ClerkPro v2.0 - User Guide

By: `T09-3` Since: `Sep 2019` Licence: `MIT`

1. Introduction

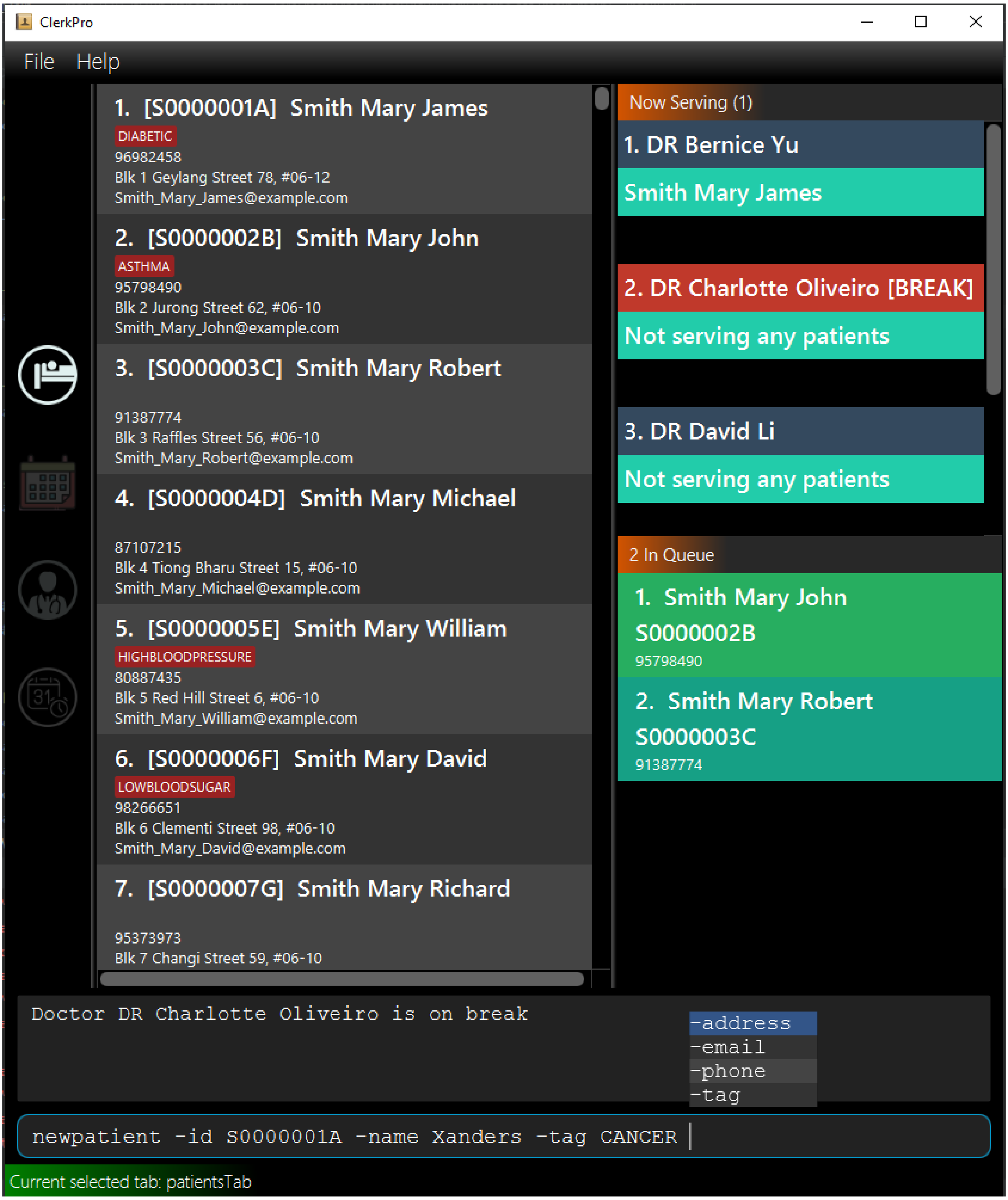
ClerkPro is an appointment and queue management system targeted at clerks working in small clinics. This desktop application is optimised for individuals who **prefer to work with a Command Line Interface (CLI).** while still having the benefits of a Graphical User Interface (GUI).

ClerkPro enables clerks to rapidly manage patient's details, appointments, queues and the staff's duty roster. Typing fast allows you to complete your tasks much faster than GUI apps.

Interested? Refer to Section 2, "Quick Start" to get started. Enjoy!

2. Quick Start

1. Ensure you have Java `11` or above installed in your Computer.
2. Download the latest `ClerkPro.jar` here.
3. Copy the file to the folder you want to use as the home folder for ClerkPro.
4. Double-click the file to start the app. The User Interface (UI) should appear in a few seconds. For testing purposes, ClerkPro is pre-loaded with sample data to allow you to explore and familiarise with the application.



1. [S0000001A] Smith Mary James
DIABETIC
96982458
Blk 1 Geylang Street 78, #06-12
Smith_Mary_James@example.com

2. [S0000002B] Smith Mary John
ASTHMA
95798490
Blk 2 Jurong Street 62, #06-10
Smith_Mary_John@example.com

3. [S0000003C] Smith Mary Robert
91387774
Blk 3 Raffles Street 56, #06-10
Smith_Mary_Robert@example.com

4. [S0000004D] Smith Mary Michael
87107215
Blk 4 Tiong Bharu Street 15, #06-10
Smith_Mary_Michael@example.com

5. [S0000005E] Smith Mary William
HIGHBLOODPRESSURE
80887435
Blk 5 Red Hill Street 6, #06-10
Smith_Mary_William@example.com

6. [S0000006F] Smith Mary David
LOWBLOODSUGAR
98266651
Blk 6 Clementi Street 98, #06-10
Smith_Mary_David@example.com

7. [S0000007G] Smith Mary Richard
95373973
Blk 7 Changi Street 59, #06-10

1. DR Bernice Yu
Smith Mary James

2. DR Charlotte Oliveiro [BREAK]
Not serving any patients

3. DR David Li
Not serving any patients

1. Smith Mary John
S0000002B
95798490

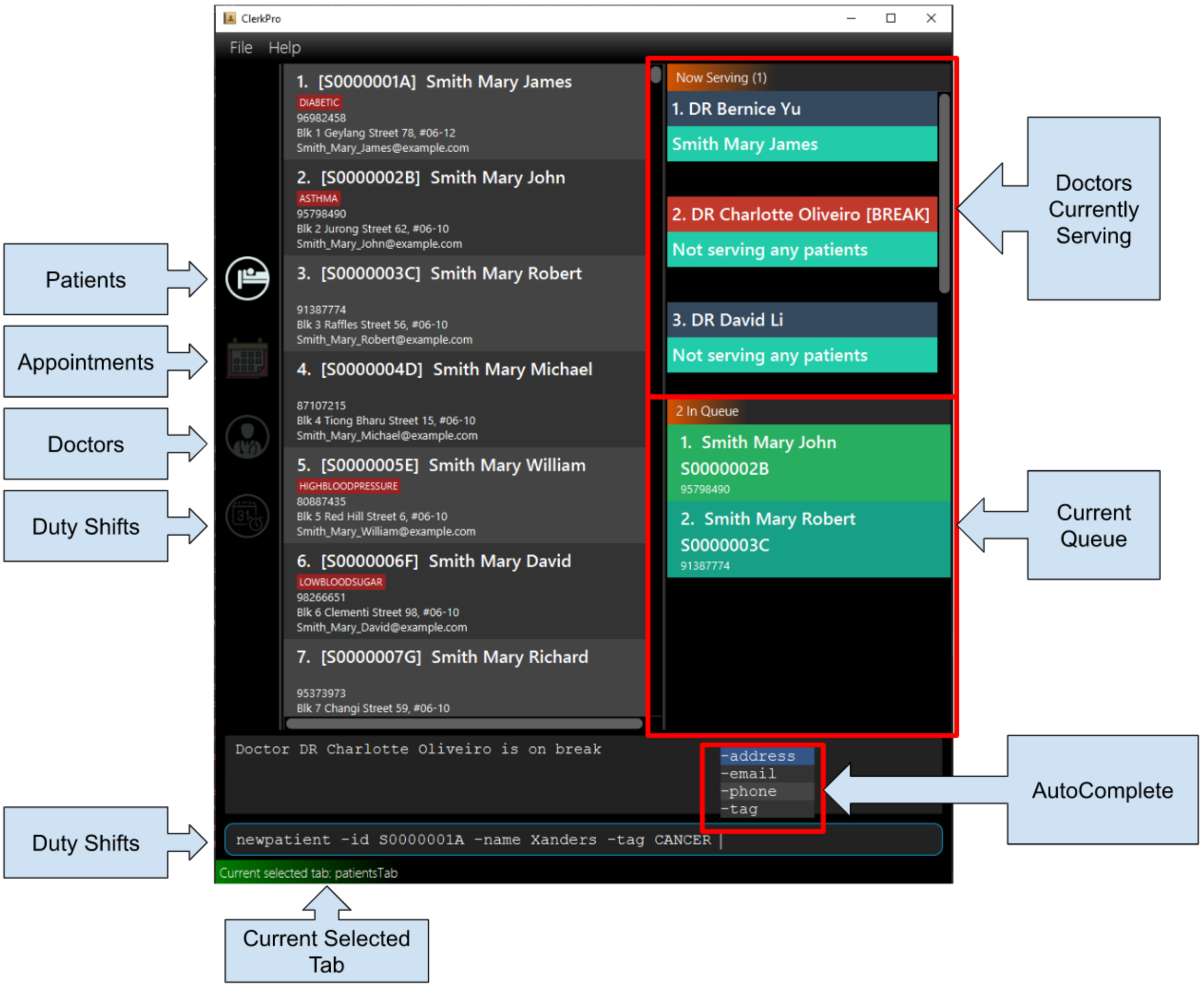
2. Smith Mary Robert
S0000003C
91387774

```
Doctor DR Charlotte Oliveiro is on break
```

```
newpatient -id S0000001A -name Xanders -tag CANCER
```

Current selected tab: patientsTab

5. You may navigate through the entire UI with `UP`, `DOWN`, `LEFT`, `RIGHT` and `TAB`.
6. The figure below shows the layout.



1. [S0000001A] Smith Mary James
DIABETIC
96982458
Blk 1 Geylang Street 78, #06-12
Smith_Mary_James@example.com

2. [S0000002B] Smith Mary John
ASTHMA
95798490
Blk 2 Jurong Street 62, #06-10
Smith_Mary_John@example.com

3. [S0000003C] Smith Mary Robert
91387774
Blk 3 Raffles Street 56, #06-10
Smith_Mary_Robert@example.com

4. [S0000004D] Smith Mary Michael
87107215
Blk 4 Tiong Bharu Street 15, #06-10
Smith_Mary_Michael@example.com

5. [S0000005E] Smith Mary William
HIGHBLOODPRESSURE
80887435
Blk 5 Red Hill Street 6, #06-10
Smith_Mary_William@example.com

6. [S0000006F] Smith Mary David
LOWBLOODSUGAR
98266651
Blk 6 Clementi Street 98, #06-10
Smith_Mary_David@example.com

7. [S0000007G] Smith Mary Richard
95373973
Blk 7 Changi Street 59, #06-10

Now Serving (1)

1. DR Bernice Yu
Smith Mary James

2. DR Charlotte Oliveiro [BREAK]
Not serving any patients

3. DR David Li
Not serving any patients

2 In Queue

1. Smith Mary John
S0000002B
95798490

2. Smith Mary Robert
S0000003C
91387774

```
Doctor DR Charlotte Oliveiro is on break
```

```
-address
-email
-phone
-tag
```

```
newpatient -id S0000001A -name Xanders -tag CANCER
```

Current selected tab: patientsTab

7. 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. Alternatively, just press `F1`.

8. Some example commands you can try:

   - `newpatient -id E0000001A -name John Doe -phone 98765432 -email johnd@example.com -address John street, block 123, #01-01` : registers a patient named `John Doe`.
   - `patient John Doe` : searches for a patient whose name contains `John Doe`
   - `enqueue E0000001A` : enqueues patient with reference id `E0000001A` into the queue. The patient `E0000001A` must be registered.
   - **`exit`** : exits the app.

9. Refer to Section 3, “Features” for details of each command.

Do not directly edit any of the following json files: e.g. `appointments.json`, `dutyRoster.json`, `patientDetails.json`, `staffDetails.json`, `preferences.json`, `config.json`

3. Features

AutoComplete & History

- When the command box is blank, `UP` & `DOWN` will auto-fill the command box with commands previously entered. `UP` gives older commands and `DOWN` gives newer commands.
- When the command box is not blank, AutoComplete will come into action and tries to suggest commands that are available for whatever that has been typed. During which, you may traverse your selection of the suggestions via `UP` & `DOWN` then press `ENTER` to confirm selection.

Full keyboard navigable UI

- `LEFT` & `RIGHT` changes focus between OmniPanel and TabBar when focus is not on CommandBox.
- `TAB` changes focus from TabBar/OmniPanel to CommandBox and from CommandBox to TabBar.

Reactive search

- The listing automatically filters to display only the most relevant results when searching for an existing entry.
- When searching for a specific patient or staff, using the `patient` or `doctor` command, the results are filtered based on whether the person's reference Id, name or phone number contains the given keyword.
- When searching for a specific appointment or duty shift, using the `appointments` or `shifts` command, the results are filtered based on whether the events involves a person whose reference Id contains the given keyword.

Validity of Appointment's timings

- Ensures that appointments can only be scheduled at timings when there are sufficient staffing.
- The scheduling of a patient's appointment will be rejected by the system, if the doctors on-duty at that timing are already pre-occupied by other appointments.
- The cancellation or changing of a doctor's duty shift will be rejected by the system, if the action results in there being insufficient staffing to accommodate already scheduled appointments.

Command Format

- Words in `<UPPER_CASE>` are the parameters to be supplied by the user e.g. in `register -name <NAME>`, `NAME` is a parameter which can be used as `register -name John Doe`.
- Items in square brackets are optional e.g `-name <NAME> [-tag <TAG>]` can be used as `-name John Doe -tag friend` or as `-name John Doe`.
- Items with `…` after them can be used multiple times including zero times e.g. `[-tag <TAG>]…` can be used as   (i.e. 0 times), `-tag friend`, `-tag friend -tag family` etc.
- Parameters can be in any order e.g. if the command specifies `-name <NAME> -phone <PHONE_NUMBER>`, `-phone <PHONE_NUMBER> -name <NAME>` is also acceptable.
- A Reference Id refers to a 9 case-insensitive alphanumeric characters, that can be used to uniquely identify a person, such as a membership number or NRIC.
- A `<PATIENT_REFERENCE_ID>` refers to the reference id which belongs to a patient. Whereas, a `<STAFF_REFERENCE_ID>` refers to the reference id of a staff doctor.
- Words in triangle brackets(`<` and `>`) are the parameters to be supplied by the user e.g. in `register -name <NAME>`, `NAME` is a parameter which can be used as `register -name John Doe`.

IMPORTANT All timings are based on the system's clock.

3.1. Patient Management

3.1.1. Searches patient by arguments: `patient`

Filters patients whose reference ids, names or phone number contains the given keyword.
If no search keyword is provided, all registered patients will be displayed.

Format: `patient [<SEARCH_KEYWORD>]`

- e.g. `patient S0000001A`

3.1.2. Registers a new patient: `newpatient`

Registers a new patient. Only patient's reference id and name are compulsory fields

Format: `newpatient -id <PATIENT_REFERENCE_ID> -name <PATIENT_NAME> [-phone <PHONE_NUM>] [-email <EMAIL>] [-address <ADDRESS>] [-tag <Tags>]…`

- e.g. `newpatient -id E0000001A -name Edmond Halley -phone 85732743 -email halley@example.com -address 12, Kent ridge Ave 3, #01-11 -tag AIDS`

IMPORTANT As a design decision, undoing the registration of a patient is allowed, via the `undo` command. However, though our target users are clerks working at clinics, our clients would be the employers and stakeholders of said clinics. Hence, the user **should not have administrative privilege to unregister any patient already in the system.**

3.1.3. Updates patients’ profiles: `editpatient`

Updates the particulars of a patients

NOTE Editing tags will overwrite all existing tags. Must not edit when patient is in queue or being served.

Format: `editpatient -entry <ENTRY_ID> [-id <PATIENT_REFERENCE_ID>] [-name <NAME>] [-phone <PHONE_NUM>] [-email <EMAIL>] [-address <ADDRESS>] [-tag <Tags>]…`

- e.g. `editpatient -entry 1 -phone 91200567 -email edmond@example.com`

3.2. Staff Personal Management

3.2.1. Searches staff doctor by arguments: `doctor`

Filters staff doctor whose reference ids, names or phone number contains the given keyword.
If no search keyword is provided, all registered doctors will be displayed.

Format: `doctor [<SEARCH_KEYWORD>]`

- e.g. `doctor S0111111A`

3.2.2. Registers a new doctor: `newdoctor`

Registers a new doctor. Only doctor’s reference id and name are compulsory fields.

Format: `newdoctor -id <STAFF_REFERENCE_ID> -name <NAME> [-phone <PHONE_NUM>] [-email <EMAIL>] [-address <ADDRESS>] [-tag <TAGS>]…`

- e.g. `newdoctor -id W0000001A -name John Doe -phone 98765432 -email johnd@example.com -address 311, Clementi Ave 2, #02-25 -tag Male`

IMPORTANT As a design decision, undoing the registration of a staff doctor is allowed, via the `undo` command. However, though our target users are clerks working at clinics, our clients would be the employers and stakeholders of said clinics. Hence, the user **should not have administrative privilege to unregister any staff doctor already in the system.**

3.2.3. Updates a doctor’s profile: `editdoctor`

If user is in the doctors listing screen, updates any change for doctor’s profiles.

NOTE Editing tags will overwrite all existing tags. Must not edit when the doctor is on duty.

Format: `editdoctor -entry <ENTRY_ID> [-id <STAFF_REFERENCE_ID>] [-name <NAME>] [-phone <PHONE_NUM>] [-email <EMAIL>] [-address <ADDRESS>] [-tag <Tags>]…`

- e.g. `editdoctor -entry 1 -phone 91234567 -email johndoe@example.com`

3.2.4. Marks the doctor as on-duty: `onduty`

Marks the doctor, based on the index given, as on-duty and adds him/her to a list of on-duty doctors.

Format: `onduty <ON_DUTY_DOCTOR_ENTRY_ID>`

- e.g. `onduty 1`

3.2.5. Marks the doctor as off-duty: `offduty`

Marks the doctor, based on the index given, as off-duty and removes him/her from the list.

Format: `offduty <ON_DUTY_DOCTOR_ENTRY_ID>`

- e.g. `offduty 1`

3.3. Queue Management

3.3.1. Adds a patient to the queue: `enqueue`

Adds a patient to the queue based based on the patient's Id. The enqueued patient must be a registered. Staff members cannot be enqueued.

Format: `enqueue <PATIENT_REFERENCE_ID>`

- e.g. `enqueue E0000001A`

3.3.2. Removes a patient from the queue: `dequeue`

Removes a patient from the queue based on their queue position.

Format: `dequeue <QUEUE_INDEX>`

- e.g. `dequeue E0000001A`

3.3.3. Assigns next patient to an available doctor : `next`

Assigns the next patient in the queue to a doctor.

Format: `next <ENTRY_ID>`

- e.g. `next 1`

3.3.4. Doctor takes a break: `break`

Avoids directing patients to a doctor. e.g. Doctor is on a lunch break

Format: `break <ENTRY_ID>`

- e.g. `break 1`

3.3.5. Doctor resumes his/her duty: `resume`

Allows patients to be directed to a doctor. e.g. Doctor is back from his/her break.

Format: `resume <ENTRY_ID>`

- e.g. `resume 1`

3.4. Appointment Management

NOTE	Before you process any commands in appointment management, please ensure you have a patient with referenceid of E0000001A by using `patient E0000001A`. You can refer to `newpatient` command to register this patient if the patient is not registered.

3.4.1. Displays appointments: `appointments`

Displays a sorted list of upcoming appointments for the patient that is associated to the given `REFERENCE_ID`. If no keyword is given, all upcoming appointments will be displayed.

Format: `appointments [<REFERENCE_ID>]`

3.4.2. Schedules an appointment: `newappt`

Schedules a new appointments for a patient.

NOTE	The appointment(s) will be rejected by the system, if there are insufficient staff doctors on duty at the time of the appointment. e.g. Cannot schedule more appointments than available doctors on duty.

If both `-reoccur <INTERVALS>` and `-num <REOCCURRING_TIMES>` fields are present, appointments of these `<REOCCURRING_TIMES>` will be added to the patient appointment schedule in `<INTERVALS>`. Otherwise, only one event will be added.

If the optional field `[-end <END_TIMING>]` is absent, default end time is 30 mins after start time of the appointment. Otherwise, end time will be `[-end <END_TIMING>]`.

The optional field `[-reoccur <INTERVALS>]` can be `-reoccur w`, `-reoccur m`, or `-reoccur y`.They represent to add weekly, monthly, yearly repeat appointment respectively.

Format: `newappt -id <PATIENT_REFERENCE_ID> -start <START_TIMING> [-end <END_TIMING>] [-reoccur <INTERVALS> -num <REOCCURRING_TIMES>]`

3.4.3. Acknowledges a appointment: `ackappt`

Acknowledges the most upcoming appointment only if patient arrives on the same day and the arriving time is before the appointment's end time.
Format: `ackappt <REFERENCE_ID>`

3.4.4. Cancels an appointment: `cancelappt`

Cancels the specified appointment.

NOTE To avoid accidental cancellation of another patient’s appointments, the user must first narrow down the search to a single patient by using the `appointments [<REFERENCE_ID>]` command.

Format: `cancelappt <ENTRY_ID>`

3.4.5. Changes the appointment date: `editappt`

Changes the timing for an existing appointment.
If optional field `[-end <END_TIMING>]` is not present, default endTiming is 30 mins after startTiming. Otherwise new endTiming will be `[-end <END_TIMING>]`.

NOTE To avoid accidental rescheduling of another patient’s appointments, the user must first narrow down the search to a single patient using the `appointments [<REFERENCE_ID>]` command.

NOTE The operation is rejected if there are insufficient staff doctors on duty at the time of the new appointment.

Format: `editappt -entry <ENTRY_ID> -start <START_TIMING> [-end <END_TIMING>]`

3.4.6. Lists patients who have missed their appointments: `missappt`

Lists all appointments that are missed.

NOTE An appointment is considered missed if the appointment was not acknowledged and the current time has passed the appointment’s end time.

Format: `missappt`

3.4.7. Sets missed appointments as settled/notified: `settleappt`

Settles and removes the missed appointment based on the given index.
Settling refers to the user following up on contacting the patient who has missed his/her appointment.

NOTE Only missed appointments can be settled. The user must first display the missed appointment listing by using `missappt`, before using this command.

Format: `settleappt <ENTRY_ID>`

3.4.8. Shows the empty slots: `slot` (v2.0)

List all the available empty slots for patients to make appointments

Format: `slot -start <START_DATE>`

3.5. Duty-shift Management

NOTE Before you process any commands in Duty-shift Management, please ensure you have a doctor with referenceId of W0000001A by using `doctor W0000001A`.
You can refer to `newdoctor` command to register this doctor if the doctor is not registered.

3.5.1. Displays duty shifts: `shifts`

Displays a sorted list of upcoming duty shifts for the staff doctors that is associated to the given `REFERENCE_ID`. If no keyword is given, all upcoming shifts will be displayed.

Format: `shifts [<REFERENCE_ID>]`

3.5.2. Adds a duty shift for a doctor: `newshift`

If both `-reoccur <INTERVALS>` and `-num <REOCCURRING_TIMES>` fields are present, duty shifts of these `<REOCCURRING_TIMES>` will be added to the doctor duty roster in `<INTERVALS>`. Otherwise, only one shift will be added.

Format: `newshift -id <STAFF_REFERENCE_ID> -start <START_TIMING> -end <END_TIMING> [-reoccur INTERVALS -num REOCCURRING_TIMES]`

3.5.3. Cancels a duty shift for a doctor: `cancelshift`

Cancels the specified duty shift for a doctor based on the given entry index.

NOTE To avoid accidental cancellation of another doctor’s shift, the user must first narrow down the search to a single doctor using the `shift [<REFERENCE_ID>]` command.

NOTE The operation is rejected if it results in there being more appointments than staff doctors on duty at any given point in time. The appointments affected by the deletion of the duty shift must be resolved first.

Format: `cancelshift <ENTRY_ID>`

3.5.4. Changes the duty shift for a doctor: `editshift`

Changes the timing of an existing duty shift based on the given entry index.

NOTE To avoid accidental cancellation of another doctor’s shift, the user must first narrow down the search to a single doctor using the `shift [<REFERENCE_ID>]` command.

NOTE The operation is rejected if it results in there being more appointments than staff doctors on duty at any given point in time.

The patient's appointment affected by the editing of the duty shift must be resolved first.

Format: `editshift -entry <ENTRY_ID> -start <START_TIMING> -end <END_TIMING>`

3.6. Inventory Commands (v2.0)

3.6.1. Views the inventory: `inventory` (v2.0)

Displays the medicine inventory.

Format: `inventory`

3.6.2. Views the inventory: `prescription` (v2.0)

Displays the prescription. A prescription is a list of medications prescribed by the doctor.

Format: `prescription <PRESCRIPTION_ID | PRESCRIPTION_NAME>`

3.7. User Accounts (v2.0)

3.7.1. Login: `login` (v2.0)

Login with your username and password. The user will then be prompted to enter their password.

Format: `login <USER_NAME>`

3.7.2. Logout: `logout` (v2.0)

Logout from your account

Format: `logout`

3.8. General Commands

Note: Search actions refers to the `patient`, `doctor`, `appointments` and `shift` commands.

3.8.1. Views help: `help`

Displays the help window.

Format: `help`

3.8.2. Exits program: `exit`

Exits the program.

Format: `exit`

3.8.3. Undo action: `undo`

Undoes an action.
All except search actions can be undone.

Format: `undo`

3.8.4. Redo action: `redo`

Redo an action.
All except search actions can be re-done.

Format: `redo`

4. Commands Summary

- **Patient Management**
  - Search for patient using reference Id, name or phone number: `patient [<SEARCH_KEYWORD>]`
  - Register new patient: `newpatient -id <PATIENT_REFERENCE_ID> -name <PATIENT_NAME> [-phone <PHONE_NUM>] [-email <EMAIL>] [-address <ADDRESS>] -num [-tag <Tags>]…`
  - Edits patient details: `editpatient -entry <ENTRY_ID>[-id <PATIENT_REFERENCE_ID>] [-name <NAME>] [-phone <PHONE_NUM>] [-email <EMAIL>] [-address <ADDRESS>] -num [-tag <Tags>]…`
- **On-Duty Doctors Management**
  - Search for doctors using reference Id, name or phone number: `doctor [<SEARCH_KEYWORD>]`
  - Register new doctor: `newdoctor -id <STAFF_REFERENCE_ID> -name <NAME> [-phone <PHONE_NUM>] [-email <EMAIL>] [-address <ADDRESS>] [-tag <TAGS>]…`
  - Edit doctor details: `editdoctor -entry <ENTRY_ID> [-id <STAFF_REFERENCE_ID>] [-name <NAME>] [-phone <PHONE_NUM>] [-email <EMAIL>] [-address <ADDRESS>]-num`
  - Mark doctor as on-duty: `onduty <ENTRY_ID>`
  - Mark doctor as off-duty: `offduty <ENTRY_ID>`
- **Queue Management**
  - enqueue: `enqueue <PATIENT_REFERENCE_ID>`
  - dequeue: `dequeue <QUEUE_INDEX>`
  - Assigns next Patient in queue to doctor: `next <DOCTOR_ENTRY_ID>`
  - Marks doctor on break: `break <DOCTOR_ENTRY_ID>`
  - Marks doctor on resuming work: `resume <DOCTOR_ENTRY_ID>`
- **Appointment Management**
  - Search for appointments: `appointments [<REFERENCE_ID>]`
  - Add new appointment: `newappt -id <REFERENCE_ID> -start <START_TIMING> [-end <END_TIMING>] [-reoccur <INTERVALS> -num <REOCCURRING_TIMES>]`
  - Edit appointment: `editappt -entry <ENTRY_ID> -start <START_TIMING> [-end <END_TIMING>]`
  - Cancel appointment: `cancelappt <ENTRY_ID>`

    - Acknowledge arrival of patient for appointment: `ackappt <REFERENCE_ID>`
    - List all missed appointments: `missappt`
    - Mark missed appointment as settled: `settleappt <ENTRY_ID>`
- **Duty-shift Management**
    - Search for shift: `shifts [<REFERENCE_ID>]`
    - Add new shift: `newshift -id STAFF_REFERENCE_ID -start <START_TIMING> -end <END_TIMING> [-reoccur <INTERVALS> -num <REOCCURRING_TIMES>]`
    - Change shift: `editshift -entry <ENTRY_ID> -start <START_TIMING> -end <END_TIMING>`
    - Cancel shift: `cancelshift <ENTRY_ID>`
- **Inventory commands (v2.0)**
    - inventory: `inventory`
    - prescription: `prescription <PRESCRIPTION_ID | PRESCRIPTION_NAME>`
- **User Accounts (v2.0)**
    - login: `login <USER_NAME>`
    - logout: `logout`
- **General Commands**
    - help: `help`
    - exit: `exit`
    - undo: `undo`
    - redo: `redo`