

+Work - User Guide

1. Introduction	1
2. Quick Start	1
3. Features	2
3.1. Universal Commands	4
3.2. Member-related Commands	5
3.3. Task-related Commands	18
3.4. Calendar Commands (Coming in v1.3)	22
3.5. Inventory Commands	24
3.6. Statistics Commands	30
3.7. Multiline Commands	32
3.8. Settings Commands	36
3.9. Saving the data	38
3.10. Command Line Autocomplete	38
4. FAQ	38
5. Command Summary	39
6. Upcoming Features	40
6.1. Cloud integration [coming in v2.0]	40
6.2. Customisable user profiles [coming in v2.0]	40

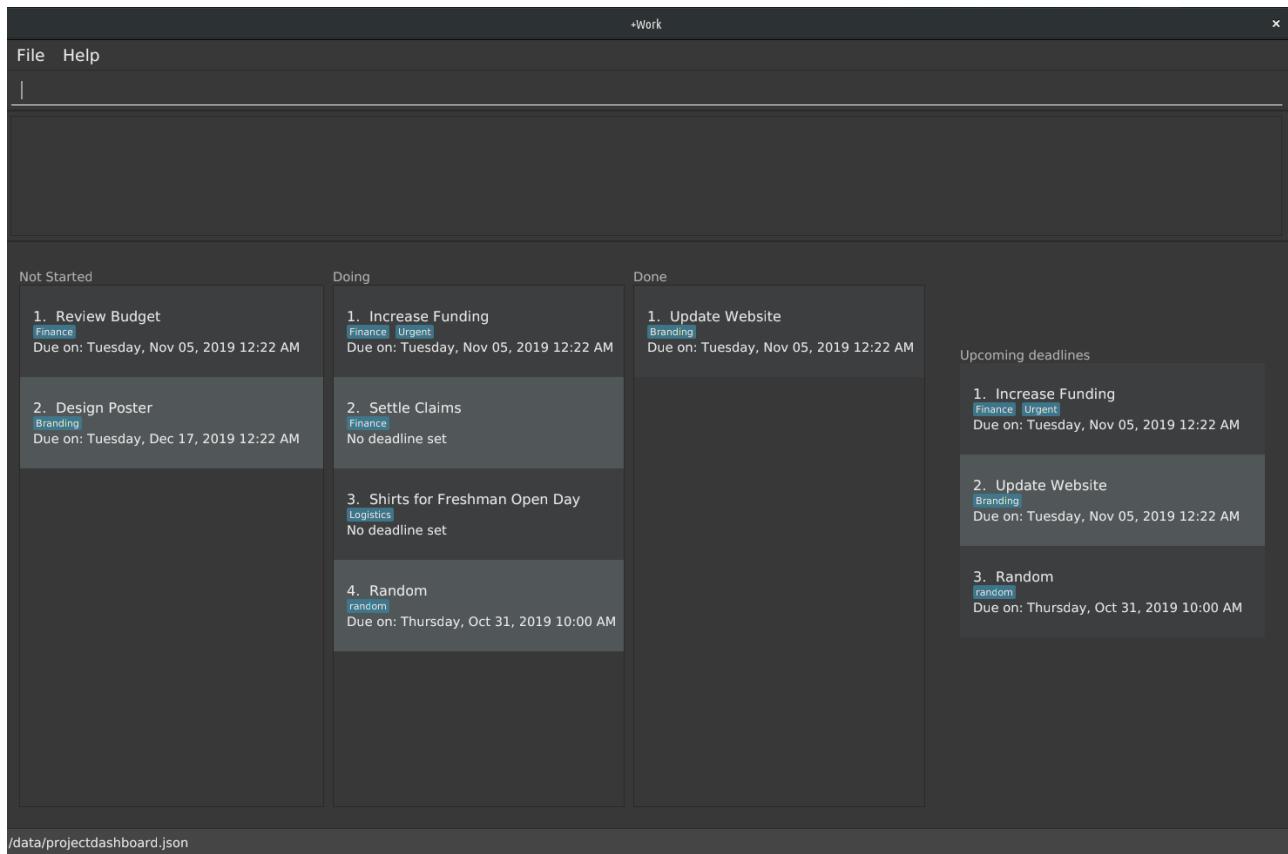
By: [CS2103T-T12-1](#) Since: [Sept 2019](#) Licence: [MIT](#)

1. Introduction

+Work is a project management tool that allows students to successfully achieve your objectives for their project. It is aimed at NUS project leaders who are in charge of a small project group of less than 20 people. +Work is a one-stop application to help you manage your members, meeting times, task allocations as well as equipment purchased. Take full advantage of our Command Line Interface (CLI) while still having the all the benefits of a Graphical User Interface (GUI). Happy working!

2. Quick Start

1. Ensure you have Java [11](#) or above installed in your Computer.
2. Download the latest [pluswork.jar](#) [here](#).
3. Copy the file to the folder you want to use as the home folder for your +Work.
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.
e.g. typing `help` and pressing `Enter` will open the help window.
6. Here are some commands you can try to get you started!:
 - `add-member mn/Adam Smith mi/AS` : Adds a member with name "Adam Smith" to the project.
 - `add-task tn/Finish up milestone setting` : Adds a task with name "Finish up milestone" to the project.
 - `assign ti/1 mi/JD` : Assigns the project member "John Doe" (with member id "JD"), to the task "Finish up milestone setting".
 - `doing-task ti/1` : Changes status of task "Finish up milestone setting" to `doing`:

3. Features

Command Format

- The command begins with the type of command to be executed
 - Example: in `remove-member mi/MEMBER_ID`, remove member is the type of command
- The words in UPPER_CASE corresponds to the parameters given by the user
 - Example: In `add-member mn/ MEMBER_NAME`, MEMBER_NAME is the input specified by the user, such as: `add-member mn/Gabriel Seow`
- You can specify the type of parameter given through a prefix. The list of allowed prefixes can be found in the table below:
 - Example: In `add-task tn/TASK_NAME s/TASK_STATUS`, the user inputs the name of the task after tn/ and inputs the status of the task after s/ , such as: `add task t/Finish Project Draft s/undone`

Table 1. List of Prefixes

Command Type	Prefix	Description + Parameters can be given in any order
Calendars	fp/	File path
Inventories	i/	Inventory name
	p/	Price
	ii/	Inventory tag
	ty/	PDF type
Meetings	meeting/	Meeting index
	start/	Start period
	end/	End period
	hours/	Duration of meeting
Members	mn/	Member name
	mi/	Member ID (set by you)
	mt/	Member tag
	im/	Image path
Tasks	tn/	Task name
	ti/	Task index
	at/	Task deadline
	s/	Task status
	tt/	Task tag

- Tags

- Tags are the only non-compulsory parameter mentioned in commands (unless otherwise stated).

- When tags are mentioned as a parameter, it is taken for granted that you can input multiple tags using the format `mt/… mt/… mt/…`

3.1. Universal Commands

3.1.1. Accessing project dashboard page: `home`

This command brings you to the project dashboard page, where tasks are displayed.

Format: `home`

Calling the `home` command will bring you to the following page:

[Home] | *Home.png*

3.1.2. Accessing time management page: `calendar`

This command brings you to the time management page where calendar and meeting times are displayed

Format: `calendar`

Calling the `calendar` command will bring you to the following page:

3.1.3. Accessing settings page: `settings` (Coming in v1.3)

This command brings you to the settings page

Format: `settings`

Entering the `settings` command will bring you to the following page:

[Settings] | *Settings.png*

NOTE You can refer to section 3.7 on specific settings-related commands to apply.

3.1.4. Viewing help: `help`

Displays a list of possible commands for the user

You can toggle through the command list (either through up down keys or mouse) and it will paste the correct syntax into the command line.

Format: `help`

[Help] | *Help.png*

3.2. Member-related Commands

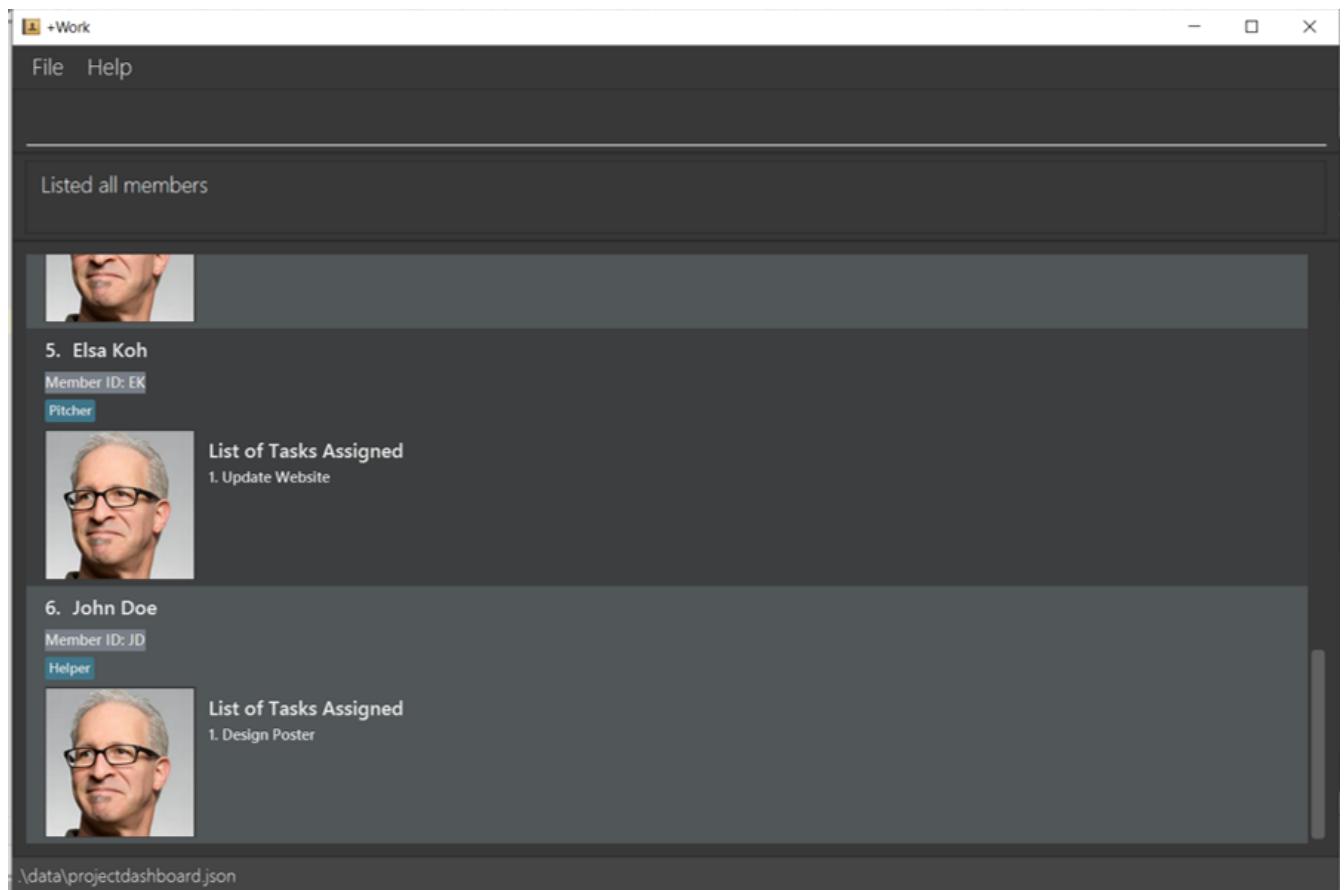
3.2.1. Adding a member: add-member

To add a member to the list of team members in +Work, use the command 'add-member' following the format below.

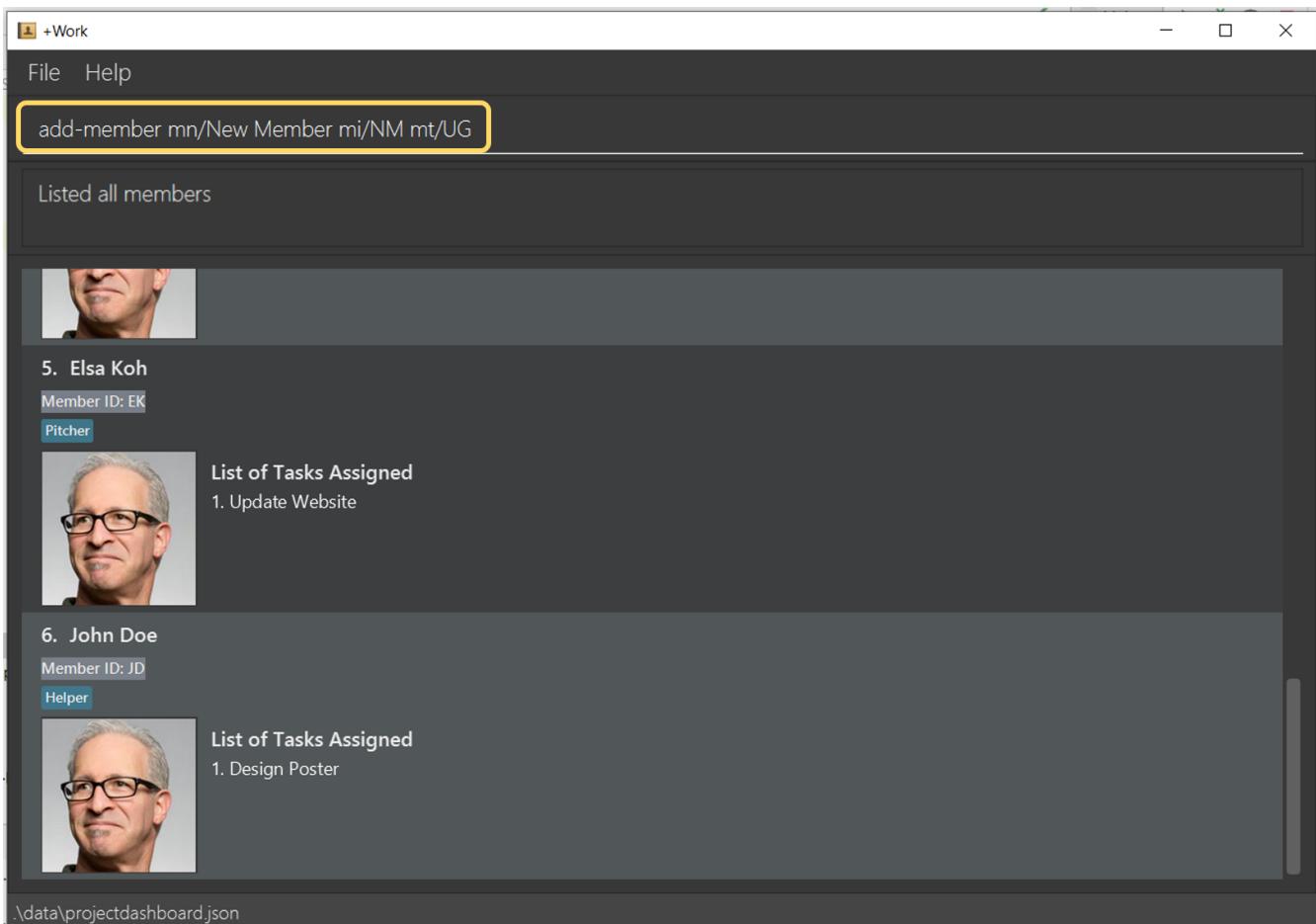
Format: `add-member mn/MEMBER_NAME mi/MEMBER_ID mt/TAGS`

Example: `add-member mn/New Member mi/NM mt/UG` can be executed as follows:

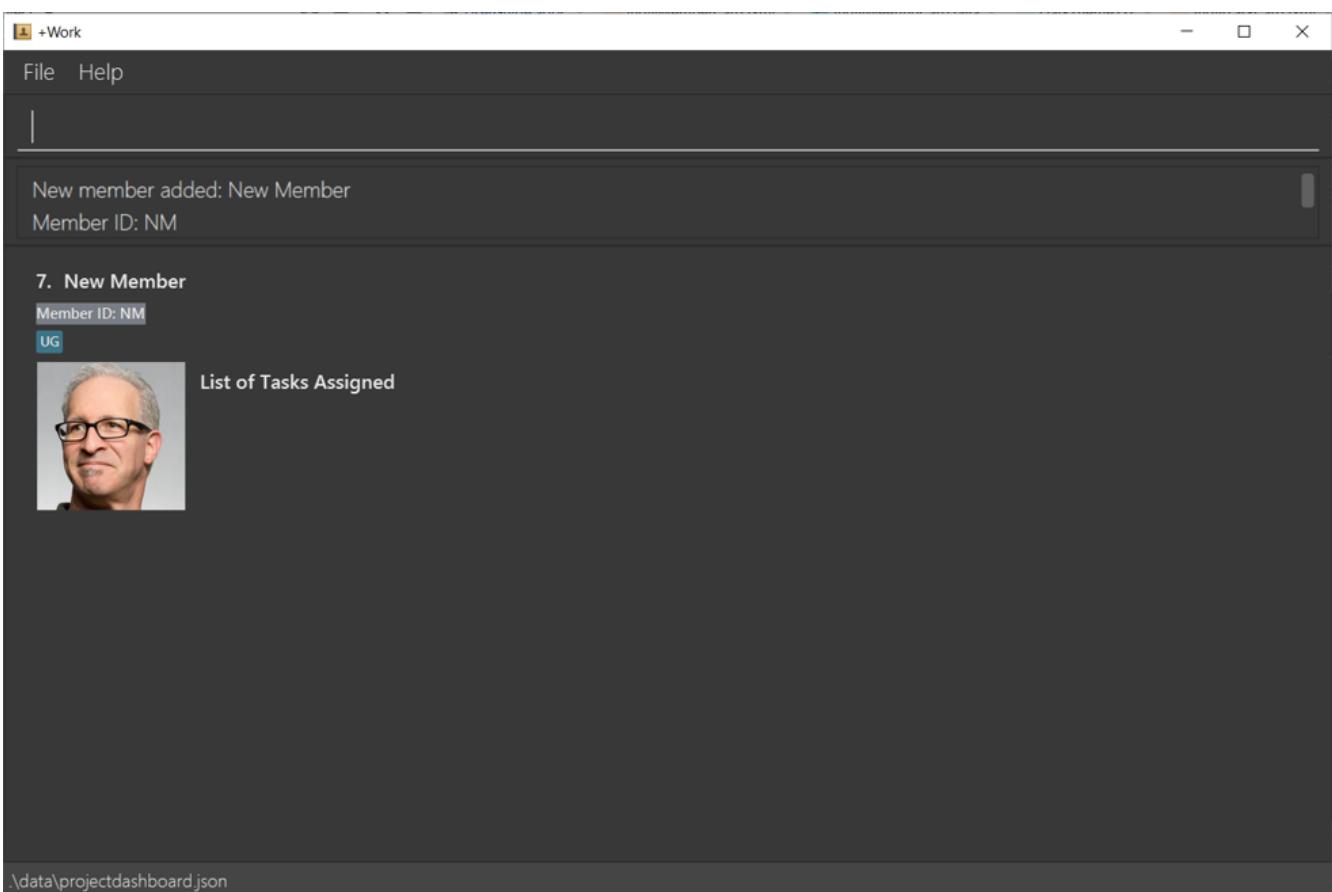
Step 1: +Work initially contains a list of 6 project members, as shown below.



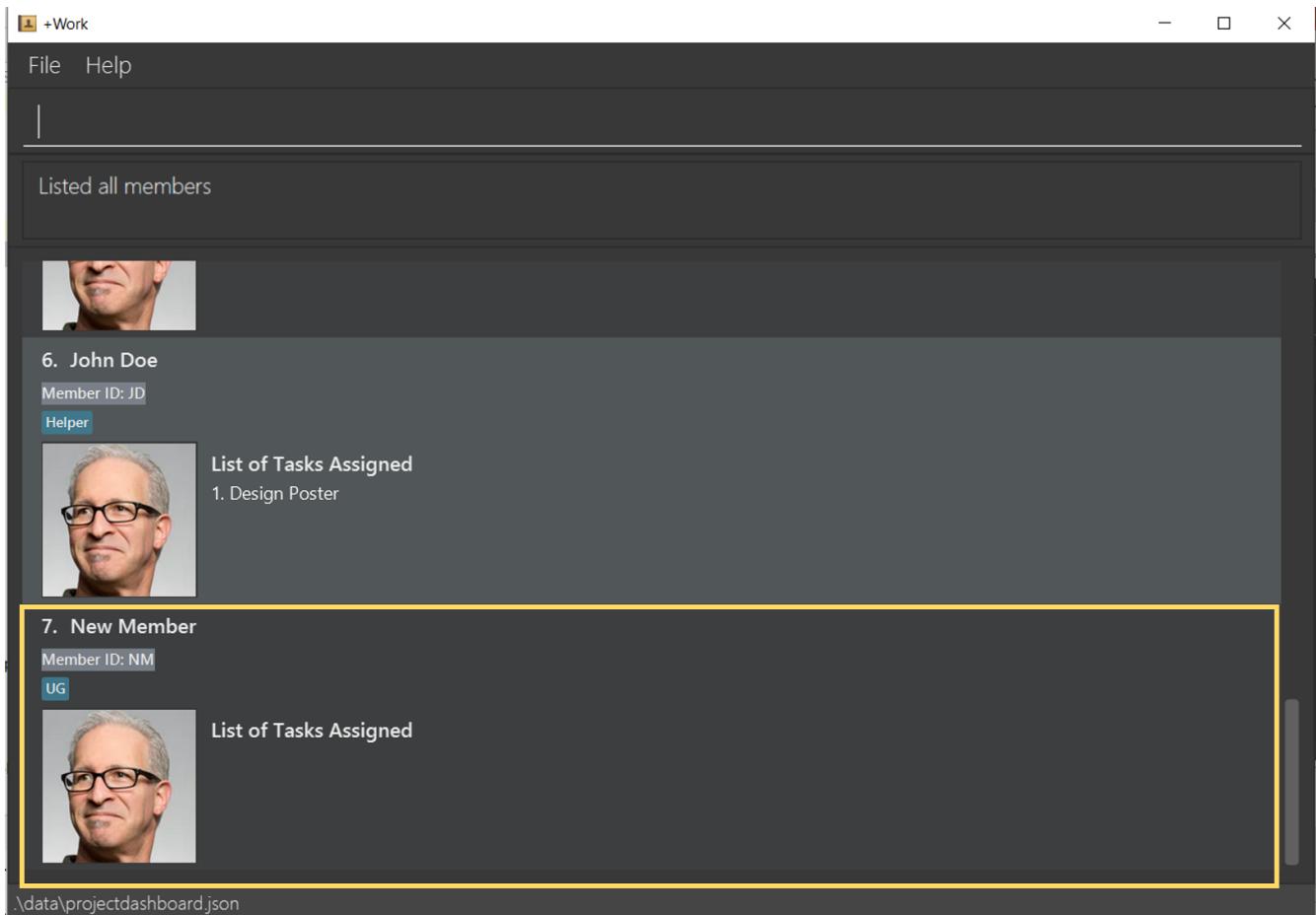
Step 2: To add a new project member into +Work, you enter the command `add-member mn/New Member mi/NM mt/UG` into the command prompt box.



Step 3: After you hit **Enter**, the result box will display the message "New member added", and a new member with name 'New Member', member ID 'NM' and tag 'UG' is added to +Work.



The addition of a new member can also be seen from the list of members as shown down below:



IMPORTANT

Member ID is an alphanumeric ID set by you, and cannot be changed once the member is created.

NOTE

- Adding a member tag is optional in the adding of a new member.
- It is possible to add a member with multiple tags following this format:
`add-member mn/New Member mi/NM mt/UG mt/DG mt/...`

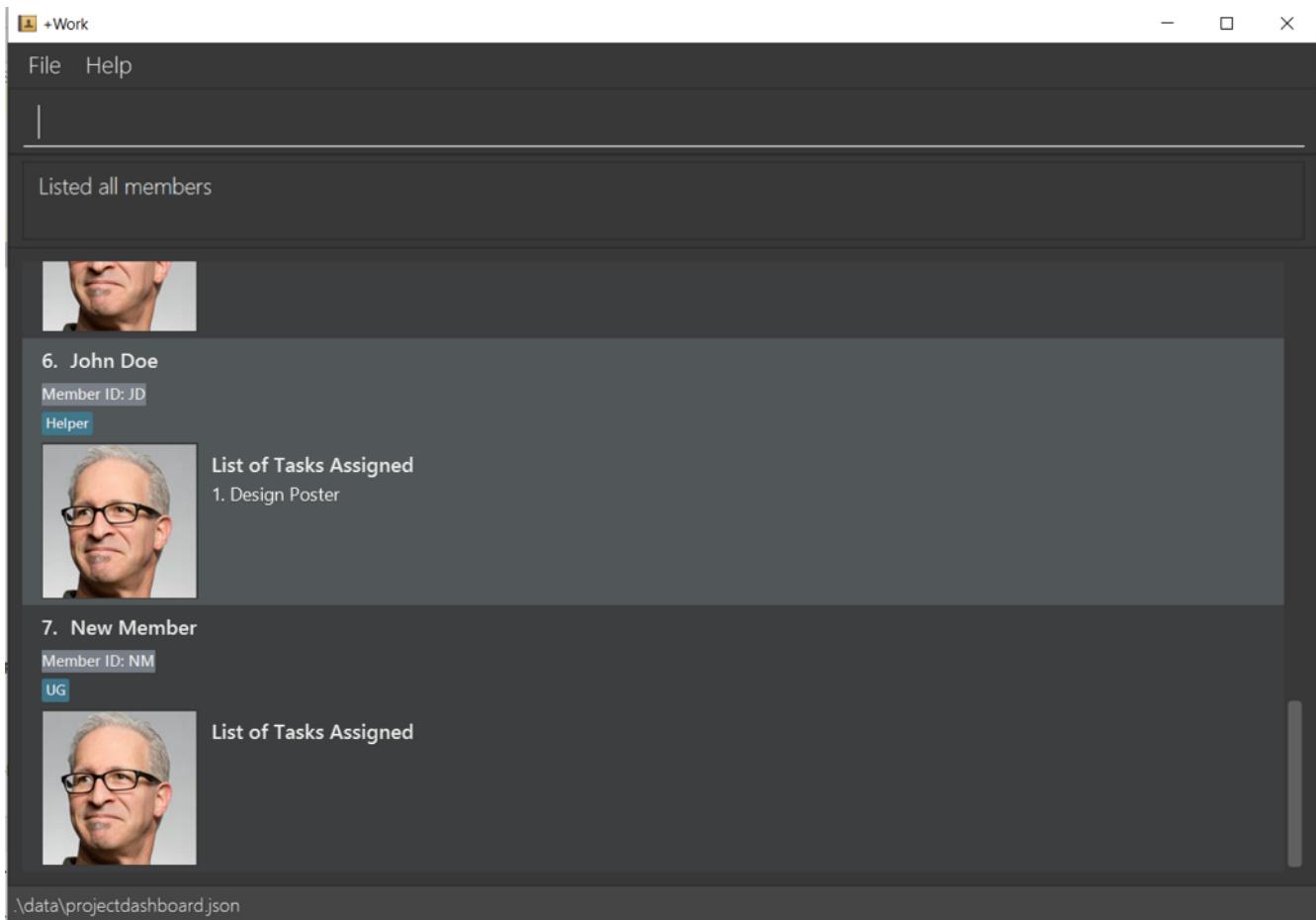
3.2.2. Set image for member: `set-image`

To set a profile picture for a member in +Work, use the command `set-image` following the format below.

Format: `set-image mi/MEMBER_ID im/IMAGE_PATH`

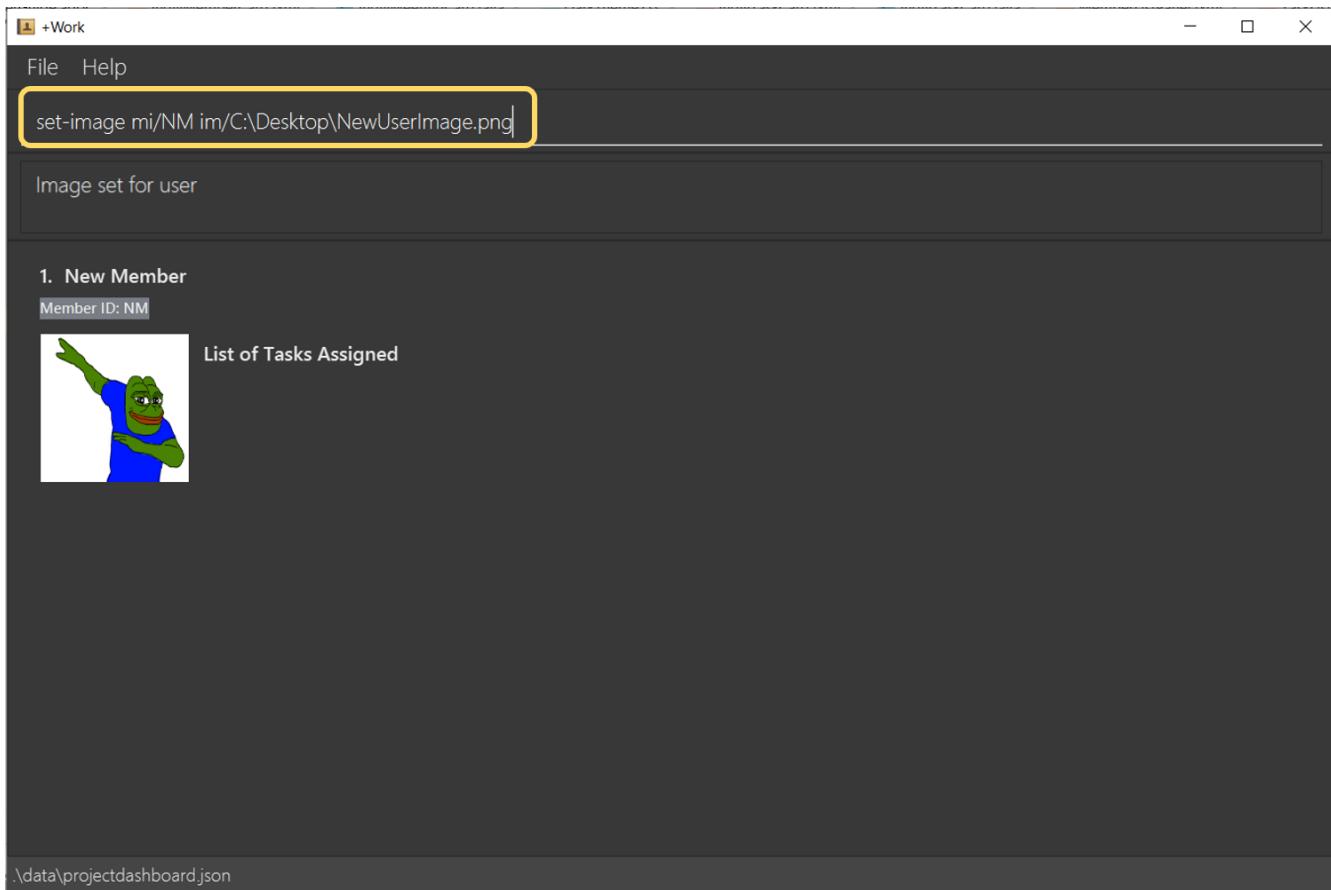
Example: `set-image mi/NM im/C:\Desktop\NewUserImage.png` can be executed as follows:

Step 1: +Work initially contains a list of project members with default profile pictures, as shown below.



Step 2: To update the profile picture of the project member with member ID 'NM' in +Work to a specified image, you enter the command `set-image mi/NM im/C:\Desktop\NewUserImage.png` into the command prompt box.

Step 3: After you **Enter** the command, the member 'New Member' with member ID 'NM' has a new profile picture, specified by the image path you entered.



NOTE

Image Path refers to the folder path of the image stored in your computer, and should end with .png

IMPORTANT

If you shift the image's location in your computer, +Work will be unable to find the image to display, and will display a warning message, before displaying the default profile picture. It is recommended that you store all the images in a central folder to prevent this from happening.

3.2.3. Editing a member : `edit-member`

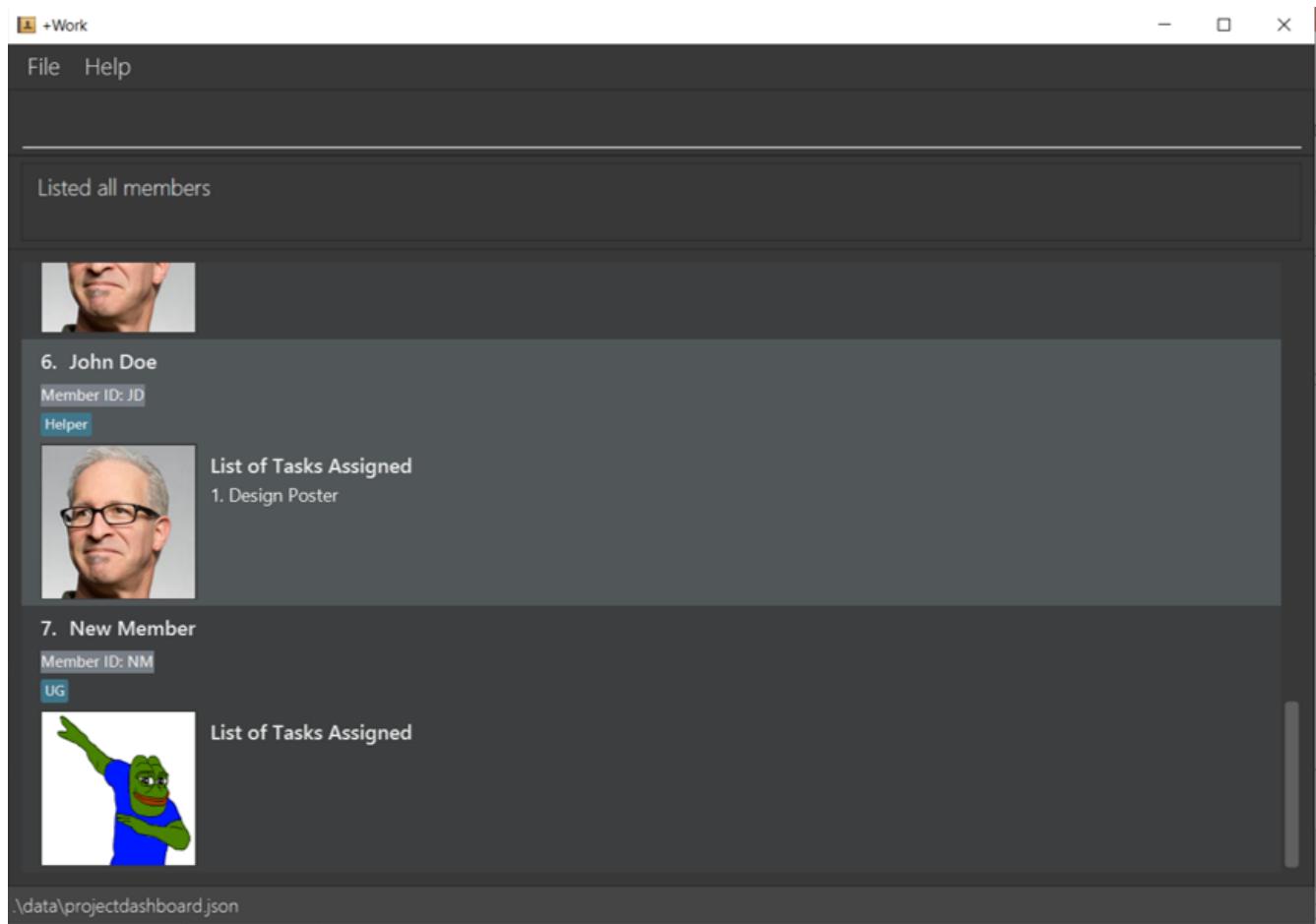
To edit a member in +Work, using the command `edit-member` following the format below.

Format: `edit-member mi/MEMBER_ID mn/MEMBER_NAME mt/MEMBER_TAG`

- Edits the member at the specified `mi/MEMBER_ID`.
- In this command, all the fields apart from `mi/MEMBER_ID` is optional. However, at least one of the optional fields must be provided.
- Existing values will be updated to the input values.
- When editing tags, the existing tags of the member will be removed i.e adding of tags is not cumulative.

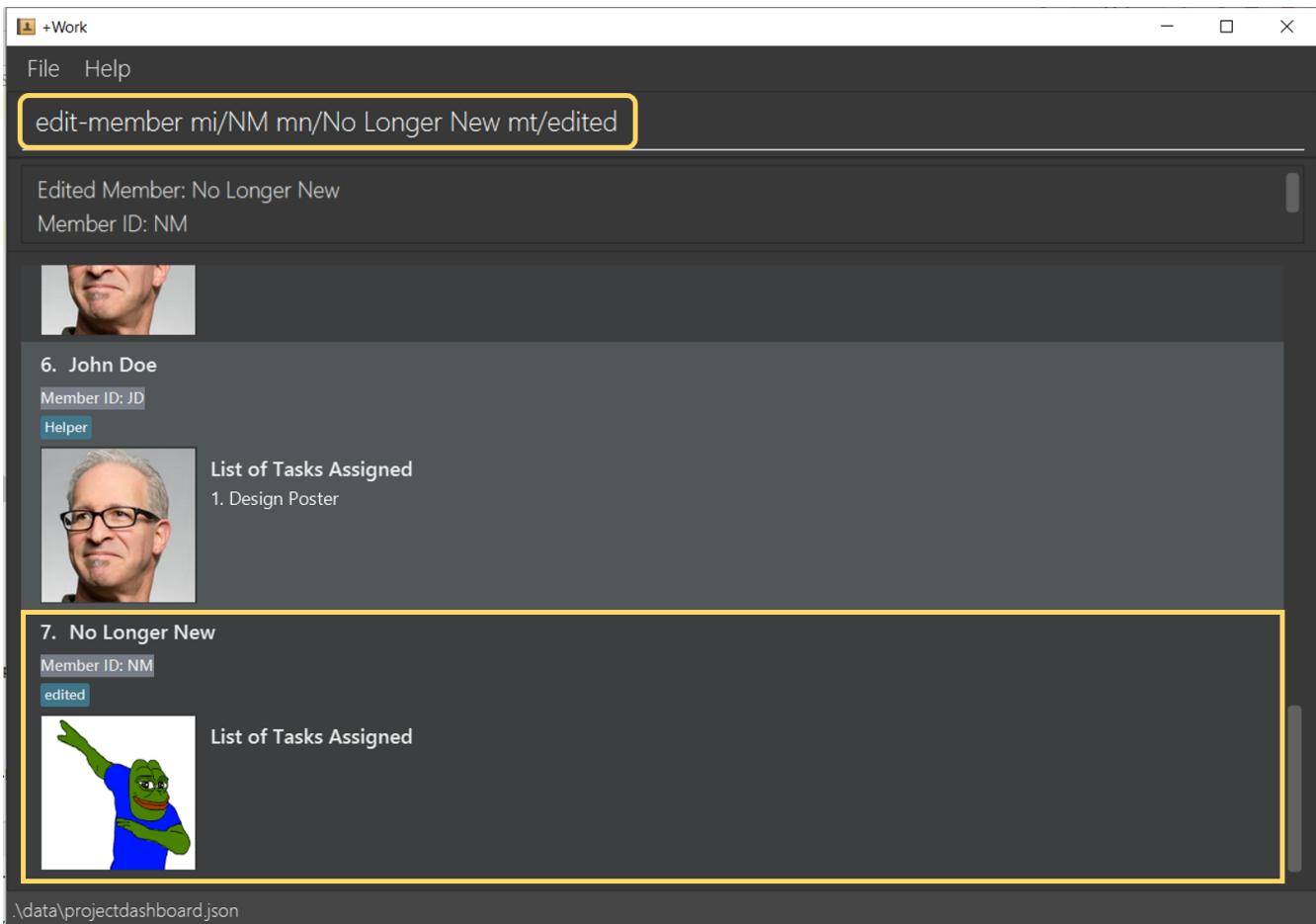
Example: `edit-member mi/NM mn/No Longer New mt/edited` can be executed as follows:

Step 1: +Work initially contains a list of 7 project members, as shown below.



Step 2: To edit the member name and tag of 'New Member' with member ID 'NM', you enter the command `edit-member mi/NM mn/No Longer New mt/edited` into the command prompt box.

Step 3: After you `Enter` the command, the member with member ID 'NM' and tag 'UG' is edited, with a new member name 'No Longer New' and new tag 'edited'.

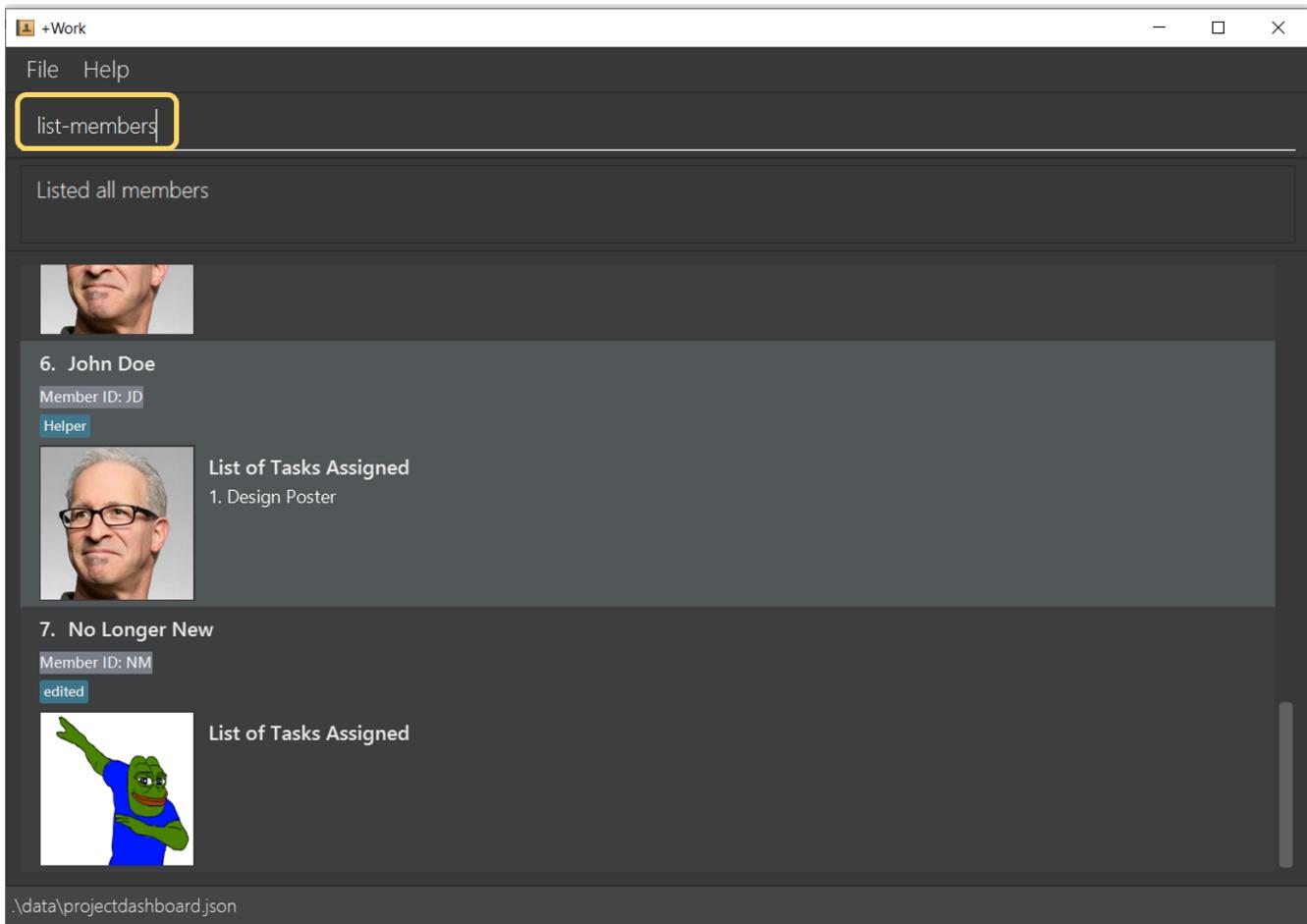


3.2.4. List existing members: `list-members`

To get a list of all members added to +Work, used the command `list-members` following the format below.

Format: `list-members`

Example: Entering `list-members` into the command prompt will result in the following:



As seen from the above, all existing project members in +Work will be listed.

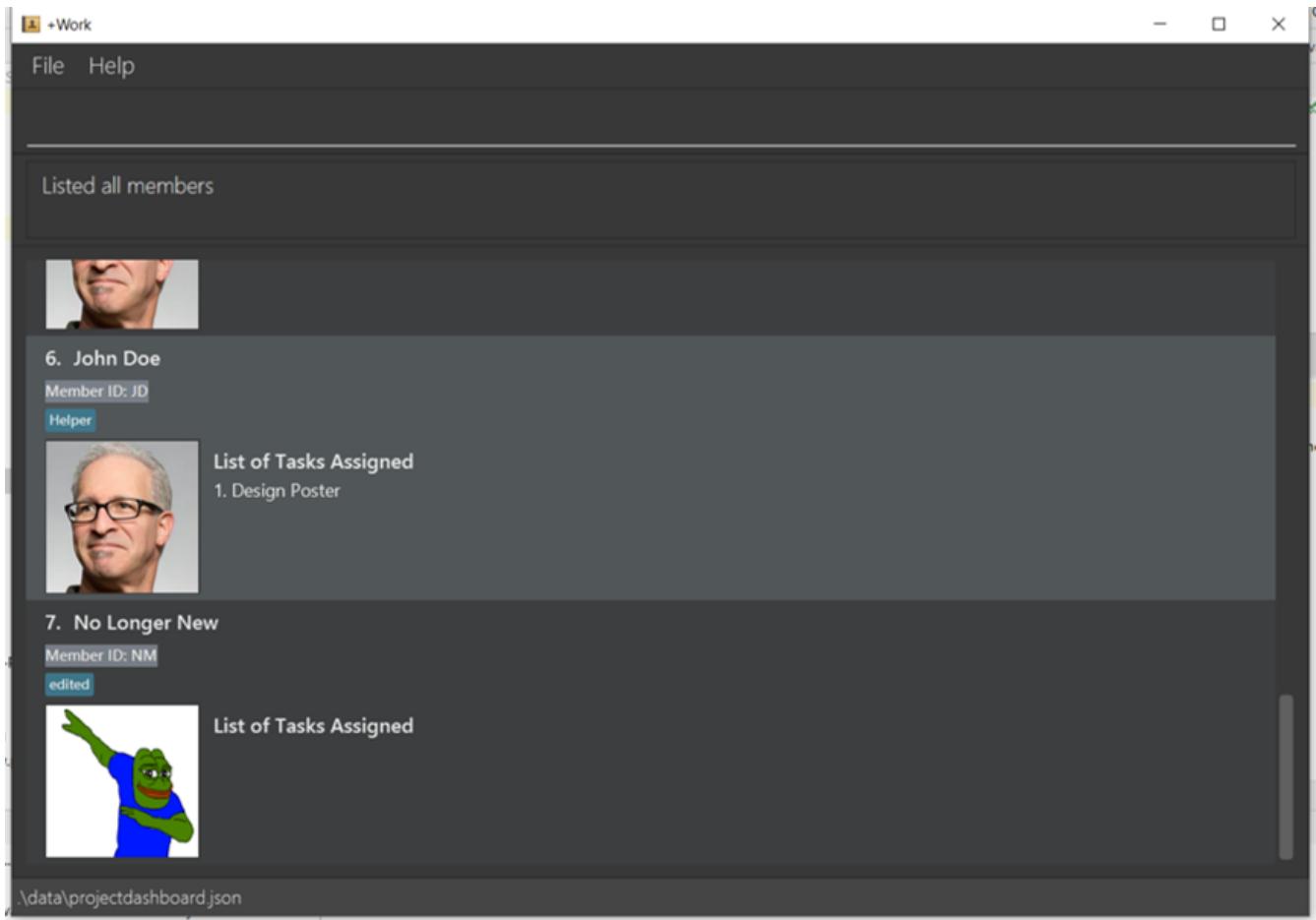
3.2.5. Removing a member: `remove-member`

To remove a member from the project, and subsequently remove him from associated tasks, use the 'remove-member' command in the format below.

Format: `remove-member [mi/MEMBER_ID]`

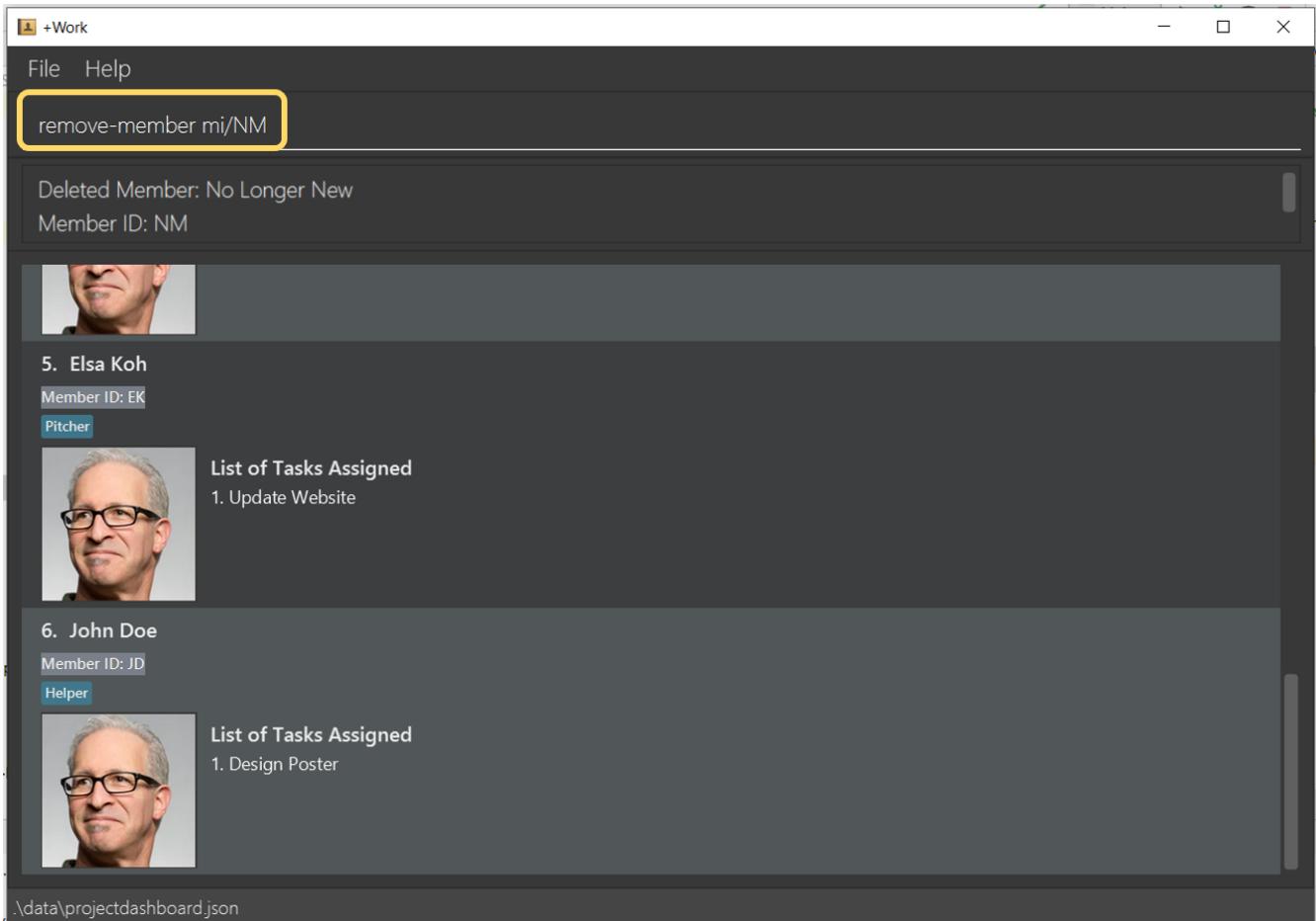
Example: `remove-member mi/GS` can be executed as follows:

Step 1: +Work now contains a list of 7 project members, as shown below.



Step 2: To remove project member 'No Longer New', with member ID 'NM' from +Work, you enter the command `remove-member mi/NM` into the command prompt box.

Step 3: After you `Enter` the command, the member 'No Longer New' is no longer a project member in +Work, as seen from the list of members below.



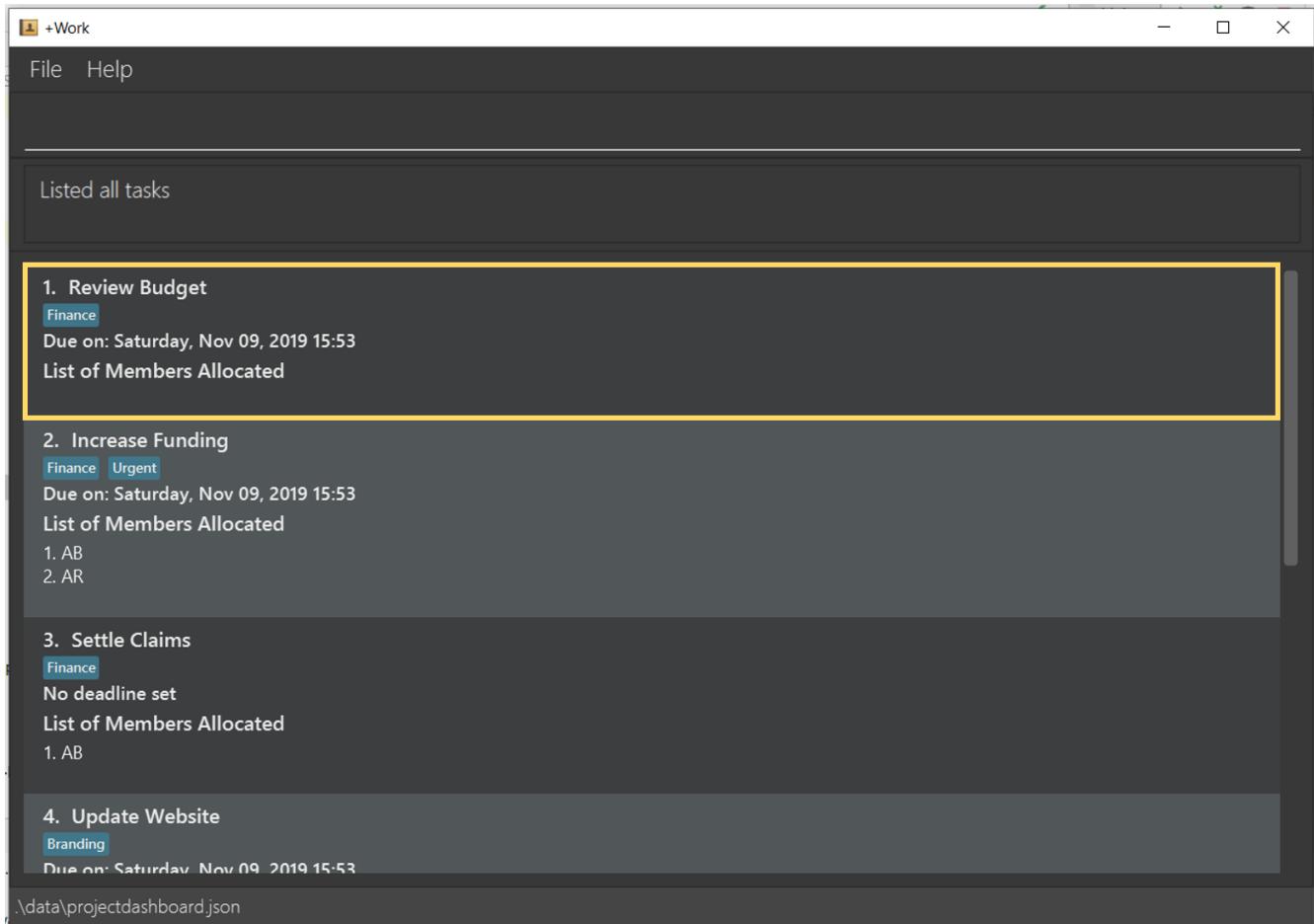
3.2.6. Assign a task to a member: `assign`

To assign a task to a specific team member, use the `assign` command in the format below.

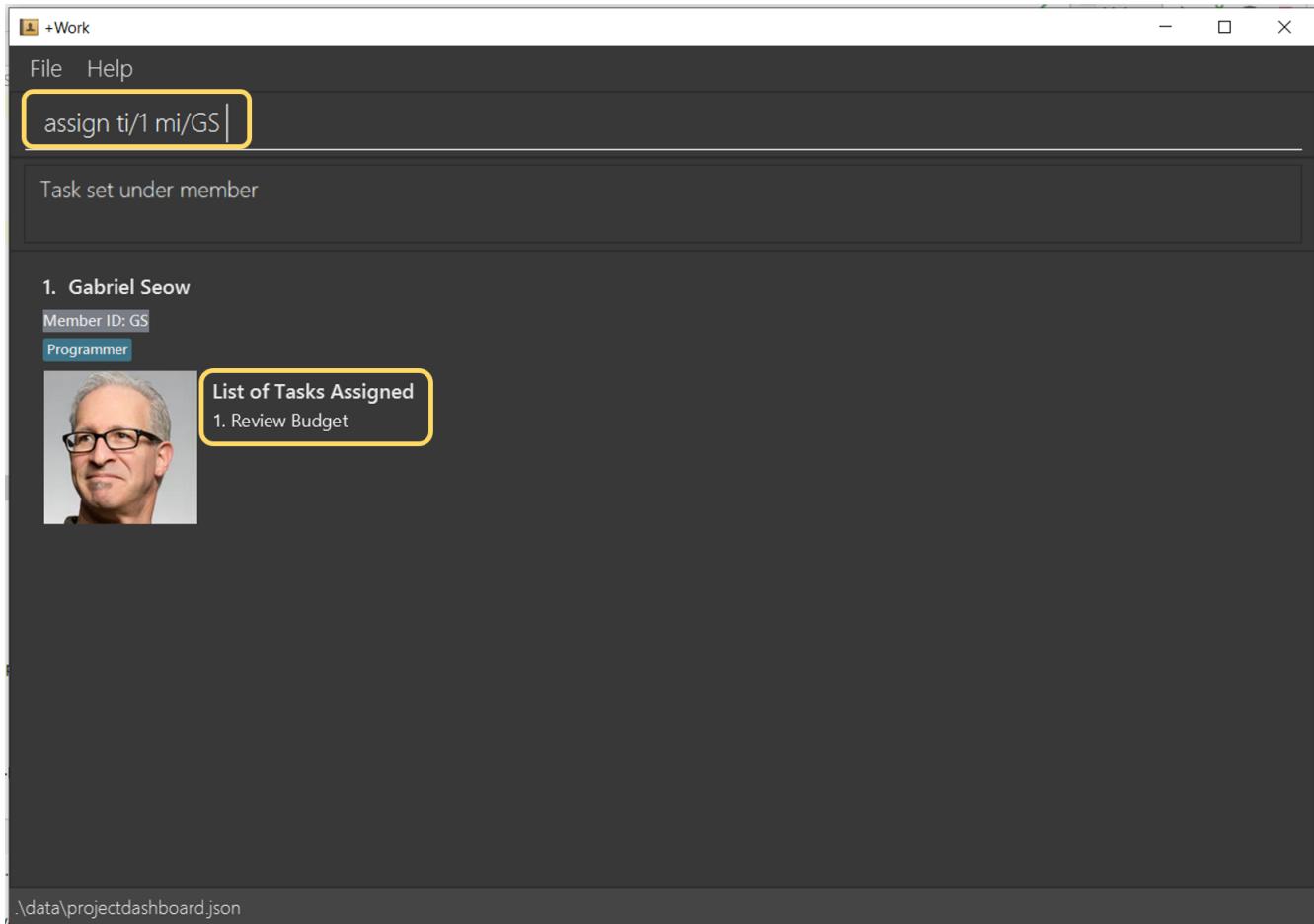
Format: `assign ti/TASK_ID mi/MEMBER_ID`

Example: `assign ti/1 mi/GS` can be executed as follows:

Step 1: From the list of tasks shown below, you decide to assign the task 'Review Budget' to project member 'Gabriel Seow' with member ID 'GS'. The task 'Review Budget' has task ID 1, prompting you to enter the command `assign ti/1 mi/GS`.



Step 2: After you **Enter** the command, the task 'Review Budget' with task ID '1' is added under member 'Gabriel Seow' with member id 'GS', as seen from the image below.



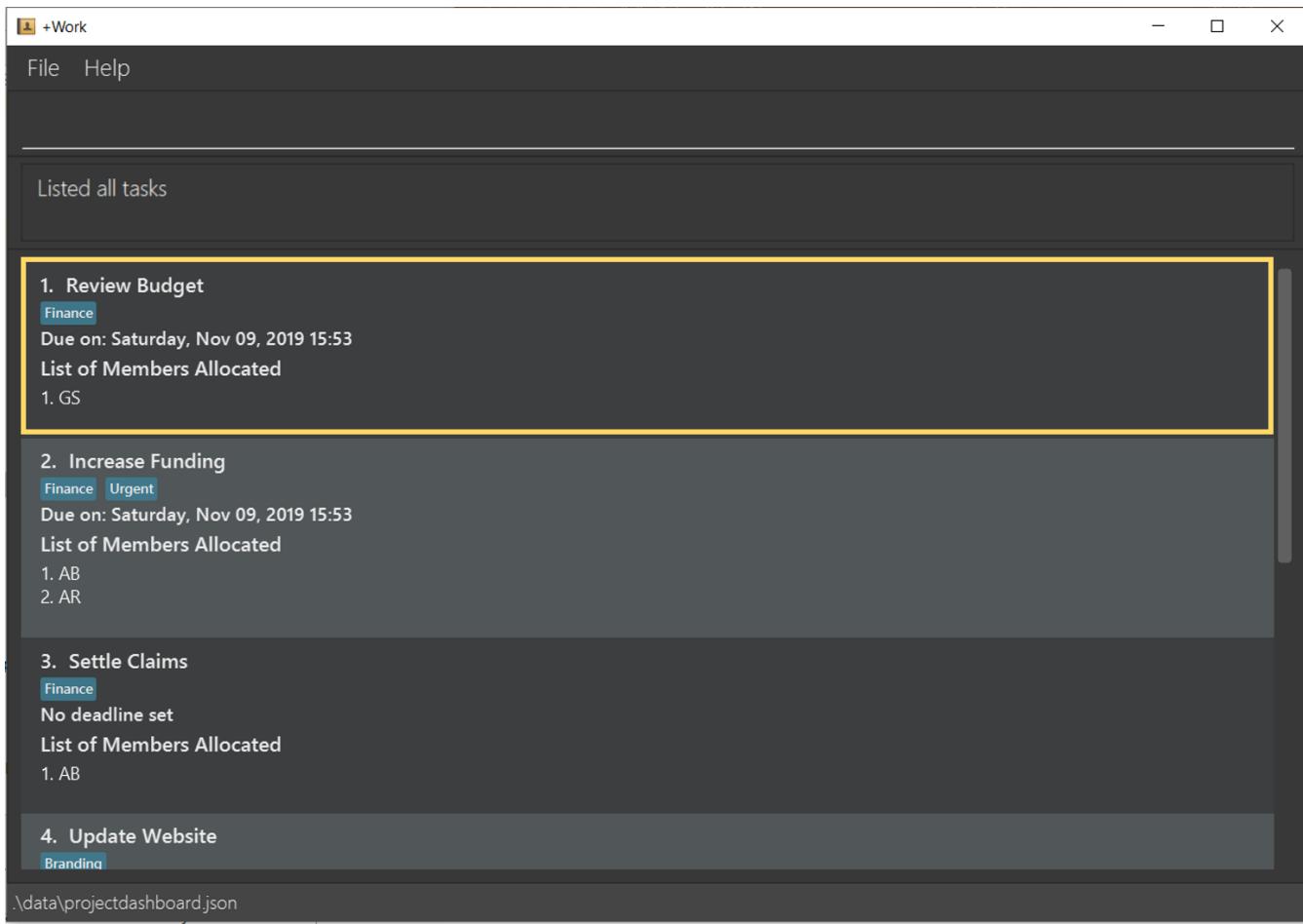
3.2.7. Removing a task from a member: **fire**

To remove a task from a specific team member, use the 'fire' command in the format below.

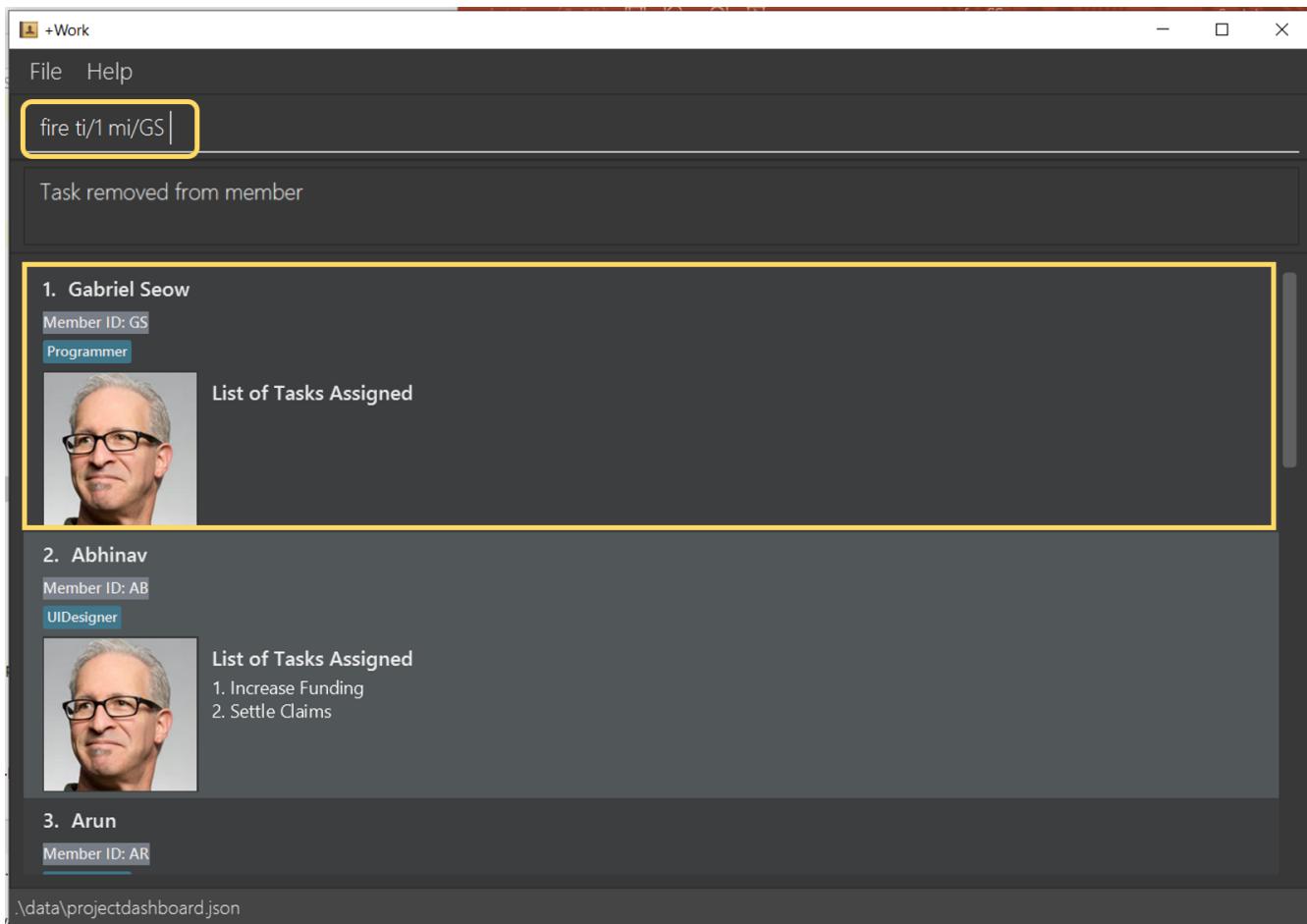
Format: **fire ti/TASK_ID mi/MEMBER_ID**

Example: **fire ti/1 mi/GS** can be executed as follows:

Step 1: From the list of tasks shown below, you decide to remove project member 'Gabriel Seow' with member ID 'GS' from being assigned to task 'Review Budget'. The task 'Review Budget' has task ID 1, prompting you to enter the command **fire ti/1 mi/GS**.



Step 2: After you **Enter** the command, the task 'Review Budget' with task id 1 is removed from member 'Gabriel Seow' with member id 'GS' as seen from the image below.



3.3. Task-related Commands

3.3.1. Adding a task: `add task`

To add a task to the project, use the 'add-task' command in the format below.

Format: `add-task [tn/TASK_NAME]`

Optional Parameters: `[s/STATUS] [tt/TAG]`

Example:

- `add-task tn/Finish up milestone setting`

A new task will be added to the project dashboard

3.3.2. Removing a task: `remove-task`

Removes a task from the dashboard

Format: `remove-task [ti/TASK_ID]`

Example:

- `remove-task 2`

Removes the 2nd task in the dashboard

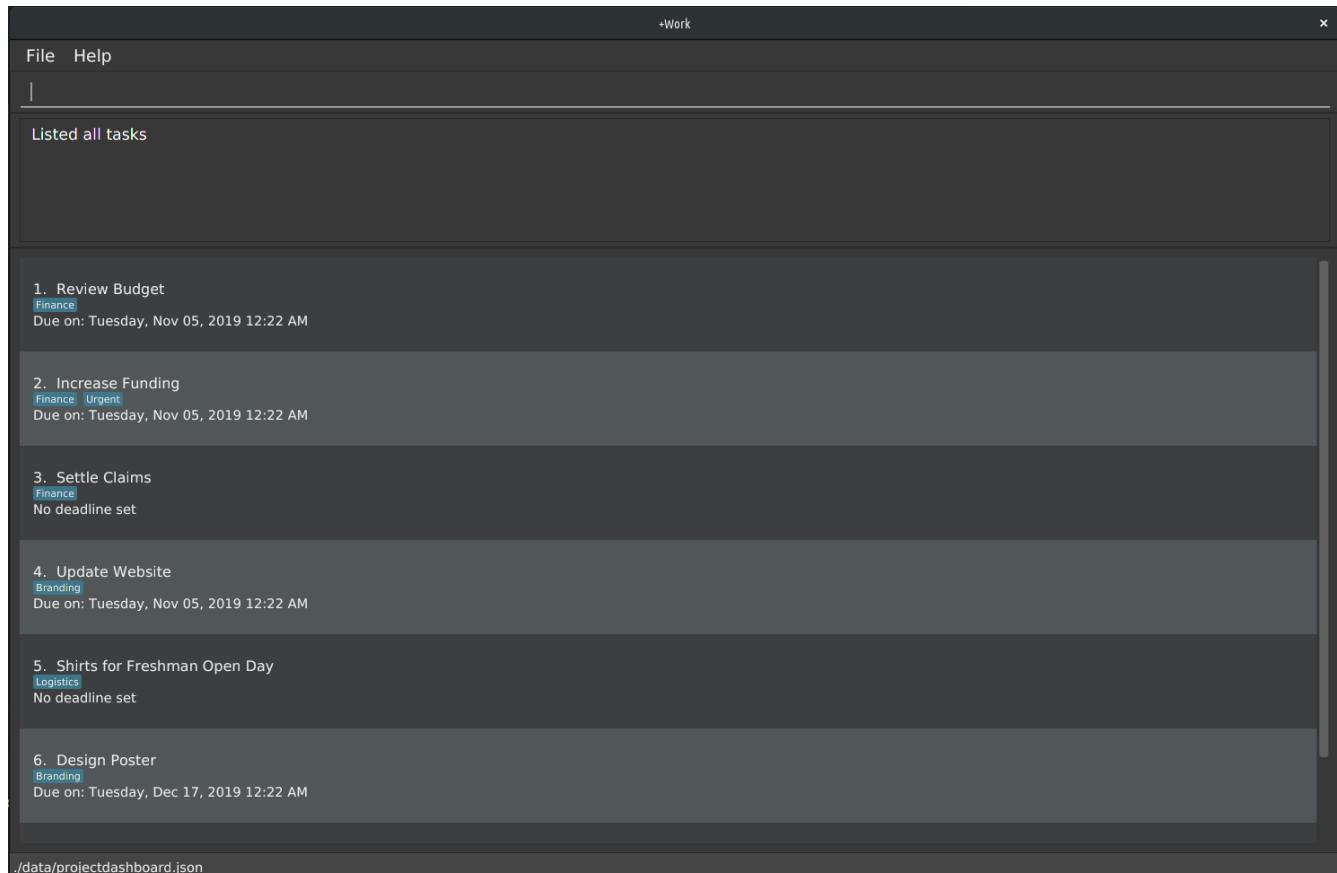
3.3.3. Listing all existing tasks: `list-tasks`

To list all the tasks created for the project, use the `list-tasks` command in the format below.

Format: 'list-tasks'

Example:

- Type `list-tasks` in the command box.
- Hit the `Enter` key and you will see all your tasks! You should see a window similar to the one below.



3.3.4. Setting a task's status to `done`: `done-task`

To update the task status to `done`, use the `done-task` command in the format below.

Format: `done-task [ti/TASK_ID]`

Example:

- After finishing the task "Shirts for Freshman Open Day" shown below, you would want to mark it as completed. To do so, first navigate to task list view using `list-tasks`.

File Help

list-tasks

Back to project dashboard.

Not Started	Doing	Done
1. Review Budget Finance Due on: Tuesday, Nov 05, 2019 00:22	1. Increase Funding Finance Urgent Due on: Tuesday, Nov 05, 2019 00:22	1. Update Website Branding Due on: Tuesday, Nov 05, 2019 00:22
2. Design Poster Branding Due on: Tuesday, Dec 17, 2019 00:22	2. Settle Claims Finance No deadline set	
	3. Shirts for Freshman Open Day Logistics Due on: Sunday, Nov 10, 2019 18:00	
	4. Random random Due on: Thursday, Oct 31, 2019 10:00	

Upcoming deadlines

1. Increase Funding Finance Urgent Due on: Tuesday, Nov 05, 2019 00:22
2. Update Website Branding Due on: Tuesday, Nov 05, 2019 00:22
3. Shirts for Freshman Open Day Logistics Due on: Sunday, Nov 10, 2019 18:00
4. Random random Due on: Thursday, Oct 31, 2019 10:00

./data/projectdashboard.json

- Type `done-task ti/5` into the command box as shown below.

File Help

done-task 5|

Listed all tasks

1. Review Budget Finance Due on: Tuesday, Nov 05, 2019 00:22
2. Increase Funding Finance Urgent Due on: Tuesday, Nov 05, 2019 00:22
3. Settle Claims Finance No deadline set
4. Update Website Branding Due on: Tuesday, Nov 05, 2019 00:22
5. Shirts for Freshman Open Day Logistics Due on: Sunday, Nov 10, 2019 18:00
6. Design Poster Branding Due on: Tuesday, Dec 17, 2019 00:22

./data/projectdashboard.json

- Hit the `Enter` key and you will see that the task has been marked as `done!` You should see a window similar to the one below when you navigate back to `home`.

[done task result dashboard] | *done-task-result-dashboard.png*

3.3.5. Setting a task's status to **doing**: doing-task

To update the task status to **doing**, use the **doing-task** command in the format below.

Format: **doing-task [ti/TASK_ID]**

Example: **doing-task ti/1**

- **doing-task ti/3**

This sets the status of task 3 to **doing**.

TIP Usage of this command is very similar to setting a task status to **doing** as explained above.

3.3.6. Setting a deadline for a task: set-deadline

This sets a deadline for an existing task in your project.

Format: **set-deadline [ti/TASK_ID] [at/DEADLINE] [at/dd-mm-yyyy hh:mm]**

TIP Remember to input using the 24 hour time format of dd-mm-yyyy hh:mm

Example:

- To set a deadline for the task "Shirts for Freshman Open Day" shown below, first navigate to the task list view using **list-tasks**.

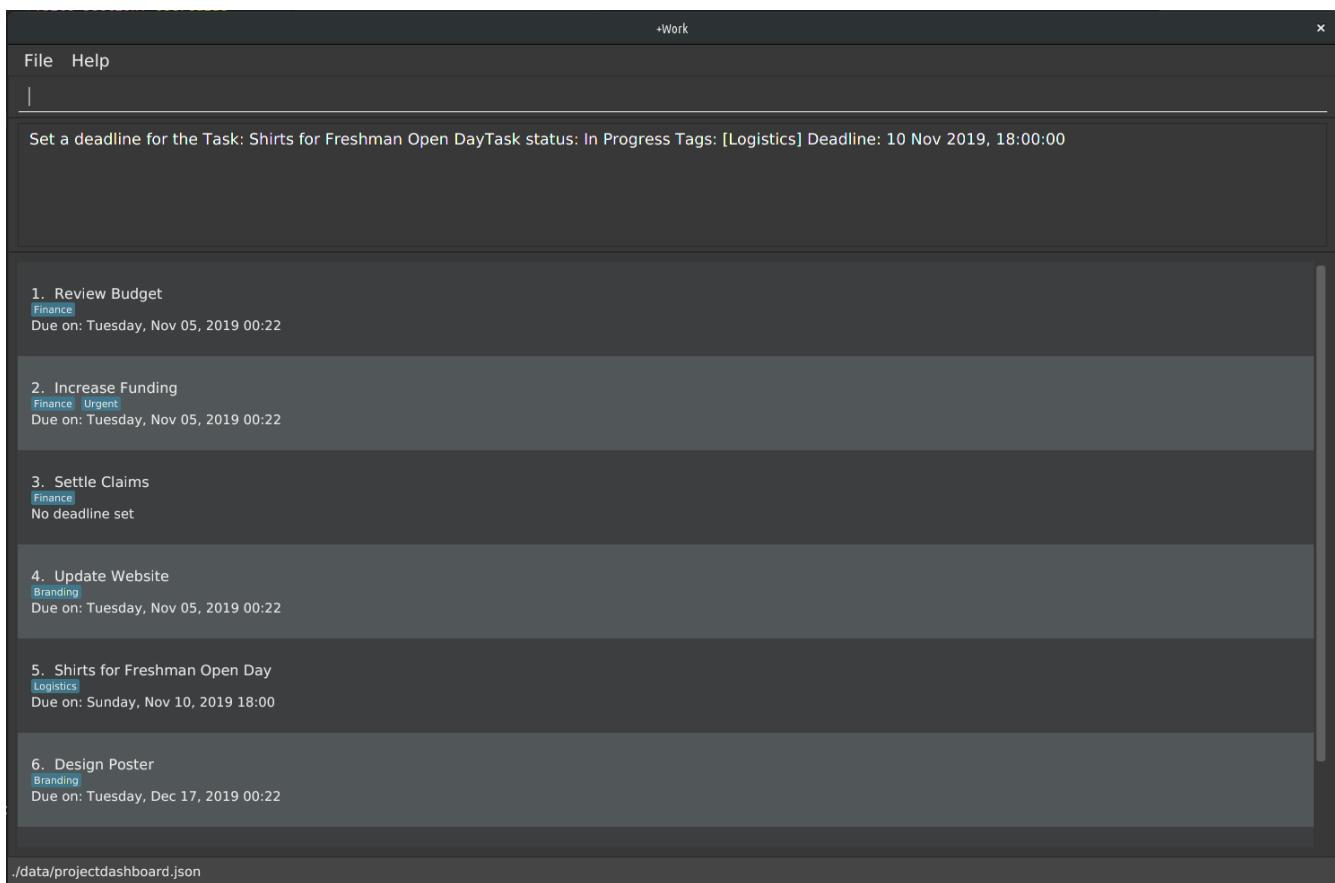
The screenshot shows a project management interface with a dark theme. At the top, there's a header bar with 'File' and 'Help' on the left and a '+Work' button on the right. Below the header, there's a search bar with a magnifying glass icon. The main area is divided into three columns: 'Not Started', 'Doing', and 'Done'. Each column contains a list of tasks with their descriptions, due dates, and categories. In the 'Doing' column, the third task, 'Shirts for Freshman Open Day', has a note 'No deadline set'. On the right side, there's a sidebar titled 'Upcoming deadlines' which lists tasks from both the 'Doing' and 'Done' columns with their respective due dates and descriptions. At the bottom of the interface, there's a footer bar with the path '/data/projectdashboard.json'.

Not Started	Doing	Done
1. Review Budget Finance Due on: Tuesday, Nov 05, 2019 12:22 AM	1. Increase Funding Finance Urgent Due on: Tuesday, Nov 05, 2019 12:22 AM	1. Update Website Branding Due on: Tuesday, Nov 05, 2019 12:22 AM
2. Design Poster Branding Due on: Tuesday, Dec 17, 2019 12:22 AM	2. Settle Claims Finance No deadline set	
	3. Shirts for Freshman Open Day Logistics No deadline set	
	4. Random random Due on: Thursday, Oct 31, 2019 10:00 AM	

Upcoming deadlines

1. Increase Funding Finance Urgent Due on: Tuesday, Nov 05, 2019 12:22 AM
2. Update Website Branding Due on: Tuesday, Nov 05, 2019 12:22 AM
3. Random random Due on: Thursday, Oct 31, 2019 10:00 AM

- Suppose the deadline for your task is on the 10th of November 2019 at 6pm, enter `set-deadline ti/5 at/10-11-2019 18:00` into the command box.
- Hit the `Enter` key and you will see that a deadline has been set!.



Now, navigate to the dashboard by entering `home`. Hit the `Enter` key and you will see that the deadline can be seen! You should see a window similar to the one below.

[set deadline res dashboard] | *set-deadline-res-dashboard.png*

TIP

The "Upcoming deadlines" sidebar (right side of above picture) helps you keep track of tasks which are due in less than two weeks.

3.4. Calendar Commands (Coming in v1.3)

Suggested steps for you to follow

1. User first exports their **.ics** file from NUSmods
2. Then import it into their google calendar
3. Add any weekly commitments to their google calendar
4. Export **.ics** file again
5. Import all to our application
6. enter command **generate-timings**
7. Choose the desired timing for the weekly meetings from the grid displayed
8. Create a new **Meeting**

3.4.1. Adding a members calendar: **import-calendar**

Format: **import-calendar [mi/MEMBER_ID] [c/PATH_TO_ICS]**

Example:

You can add your project's member John Doe's **ics** file like this,

- **import-calendar mi/JD c/data/john_ics_file**
Adds a calendar to John's profile

3.4.2. Generating time across all calendars uploaded: **generate-timings**

Format: **generate-timings**

Example:

- **generate-timings**
Generates a grid showing the number of people who can make it for a particular time slot during the week (mock-up needed)

3.4.3. Adding a meeting: **add meeting**

Format: **add-meeting [at/ dd-mm-yyyy hh:mm] [l/LOCATION]**

to add a new meeting

Example:

- **add-meeting at/10-10-2018 19:00 l/COM2-0204**
User chooses the meeting time from the grid displayed from generate timings commands (3.4.2)
. A meeting is added to the internal calendar of the application, which will be displayed on the dashboard.

3.5. Inventory Commands

3.5.1. Adding an inventory: add-inv

This command allows you to add an inventory bought or retrieved for a specific task by a specific member. This command allows you to add an inventory bought or retrieved for a specific task by a specific member.

Format: `add-inv [i/ITEM_NAME] [p/PRICE(optional)] [ti/TASK_ID] [mi/MEMBER_ID]`

NOTE

The item name, task id and member id are compulsory inputs. An input without any price value will automatically set the price to \$0. Ensure that an existing task id and member id (as displayed by list-tasks and list-members respectively) is being typed into the command box.

Examples:

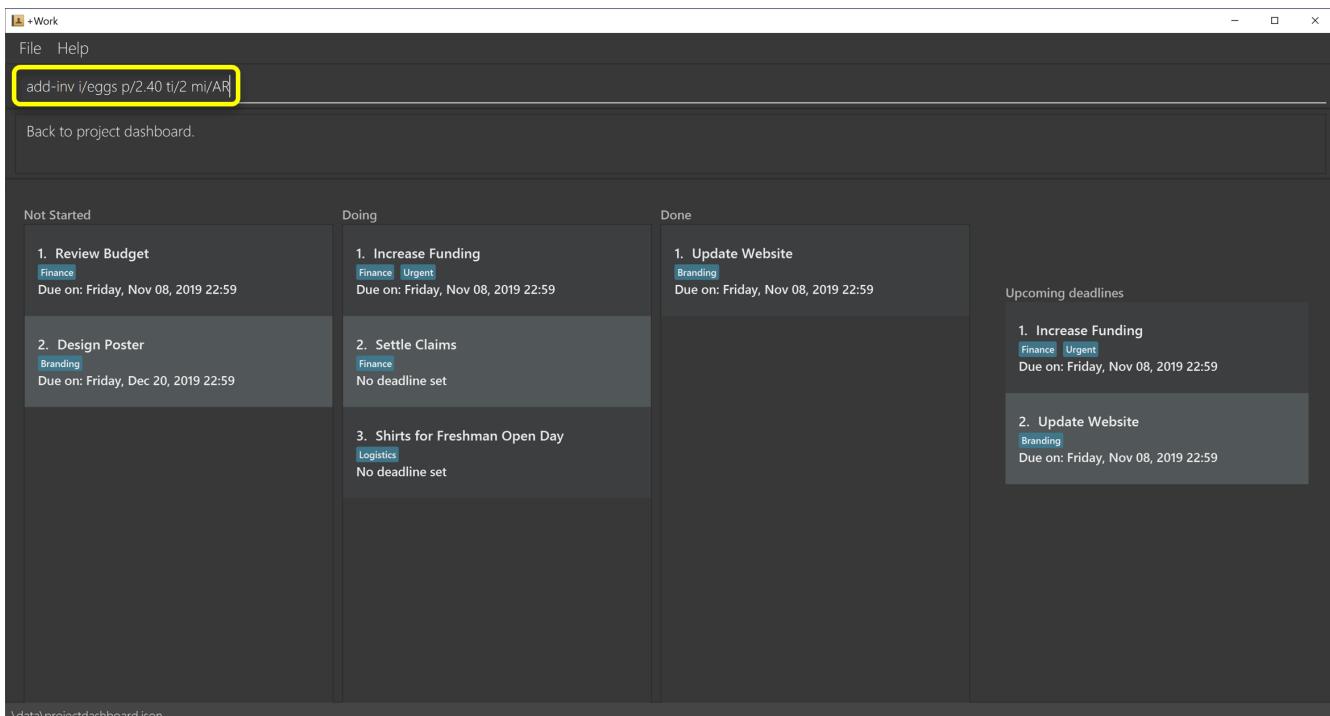
- `add-inv i/scissors ti/4 mi/GS`

Entering this command adds the inventory “scissors” to the inventory list. The item is tagged to task with id as 4 (id is shown by list-tasks) and is provided by member with the member id “GS” for a price of \$0.

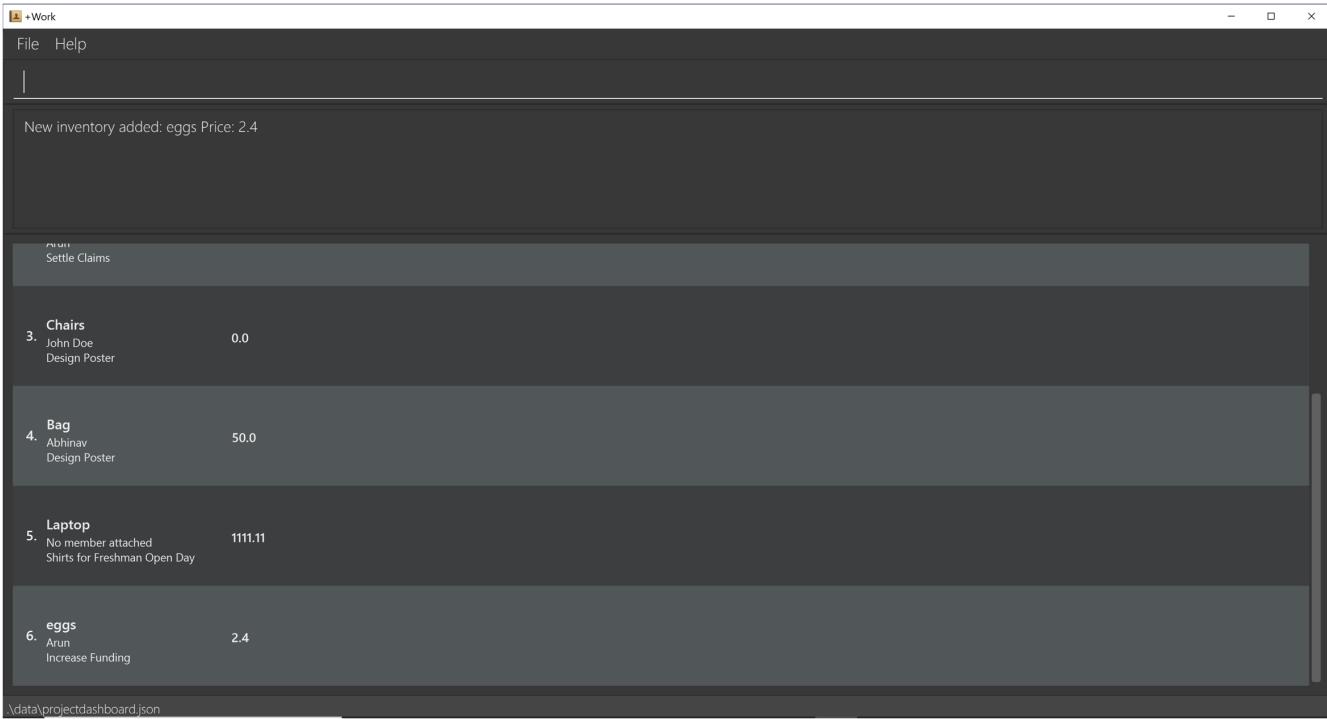
- `add-inv i/eggs p/2.40 ti/2 mi/AR`

Entering this command adds the item “eggs” for \$2.40 to the inventory list. This item is tagged to task tagged to task with id as 2 (id is shown by list-tasks) and was paid for by the member with member id “AR”. The following pictures show how this command is to be executed.

- First, type `add-inv i/eggs p/2.40 ti/2 mi/AR` into the command box as shown below.



- Then, hit the `Enter` key and you will see that the inventory has been added! You should see a window like the one below.



3.5.2. Deleting an inventory: `delete-inv`

This command allows you to delete an inventory.

Format: `delete-inv [ii/ITEM_ID]`

Examples:

- `delete-inv ii/3`

This command deletes the third item from the inventory list. The following pictures show how this command is to be executed

- First, find the index of inventory that you want to delete, by navigating to inventory list view using `list-inv`.

Not Started

- 1. Review Budget
Finance
Due on: Saturday, Nov 09, 2019 15:12
- 2. Design Poster
Branding
Due on: Saturday, Dec 21, 2019 15:12

Doing

- 1. Increase Funding
Finance Urgent
Due on: Saturday, Nov 09, 2019 15:12
- 2. Settle Claims
Finance
No deadline set
- 3. Shirts for Freshman Open Day
Logistics
No deadline set

Done

- 1. Update Website
Branding
Due on: Saturday, Nov 09, 2019 15:12

Upcoming deadlines

- 1. Increase Funding
Finance Urgent
Due on: Saturday, Nov 09, 2019 15:12
- 2. Update Website
Branding
Due on: Saturday, Nov 09, 2019 15:12

\data\projectdashboard.json

- Then, type delete-inv ii/6 into the command box as shown below.

File Help

delete-inv ii/6

Listed all inventories

Settle Claims

- 3. Chairs
Gabriel Seow
Design Poster
- 4. Bag
Abhinav
Design Poster
- 5. Canola Oil
Arun
Settle Claims
- 6. Laptop
Gabriel Seow
Shirts for Freshman Open Day

\data\projectdashboard.json

- Hit the **Enter** key and you will see that the inventory has been deleted! You should see a window like the one below.

The screenshot shows the +Work application interface. At the top, there's a menu bar with 'File' and 'Help'. Below the menu, a message says 'Deleted Inventory: Laptop Price: 1111.11'. The main area displays a list of inventories:

Rank	Item Name	Description	Price
2.	Bench	Arun Settle Claims	59.9
3.	Chairs	Gabriel Seow Design Poster	0.0
4.	Bag	Abhinav Design Poster	50.0
5.	Canola Oil	Arun Settle Claims	17.9

At the bottom left, it says '\data\projectdashboard.json'.

3.5.3. List existing inventories: `list-inv`

To get a list of all inventories added to +Work, used the command `list-inv` following the format below.

Format: `list-inv`

Example: Entering `list-inv` into the command prompt will result in the following:

The screenshot shows the +Work application interface. At the top, there's a menu bar with 'File' and 'Help'. Below the menu, a message says 'Listed all inventories'. The main area displays a list of inventories:

Rank	Item Name	Description	Price
1.	Toy	Arun Design Poster	8.9
2.	Bench	Arun Settle Claims	59.9
3.	Chairs	Gabriel Seow Design Poster	0.0
4.	Bag	Abhinav Design Poster	50.0
5.	Canola Oil	Arun Settle Claims	17.9

3.5.4. Editing a inventory: `edit-inv`

This command allows you to edit the details of an existing inventory.

Format: `edit-inv [ii/ITEM_ID] (i/ITEM_NAME) (p/PRICE) (ti/TASK_ID) (mi/MEMBER_ID)`

NOTE

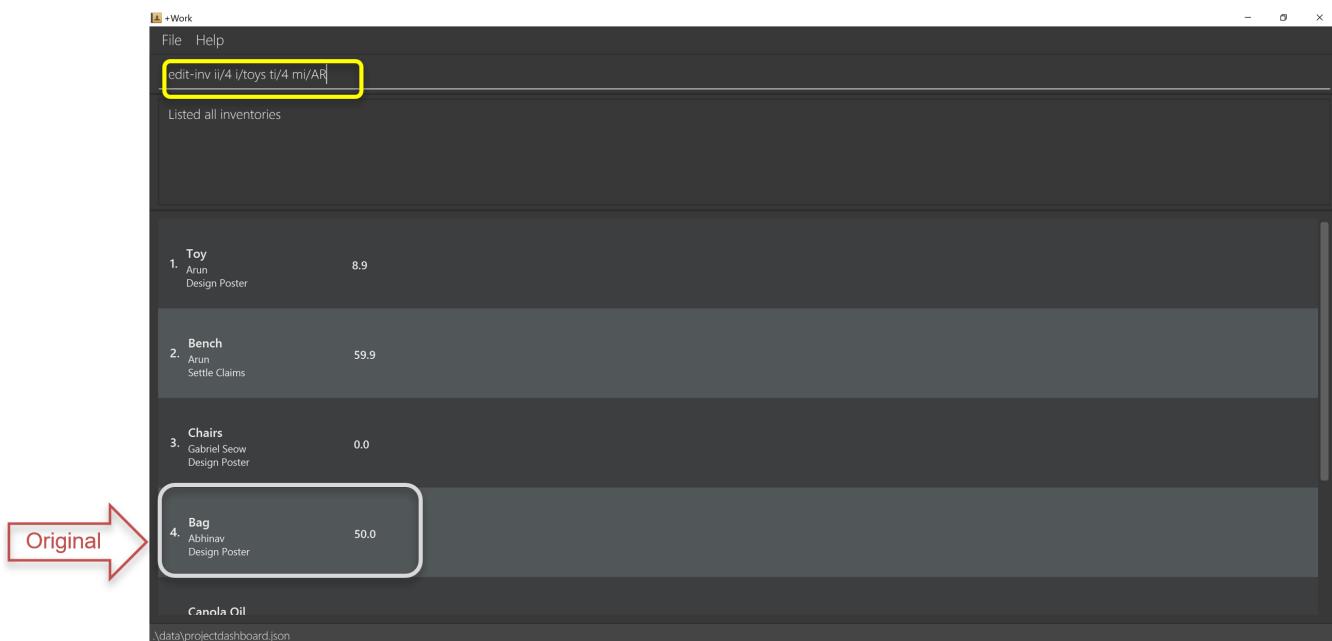
Multiple attributes can be changed at the same time, but at least one of the inventory attributes must be changed. Therefore, this command is invalid: `edit-inv ii/4` However, this command is valid: `edit-inv ii/4 i/toys ti/4 mi/AR`

Example:

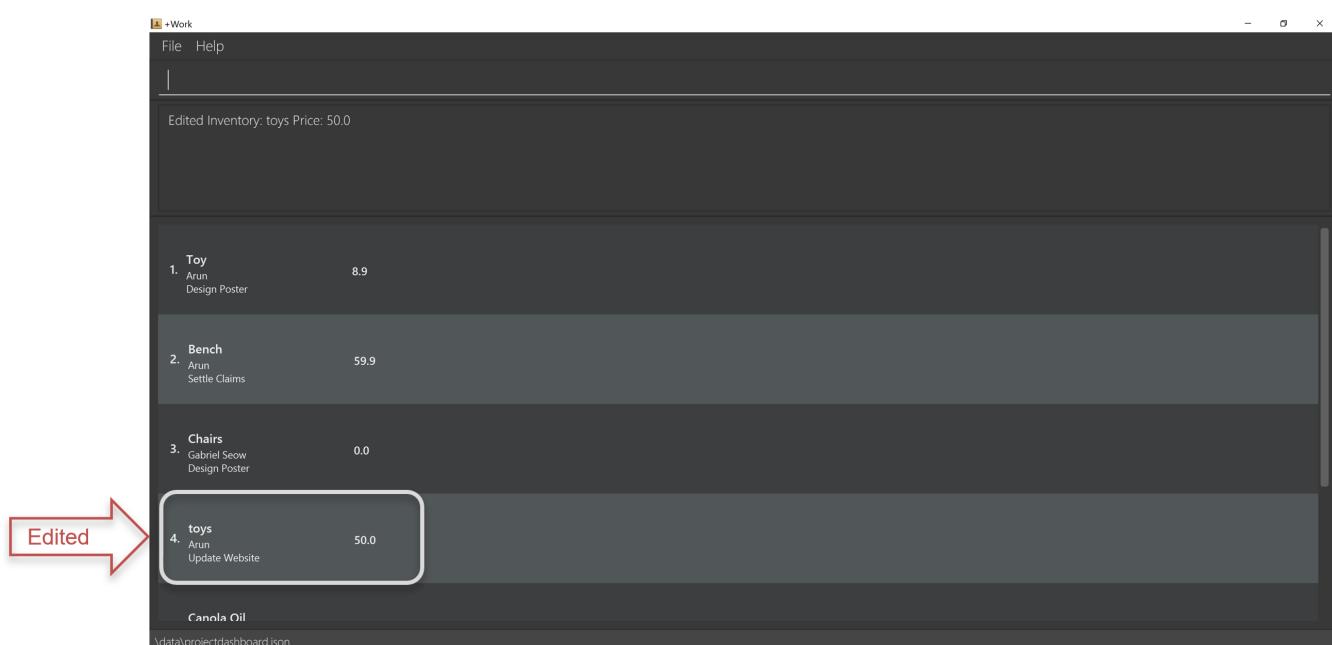
- `edit-inv ii/4 i/toys ti/4 mi/AR`

This command edits the 4th inventory from the inventory list as shown by list-inv. It changes the inventory name to toys, the task attached to 4, and the member attached to the member with member ID “AR”. The following pictures show how this command is to be executed.

- First, type `edit-inv ii/4 i/toys ti/4 mi/AR` into the command box as shown below.



- Hit the `Enter` key and you will see that the 4th inventory has been edited! The name has been changed to toys, task has been changed to the task with index 4, and finally member has been changed to member with index “AR”. You should see a window like the one below.



3.5.5. Creating a report of inventories: pdf

This command allows you to create a PDF report of the existing inventories classified either by the member attached or by the task attached.

Format: **pdf [ty/TYPE]**

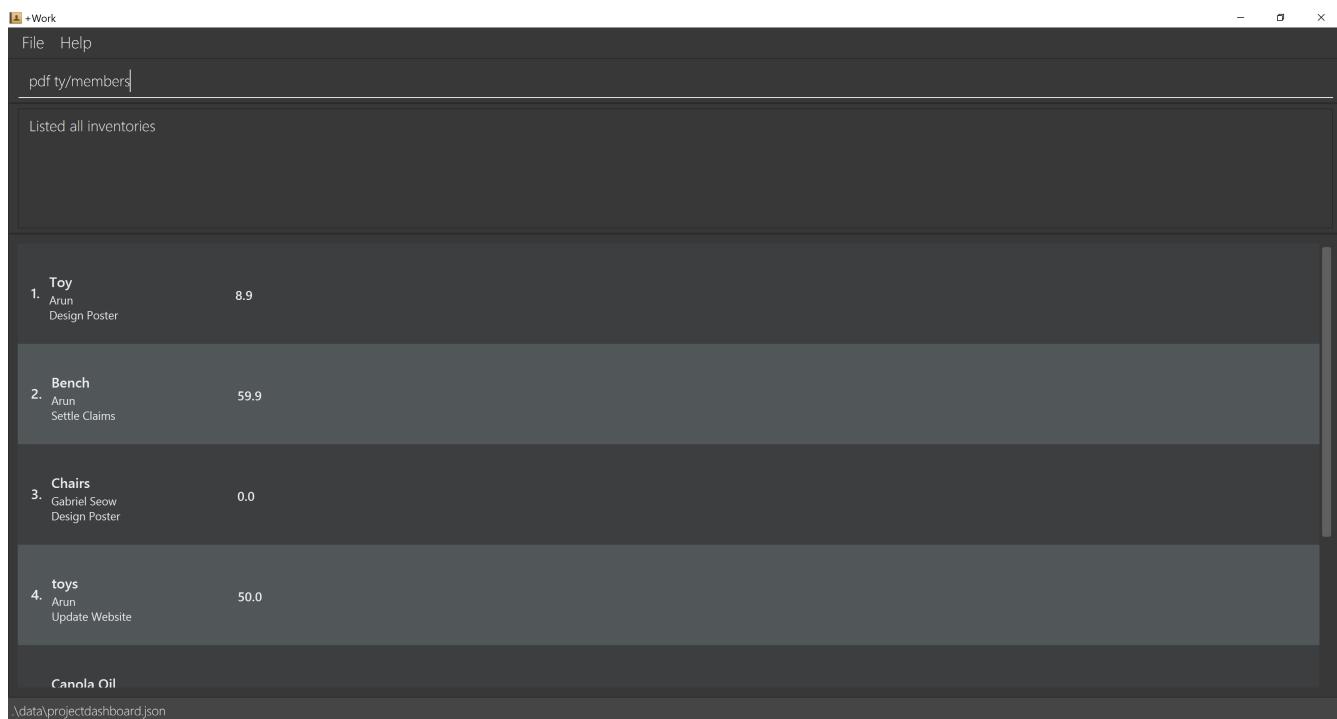
NOTE The only two attributes for TYPE are **members** and **tasks**. Other inputs will not work.
If a PDF created and is currently open, it has to be closed before another pdf can be created.

Example:

- **pdf ty/members**

This command creates and opens a pdf file of inventories that is classified according to the member attached. The following pictures show how this command is to be executed.

- First, type **pdf ty/members** into the command box as shown below.



- Hit the **Enter** key and you will see that the default PDF viewer will open to show the PDF report. You should see a report like the one below.

Inventories (by members)		
no.	Arun's inv	Price(\$)
1	Toy	8.9
2	Bench	59.9
3	Canola Oil	17.9
4	toys	50.0
Total		136.70
no.	Gabriel Seow's inv	Price(\$)
1	Chairs	0.0
Total		0.00
no.	Seah Lynn's inv	Price(\$)
1	buns	2.4
Total		2.40
Total Price:		139.10

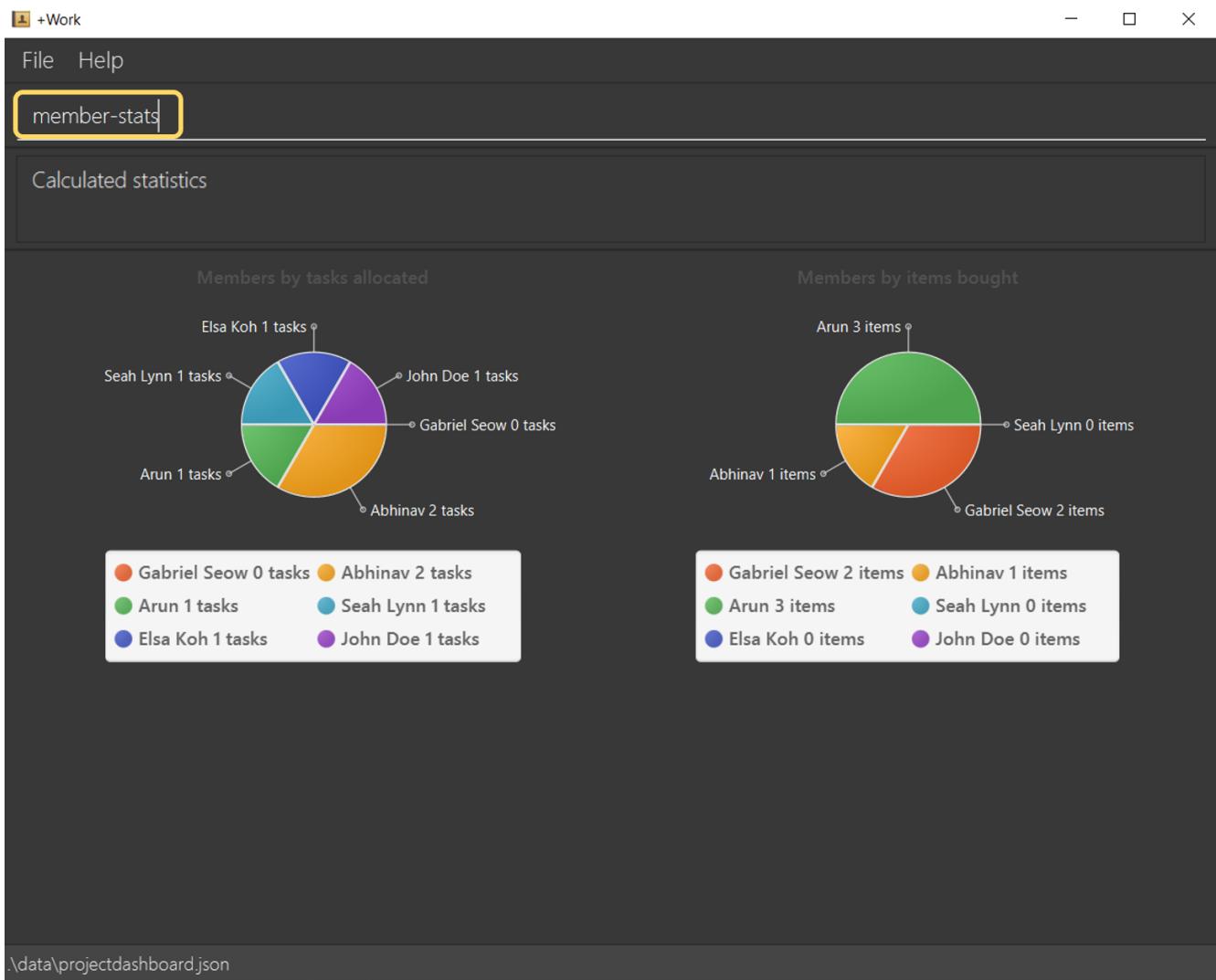
3.6. Statistics Commands

3.6.1. Getting statistics of members: `member-stats`

To get statistics relating to the members in +Work, use the statistics command following the format below.

Format: `member-stats`

Calling the `member-stats` command will result in the statistics being displayed as follows:



NOTE

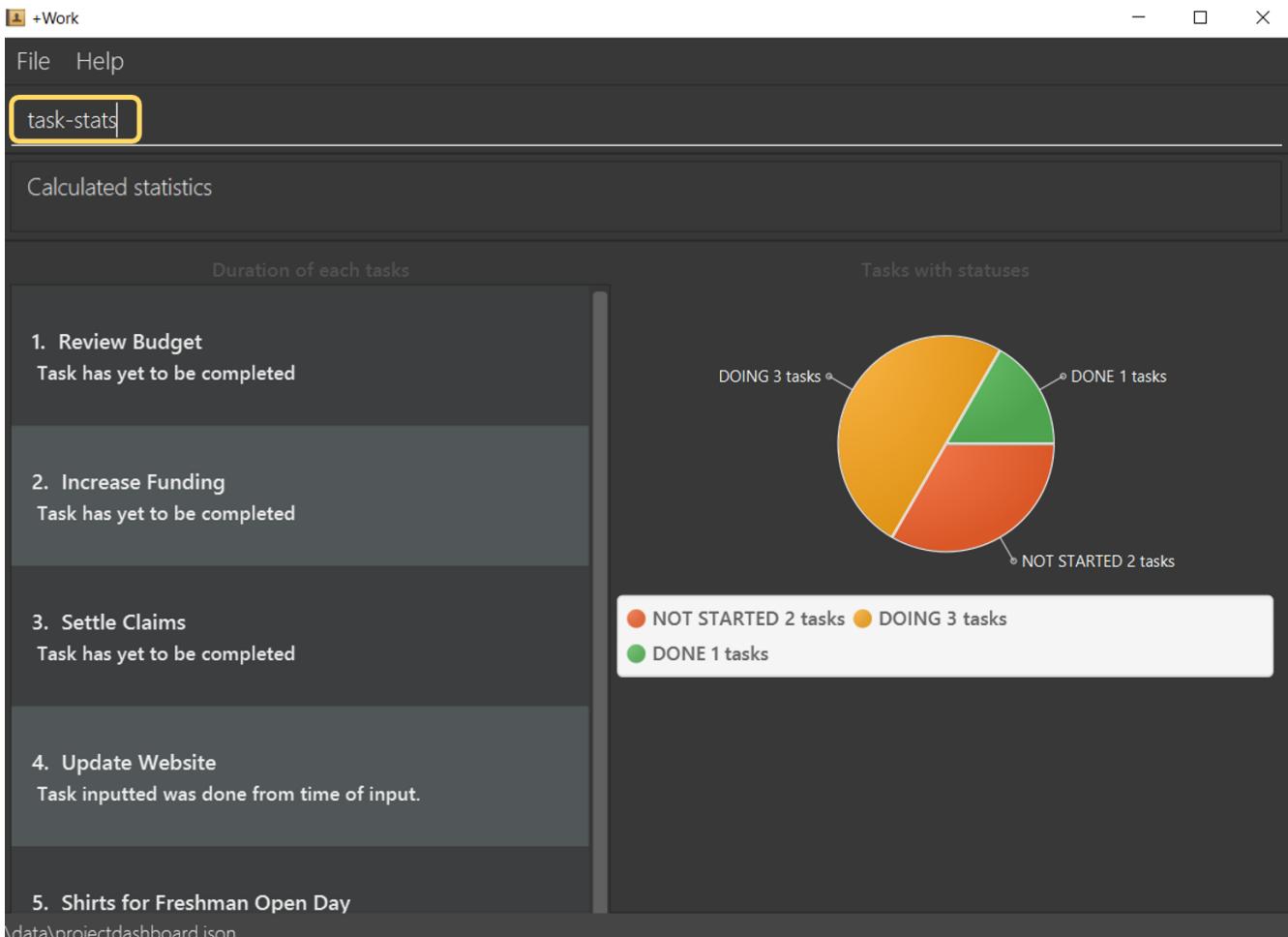
The resultant statistics displayed shows the proportion and number of tasks and inventory items allocated to each project member in +Work.

3.6.2. Getting statistics of tasks: **task-stats**

To get statistics relating to the tasks in +Work, use the statistics command following the format below.

Format: **task-stats**

Calling the **task-stats** command will result in the statistics being displayed as follows:



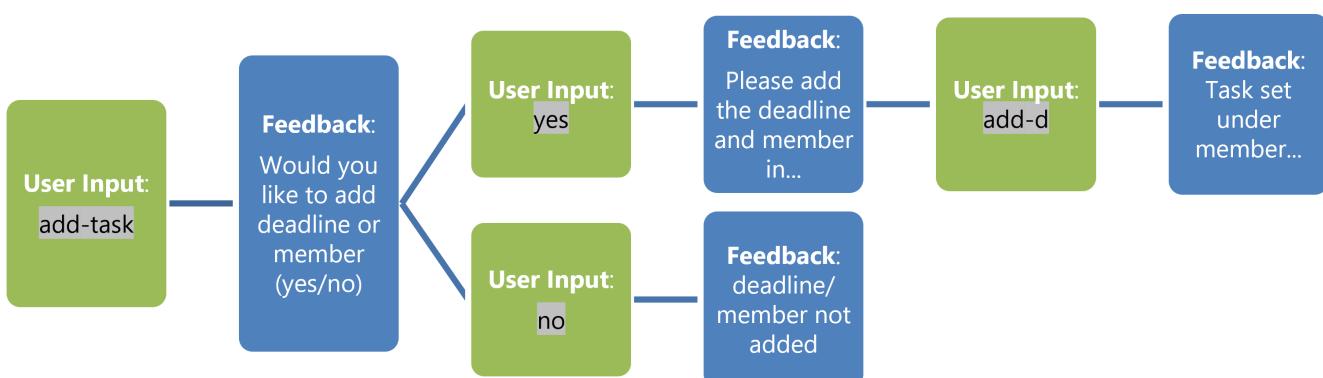
NOTE

The resultant statistics displayed shows the proportion of tasks that are undone, in progress and completed, as well as the time taken for each task inputted into +Work.

3.7. Multiline Commands

3.7.1. Adding a task: add-task

As mentioned in the earlier part of user guide, to add a task to the project, use the add-task command in the format below. This will result in a series of questions that allow you to add deadline and a member to the task. The following flowchart shows this series of feedbacks from the application and user inputs.



Format (to add task): **add-task [t/TASK_NAME]**

Format (for yes): **yes**

Format (for no): **no**

Format (to add deadline and member): **add-d [at/DEADLINE] [mi/MEMBER_ID]**

add-d command does not work unless it is typed after **add-task** and **yes**.

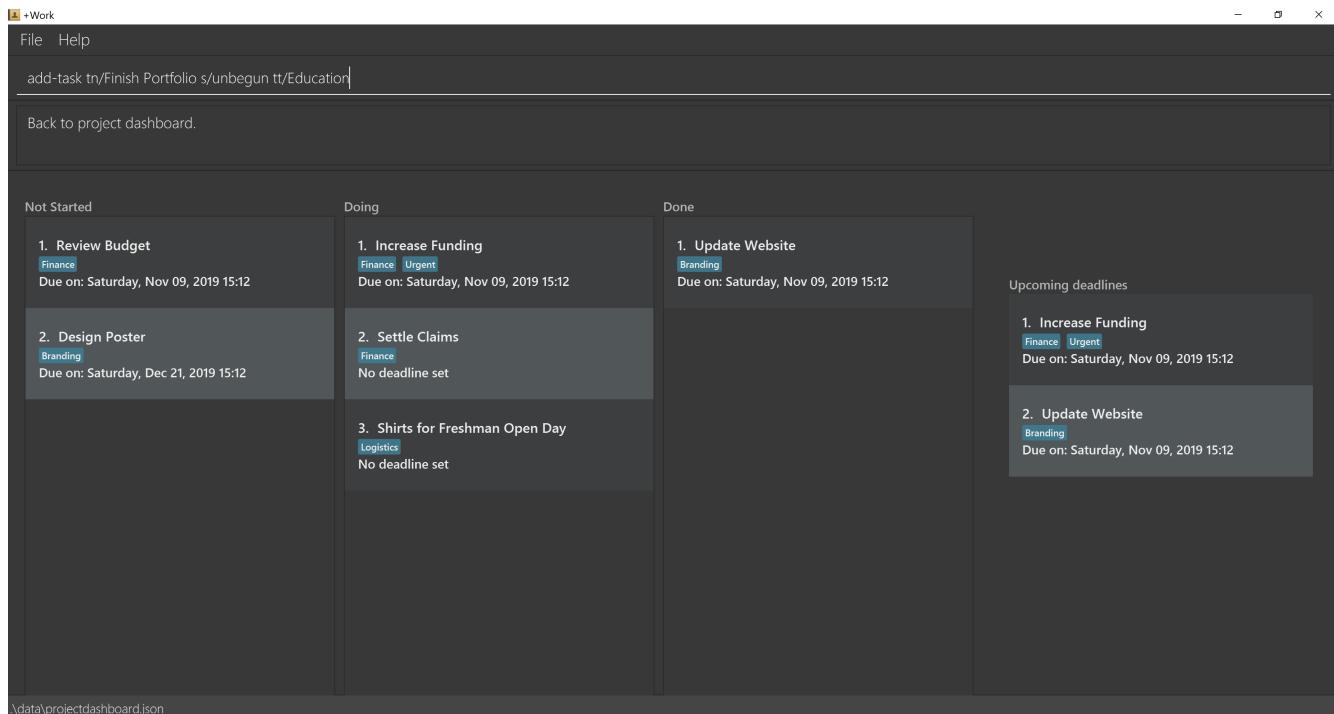
NOTE Either deadline or member or both can be added using the **add-d** command. However at least one attribute must be present.

Example:

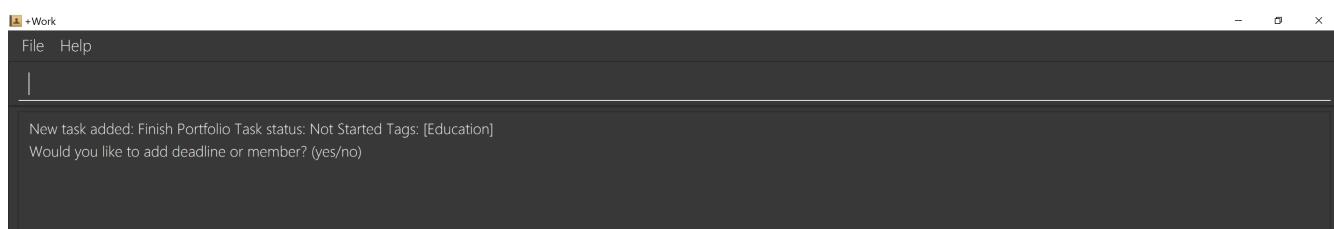
- **add-task tn/Finish Portfolio s/unbegun tt/Education**

A new task will be added to the project dashboard, which prompts a sequence of questions, as shown in the following pictures

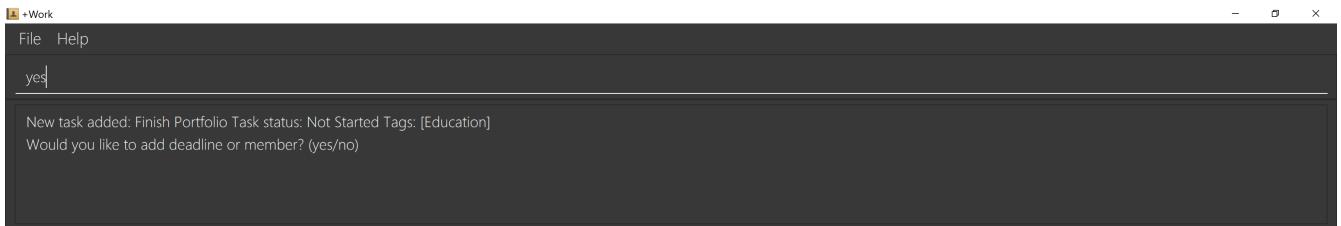
- First, type the add-task command



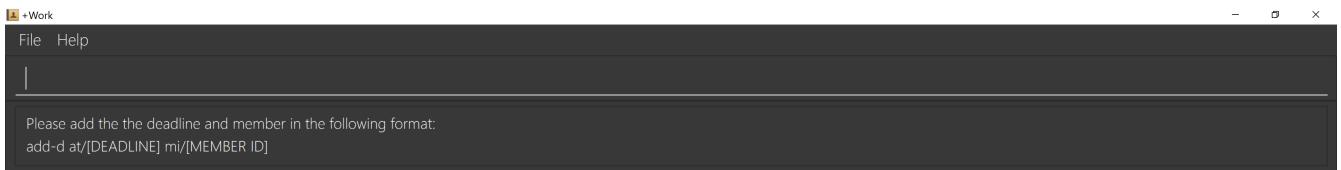
- This prompts the feedback asking for your choice to add deadline and member



- Type yes to add deadline or member



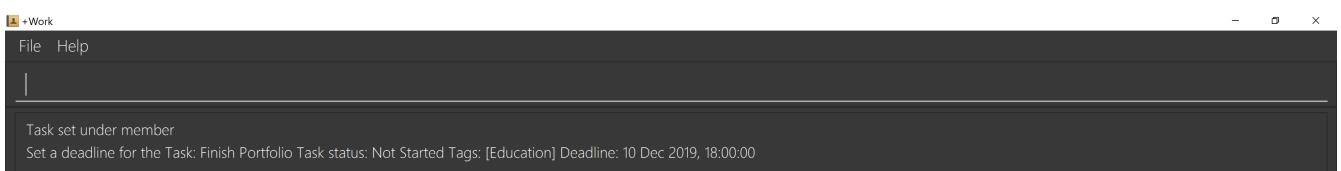
- This prompts to add the details



- Type `add-d at/10/12/2019 18:00 mi/AR` to set deadline at 10/10/2019 18:00 and assign the member with member id “AR” to the task.



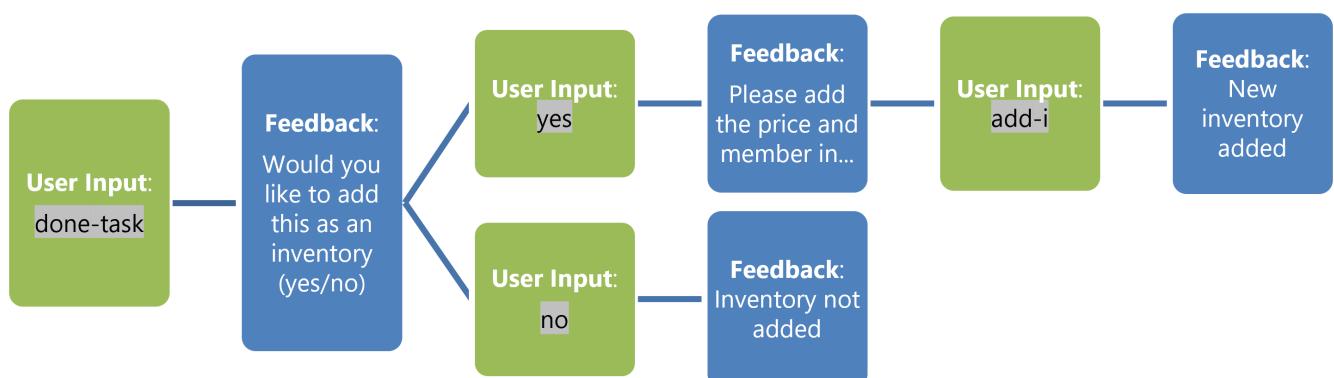
- This sets the deadline and assigns the member successfully! You should see a window like this.



3.7.2. Changing a task status to done: `done-task`

(for tasks that are tagged “Inventory”)

As mentioned in the earlier part of user guide, to change the task status to “done”, use the `done-task` command in the format below. If the task is tagged as `Inventory`, this will result in a series of questions that allow you to add the task as an inventory. The following flowchart shows this series of feedbacks from the application and user inputs.



Format (for done task): `done-task [ti/TASK_ID]`

Format (for yes): `yes`

Format (for no): `no`

Format (to add price and member): **add-i [p/PRICE] [mi/MEMBER_ID]**

NOTE

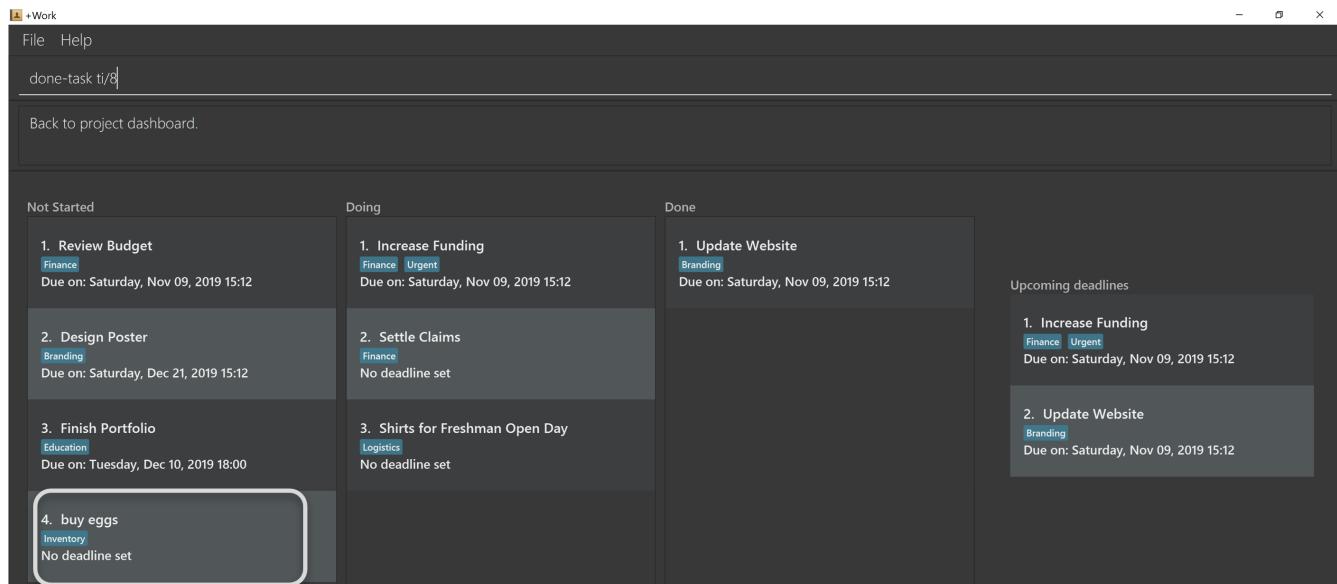
add-i command does not work unless it is typed after **done-task** and **yes**.
Unlike **add-task**, in this case both attributes price and member id are required.

Example:

- **done-task ti/8**

The task with index 8 will have its task status changed to done. Since the task is tagged as “Inventory”, this will prompt a series of questions, as shown in the pictures below.

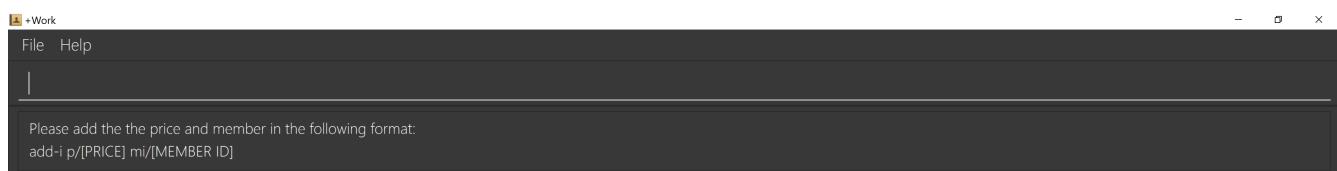
- First, type **done-task ti/8**



- This prompts the feedback asking for your choice to add it as an inventory.



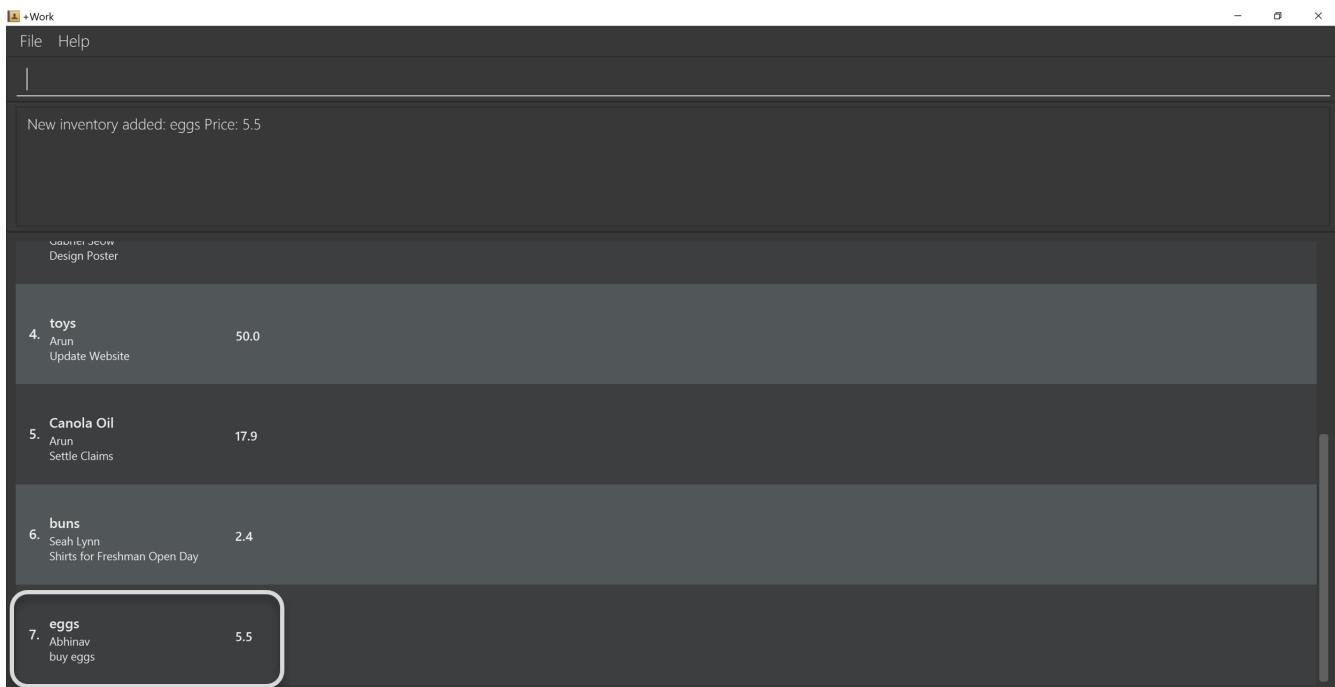
- After typing **yes** as the user input, feedback will prompt you to add the details.



- Type **add-i p/5.50 mi/AB** to set the price at \$5.50 and to assign the member with member id “AB” to the inventory



- This sets the price and assigns the member successfully! You should see a window like this.



3.8. Settings Commands

3.8.1. Switching the theme of +Work: `theme`

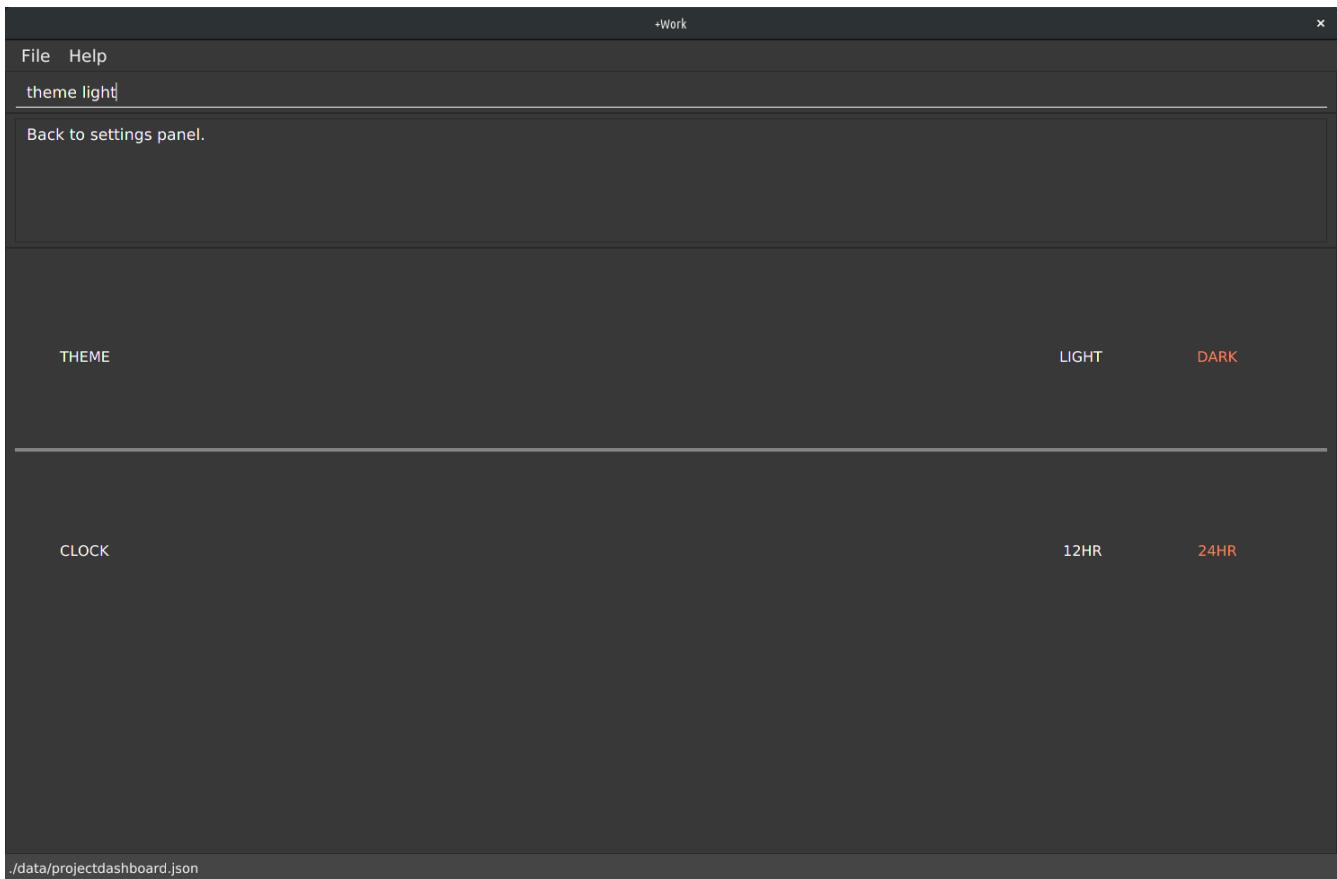
This command helps you toggle the theme of +Work between light and dark to suit your viewing preferences.

TIP By default, the theme is set to dark.

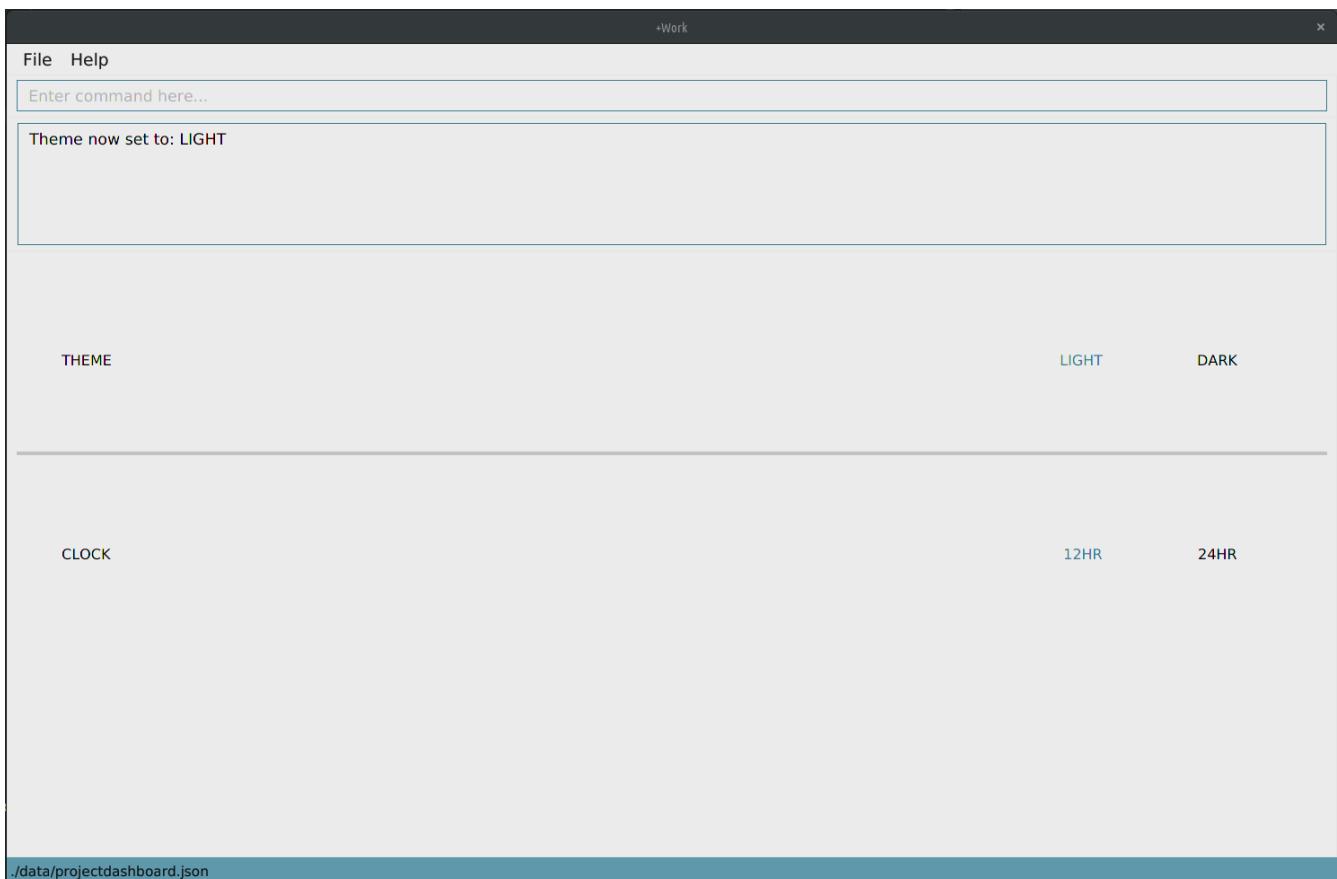
Format: `theme light`

Examples:

- Type `theme light` as shown below.



- Hit the `Enter` key and +Work switches to the light theme!



- Similarly, `theme dark` switches to the dark theme, and your choice is highlighted.

3.8.2. Switching the clock format of +Work: `clock`

This command helps you toggle the clock format of +Work between 24 hour and 12 hour format. The clock format is set to 24 hour by default.

NOTE

This does not affect the input format of deadlines for tasks, you still need to enter them in the 24 hour format!

Format: `clock twenty_four`

Examples:

- `clock twelve`
Switches to the 12 hour clock
- `clock twenty_four`
Switches to the 24 hour clock

3.9. Saving the data

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

3.10. Command Line Autocomplete

+Work will automatically prompt you on the various possible commands based on input and help you paste the correct command format into the command line if chosen.

For example, when you type `add-mem` into the command-line, +Work will prompt you to select `add-member` from the drop down and paste `add-member mn/ mi/ mt/` into your command-line automatically.

4. FAQ

Q: Can I use file formats other than ics for the calendar feature?

A: No, the file format has to be in ics folder, downloaded either from NUSmods or Google Calendar.

Q: Can I export the claims report as a word document?

A: No, the application only supports exporting of files in PDF format.

Q: What if the calendar cannot find a timing where everyone is free?

A: There will be a grid showing the number of available members by time, so it will be easier for the user to choose the best possible time for the meeting.

Q: My project has a budget, does the application help me keep track of the projects financial status?

A: Using `/inventory` the application can keep track of current expenses for each task and the member who purchased it, however, there are currently no accounting services available.

Q: Some of the tasks for my project require sub tasks to be completed, is there a way to add them?

A: Ideally since the user is the project leader, they should only see the main tasks to be accomplished for the project. This would help facilitate their managerial role within the project. Hence, the application does not support subtasks.

Q: How can I add multiple users to the same task?

A: Since +Work is member-oriented, the application displays information according to members. Hence, you have to assign task to all the different members that are working on the task.

Q: What happens when a task has been completed?

A: You can mark the task as **done**, which will move the task to the bottom of the task list. In the case where you do not want to keep the task in the list, using 'Remove Task' will delete it.

5. Command Summary

- **add-inventory [i/NAME] [p/PRICE] [ti/TASKID] [mi/MEMBER_ID]: Adds an inventory**
- **add-meeting [at/dd-mm-yyyy hh:mm - hh:mm] [l/LOCATION]: Adds a meeting**
- **add-member [mn/MEMBER_NAME] [mi/MEMBER_ID] [mt/MEMBER_TAG]: Adds a member**
- **add-task [tn/TASK_NAME] s/TASK_STATUS tt/TASK_TAG: Adds a task**
- **assign [ti/TASK_ID] [mi/MEMBER_ID]: Adds a task to a member**
- **calendar: Switches to the time management page**
- **clock twelve: Switches deadline display to the 12 hour clock format**
- **clock twenty_four: Switches deadline display to the 24 hour clock format**
- **set-deadline [ti/TASK_ID] [at/dd-mm-yyyy hh:mm]: Assigns a deadline to a task**
- **delete-inventory [ii/ITEM_ID]: Deletes an inventory**
- **doing-task [ti/TASK_ID]: Marks a task as doing**
- **done-task [ti/TASK_ID]: Marks a task as done**
- **fire-member [ti/TASK_ID] [mi/MEMBER_ID]: Removes a task from a member**
- **generate-inventory /task: Generates report of inventory by task**
- **generate-inventory /person: Generates report of inventory by person**
- **generate-timings: Generates free times among members**
- **home: Switches to the project dashboard**
- **help: Accesses the help window**
- **import-calendar [mi/MEMBER_ID] [c/PATH_TO_ICS]: (Coming in v1.3) Imports a calendar**
- **list-members: Lists all members**
- **list-tasks: Lists all tasks**
- **remove-member [mi/MEMBER_ID]: Removes a member**
- **remove-task [ti/TASK_ID]: Removes a task**
- **settings: Switches to the settings configuration page**

- `theme dark`: Switches to the dark theme
- `theme light`: Switches to the light theme

6. Upcoming Features

6.1. Cloud integration [coming in v2.0]

Collaborate online with your teammates, and store your project details such as members calendars in the cloud.

6.2. Customisable user profiles [coming in v2.0]

Create more detailed profiles for members and choose who has authority to change details of the project.