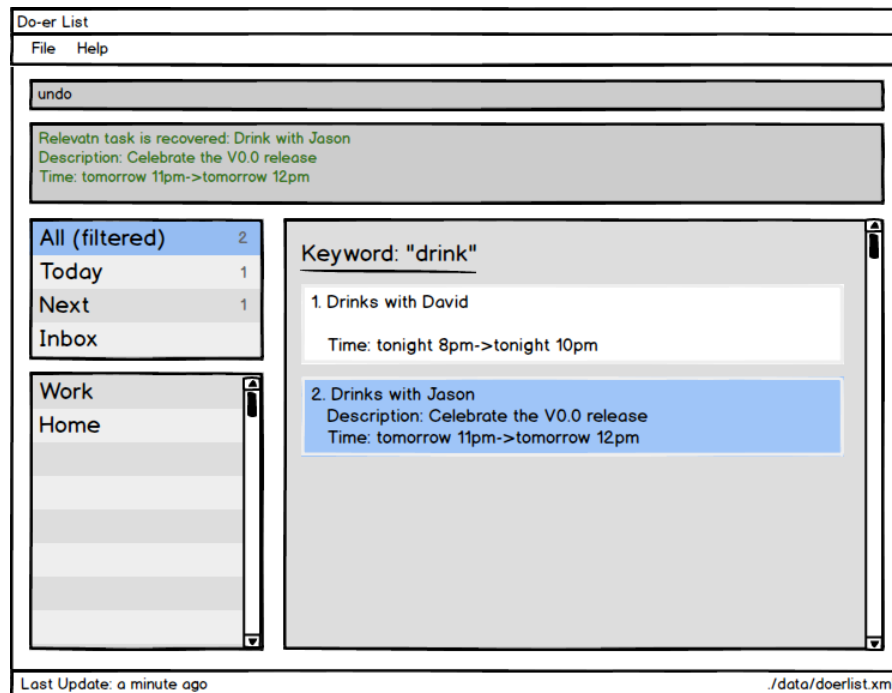```

Do-*er* List will revert back to its previous database.



Figure 12: Using the "undo" command

General Usage:

```
>> undo
```

Understanding the "undo" command:
- Undo the most recent operation that modifies the data in the Do-*er* List.

A sample on how to use the "undo" command:

```
>> undo
```

3.1.12: Redo Operation

Jason just told you he is unable to attend the drinking session again. Hence you need to redo the operation. Simply type this command:

```
>> redo
```
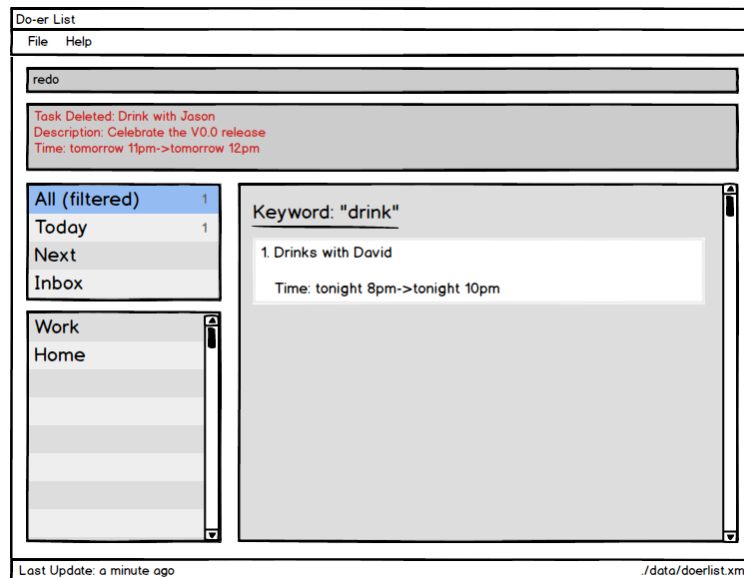
Do-*er* List will revert to its modified database.

Figure 13: Using the "redo" command

General Usage:

```
>> redo
```

Understanding the "`redo`" command:
  • Redo the operation caused by using undo command.

A sample on how to use the "`redo`" command:

```
>> redo
```

3.1.13: Find All Due Task

After the drinking session with Jason, you need to get back to your work. You want to check all of the tasks due by tomorrow. To do this, simply type this command in the command console:

```
>> taskdue tomorrow
```

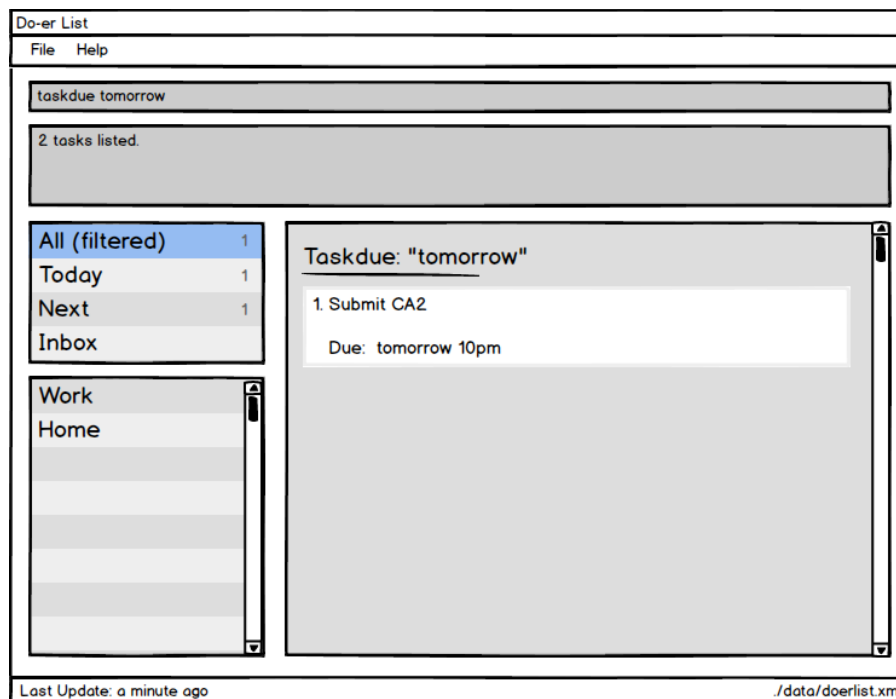Do-*er* List will display all the tasks that are due by tomorrow.

Figure 14: Using the "taskdue" command

General Usage:

```
>> taskdue tomorrow
```

*Note: This just a general summary of how the command is used and may not be used in the following way. On how to use it, please look at some of our samples below.

Understanding the **"taskdue"** command:
- This command will list all tasks due by *END*.

- *END*
  o Your dates should be in the following format: "**yyyy-mm-dd**" and "**hh:mm**".
    ▪ The time specified should be in a 24-hour clock format
  o Supports several layman words such as, regardless of letter-case:
    ▪ next X hours / days / weeks / months

      • *("X" can be any number: 1, 2, 3...)*

- today

- tomorrow

- next week / month

Some samples on how to use the **"taskdue"** command:

```
>> taskdue today
```

```
>> taskdue next 5 hours
```

```
>> taskdue 2016-11-11 21:03
```

*Note: This command lists all tasks by **END** while the list command lists all tasks for the day.

3.1.14: Exiting the Program

You can exit the program by typing the following command in the command console:

```
>> exit
```

3.2: 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.

4: Frequently Asked Questions

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 Do-*er* List folder.

Q: Where is the save button for me to save my schedule in this program?
A: Your data are saved in the hard disk automatically after any command that changes the data as aforementioned in the guide. There is no need for you to save it manually.

Q: Running "doerlist.jar" gives and error or does not seem to work.
A: You will need to install the latest version of Java. Refer to the installation guide here.

5: Command Summary

Command	Format
Help	help [COMMAND]
Add	add -t TITLE [-d DESCRIPTION] [{[START]->[END]}] [-c CATEGORY] ...
Edit	edit INDEX [-t TITLE] [-d DESCRIPTION] [{[START]->[END]}] [-c CATEGORY] ...
Mark Done	mark TASK_NUMBER
Mark Undone	unmark TASK_NUMBER
List	list [CATEGORY]
Find	find KEYWORD [MORE_KEYWORDS]
View	view INDEX
Delete	delete INDEX
Undo	undo
Redo	redo
Task Due	taskdue END_DATE
Exit	exit