HealthyBaby - User Guide

By: `Team HYBB` Since: `Mar 2020` Licence: `MIT`

1. Introduction

HealthyBaby (HYBB) is a desktop application that provides you with a database of quick, easy, and healthy recipes. Are you a **university student** staying on campus and are tired of purchasing the same few unhealthy meals? Looking for healthier options but searching for and planning recipes are a hassle for you? HYBB is just the app for you! HYBB helps alleviate the challenges faced while maintaining healthy eating habits, easing the transition from eating out everyday to cooking healthy meals regularly.

Value proposition: Focuses on healthy, simple recipes with short cooking time with ingredients filter to minimise food wastage.

HYBB is **optimized for those who prefer to work with a Command Line Interface** (CLI) while still having the benefits of a Graphical User Interface (GUI). Through our convenient filters, searching and planning for new healthy recipes will be a breeze.

Get motivated through our eye-catching goals and achievements.

Interested? Jump to the Section 3, “Quick Start” to get started. Enjoy!

2. Overview (Harshita)

The main features of HYBB are as follows:

2.1. Filter

HYBB enables filtering by specific components in the recipes such as goals, favourites, and ingredients in order to reduce the time needed to find recipes that suit your needs.

2.2. Planning

This feature enables students to plan their meals in advance and provides an easy grocery list for your shopping needs.

2.3. Goals

This feature enables students to monitor their healthy eating progress, HYBB provides a history of all cooked meals along with statistics based on 4 main goals: “Wholesome Wholemeals”, “Bulk like a Hulk”, “Herbivore” and "Fruity Fiesta".

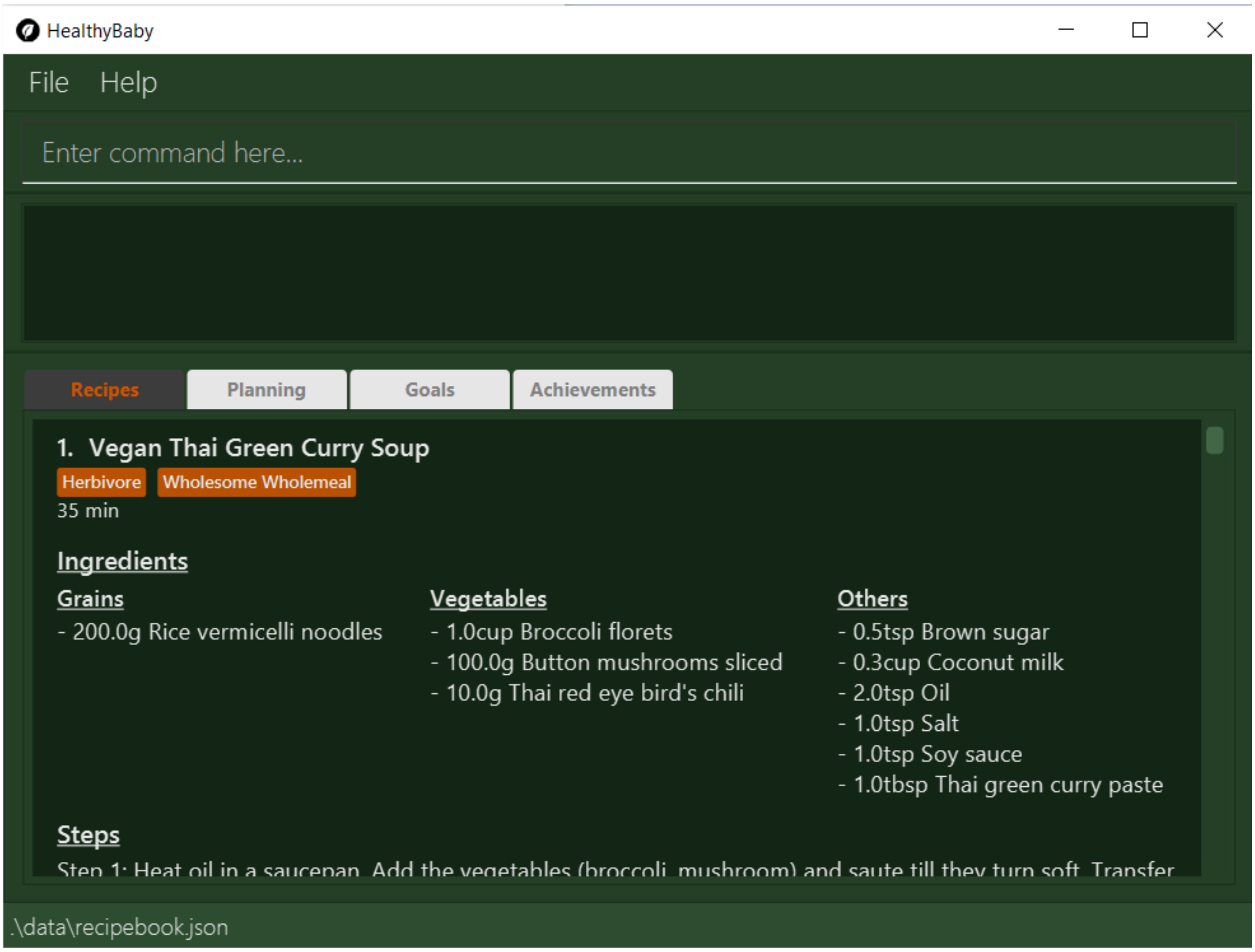
2.4. Achievements

This feature aims to encourage students to continue their healthy eating by showing visual evidence of their progress through streaks and daily motivational quotes.

Back to Top

3. Quick Start

1. Ensure you have `Java 11` or above installed in your Computer.
2. Download the latest `healthybaby.jar` here.
3. Copy the file to the folder you want to use as the home folder for your HealthyBaby application.
4. Double-click the file to start the app. The Graphical User Interface should appear in a few seconds.



1. Vegan Thai Green Curry Soup

Herbivore Wholesome Wholemeal

35 min

Ingredients

Grains	Vegetables	Others
- 200.0g Rice vermicelli noodles	- 1.0cup Broccoli florets	- 0.5tsp Brown sugar
	- 100.0g Button mushrooms sliced	- 0.3cup Coconut milk
	- 10.0g Thai red eye bird's chili	- 2.0tsp Oil
		- 1.0tsp Salt
		- 1.0tsp Soy sauce
		- 1.0tbsp Thai green curry paste

Steps

Step 1: Heat oil in a saucepan. Add the vegetables (broccoli, mushroom) and saute till they turn soft. Transfer

.\data\recipebook.json

5. Enter a command in the command box to execute it. The result of the command will appear in the result box directly below the command box.

   e.g. Entering `help` will open the help window.

6. Some example commands you can try:

   - `list` : lists all recipes
   - `delete 3` : deletes the 3rd recipe shown in the current list
   - `undo` : undoes the previous action
   - `exit` : exits the app

7. Refer to Section 7, “Basic Commands” for details of each command.

Back to Top

4. The Anatomy of a Recipe

You can store the following information in a recipe:
Name - Name of the recipe
Time - Time taken for the recipe to be cooked, measured in minutes
Ingredients - Ingredients used in the recipe, broken down into five categories (Grains, Vegetables, Proteins, Fruits, Others)
Steps - Steps taken to cook the meal

Goals - Goal that the recipe falls under

4.1. Goals (Harshita)

HYBB supports the following goals that model the Healthy Eating Plate. Goals are automatically added to a recipe you add or edit if they meet the recommended nutritional quantity specified for their respective food group.

Goal	Food Group represented by Goal
Herbivore	Vegetables
Fruity Fiesta	Fruits
Bulk like the Hulk	Healthy Proteins
Wholesome Wholemeals	Whole Grains

4.2. Ingredients (Beatrice)

You can use the following units to represent the required amount for each ingredient.

Unit	Description
g	Grams
ml	Milliliters
tbsp	Tablespoon
tsp	Teaspoon
cup	Cups

Back to Top

5. HealthyBaby’s User Interface (Beatrice)

Upon opening **HYBB**, you will see our Graphical User Interface.

[AnnotatedUserInterface] | *AnnotatedUserInterface.png*

Figure 1. HealthyBaby’s User Interface

- The **Command Box** is where you can enter commands to tell HYBB what to do.
- The **Result Box** is where you can see whether the command you have given was successful or unsuccessful.
- The **Tab Panel** can be clicked to switch between tabs.
  Alternatively, if you wish to switch between tabs by typing instead of clicking, you can type `switch [tab name]` into the Command Box instead.
  Click here for more information on the `Switch` command.
- The **Display Panel** displays information for the current tab you are on.

Back to Top

6. Notations Used

6.1. List of Tags

You can use the following tags to converse with HYBB.

Tag	Description
n/	Name of recipe
t/	Time taken to prepare and cook recipe, measured in minutes
ig/	Grain ingredient (e.g. rice, bread, spaghetti)
iv/	Vegetable ingredient (e.g. spinach, cabbage, carrot)
ip/	Protein ingredient (e.g. chicken, salmon, tofu)
if/	Fruit ingredient (e.g. grapes, honeydew, watermelon)
io/	Other ingredient, for any other ingredient that do not belong in the above 4 categories (e.g. oyster sauce, pepper, sesame oil)
s/	Steps of the recipe

6.2. Legend

From sections 6 to 11, the following notations may be used.

Notation	Meaning
[ ]	Necessary field
< >	Optional field. e.g. `n/name <s/step>` can be used as `n/Spicy Chicken s/Step 1` or just `n/Spicy Chicken`
[< >]	At least one of the optional fields is necessary
…	One or more of this same field can be added. e.g. if the command specifies `<ip/protein>…`, you may either leave the field empty, or you may specify one or more protein ingredients. e.g. if the command specifies `[recipe index]…`, you may specify one or more *recipe indexes*.

Notation	Meaning
▯	Tips for usage and things to take note of

TIP Parameters can be in any order e.g. if the command specifies `n/name t/time`, `t/time n/name` is also acceptable.
Also, commands are case-insensitive, but tags are case-sensitive.

Back to Top

7. Basic Commands

7.1. Viewing help : `help`

Gives you a short summary of all the available commands.
Format: `help`

7.2. Switch the current tab: `switch` (Harshita)

Switches the current tab you are in and changes the display.
Format: `switch` [tab name]

Example: `switch planning`
Switches the current tab to the planning tab.

TIP Tabs available: recipes, planning, goals, achievements.

7.3. List all recipes: `list`

Lists all the recipes present in the database.
Format: `list`

7.4. Find a recipe: `find` (Brian)

Searches for existing recipes by their names using the keyword(s) that you have specified.
Format: `find </strict> [keyword] <keyword>…`

Example 1: `find /strict Avocado Chicken`
Finds recipes that contain the words "Avocado" **or** "Chicken" in their names (ie. only one of them has to be present).

TIP

Using `/strict` will treat all subsequent keywords as **separate keywords** (separated by the space), as seen in Example 1.

It will also search for an **exact match** of each specified keyword (e.g. If you are searching for a recipe that has "Avocado" in its name, `find /strict Avo` *will not* be able to find it. Instead, use `find /strict Avocado` or see Example 2)

Example 2: `find Avocado Chicken`
Finds recipes that contain the single keyword "Avocado Chicken".

TIP

Without `/strict`, all the keywords will be taken as a **single keyword** and it **does not** require an exact match for a recipe to be found (e.g. `find Avo` will be able to find recipes that has "Avocado" in its name)

7.5. Add a recipe: `add`

Adds a recipe to the recipe book
Format: `add [n/name] [t/time] [<ig/grain>… <iv/vegetable>… <ip/protein>… <if/fruit>… <io/other>…] <s/step>…`

Example: `add n/Chicken Rice t/30 ip/300g, Chicken Thigh ig/300g, Rice s/Boil chicken s/While chicken is cooking, add sesame oil and crushed ginger into rice and cook it s/When chicken is done, dip it into iced water s/Serve while rice is hot`
Adds a new recipe entry that contains the following description:

[AddExampleOne] | *AddExampleOne.png*

Figure 2. Expected output

WARNING

Multiple steps and multiple ingredients for a recipe can be added.
However, multiple 'name' and 'time' cannot be added.
HYBB will take the latter input for these fields.
For example, `add n/Chicken Rice n/Chicken with Rice t/25 t/30, ip/300g, Chicken Thigh` would create a new recipe with the name Chicken with Rice and the time 30 minutes.

TIP

To create a recipe, you need to specify the **name and time** of the recipe and **at least one ingredient** that was used.

TIP

For recipes with many ingredients and steps, you can add them separately using our recipe customation commands.

7.6. Edit a recipe : `edit`

Edits an existing recipe. This is the command used for editing entire fields at a time.
Format: `edit [recipe index] [<n/name> <t/time> <ig/grain>… <iv/vegetable>… <ip/protein>… <if/fruit>… <io/other>… <s/step>…]`

Example: `edit 4 n/Chicken Rice t/20`
Renames the 4th recipe to Chicken Rice, and sets the preparation time to 20 minutes (from whatever amount it was before).

WARNING Using this command to edit a field that might have multiple entities like ‘vegetable ingredients’ or 'steps' will overwrite the entire field. For example, if the 4th recipe currently has a list of 5 vegetable ingredients, running `edit 4 iv/50g, Lettuce` will replace the **entire** list of vegetable ingredients with only 50g of Lettuce.

TIP To make changes to a single entity in a field without having to rewrite everything, please refer to our recipe customation commands.

WARNING HYBB will take the latter input if more than one 'name' or more than one 'time' is given.
For example, `edit 4 n/FirstName n/SecondName t/10 t/20` would change the name to SecondName and the time to 20 minutes for the recipe at *recipe index* 4.

7.7. Delete a recipe : `delete` (Brian)

Deletes the recipe(s) that you have specified.
Format: `delete [recipe index] <recipe index>…`

Example: `delete 1 3 4`
Deletes recipe 1, 3, and 4 from the recipe book.

TIP You can **delete, favourite, or unfavourite** multiple recipes at the same time (at least one recipe must be selected).

TIP If you specify a recipe number that does not exist in the list of recipes *that you are currently viewing*, none of your selected recipes will be deleted, favourited, or unfavourited.

7.8. Favourite a recipe: `favourite` (Brian)

Favourites the recipe(s) that you have specified.
Format: `favourite [recipe index] <recipe index>…`

Example: `favourite 1 3 4`
Favourites recipes 1, 3, and 4.

7.9. Unfavourite a recipe: `unfavourite` (Brian)

Unfavourites the recipe(s) that you have specified.
Format: `unfavourite [recipe index] <recipe index>…`

Example: `unfavourite 1 3 4`
Unfavourites recipes 1, 3, and 4.

7.10. Undo a previous action: `undo` (Brian)

Undoes a previous action.
Format: `undo <number of actions> or <all>`

Example 1: `undo`
Undoes only the previous action.

Example 2: `undo 3`
Undoes the 3 previous actions. If there are less than 3 actions to undo, you will not be allowed to undo.

Example 3: `undo all`
Undoes all previous actions.

TIP For both undo and redo, the number of actions that you specify (if any) has to be a non-zero unsigned integer!

7.11. Redo a previous undo: `redo` (Brian)

Restores actions that have been previously undone using `undo`. Format: `redo <number of actions> or <all>`

Example 1: `redo`
Restores the actions that have been undone by the previous undo.

Example 2: `redo 3`
Restores the actions that have been undone by the previous 3 undo-s. If there are less than 3 actions to restore, you not be allowed to redo.

Example 3: `redo all`
Restores the actions that have been undone by all previous undo-s.

7.12. Clear the recipe book : `clear`

Clears the recipe book. After entering this command, the recipe book will be empty and all planned recipes will be removed.
Format: `clear`

WARNING All recipes and plans, including default ones, will be removed.

TIP You can `undo` this command if it was performed by accident.

7.13. Exit the program : `exit`

Saves HYBB and exits the program.
Format: `exit`

7.14. Saving the data

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

Back to Top

8. Recipe Customisation Commands (Brian)

As we see from the `edit` command in section 6.6, if you want to add, edit, or delete a single ingredient or step, you would have to rewrite the whole field that you wish to edit. This would be troublesome if the field contains multiple ingredients or steps that you may not necessarily want to edit. Therefore, the following commands are used in occasions like these!

8.1. Add Ingredients to a Recipe: `addIngredient` (Brian)

Adds more ingredients to an existing recipe.
Format: `addIngredient [recipe index] [<ig/grain>… <iv/vegetable>… <ip/protein>… <if/fruit>… <io/other>…]`

Example: `addIngredient 2 ig/50g, Bread io/5g, Butter`
Adds 50g of Bread and 5g of Butter to recipe 2.

TIP You can add multiple ingredients at a time (at least one ingredient must be added).

TIP If you add an ingredient that already exists in the recipe, that existing ingredient will be replaced with the new one.

8.2. Edit Ingredient Quantity in a Recipe: `editIngredient` (Brian)

Edits the quantity of an ingredient in an existing recipe.
Format: `editIngredient [recipe index] [<ig/grain>… <iv/vegetable>… <ip/protein>… <if/fruit>… <io/other>…]`

Example: `editIngredient 3 ig/50g, Bread`
Searches for Bread in recipe 3 and changes its quantity to 50g. An error message will appear if Bread does not exist in recipe 3's ingredients set.

TIP You can edit multiple ingredients at a time (at least one ingredient must be edited).

8.3. Delete Ingredients in a Recipe: `deleteIngredient` (Brian)

Deletes the specified ingredient(s) from an existing recipe.
Format: `deleteIngredient [recipe index] [<ig/grain name>… <iv/vegetable name>… <ip/protein name>… <if/fruit name>… <io/other name>…]`

Example: `deleteIngredient 3 ig/Rice iv/Kailan`
Searches for Rice and Kailan in recipe 3 and deletes them. An error message will appear if Rice and/or Kailan does not exist in recipe 3's ingredients set.

TIP You can delete multiple ingredients at a time (at least one ingredient must be deleted).

TIP There is no need to specify quantity here. Just the ingredient name will do!

TIP If you type an ingredient prefix (e.g. "ig/") but don't specify any ingredients, all ingredients of that type will be deleted.

8.4. Add Steps to a Recipe: `addStep` (Brian)

Adds more steps to an existing recipe.
Format: `addStep [recipe index] [s/step] <s/next step>…`

Example: `addStep 1 s/New step s/Another new step`
Adds 2 new steps to recipe 1.

TIP You can add multiple steps at a time (at least one step must be added).

8.5. Edit a Step in a Recipe: `editStep` (Brian)

Edits the specified step in an existing recipe.
Format: `editStep [recipe index] [step index] [s/new step]`

Example: `editStep 3 4 s/Edited new step`
Replaces step 4 of recipe 3 with "Edited new step". If you specify more than one step, only the first one will be used to replace the old step.

8.6. Delete Steps in a Recipe: `deleteStep` (Brian)

Deletes the specified step(s) from an existing recipe.
Format: `deleteStep [recipe index] [step index] <step index>…`

Example: `deleteStep 3 2 3 5`

Deletes steps 2, 3, and 5 of recipe 3.

TIP You can delete multiple steps at a time (at least one step must be deleted).

Back to Top

9. Filter Command (Brian)

This command is not to be confused with the `Find` command, which only searches for recipes by their names. The `Filter` command is a more robust search command that allows you to search for recipes using *various criteria*.

TIP You can combine the input of the next few subsections to filter the recipes by multiple criteria!

9.1. Filter by ingredients (Brian)

Finds recipes that contains the specified ingredients.
Format: `filter <ig/grain>… <iv/vegetable>… <ip/protein>… <if/fruit>… <io/other>…`

Example 1: `filter ig/Rice iv/Cabbage`
Finds recipes that contains Rice and Cabbage.

Example 2: `filter ig/exclude Pasta ip/Chicken`
Finds recipes that **does not** contain Pasta and contains Chicken.

TIP Notice the use of the keyword "exclude" in Example 2? Use this to exclude ingredients that you do not want!

9.2. Filter by goals (Brian)

Finds recipes that are tagged with the specified goal.
Format: `filter [g/goal] <g/goal>…`

Example: `filter g/Herbivore`
Finds recipes that are tagged with the `Herbivore` goal.

9.3. Filter by favourites (Brian)

Finds recipes that are tagged as favourites.
Format: `filter favourites`

9.4. Filter by preparation time (Brian)

Finds recipes that have preparation time less than or equals to the specified time (in minutes).
Format: `filter [t/time] or [t/time range]`

Example 1: `filter t/15`
Finds recipes that have 15 minutes or less of preparation time.

Example 2: `filter t/20-30`
Finds recipes that have 20 to 30 minutes (inclusive) of preparation time.

Back to Top

10. Planning Command (Beatrice)

Most of the planning commands take place in the planning tab.
Switch to the planning tab by typing `switch planning` or by clicking on 'Planning' from the **Tab Panel**.

After switching to the 'Planning' tab, you should get the following interface:

[PlanTab] | *PlanTab.png*

Figure 3. User Interface of the Planning Tab

The **Display Panel** for the 'Planning' tab lists all the plans that you have made in chronological order, with the plans with earlier dates at top of the list.

You can the see the details of each plan from this list:

[PlanTab Plan] | *PlanTab_Plan.png*

Figure 4. Details of each plan

WARNING

Look out for the two different indexes used in this section: *plan indexes* and *recipe indexes*.
The indexes in the 'Planning' tab are *plan indexes*, whereas the indexes in the 'Recipes' tab are *recipe indexes*.
Each type of index is intended to be used for different commands.
Still unclear? Click here to find out more.

10.1. Plan recipes: `plan` (Beatrice)

Plan recipe(s) that you would like to cook on a certain day.
Format: `plan [recipe index]… [d/yyyy-mm-dd]`

Example 1: `plan 1 d/2020-05-27`
Adds a new plan on 27 May 2020 for the recipe at *recipe index* 1.

Example 2: `plan 1 2 3 d/2020-05-27`
Adds three new plans on 27 May 2020 for the recipes at *recipe indexes* 1, 2 and 3.

WARNING

At least one *recipe index* is required.
Valid indexes are integer numbers ranging from 1 to the current size of the plan list. The command will not be successful if one of the given *recipe indexes* was invalid.

WARNING

Valid dates start from today's date and are in the format yyyy-mm-dd. This means that we cannot make a plan on dates later than today.

TIP

You can specify as many *recipe indexes* as you want. New plans will be added for each recipe.

10.2. Delete plans: `deletePlan` (Beatrice)

Delete plan(s) and remove them from the plan list.
Format: `deletePlan [plan index]…`

Example 1: `deletePlan 3`
Deletes the plan on *plan index* 3.

Example 2: `deletePlan 3 5 8`
Deletes the plans on *plan indexes* 3, 5 and 8.

WARNING

At least one *plan index* is required.
Valid indexes are integer numbers ranging from 1 to the current size of the plan list. The command will not be successful if one of the given *plan indexes* was invalid.

TIP

You can specify as many *plan indexes* as you want. Each plan will be deleted.

TIP

Using the `cooked` command will automatically delete today's plan for the cooked recipe.

10.3. Clear all plans: `clearPlan` (Beatrice)

Clears all the plans that you have made.
Format: `clearPlan`

WARNING

All plans will be deleted.

TIP

You can `undo` this command if it was performed by accident.

10.4. Obtain ingredients for plans: `groceryList` (Beatrice)

Lists the ingredients needed for all recipes that have been planned.
You can click on the 'Copy' button to copy all the ingredients into your clipboard.

Paste this list onto a notepad and bring it with you for a convenient grocery list at the market.

Format: `groceryList`

WARNING If no plans have been made, the grocery list will not be generated.

10.5. View this week's plans: `viewWeek` [Coming in v2.0]

Display only plans that were made for this week.
Format: `viewWeek`

10.6. View this month's plans: `viewMonth` [Coming in v2.0]

Display only plans that were made for this month.
Format: `viewMonth`

Back to Top

11. Goals Command (Harshita)

11.1. Remove goal: `deleteGoal` (Harshita)

Deletes a goal that has been auto-generated through the addition or editing of a recipe from a specific recipe.
Format: `deleteGoal [recipe index] [goal]`

Example 1: `deleteGoal 2 Bulk like the Hulk`
Deletes goal named Bulk like the Hulk from the 2nd recipe if it exists.

TIP Ensure that the capitalisation of the goal is the same! Check out `listGoals` if unsure.

11.2. List goals being tracked: `listGoals` (Harshita)

Lists all goals tracked by HYBB.
Format: `listGoals`

Example 1: `listGoals`
Displays the 4 main goals being tracked in the **Result Box**.

11.3. Track cooked meals: `cooked` (Harshita)

Tracks all cooked meals and adds to the progress of their respective goals.
Format: `cooked [recipe index] <recipe index>…`

Example 1: `cooked 2`
Marks the 2nd recipe as cooked.

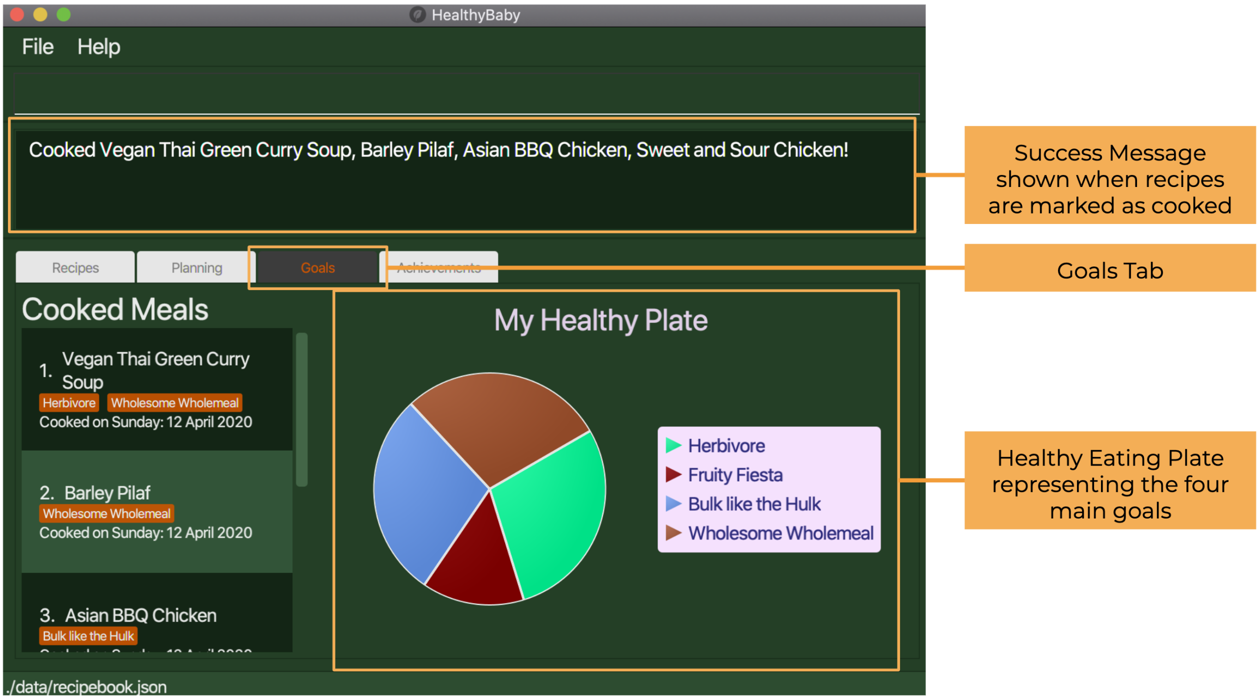
> **TIP** You can mark multiple recipes as cooked at the same time (at least one recipe must be marked).

Example 2: `cooked 3 4 8`
Marks the 3rd, 4th and 8th recipe as cooked.

> **WARNING** If you specify a recipe number that has already been cooked within the day or does not exist in the list of recipes *under the recipes tab*, none of your selected recipes will be marked as cooked.

> **WARNING** If a plan for the recipe being cooked exists today, using `cooked` will automatically delete these plans.

> **TIP** A history of all cooked meals, along with a pie chart based on your meals cooked, can be found under the goals tab.



Cooked Vegan Thai Green Curry Soup, Barley Pilaf, Asian BBQ Chicken, Sweet and Sour Chicken!

Cooked Meals

1. Vegan Thai Green Curry Soup
Herbivore Wholesome Wholemeal
Cooked on Sunday: 12 April 2020

2. Barley Pilaf
Wholesome Wholemeal
Cooked on Sunday: 12 April 2020

3. Asian BBQ Chicken
Bulk like the Hulk

./data/recipebook.json

Success Message shown when recipes are marked as cooked

Goals Tab

Healthy Eating Plate representing the four main goals

Figure 5. Goal tab

> **TIP** The pie chart represents the goals tally according to your cooked meal history and models the Healthy Eating Plate!

11.4. Show progress of goals: `chart` [Coming in v2.0]

Shows weekly progress of respective goal through graph.
Format: `chart [g/goal]`

11.5. Analyse images added: `analyse` [Coming in v2.0]

Auto-generates goals and statistics through the use of Artificial Intelligence.

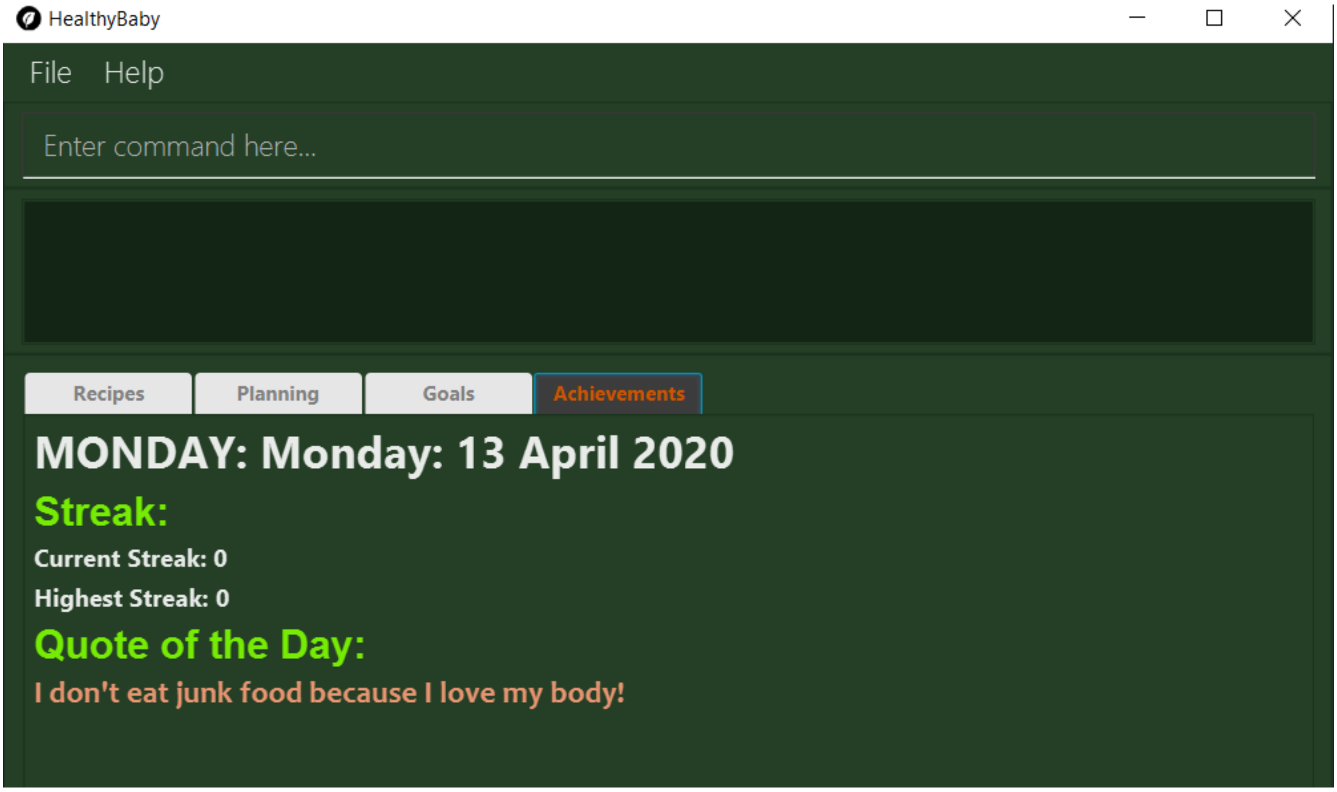
Back to Top

12. Achievement Command (Yue Tong)

12.1. Add motivational quote : `quote`

Adds a quote by the user to the quote database to store for future(randomised) display of quote in Achievement tab. Format: `quote` [quote]

Example 1: `quote Today was yesterdays tomorrow`
Adds quote "Today was yesterday’s tomorrow" to quote database.



MONDAY: Monday: 13 April 2020

Streak:

Current Streak: 0

Highest Streak: 0

Quote of the Day:

I don't eat junk food because I love my body!

Figure 6. Achievement tab

12.2. Display streak: `streak` [Coming in v2.0]

Displays line chart of all logged streaks by user Format: `streak`

12.3. Remove quote: `remove quote` [Coming in v2.0]

Removes quote indicated by user
Format: `remove quote` [quote]

12.4. Remove quote: `change quote` [Coming in v2.0]

Changes currently displayed quote to a new randomised one
Format: `change quote`

Back to Top

13. FAQ (Harshita)

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 HYBB folder.

Q: How does the auto-generation of goals work?
A: HYBB analyses the ingredients added or edited in a recipe and auto-generates goals for you. The recipe being tagged by a goal suggests that the minimum requirement per meal for the respective food group, as recommended by the National Institute of Health, has been met.

Q: Why is there a deleteGoal command and no addGoal command?
A: As this is the beta version of HYBB, the auto-generation of goals is designed to either accurately calculate quantity or err on the side of overestimation. This would allow you to remove any goal you deem inaccurate based on your own judgement. You can look forward to future releases which will integrate artificial intelligence into HYBB to auto-generate goals more accurately!

Q: Why are there no safeguards for adding ingredients?
A: Currently, HYBB is designed to help you manage your customized recipe list. Hence, the addition of ingredients and under which tag(ip/ iv/ ig/ if/ io/) they fall would be solely based on your judgement! For future releases, we will ensure more safeguards to prevent invalid inputs based on common knowledge.

Q: What are *recipe indexes*, *step indexes* and *plan indexes*?
A: Recipe indexes_ are the numbers beside each recipe in the 'Recipes' tab.
Step indexes_ are the numbers beside each step in a Recipe.
Plan indexes_ are the numbers beside each plan in the 'Planning' tab.

For example, in the diagram below, 'Asian BBQ Chicken' is at *recipe index* 1, with the step "In a bowl, mix all the ingredients ..." at *step index* 1.

[Recipe Index] | *Recipe_Index.png*

Figure 7. Recipe Index in Planning Tab

In the diagram below, 'Vegan Thai Green Curry Soup' is at *plan index* 1.

[Plan Index] | *Plan_Index.png*

Figure 8. Plan Index in Planning Tab

14. Glossary (Harshita)

Term	Definition
Command Line Interface	Text-based interface used for entering commands.
Healthy Eating Plate	Refers to My Healthy Plate designed for Singaporeans by the Health Promotion Board. It is a visual guide that shows you what to eat in the right amounts for each meal, so that you can plan your portions accordingly.

Back to Top

15. Command Summary

Command	Action
`help`	View help
`switch`	Switch to the desired tab
`list`	List all recipes
`find`	Find recipes by name
`add`	Add a recipe
`edit`	Edit a recipe
`delete`	Delete recipes
`favourite`	Favourite recipes
`unfavourite`	Unfavourite recipes
`clear`	Clear the recipe book
`exit`	Exit the programme
`addIngredient`	Add ingredients to a recipe
`editIngredient`	Edit ingredients’ quantities in a recipe
`deleteIngredient`	Delete ingredients in a recipe
`addStep`	Add steps to a recipe
`editStep`	Edit a step in a recipe
`deleteStep`	Delete steps in a recipe

Command	Action
`filter`	Search for recipes by (multiple) criteria
`plan`	Plan recipes
`deletePlan`	Delete plans
`clearPlan`	Clear all plans
`groceryList`	List all ingredients used in the planned recipes
`deleteGoal`	Deletes specified goal from a recipe
`listGoals`	Lists all available goals
`cooked`	Records a recipe as cooked
`quote`	Adds a new quote

Back to Top