

Event Planning isn't Complicated (EPIC) - User Guide

1. Introduction	1
2. Quick Start	1
3. Features	2
3.1. Viewing help : help	2
3.2. Adding a person: add	2
3.3. Listing all persons : list	3
3.4. Editing a person : edit	3
3.5. Locating persons by name: find	3
3.6. Deleting a person : delete	4
3.7. Selecting a person : select	4
3.8. Adding an event: add-event [Since v1.2]	5
3.9. Listing all events: list-events [Since v1.2]	5
3.10. Editing an event : edit-event [Since v1.2]	5
3.11. Locating events by name: find-event [Since v1.2]	5
3.12. Deleting an event : delete-event [Since v1.2]	6
3.13. Registering a person to an event: register [Since v1.2]	6
3.14. Deregistering a person from an event: deregister [Since v1.2]	7
3.15. Listing all persons registered in an event: list-registered [Since v1.2]	7
3.16. Listing entered commands : history	8
3.17. Undoing previous command : undo	8
3.18. Redoing the previously undone command : redo	8
3.19. Clearing all entries : clear	9
3.20. Exiting the program : exit	9
3.21. Saving the data	9
3.22. Marking attendance for event participant : mark [coming in v1.5]	9
3.23. Unmarking attendance for event participant : unmark [coming in v1.5]	9
3.24. Exporting contact information to a csv file : export [coming in v1.5]	9
3.25. Importing contact information from a csv file : import [coming in v1.5]	10
3.26. Adding feedback for an event from a participant : add-feedback [coming in v2.0]	10
3.27. Exporting event details to a csv file : export-event [coming in v2.0]	10
3.28. Registering all persons with a particular tag to an event : register-by-tag [coming in v2.0]	10
3.29. Adding automatic email reminder to event participants : add-reminder [coming in v2.0] ..	10
3.30. Adding hotkeys for commands : add-hotkey [coming in v2.0]	10
4. FAQ	10
5. Command Summary	11

1. Introduction

Event Planning isn't Complicated (EPIC) is for those who **prefer to use a desktop app for event plannings and streamline event registration**. More importantly, EPIC is **optimized for event planners who prefer to work with a Command Line Interface (CLI)** while still having the benefits of a Graphical User Interface (GUI). If you can type fast, EPIC can get your event planning tasks and registration processes for events done faster than traditional GUI apps or pen and paper registration. Interested? Jump to the [Section 2, "Quick Start"](#) to get started. Enjoy!

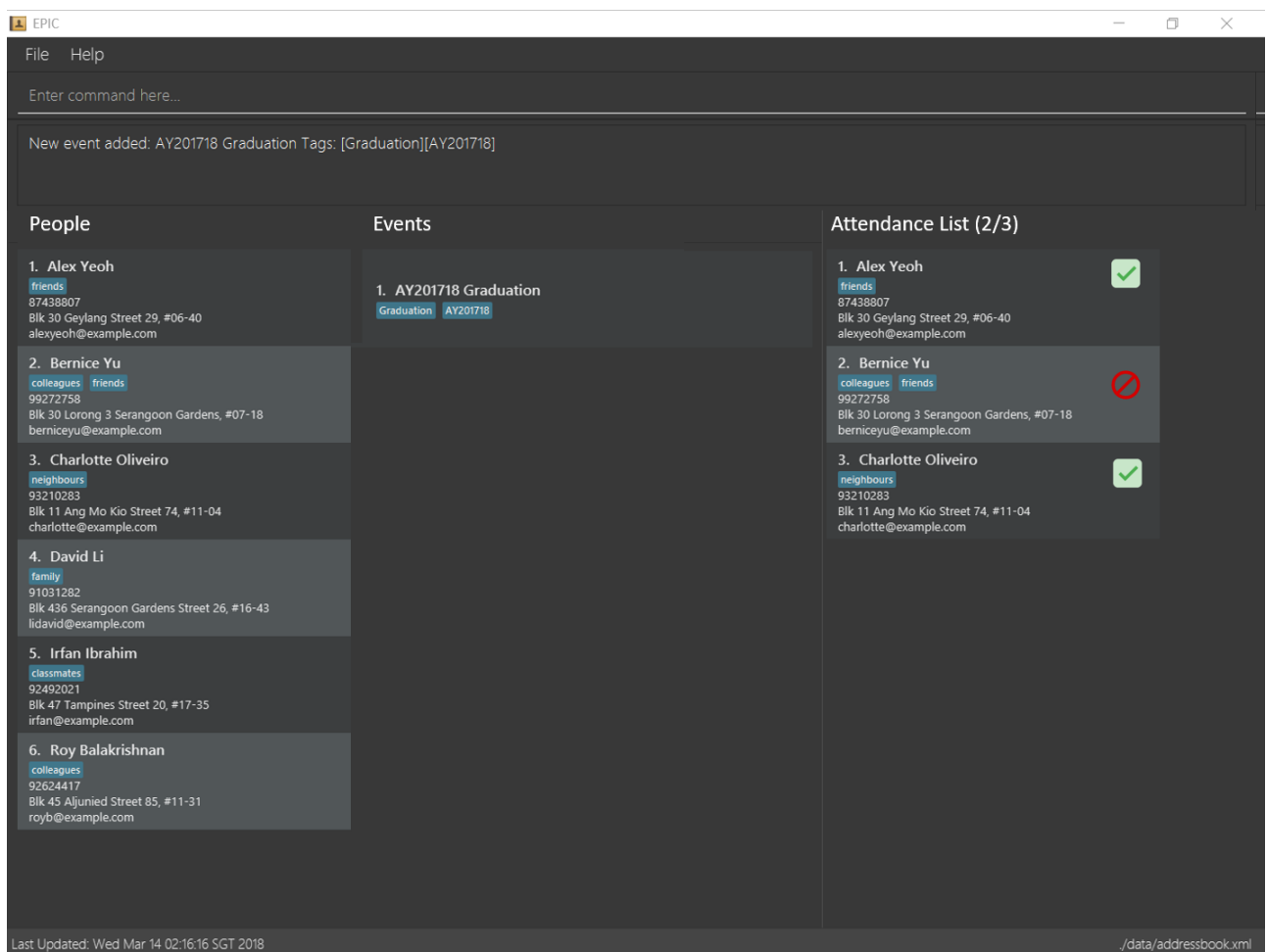
2. Quick Start

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

NOTE

Having any Java 8 version is not enough.
This app will not work with earlier versions of Java 8.

2. Download the latest **EPIC.jar** [here](#).
3. Copy the file to the folder you want to use as the home folder for your Event Planner.
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. Some example commands you can try:
 - **list** : lists all contacts
 - **addn/John Doe p/98765432 e/johnd@example.com a/John street, block 123, #01-01** : adds a contact named **John Doe** to the Event Planner.
 - **delete3** : deletes the 3rd contact shown in the current list
 - **exit** : exits the app
7. Refer to [Section 3, “Features”](#) for details of each command.

3. Features

Command Format

- Words in **UPPER_CASE** are the parameters to be supplied by the user e.g. in **add n/NAME**, **NAME** is a parameter which can be used as **add n/John Doe**.
- Items in square brackets are optional e.g **n/NAME [t/TAG]** can be used as **n/John Doe t/friend** or as **n/John Doe**.
- Items with **...** after them can be used multiple times including zero times e.g. **[t/TAG]...** can be used as (i.e. 0 times), **t/friend**, **t/friend t/family** etc.
- Parameters can be in any order e.g. if the command specifies **n/NAME p/PHONE_NUMBER**, **p/PHONE_NUMBER n/NAME** is also acceptable.

3.1. Viewing help : **help**

Format: **help**

3.2. Adding a person: **add**

Adds a person to the event planner

Format: **add n/NAME p/PHONE_NUMBER e/EMAIL a/ADDRESS [t/TAG]...**

TIP | A person can have any number of tags (including 0)

Examples:

- **add n/John Doe p/98765432 e/johnd@example.com a/John street, block 123, #01-01**
- **add n/Betsy Crowe t/friend e/betsycrowe@example.com a/Newgate Prison p/1234567 t/criminal**

3.3. Listing all persons : **list**

Shows a list of all persons in the event planner and changes the focus to persons list tab.

Format: **list**

3.4. Editing a person : **edit**

Edits an existing person in the event planner.

Format: **edit** INDEX [n/NAME] [p/PHONE] [e/EMAIL] [a/ADDRESS] [t/TAG]...

- Edits the person at the specified **INDEX**. The index refers to the index number shown in the last person listing. The index **must be a positive integer** 1, 2, 3, ...
- 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 person will be removed i.e adding of tags is not cumulative.
- You can remove all the person's tags by typing **t/** without specifying any tags after it.

Examples:

- **edit 1 p/91234567 e/johndoe@example.com**
Edits the phone number and email address of the 1st person to be **91234567** and **johndoe@example.com** respectively.
- **edit 2 n/Betsy Crower t/**
Edits the name of the 2nd person to be **Betsy Crower** and clears all existing tags.

3.5. Locating persons by name: **find**

Finds persons whose names contain any of the given keywords.

Format: **find** KEYWORD [MORE_KEYWORDS]

- The search is case insensitive. e.g **hans** will match **Hans**
- The order of the keywords does not matter. e.g. **Hans Bo** will match **Bo Hans**
- Only the name is searched.
- Only full words will be matched e.g. **Han** will not match **Hans**
- Persons matching at least one keyword will be returned (i.e. **OR** search). e.g. **Hans Bo** will return **Hans Gruber, Bo Yang**

Examples:

- **find John**
Returns **john** and **John Doe**

- `find Betsy Tim John`
Returns any person having names `Betsy`, `Tim`, or `John`

3.6. Deleting a person : `delete`

Deletes the specified person from the event planner.

Format: `delete INDEX`

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

Examples:

- `list`
`delete 2`
Deletes the 2nd person in the event planner.
- `find Betsy`
`delete 1`
Deletes the 1st person in the results of the `find` command.

3.7. Selecting a person : `select`

Selects the person identified by the index number used in the last person listing.

Format: `select INDEX`

- Selects the person and loads the Google search page the person at the specified `INDEX`.
- The index refers to the index number shown in the most recent listing.
- The index **must be a positive integer** 1, 2, 3, ...

Examples:

- `list`
`select 2`
Selects the 2nd person in the event planner.
- `find Betsy`
`select 1`
Selects the 1st person in the results of the `find` command.

3.8. Adding an event: `add-event` [Since v1.2]

Adds an event to the event planner.

Format: `add-event n/NAME [t/TAG]...`

TIP | An event can have any number of tags (including 0)

Examples:

- `add-event n/AY201718 Graduation t/Graduation`
- `add-event n/Computing Seminar`

3.9. Listing all events: `list-events` [Since v1.2]

Shows a list of all events in the event planner and changes the focus to events list tab.

Format: `list-events`

3.10. Editing an event : `edit-event` [Since v1.2]

Edits an existing event in the event planner.

Format: `edit-event INDEX [n/NAME] [t/TAG]...`

- Edits the event at the specified **INDEX**. The index refers to the index number shown in the last event listing. The index **must be a positive integer** 1, 2, 3, ...
- 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 event will be removed i.e adding of tags is not cumulative.
- You can remove all the event's tags by typing `t/` without specifying any tags after it.

Examples:

- `edit-event 1 n/IoT Talk t/talk`
Edits the name and tag of the 1st event to be `IoT Talk` and `talk` respectively.
- `edit-event 2 n/Graduation Day t/`
Edits the name of the 2nd event to be `Graduation Day` and clears all existing tags.

3.11. Locating events by name: `find-event` [Since v1.2]

Finds events whose names contain any of the given keywords.

Format: `find-event KEYWORD [MORE_KEYWORDS]`

- The search is case insensitive. e.g `talks` will match `Talks`
- The order of the keywords does not matter. e.g. `Food Talk` will match `Talk Food`
- Only the name is searched.
- Only full words will be matched e.g. `Talk` will not match `Talks`
- Events matching at least one keyword will be returned (i.e. `OR` search). e.g. `Talk` will return `Food Talk`, `IoT Talk`

Examples:

- `find-event Talk`
Returns `Talk` and `Food Talk`
- `find-event Talk Competition`
Returns any event having name `Talk`, or `Competition`

3.12. Deleting an event : `delete-event` [Since v1.2]

Deletes the specified event from the event planner.

Format: `delete-event INDEX`

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

Examples:

- `list-events`
`delete-event 2`
Deletes the 2nd event in the event planner.
- `find-event Talk`
`delete-event 1`
Deletes the 1st event in the results of the `find-event` command.

3.13. Registering a person to an event: `register` [Since v1.2]

Registers the specified person to an event in the event planner.

Format: `register INDEX EVENT_NAME`

- Registers the person at the specified **INDEX**.
- The index refers to the index number shown in the most recent listing.
- The index **must be a positive integer** 1, 2, 3, ...
- **EVENT_NAME** must match the name of an event in the EventPlanner exactly.

Examples:

- **list**
`register 2 AY201718 Graduation`
 Registers the 2nd person in the event planner to AY201718 Graduation.
- **find Betsy**
`register 1 Computing Seminar`
 Registers the 1st person in the results of the **find** command to Computing Seminar.

3.14. Deregistering a person from an event: **deregister** [Since v1.2]

Deregisters the specified person from an event in the event planner.

Format: **deregister** **INDEX** **EVENT_NAME**

- Deregisters the person at the specified **INDEX**.
- The index refers to the index number shown in the most recent listing.
- The index **must be a positive integer** 1, 2, 3, ...
- **EVENT_NAME** must match the name of an event in the EventPlanner exactly.
- The person to be deregistered must be already in the event.

Examples:

- **list**
`deregister 2 AY201718 Graduation`
 Deregisters the 2nd person in the event planner from AY201718 Graduation.
- **find Betsy**
`deregister 1 Computing Seminar`
 Deregisters the 1st person in the results of the **find** command to Computing Seminar.

3.15. Listing all persons registered in an event: **list-registered** [Since v1.2]

Lists all persons registered for the specified event.

Format: **list-registered** **EVENT_NAME**

3.16. Listing entered commands : **history**

Lists all the commands that you have entered in reverse chronological order.

Format: **history**

NOTE

Pressing the up and down arrows will display the previous and next input respectively in the command box.

3.17. Undoing previous command : **undo**

Restores the event planner to the state before the previous *undoable* command was executed.

There is no guarantee that relative ordering of persons/events will be maintained after the undoing of a **delete** or **delete-event** command.

Format: **undo**

NOTE

Undoable commands: those commands that modify the event planner's content. Currently, the undoable commands are **add**, **delete**, **add-event**, **delete-event**. More commands will be added to this list by **v1.5**

Examples:

- **delete 1**
list
undo (reverses the **delete 1** command)
- **select 1**
list
undo
The **undo** command fails as there are no undoable commands executed previously.
- **delete 1**
clear
undo (reverses the **clear** command)
undo (reverses the **delete 1** command)

3.18. Redoing the previously undone command : **redo**

Reverses the most recent **undo** command.

Format: **redo**

Examples:

- **delete 1**
undo (reverses the **delete 1** command)
redo (reapplies the **delete 1** command)
- **delete 1**
redo
The **redo** command fails as there are no **undo** commands executed previously.

- `delete 1`
`clear`
`undo` (reverses the `clear` command)
`undo` (reverses the `delete 1` command)
`redo` (reapplies the `delete 1` command)
`redo` (reapplies the `clear` command)

3.19. Clearing all entries : `clear`

Clears all entries from the event planner.

Format: `clear`

3.20. Exiting the program : `exit`

Exits the program.

Format: `exit`

3.21. Saving the data

Event planner data are saved in the hard disk automatically after any command that changes the data.

There is no need to save manually.

3.22. Marking attendance for event participant : `mark` `[coming in v1.5]`

Marks the attendance of a participant to an event.

Format: `mark INDEX EVENT_NAME`

3.23. Unmarking attendance for event participant : `unmark [coming in v1.5]`

Unmarks the attendance of a participant to an event.

Format: `unmark INDEX EVENT_NAME`

3.24. Exporting contact information to a csv file : `export [coming in v1.5]`

Exports the list of all persons in the event planner to a csv file.

Format: `export FILE_NAME`

3.25. Importing contact information from a csv file :

import [coming in v1.5]

Imports a list of persons from a csv file and adds them to the event planner.

Format: **import** FILE_NAME

3.26. Adding feedback for an event from a participant :

add-feedback [coming in v2.0]

Adds feedback for an event from a participant of the event.

Format: **add-feedback** INDEX EVENT-NAME FEEDBACK

3.27. Exporting event details to a csv file : **export-event** [coming in v2.0]

Exports event details for a single event into a csv file.

Format: **export-event** EVENT_NAME FILE_NAME

3.28. Registering all persons with a particular tag to an event : **register-by-tag** [coming in v2.0]

Registers all persons with a particular tag to an event in the event planner.

Format: **register-by-tag** TAG EVENT_NAME

3.29. Adding automatic email reminder to event participants : **add-reminder** [coming in v2.0]

Adds an automatic email reminder at the stipulated time to all event participants.

Format: **add-reminder** DAYS_OFFSET EVENT_NAME

3.30. Adding hotkeys for commands : **add-hotkey** [coming in v2.0]

Adds a hotkey for a command.

Format: **add-hotkey** HOTKEY COMMAND_NAME

4. FAQ

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 Event Planner folder.

5. Command Summary

- **Add Event** `add-event n/NAME [t/TAG]...`
e.g. `add-event n/AY201718 Graduation t/Graduation`
- **Add Person** `add n/NAME p/PHONE_NUMBER e/EMAIL a/ADDRESS [t/TAG]...`
e.g. `add n/James Ho p/22224444 e/jamesho@example.com a/123, Clementi Rd, 1234665 t/friend t/colleague`
- **Clear Persons and Events** : `clear`
- **Delete Event** : `delete-event INDEX`
e.g. `delete-event 3`
- **Delete Person** : `delete INDEX`
e.g. `delete 3`
- **Deregister Person** : `deregister INDEX EVENT_NAME`
e.g. `deregister 1 Computing Seminar`
- **Edit Event** : `edit-event INDEX [n/NAME] [t/TAG]...`
e.g. `edit-event 1 n/IoT Talk t/talk`
- **Edit Person** : `edit INDEX [n/NAME] [p/PHONE_NUMBER] [e/EMAIL] [a/ADDRESS] [t/TAG]...`
e.g. `edit 2 n/James Lee e/jameslee@example.com`
- **Find Event** : `find-event KEYWORD [MORE_KEYWORDS]`
e.g. `find-event Talk Competition`
- **Find Person** : `find KEYWORD [MORE_KEYWORDS]`
e.g. `find James Jake`
- **Help** : `help`
- **History** : `history`
- **List Events** : `list-events`
- **List Persons** : `list`
- **List Registered Persons** : `list-registered`
- **Redo** : `redo`
- **Register Person** : `register INDEX EVENT_NAME`
e.g. `register 1 Computing Seminar`
- **Select Person** : `select INDEX`
e.g. `select 2`
- **Undo** : `undo`