

Coupon Stash v1.4 - User Guide

1. Introduction	3
2. About This Document	3
3. Quick Start	4
4. User Interface	5
4.1. Tabs	5
4.2. Calendar	10
4.3. Command Input Box	11
4.4. Command Result Box	11
5. Features	11
5.1. Adding a coupon: add	12
5.2. Editing a coupon: edit	14
5.3. Deleting a coupon: delete	14
5.4. Listing coupons: list (Denise)	15
5.5. Using a coupon: used (Chester)	15
5.6. Viewing savings: saved (Alden)	16
5.7. Archiving a coupon: archive (Chester)	17
5.8. Unarchiving a coupon: unarchive (Chester)	17
5.9. Finding coupons by name: find	18
5.10. Listing all expiring coupons: expiring (Denise)	18
5.11. Sorting coupons: sort (Nelson)	19
5.12. Copying a coupon: copy (Denise)	20
5.13. Sharing a coupon: share (Chester)	21
5.14. Undoing the previous command: undo (Nelson)	21
5.15. Reversing an undo: redo (Nelson)	22
5.16. Going to a month-year on the calendar: goto (Denise)	23
5.17. Expanding a coupon: expand (Alden)	23
5.18. Setting currency symbol: setcurrency (Alden)	24
5.19. Viewing help: help (Alden)	25
5.20. Clearing all entries: clear	25
5.21. Exiting the program: exit	25
5.22. Reminding of coupon expiry (Kevin)	25
5.23. Saving the data	27
6. Keyboard Shortcuts (Nelson)	27
6.1. Restore previously entered command texts: ↑ and ↓	27
6.2. Close expanded coupon or reminder windows: Ctrl +q	30
6.3. Cycle through tabs: Ctrl +Tab	34
7. FAQ	34
8. Command Summary	35

9. Appendix	36
9.1. Query expressions (coming in v2.0)	36



By: Team F09-1

Welcome to the Coupon Stash User Guide! Thank you so much for choosing Coupon Stash, the world's most popular command-line coupon management desktop application*!

This guide will help you get started quickly, providing useful tips and hints along the way.

*as of April 2020



Figure 1. Picture of Stasher (Coupon Stash's Mascot)

Hi! I'm Stasher, the mascot of Coupon Stash. Together, we are going to save so much money! Let me guide you to use this wonderful application.

1. Introduction

Coupon Stash is meant for School of Computing (SoC) students and staff who enjoy managing their coupons with a desktop command-line interface. It is optimised for efficient usage in this aspect, while still offering a sleek, beautiful Graphical User Interface (GUI).

Jump to the [Quick Start section](#) to get started now. We sincerely hope you enjoy using Coupon Stash!

2. About This Document

This document introduces the features of Coupon Stash.

The following are symbols and formatting used in this document:



Tips are used to describe small features or techniques that may come in handy when using Coupon Stash!



Warnings bring your attention to certain practices that may have unintended consequences!



Notes describe certain features or behaviour that may not be so obvious!

Boxes like these contain useful information, for emphasis and easy reference!

- Coupon Stash command words are formatted with **code blocks**.
- Blue, underlined words are [links](#)! Try clicking on the link on the left!
- Keyboard buttons are represented in a box, and used to represent certain shortcuts in Coupon Stash. For example, the `Ctrl` key, which is used for most shortcuts.

3. Quick Start

Getting started! Look no further! The simple steps below will get you up to speed on this program faster than a beetle!

1. Ensure you have Java (version 11 or above) installed in your Computer. Java is an application development platform and version 11 can be downloaded [here](#).
2. Download the latest **CouponStash.jar**.
3. Copy the file to the folder you want to use as the home folder for your Coupon Stash.
4. Double-click the file to start the app. The Graphical User Interface (GUI) should appear in a few seconds. If nothing occurs, use the command **java -jar CouponStash.jar** on your computer's Command Line Interface (CLI) to open the file up.
5. Type the command in the command box and press Enter to execute it.
6. Some example commands you can try:
 - **add:** **add n/Domino's Pizza Extra Large s/40% p/ILOVEDOMINOS e/30-4-2020 l/5 t/pizza t/delivery**
 - adds a coupon named Domino's Pizza Extra Large to the Coupon Stash, with an expiry date of 30th April 2020, 40% savings with promo code ILOVEDOMINOS, and a limit of 5 usages.
 - **used:** **used 3** : marks the third coupon shown in the current list as used, and automatically archives it if the limit is reached
 - **saved:** **saved** : displays how much you have saved from all coupons
 - **exit** : exits the app (your data will be saved automatically)
7. Refer to the [Features](#) section below for the details of each command.



There is some default sample data in Coupon Stash on your first startup of Coupon Stash. They are for you to experiment with Coupon Stash's [features](#)! Once you are ready, use the **clear** command to clear them and start your couponing journey with Coupon Stash!

4. User Interface

In this section, we look at the key features of Coupon Stash's graphical user interface. The interface can be broadly split into 4 different parts: the different tabs present in the program (Coupons, Summary and Help), the calendar, the command input box and the command result box.



Figure 2. Coupon Stash main screen.

4.1. Tabs

Tab selection determines the content shown in the main window of Coupon Stash.

The tabs can be switched by pressing **Ctrl** + **Tab** or by clicking on the tabs bar on the left side of the application with the name of the tab you want to switch to. The current selected tab will be highlighted with a bolder text, and a darker blue.

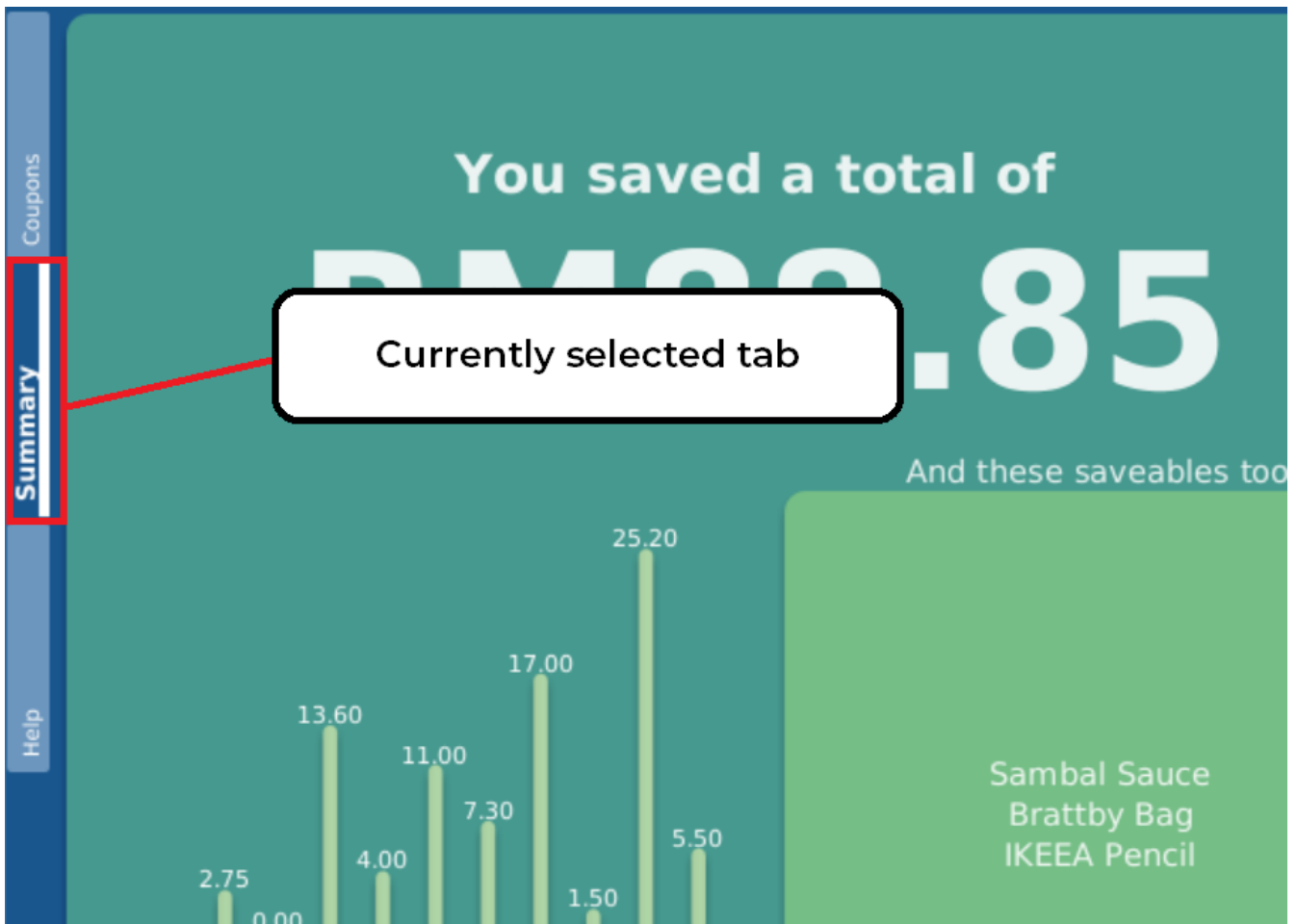


Figure 3. Coupon Stash summary tab.



The tabs bar is visible on every single tab! It serves as a good visual reference to tell you which tab is currently being displayed.

The following sections explain the different features of each tab of Coupon Stash, as well as how to use them.

4.1.1. Coupons tab

When you first start Coupon Stash, the default selected tab is the coupons tabs. The coupons tab contains a visual overview of the coupons which you have added to the application.

Coupon card

The coupon card displays information about each coupon, and it is styled to look like a coupon itself!



Figure 4. Sample coupon card.

Coupon Fields

- Name: Shopee - Name of the coupon.
- Promo Code: NEWUSER - Promo code required to use the coupon at the merchant.
- Expiry Date: 31-10-2020 - Expiry date of the coupon.
- Start Date: 01-01-2020 - Start date of when the coupon can be used.
- Remind Date: 28-10-2020 - A reminder will pop up when Coupon Stash is opened on this date.
- Usage: 1/2 - Number of times the coupon was used (1), and the number of times the coupon can be used (2).
- T&C: New users only - Terms and Condition applied for the valid usage of the coupon.
- Savings (blue box): \$7.00, USB-C cable, iPhone case - Savings (\$7) generated from the use of this coupon. USB-C Cable and iPhone Case are saveables; free items that come along with the use of this coupon.
- Tags (orange label): newuser - Tags to categorize coupons.
- Index (beside Tags): 1 - Index of the coupon on the displayed list. Used accordingly with some commands of Coupon Stash.



Duplicate coupons cannot exist in the active coupon list! Coupons are duplicates when they have the same name, promo code, expiry date and savings.

4.1.2. Summary tab

The summary tab provides an overview of how much you have saved from using coupons in Coupon Stash.

This tab provides an estimate of all the savings you have accumulated, provided the coupon that resulted in those savings is still present in Coupon Stash (this includes [archived](#) coupons). This estimate is displayed in the form of a monetary amount at the top of the page, with saved items that may not be easily quantified displayed in a lime-green box at the side, preceded by the text "And these saveables too!".

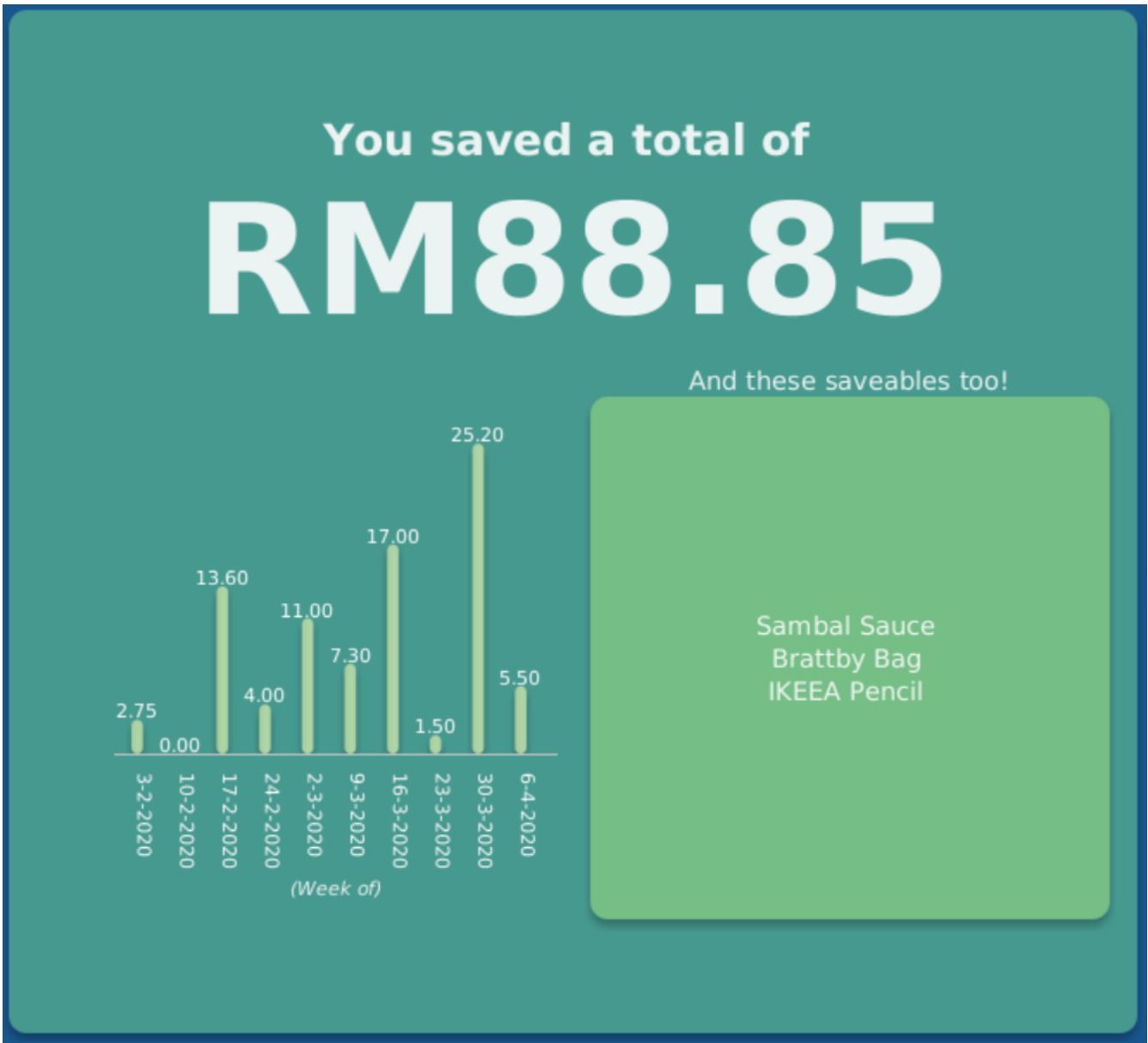


Figure 5. Coupon Stash summary tab.

Also, a graph that shows the monetary amount saved every week is also displayed in the summary tab. The next section will tell you more about how to interpret this graph.

Savings graph

The savings graph provides a bar chart which holds statistics on the amount of money saved per week. The taller the bar, the more savings earned that week. The exact amount is also shown in white text at the top of each bar.

In the graph, Monday is taken to be the first day of a week, and Sunday the last. The date displayed on the bottom axis corresponds to the first day of that week (Monday). Earlier weeks are displayed towards the left, while later weeks are displayed towards the right (direction of time axis is from left to right).

The graph is drawn based on the latest week in which coupon usage was recorded by Coupon Stash, and shows a maximum of 10 bars or 10 weeks of savings before that latest date.



Any savings that were recorded by Coupon Stash before the time period of 10 weeks will not be shown in the graph. However, they can still be calculated using the [saved command](#).

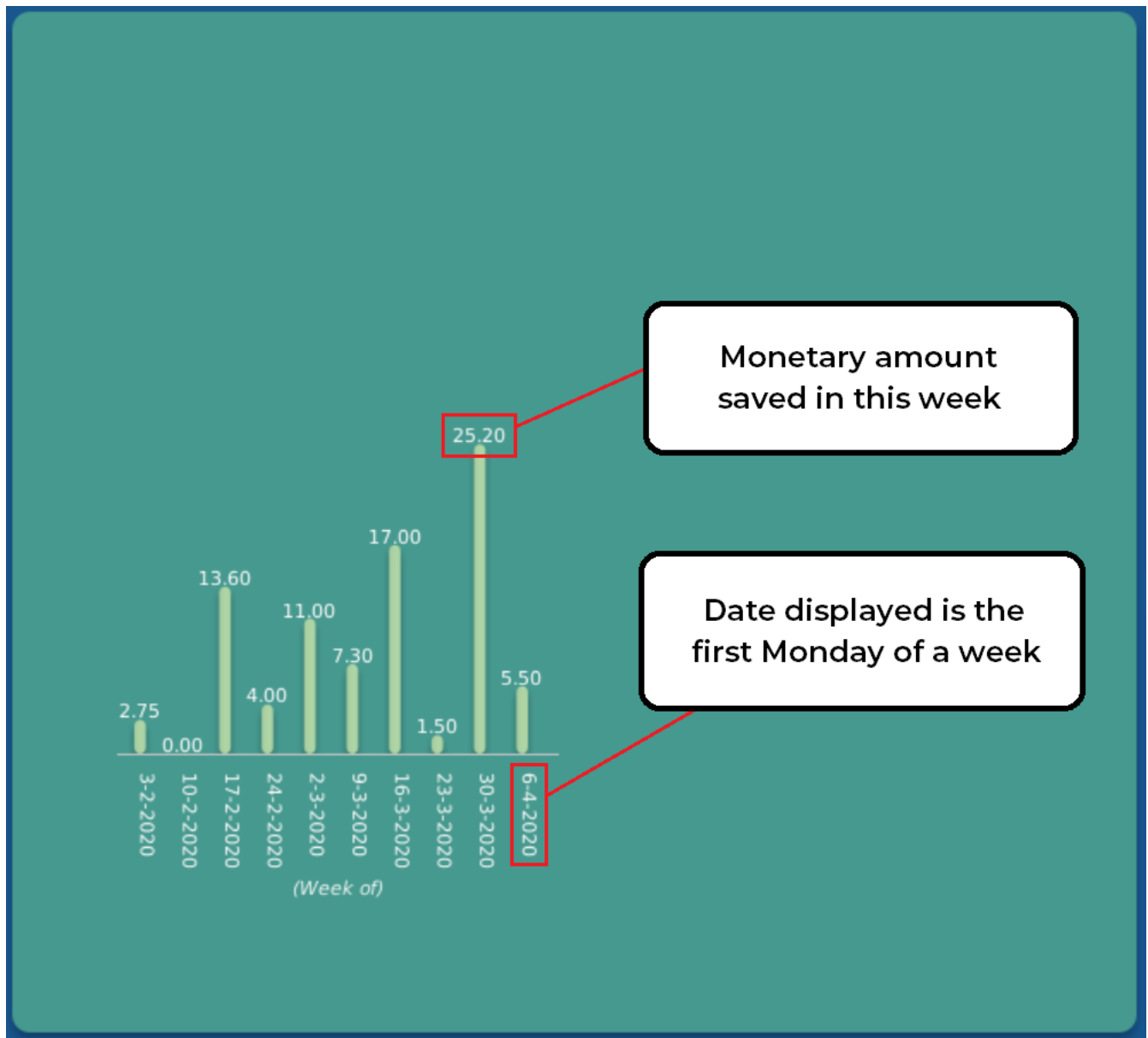


Figure 6. Coupon Stash savings graph.



Although the total monetary amount and saveables list in the summary tab is obtained by adding up all the savings from every coupon regardless of the date, the savings graph may not display every weeks' savings as it only displays up to a maximum of 10 weeks.

4.1.3. Help tab

The help tab shows you a short description of how to use all the commands in Coupon Stash.



As compared to the User Guide's descriptions, the descriptions provided in the help tab are less detailed!

4.2. Calendar

The calendar provides you with an overview of all your coupon's expiry dates within a month year. It reflects the expiry dates of the list of coupons in the **coupons tab**.

The calendar changes accordingly to the changes in the list of coupons in the **coupons tab**. Hence, the following commands that alters the list of coupons may also alter the calendar:

- **add**
- **archive**
- **clear**
- **delete**
- **edit**
- **expiring**
- **find**
- **list**
- **redo**
- **unarchive**
- **undo**
- **used**

4.2.1. Calendar Title

The calendar title indicates the current month year on display in the calendar .

4.2.2. Previous & Next Arrows

The left and right arrows enable you to change the month year on display on the calendar. A click on the left and right arrows displays the previous month and the next month respectively.



You may also change the month year on display using the **goto** command!

4.2.3. Dates

The numbers on the calendar represent each date in the current month year on display in the calendar.

Some date cells are highlighted. The following are the meanings of the various highlights:

- Blue highlight: Indicates the system's date (usually today's date)
- Red highlight: Indicates expiring coupon(s) on that date

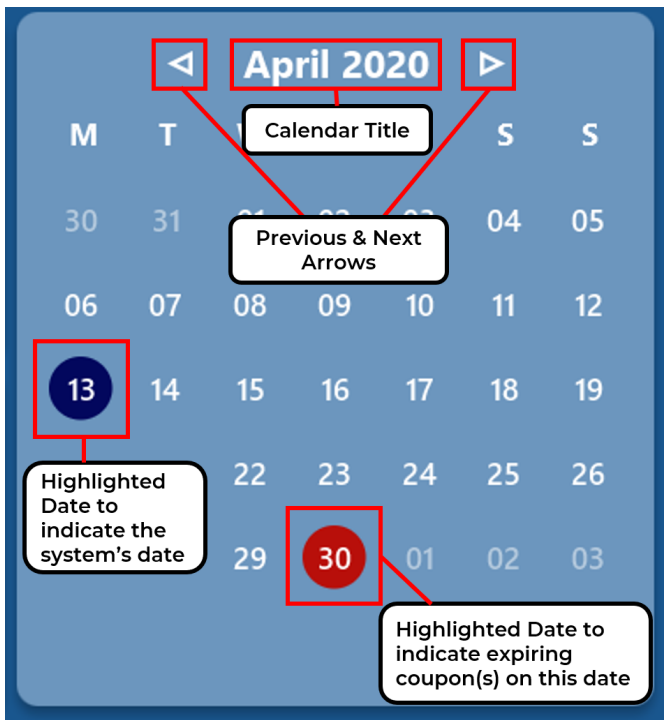


Figure 7. Elements in the Calendar

4.3. Command Input Box

The command input box is the main way you will interact with Coupon Stash. Simply type in your commands and hit !

4.4. Command Result Box

The command result box shows you messages depending on whether your command was successful or not.

5. Features

This section introduces the features of Coupon Stash. There are some examples for you to try out in Coupon Stash too!

Also, take note of the general format of all the commands below!

Command Format

- Words in UPPERCASE: parameters to be supplied by you (the user)
 - e.g. `add n/NAME`
- **INDEX**: positive integer that specifies the index number shown in the displayed coupon list. The index must be a positive integer
 - e.g. 1, 2, 3, ...
- **NAME** is a parameter which can be used as `add n/The Deck Chicken Rice`
- Letter(s) followed by a `/` : prefixes to come before some parameters
 - e.g. `n/NAME`
 - `n/` is the prefix for the **NAME** parameter
- Items in square brackets: optional
 - e.g. `n/NAME [t/TAG]`
 - Can be used as `n/The Deck Chicken Rice t/value` or `n/The Deck Chicken Rice`
- Items with ... after them: can be used multiple times (including zero times)
 - e.g. `[t/TAG]...`
 - can be used as `t/pizza`, `t/friend t/value` or not used at all etc.
- Parameters can be in any order
 - e.g. specifying `n/NAME s/SAVINGS` is the same as `s/SAVINGS n/NAME`
- Dates are all in the D-M-YYYY format (Coupon Stash date format)
 - D and M can be one or two digits, but YYYY has to be four digits
 - e.g. `1-1-2020` and `01-01-2020` are valid dates
- MonthYear are all in the M-YYYY format (Coupon Stash MonthYear format)
 - M can be one or two digits, but YYYY has to be four digits
 - e.g. `1-2020` and `01-2020` are valid MonthYears

5.1. Adding a coupon: `add`

Adds a coupon. Some fields like `name`, `expiry date` and `savings` are required. Optional fields like `promo code` and `tags` may be provided as well.

Format: `add n/NAME e/EXPIRY_DATE s/SAVINGS_OR_FREE_ITEM [sd/START_DATE] [r/REMIND_DATE] [p/PROMO_CODE] [c/CONDITIONS] [u/USAGE] [l/USAGE_LIMIT] [s/EXTRA_FREE_ITEMS]... [t/TAG]...`



A coupon can have any number of tags (including 0)



If unspecified, the default remind date of a coupon will be set to 3 days before the expiry date. Go to [\[Reminding of coupon expiry\]](#) to learn more about remind dates.

- Coupons must have at least one "savings" value, whether it is a flat monetary amount (e.g. \$1.00), a percentage (e.g. 10%) or an item. (e.g. Free Water Bottle)
- Savings can be represented by multiple free items, but not multiple monetary amounts or percentage amounts.
- Free item names cannot contain any numbers. (e.g. iPhone 11 is an invalid free item name as it contains the number 11)
- Coupons cannot have both a monetary amount and a percentage amount.
- To add more free items, use the same prefix as before!

Examples:

- `add n/McDonald's McGriddles p/ILOVEMCGRIDDLES e/31-12-2020 s/$2 sd/1-4-2020 l/2 t/value t/savoury`

Adds a new Coupon:



Figure 8. Newly added McDonald's coupon.

- `add n/Grabfood s/40% e/30-4-2020 p/GRAB40 t/delivery`

Adds a new Coupon:

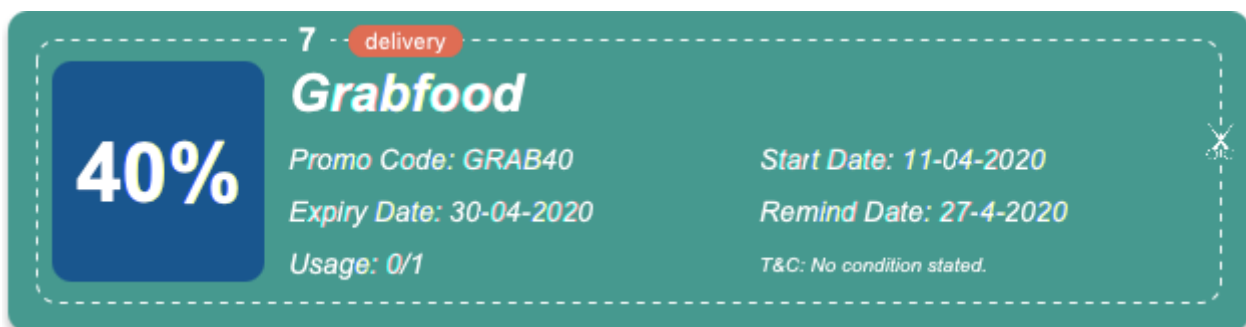


Figure 9. Newly added Grabfood coupon.



If your coupon allows for unlimited usage, set the limit of the coupon to a number less than 1! E.g. `l/0`

5.2. Editing a coupon: **edit**

Edits an existing coupon in the coupon book.

Format: **edit** INDEX [n/NAME] [e/EXPIRY_DATE] [sd/START_DATE] [r/REMIND_DATE] [p/PROMO_CODE] [c/CONDITION] [l/USAGE_LIMIT] [s/SAVINGS_OR_FREE_ITEMS]... [t/TAG]...

- Edits the coupon at the specified **INDEX**.
- 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 coupon will be removed. (i.e adding of tags is not cumulative)
- You can remove all of a coupon's tags by typing **t/** without specifying any tags after it.
- When editing savings, just like tags, existing savings of the coupon will also be removed.
- Usage cannot be edited.



Savings cannot be cleared in the same way as clearing tags, as a coupon must have at least 1 saving.



Remind date cannot be removed or cleared. Go to [\[Reminding of coupon expiry\]](#) to learn more about remind dates.

Examples:

- **edit 1 s/50%**
Edits the savings of the first coupon in the coupon list to be 50% off.
- **edit 2 n/The Deck Nasi Ayam Hainan t/**
Edits the name of the second coupon in the coupon list to be The Deck Nasi Ayam Hainan and clears all existing tags.

5.3. Deleting a coupon: **delete**

Deletes the specified coupon from the Coupon Stash.

Format: **delete** INDEX

- Deletes the coupon at the specified **INDEX**.

Examples:

- **list**
delete 2
Deletes the second coupon in the coupon list.

- `find rice`

`delete 1`

Deletes the first coupon in the resulting coupon list of the find command.

5.4. Listing coupons: `list` (Denise)

Shows a list of coupons in the Coupon Stash.

There are 3 types of lists:

- List of active coupons
- List of archived coupons
- List of used coupons



This command will change the month year on display in the [calendar](#) to the system's date (usually today's date).

Format: `list [PREFIX]`

- `PREFIX` can be either `a/` or `u/`.
- Using `a/` lists all coupons that are archived, while `u/` lists all used coupons (including [archived](#) coupons at the end of the list).



`PREFIX` can be left blank to list all active coupons (coupons that are not fully used/not archived (except for coupon(s) that are [unarchived](#) on purpose)).

Examples:

- `list`
Shows a list of all **active** coupons in Coupon Stash.
- `list a/`
Shows a list of all **archived** coupons in Coupon Stash.
- `list u/`
Shows a list of all **used** coupons in Coupon Stash.

5.5. Using a coupon: `used` (Chester)

Use a coupon if its usage has yet to reach its limit. Requires an original amount of purchase if the coupon has savings in a percentage value.

Format: `used INDEX` or `used INDEX MONETARY_AMOUNT` where `MONETARY_AMOUNT` is a number prefixed by a user-defined currency symbol e.g. (\$4.50, where the currency symbol is "\$").

- Uses the coupon at the specified [INDEX](#).

Examples:

- `used 1`
Uses the first coupon in the coupon list.
- `used 1 $10.0`
Uses the first coupon in the coupon list, which also has a percentage savings. The total savings of the coupon will be calculated, and can be seen with the command `saved`.



A coupon cannot be used if it has not reached its start date, or it is archived. If there is an intention to use the coupon, you can `edit` the start date or `unarchive` the coupon.

5.6. Viewing savings: `saved` (Alden)

Shows you how much you have saved by using coupons in Coupon Stash. There are three ways to use this command:

- If just the word "saved" is entered, the total savings accumulated since you started using Coupon Stash will be shown.

Format: `saved`

- If a specific date is given, Coupon Stash will show you savings earned only on that day.

Format: `saved d/DATE_TO_SHOW`

- If a start date and end date are given, Coupon Stash will show you the total savings accumulated over all the dates between that start date and end date, inclusive of those dates as well.

Format: `saved sd/START_DATE e/END_DATE`

- Dates are given in [Coupon Stash date format](#), namely D-M-YYYY where D and M can be single or double digits.
- Shows you a numeric value (e.g. 12.00 to represent twelve dollars/euros/pounds/pesos) that represents how much money you saved since a certain date, as well as certain items that you might have saved.
- This value changes depending on which coupons were marked as used during the time period specified.

Examples:

- `saved d/1-3-2020`
A message with your total savings and saveables saved on 1 March 2020 will be displayed in the

Command Result Box.

- `saved sd/1-5-2019 e/20-3-2020`

A message with your total savings and saveables saved between 1 May 2019 and 20 March 2020 will be displayed in the [Command Result Box](#).

Here are instructions for viewing your savings with the `saved` command:

1. First, ensure that your coupons have been added to Coupon Stash. This can be done with the `add` [command](#).
2. Use your coupons to make a purchase!
3. Register your usage of the coupon with the `used` [command](#)
4. Use the `saved` command to view your savings!
5. Navigate to the [summary tab](#) to view a graph of your savings per week.

5.7. Archiving a coupon: `archive` (Chester)

Archives a coupon when you want to keep a record of it, without cluttering your current stash of coupons.



To view archived coupons, the user can run the `list a/` command to list all archived coupons. Alternatively, the `find` command explained below will search all archived coupons, thus displaying archives that match the supplied keywords below unarchived coupons. Finally, the `expiring` command will also include all archived coupons that are expiring in its results. Read the section on `find` and `expiring` respectively to find out more.

Format: `archive INDEX`

- Archives the coupon at the specified [INDEX](#).

Examples:

- `archive 1`
Archives the first coupon in the coupon list.



To keep your coupons more organized, Coupon Stash will automatically archive your coupons once their usage limit has been reached, or when they have expired.

5.8. Unarchiving a coupon: `unarchive` (Chester)

Unarchive a coupon, thus bringing it back to your active coupons list.



To unarchive a coupon, you have to make sure that the archived coupon is visible in the Coupon Stash first. Read the section on the **archive** command to find out the ways that you can display archived coupons.

Format: **unarchive** INDEX

- Unarchive the coupon at the specified **INDEX**.

Examples:

- **unarchive 1**
Unarchive the first coupon in the coupon list.

5.9. Finding coupons by name: **find**

Find coupon(s) whose names contain any of the given keywords.

Format: **find** KEYWORD [MORE_KEYWORDS]...

- The search is case insensitive. e.g mALa will match Mala.
- The order of the keywords does not matter. e.g. Rice Chicken will match Chicken Rice.
- Only full words will be matched e.g. Chicken will not match Chickens.
- Coupons matching at least one keyword will be returned. e.g. Chicken Rice will return Duck Rice, Chicken Chop.



For easy finding of coupons, include the store or brand name in your coupon name!

Examples:

- **find ken**
For example, coupons with the keyword **ken** would be returned:
Ken Kitchen, **KEN** RIDGE Optical, **ken** you build a snowman.



The **find** command searches the **archive** for matches too. Coupons that are found in the **archive** are displayed **below** the non-archived matches.

5.10. Listing all expiring coupons: **expiring** (Denise)

Shows a list of all your coupons expiring on a date or during a **month year**.



This command will change the month year on display in the [calendar](#) to the month year of specified [expiry date](#) or [month year](#).
For example, `expiring e/30-9-2020` & `expiring my/9-2020` both changes the month year on display in the [calendar](#) to September 2020.

Format: `expiring e/EXPIRY_DATE` or `expiring my/MONTH_YEAR`

- Lists all coupons that are expiring on the specified expiry date or during the specified month year.
- The [expiry date](#) or [month year](#) must follow the [Coupon Stash date format](#) or the [Coupon Stash MonthYear format](#) respectively.

Examples:

- `expiring e/30-9-2020`
Shows you all the coupons that expire on 30 September 2020.
- `expiring my/9-2020`
Shows you all the coupons that expire during September 2020.



The `expiring` command will include [archived](#) coupons in its results too. Coupons that are expiring in the archives are displayed **below** the non-archived matches.

5.11. Sorting coupons: `sort` (Nelson)

Sorts the coupons in Coupon Stash in **ascending** order. It is possible to sort by coupon name, expiry date, or remind date.



When you `edit` or `add` a coupon, the sorting order will revert to the original order. Otherwise, the sorting order will persist throughout the runtime of the program.



The command will sort any coupons currently on screen, including [archived](#) coupons, if they are present on screen. [Archived](#) coupons will always appear **below** active coupons.

Format: `sort PREFIX`

- The `PREFIX` can be either `n/`, `e/` or `r/`.
- Using `n/` would mean sorting by name, `e/` would mean sorting by expiry date, and `r/` would mean sorting by reminder date.

Examples:

- `sort n/`
Sorts the Coupon Stash by name in ascending order.

- `sort e/`
Sorts the Coupon Stash by expiry date from earliest to latest.
- `sort r/`
Sorts the Coupon Stash by reminder date from earliest to latest.

5.12. Copying a coupon: `copy` (Denise)

Copies a coupon as an `add` command to your clipboard so that you can easily share it with your friends/family!

Format: `copy INDEX`

- Copies the coupon at the specified `INDEX`.
- The `add` command of the coupon will be copied to your clipboard. Simply `Ctrl` + `V` to paste it! (`Cmd` + `V` for macOS)
- Statistics like number of times used or amount of savings accumulated will not be copied.
- The copied `add` command will only contain the following fields:
 - Name
 - Savings
 - Expiry Date
 - Limit
 - Promo Code (if any)
 - Condition (if any)

Below is a short demonstration of the `copy` command:

[copy command] | *gifs/copy command.gif*

Figure 10. Animation of `copy` command operation.

Examples:

- `list`
`copy 2`
Copies the second coupon in the resulting coupon list of the `list` command to an `add` command and copies it to your clipboard.
- `find chicken`
`copy 1`
Copies the first coupon in the resulting coupon list of the `find` command to an `add` command and copies it to your clipboard.

5.13. Sharing a coupon: **share** (Chester)

Shares a coupon as a **.png** image file (portable network graphic). Now you can post your coupons on your social media platforms like Instagram!

Format: **share** INDEX

- Shares the coupon at the specified INDEX.
- A message with the file path to the **.png** file will be displayed in the **Command Result Box**.



Image generated will have an index of 1 regardless of its true index! Below is a sample of how a shared coupon image looks like:

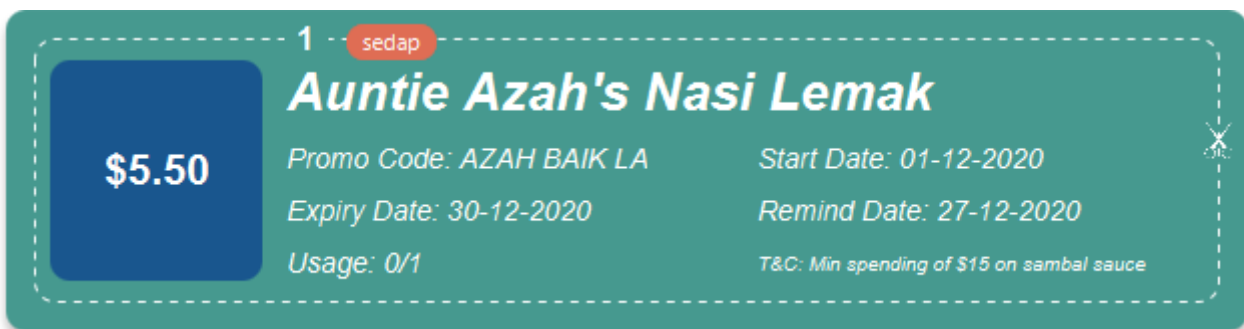


Figure 11. Example of a shared coupon



Do not try entering this command with **MAX_INT** value as INDEX. This may lead to a bug.

Examples:

- **list**
share 2
Shares the second coupon in the resulting coupon list of the **list** command and save it as a **.png** file.
- **find chicken**
share 1
Shares the first coupon in the resulting coupon list of the **find** command and save it as a **.png** file.

5.14. Undoing the previous command: **undo** (Nelson)

Undo the most recent operation on the Coupon Stash. Only operations that change the coupons in the Coupon Stash can be undone.

Format: **undo**

Undo only works on the following commands: (i.e. commands that change the coupons in the Coupon Stash)

- `add`
- `edit`
- `delete`
- `archive`
- `unarchive`
- `clear`
- `used`



Changes to user preferences, such as the [setting of the currency symbol](#), cannot be undone.



You can only undo commands which you have entered since you opened Coupon Stash. This means that after you close the application, your undo history will be lost!

Examples:

- `edit 1 r/ 25-12-2020`
`undo`

Undo the `edit` command. Remind date of first coupon in the coupon list reverts back to its previous value.

- `delete 1`
`undo`

Revert the `delete` that was performed. Deleted coupon is restored.

5.15. Reversing an undo: `redo` (Nelson)

Reverse your last `undo`. This is akin to undoing an `undo`.

Format: `redo`

Examples:

- `edit 1 r/ 25-12-2020`
`undo`
`redo`

Un-undo the `edit` command. Remind date of first coupon that was in the resulting coupon list is edited.

- `delete 1`
`undo`
`redo`

Un-undo the **delete** command. First coupon that was in the resulting coupon list is deleted.

5.16. Going to a month-year on the calendar: **goto** (Denise)

Goes to the specified month and year on the **calendar** by changing the month year on display.

Format: **goto** my/MONTH_YEAR

Example:

- **goto** my/12-2020

Goes to December 2020 on the **calendar** by changing the month year on display to December 2020.

5.17. Expanding a coupon: **expand** (Alden)

Displays a coupon in its own window, giving you a complete expanded view of it.

Format: **expand** INDEX

- Expands the coupon at the specified **INDEX**.

Examples:

- **expand** 1

Opens the first coupon in the coupon list in a new window.



Figure 12. Example of an expanded coupon window.

5.18. Setting currency symbol: `setcurrency` (Alden)

Sets the currency symbol used in Coupon Stash. This will force Coupon Stash to reload all coupons to use the new symbol instead.

- The currency symbol may affect the visual look of the program, as well as modify commands entered!
- For example, [adding a coupon](#) with savings of a certain monetary amount will require you to use the new symbol instead.

Format: `setcurrency ms/NEW_MONEY_SYMBOL`

Examples:

- `setcurrency ms/€`

Sets the money symbol used in commands, as well as to display coupons' savings, to € (Euro). The changes will be visible immediately.

5.19. Viewing help: `help` (Alden)

Open a browser window that accesses an offline copy of this user guide.



First run of this command may take some time as the program needs to extract an offline copy of the document.

Format: `help`

5.20. Clearing all entries: `clear`

Deletes every single coupon in Coupon Stash. This includes archived and non-archived coupons!

Format: `clear`



If you clear your Coupon Stash on accident, you can use `undo` to restore your Coupons! Just make sure you don't close the application immediately after you cleared your Coupons.

5.21. Exiting the program: `exit`

This command immediately exits the program when entered. When you exit the program, all coupon data is saved to your computer.

Format: `exit`



There are many ways of exiting the program! One other way is to click the cross at the top of the application (position of this cross button differs depending on operating system).

5.22. Reminding of coupon expiry (Kevin)

By default, all added coupons have a remind date that is 3 days before its stated expiry date. This default value can be changed during the process of `adding` a new coupon or by `editing` an existing coupon. The remind dates for all coupons are checked on program launch. All coupons that have a remind date which falls on the date of program launch will be listed in a popup window on launch. The popup window can be closed by pressing `Ctrl + q`.

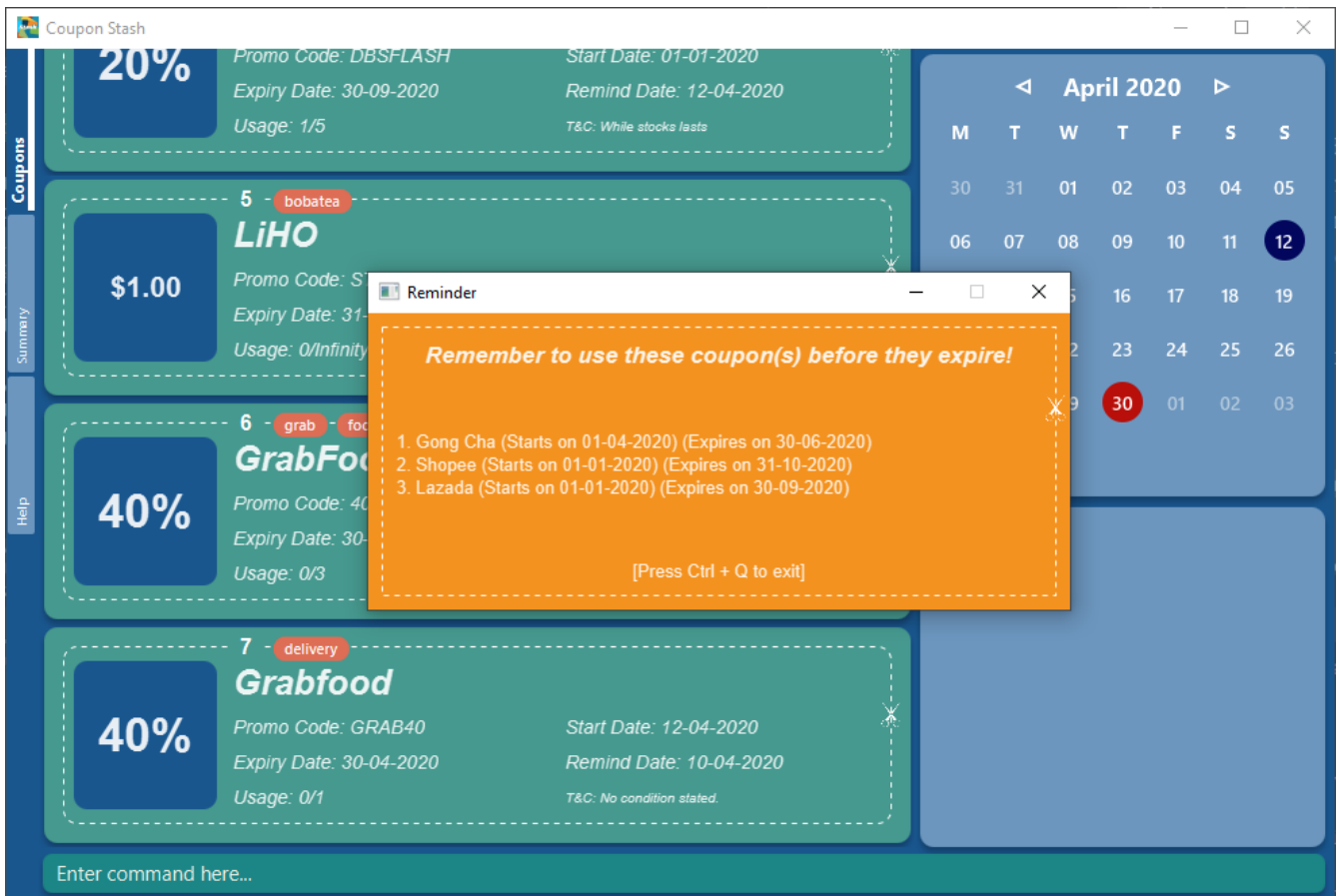


Figure 13. Popup window showing all coupons that have their remind dates set on the date of program launch.



Remind dates are **only** checked during program launch. Thus, if a day passes while the program is open, even if there are coupons expiring on the new day, no new reminder window will open.

As of now, there is no functionality for the disabling of reminders for coupons. This feature will be made available in version 2.0.

- The format of dates entered via the `r/` field is in the D-M-YYYY format. ([Coupon Stash date format](#))

Examples:

- `add n/McDonald's McGriddles p/ILOVEMCGRIDDLES e/31-12-2020 s/$2 sd/1-4-2020 l/2 t/value t/savoury`

Add a new Coupon without specifying a remind date. Note that the default remind date is 3 days before the state expiry date:



Figure 14. New coupon with a default remind date.

- `add n/Grabfood s/40% e/30-4-2020 p/GRAB40 t/delivery r/10-4-2020`

Add a new Coupon while specifying a remind date of **10-04-2020**:

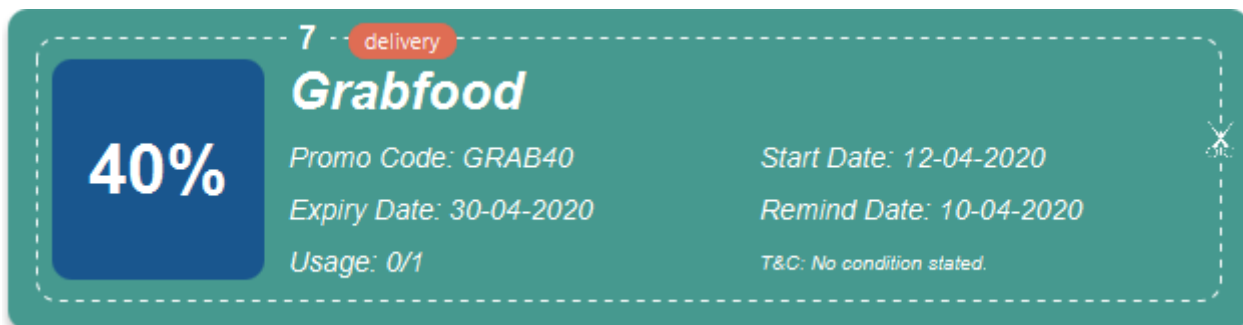


Figure 15. New coupon with remind date set to **10-04-2020**.

- `edit 1 r/10-4-2020`

Edit the coupon at index 1 and change its remind date to **10-4-2020**.

5.23. Saving the data



Coupon data is saved in the hard disk automatically after any command that changes the data.

There is no need to save manually.

6. Keyboard Shortcuts (Nelson)

With Coupon Stash being optimized for efficient command line usage, how can we not include some nifty keyboard shortcuts to further streamline your workflow! This section introduces some of the keyboard shortcuts available for use in this program.

6.1. Restore previously entered command texts: and

Pressing the  keyboard button while the focus is on the command box allows you to restore the text of the previous executed command. Pressing the  keyboard button allows you to revert to a more recently entered command.

Here are instructions for restoring previously entered command texts with the  and  keys:

1. Suppose you ran these 3 commands in order in Coupon Stash:

- `edit 1 n/ Adidas 31 Percent`

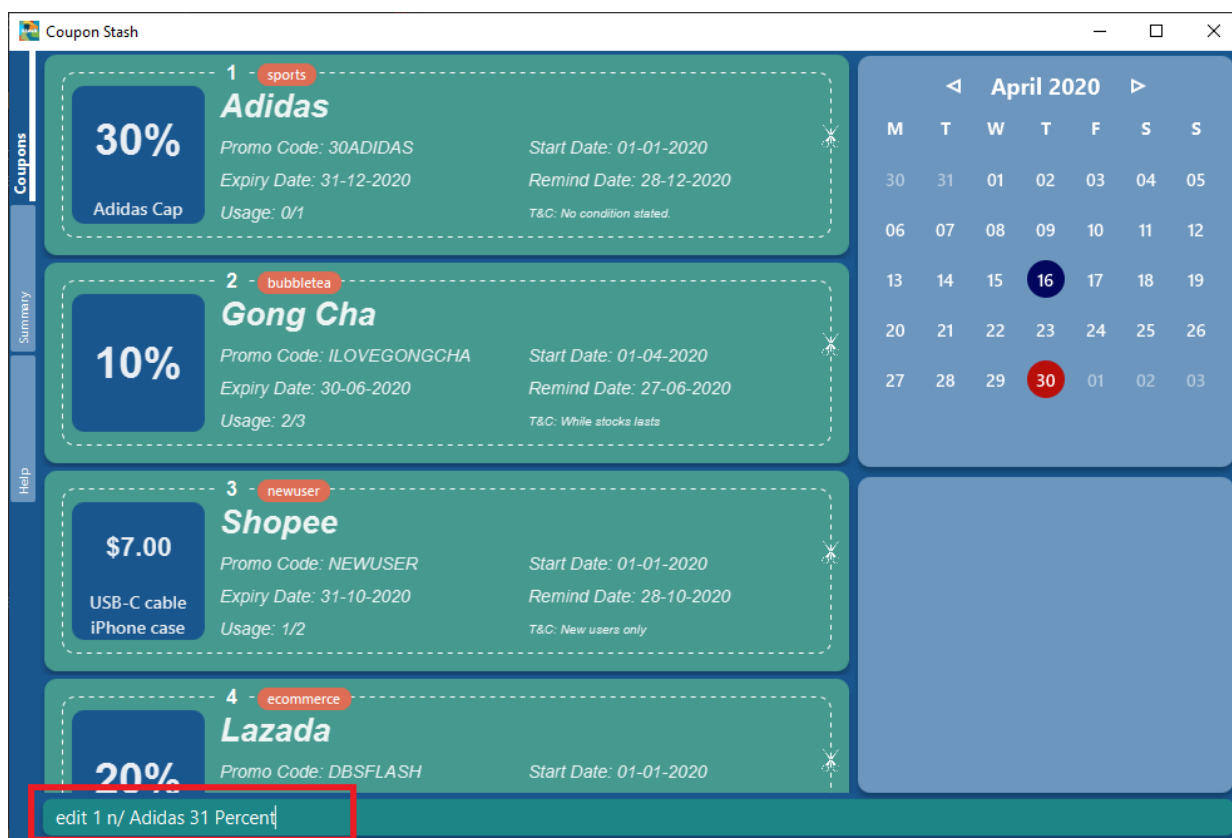


Figure 16. Before running the `edit` command.

- `archive 2`



Figure 17. Before running the `archive` command.

- `add n/McDonald's McGriddles p/ILOVEMCGRIDDLES e/31-12-2020 s/50% sd/1-4-2020 l/2 t/Value t/Savoury`

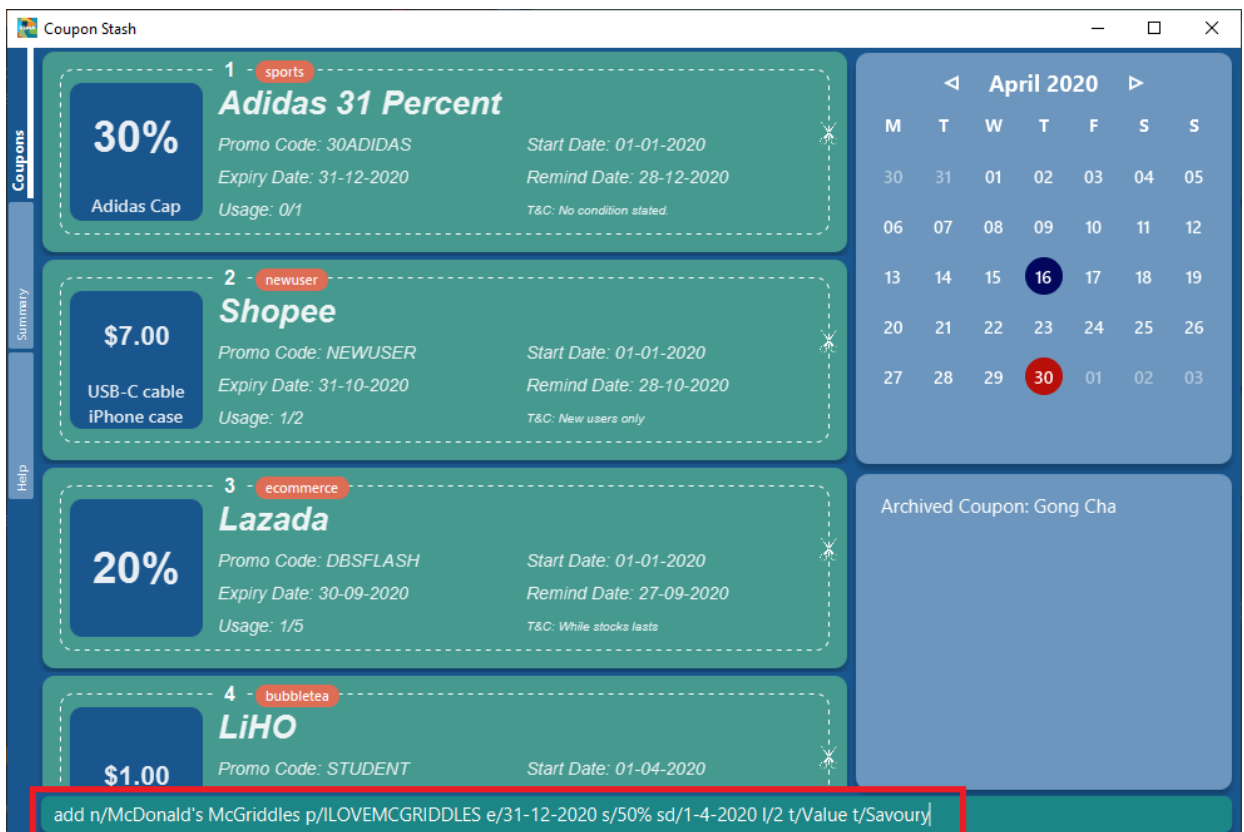


Figure 18. Before running the `add` command.

2. We realise that in the first command we ran (the `edit` command), we renamed the first coupon wrongly.

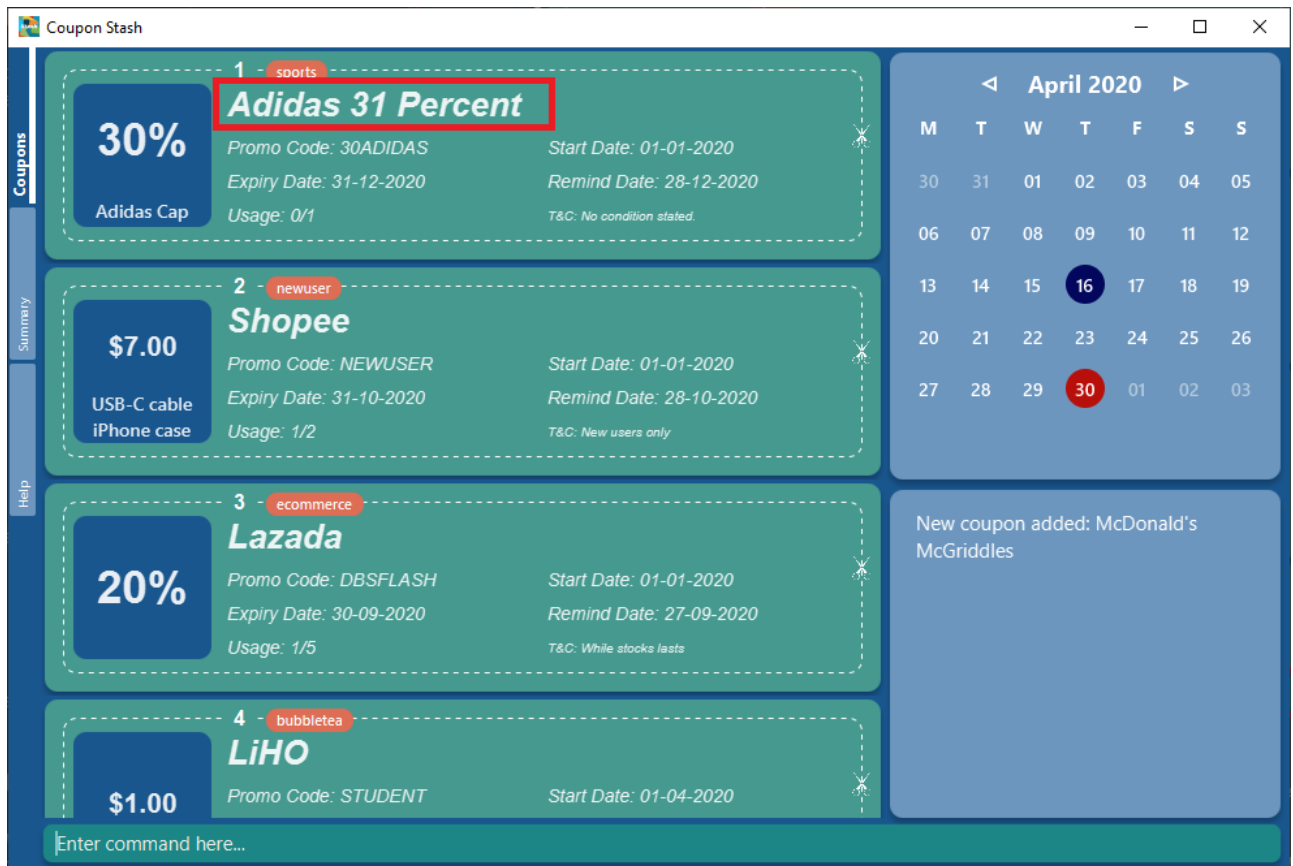


Figure 19. Name was edited erroneously, should be "Adidas 30 Percent" instead of "Adidas 31 Percent".

3. To save ourselves the hassle of retyping the entire `edit` command again, we can access the previously entered command via the `↑` keyboard button.
4. Press `↑` thrice to retrieve the `edit` command.
 - We press `↑` thrice as the `edit` command was run three commands ago.

6.2. Close expanded coupon or reminder windows:

`Ctrl+q`

After [expanding](#) a coupon, you can close the expanded coupon window by keeping the focus on the coupon window and pressing the keyboard buttons `Ctrl` and `q` simultaneously. This shortcut also works for closing [reminder](#) windows.

Demonstration:

[68747470733a2f2f6d656469612e67697068792e636f6d2f6d656469612f4c6f497a67795237354e72504f

<https://camo.githubusercontent.com/2229a99e9c948cd605884c51defe362a1d3c3259/68747470733a2f2>

f6d656469612e67697068792e636f6d2f6d656469612f4c6f497a67795237354e72504f56456132432f67697

[ctrl q] | *gifs/ctrl q.gif*

6.3. Cycle through tabs: **Ctrl**+**Tab**

You can press the keyboard buttons **Ctrl** and **Tab** simultaneously to switch tabs. Do note that the focus has to be on the main panel before the key presses would work.

Demonstration:

[ctrl tab] | *gifs/ctrl tab.gif*

*Figure 20. Animation of switching tabs with **Ctrl**+**Tab**.*

7. FAQ

Q: How do I transfer my data to another computer?

A: Simply install Stash in the other computer and overwrite the empty data file with your intended data file.

Q: Can I add multiple coupons using a command line?

A: Sorry, not at the moment. We may consider adding this feature in version 2.0.0.

Q : Why do I need to tag my coupons?

A : Tagging is not compulsory. However, it allows you to group similar coupons together for easier identification. For example, you can easily identify all the coupons that are tagged "cheap", as these tags are displayed in bright orange at the top of each coupon. Do refer to [section 4.1.1](#) for more details about the coupon card.

Q : How do I store coupons with no expiry date?

A : Sorry, at the moment, coupon entries with no expiry date cannot be stored. You would need to enter a much further date like 1-1-2030. To be added in version 2.0.0

Q : How do you calculate the savings value?

A : Whenever a coupon is marked as used, Coupon Stash will automatically calculate the values saved based on the details of the coupon. You can use the [saved command](#) to view the amount saved.

Q : What animal is Stasher?

A : Stasher is a beetle that loves discounts and hot deals! Beetles are resourceful insects that know

how to scavenge for the best coupons, as well as remember these coupons safely inside their tough exoskeleton and elytra.

8. Command Summary

This section has a summary of all the commands.

Action	Format
Add	add n/NAME e/EXPIRY_DATE s/SAVINGS [u/USAGE] [l/USAGE_LIMIT] [p/PROMO_CODE] [sd/START_DATE] [r/REMIND_DATE] [c/CONDITION] [s/SAVINGS]... [t/TAG]... e.g. add n/The Deck Chicken Rice s/20% sd/2-3-2020 e/30-8-2020 t/friend t/value
Edit	edit INDEX [n/name] [e/EXPIRY_DATE] [l/USAGE_LIMIT] [p/PROMO_CODE] [sd/START_DATE] [r/REMIND_DATE] [c/CONDITION] [s/SAVINGS]... [t/TAG]... e.g. edit 2 n/Chicken Up s/50%
Delete	delete INDEX e.g. delete 3
List	list [PREFIX] e.g. list
Find	find KEYWORD [MORE_KEYWORDS] e.g. find western chick
Sort	sort PREFIX e.g. sort n/
Expiring	expiring e/EXPIRY_DATE or expiring my/MONTH_YEAR e.g. expiring e/30-9-2020 or expiring my/9-2020
Used	used INDEX or used INDEX MONETARY_SYMBOL + ORIGINAL_AMOUNT e.g. used 1 or used 1 \$10
Saved	saved d/DATE_TO_TRACK_FROM e.g. saved d/ 1-5-2020
Archive	archive INDEX e.g. archive 1
Unarchive	unarchive INDEX e.g. unarchive 1
Copy	copy INDEX e.g. copy 1
Share	share INDEX e.g. share 1
Undo	undo
Redo	redo
Goto	goto my/MONTH_YEAR e.g. goto 9-2020
Expand	expand INDEX e.g. expand 1

Action	Format
SetCurrency	<code>setcurrency [ms/NEW_MONEY_SYMBOL]</code> e.g. <code>setcurrency ms/€</code>
Help	<code>help</code>
Clear	<code>clear</code>
Exit	<code>exit</code>

9. Appendix

This section talks about features that have not been implemented in Coupon Stash yet, but features that the developers are keen to add in upcoming versions.

9.1. Query expressions (coming in v2.0)

Query expressions serve to enable users to perform efficient batch operations on stored coupons using SQL-like syntax.

Examples:

- `edit favorites set tag=hated where expiry < 19-2-2020`
For all coupons tagged as `favorites''`, set their tag to `hated` if they expire before the 19th Feb 2020.
- `delete favorites where expiry < 19-2-2020`
For all coupons tagged as `favorites''`, delete them if they expire before the 19th Feb 2020.

9.1.1. Bulk edit

Format: `edit tag set field1=value1, field2=value2, ... [where condition]`

- The `edit` keyword is compulsory.
 - If selecting all tags, put tag as `*`.
- The `set` keyword is compulsory.
 - If the field is present in a coupon, update the value, else create the field and value.
 - At least 1 field must be updated/added per edit operation.
- The `where` keyword is optional.
 - The `==, >=, <=, >, <, !=`, operators are supported in the condition.
 - Only default fields can be compared.
 - Condition syntax: `DEFAULT_FIELD operator LITERAL_VALUE`
 - `AND/OR` functionality is not available.

9.1.2. Bulk delete

Version 2.0 will bring new features like confirmation of coupons before they are deleted!

Also, a "recycle bin" is in the works for version 3.0, similar to the Recycle Bin in Microsoft Windows.

Format: `delete tag [where condition]`

- The `delete` keyword is compulsory.
 - If selecting all tags, put tag as `*`.
- The `where` keyword is optional.
 - If the `where` keyword is omitted, deletes all coupons with that tag.
 - Follows syntax explained above.