User Guide: MediNote

Introduction

MediNote is a desktop app for managing hospital information such as doctor and patient records, optimised for use via a Command Line Interface (CLI).

💡 Quick Start 💡

1. Ensure that you have **Java 17** or above installed.
2. Down the latest version of ▶️ `MediNote` from here.
3. Copy the application to the directory you want to use as the **working directory**.
4. In your terminal, move to the directory containing the application and run it with `java -jar tP_V2.jar`.

Features

- Show a list of commands: `help`
- Listing all patients: `list patient`
- Listing all doctors: `list doctor`
- Viewing specific doctors: `doctor`
- View patient information: `patient`
- Register a new patient: `register`
- Add a new doctor: `oncall`
- Discharge a patient: `discharge patient <NAME>`
- Delete a doctor: `delete doctor <NAME>`
- Updates Patient or Doctor in question: `update`
- Display popular doctor or specialisation: `popular`
- View specific patient attribute across all patients: `list patient <ATTRIBUTE>`
- View specific patient attribute individually: `view patient <ATTRIBUTE>`
- View specific doctor attribute individually: `view doctor <ATTRIBUTE>`
- Display Hospital Summary Stats: `stats`
- To Exit Application: `exit`

! Note:

For NAME, format MUST BE:

- `Mr-Tan`

- John-Doe

- Avery-Miller-Dove

- Dr-Hudson

Show a list of commands: `help`

Shows a list of commands for navigating around the database. Provides an explanation on how different commands work to bring out or add the desired information from or into the database.

Format: `help`

Example output:

```
=============================================================================================
Keyword             | Action
=============================================================================================
help                | Shows a list of commands for navigating the database
                    |
register            | Registers new patient with necessary details:
                    | • <NAME>
                    | • <SYMPTOMS>
                    | • <ADMISSION TIME>
                    | • <MEDICAL HISTORY>
                    | Format: register <NAME> / <SYMPTOMS> / <ADMISSION TIME> / <MEDICAL HISTORY>
                    |
oncall              | Register on-call doctor(s) with necessary details:
                    | • <NAME>
                    | • <SPECIALISATION>
                    | Format: oncall <NAME> / <SPECIALISATION>
                    |
patient             | Displays specific patient information such as:
                    | • Name
                    | • Symptoms
                    | • Admission time
                    | • Medical history
                    | • Treatment status (if any)
                    | • Doctors assigned (if any)
                    | Format: patient <NAME_1> / ... / <NAME_X>
                    | Note: <NAME> inputted does not need to be case-sensitive
                    |
view patient        | Displays each patient's specific attribute one by one:
                    | Available attributes:
                    |  • name
                    |  • symptoms
                    |  • timestamp
                    |  • history
                    |  • treatment
                    |  • doctor
                    | Format: view patient <ATTRIBUTE>
                    |
```

```
  doctor               | Displays specific doctor information such as:
                       | • Specialisation
                       | • Availability
                       | • Patients under the doctor (if any)
                       | Format: doctor <NAME_1> / ... / <NAME_X>
                       | Note: <NAME> inputted does not need to be case-sensitive
                       |
  view doctor          | Displays each doctor's specific attribute one by one:
                       | Available attributes:
                       |  • name
                       |  • specialisation
                       |  • availability
                       |  • treating
                       |  • numtreated
                       | Format: view doctor <ATTRIBUTE>
                       |
  list patient         | Displays all the patients in the patients class array that are yet to be discha
  list doctor          | Displays all doctors available in the hospital
                       |
                       |
  update patient       | Updates corresponding patient's information on:
                       | • <TREATMENT STATUS> whether they are still in queue or being treated
                       | • <DOCTOR ASSIGNED> the doctor that they are assigned
                       |
  update doctor        | Updates corresponding doctor information on:
                       | • <AVAILABILITY> their current working status
                       | • <PATIENT NAME> the patient they are currently treating (if any)
                       |
  discharge patient    | Deletes patient and corresponding information from the database.
  delete doctor        | Deletes doctor and corresponding information from the database
                       |
  clear patients       | Clears all patient entries from the database
  clear doctors        | Clears all doctor entries from the database
                       |
  popular              | Displays most popular doctors based on specified attributes
                       | Available options:
                       |  • doctor type
                       |  • visited doctor
                       |
  stats                | Prints an overview of hospital metrics
                       |  • Total number of patients
                       |  • Number of patients currently being treated
                       |  • Total number of doctors
                       |  • Most active doctor (by patient count)
                       |  • Average patients per doctor
                       |
  exit                 | Exits the program
========================================================================================
```

Listing all current patients: `list patient`

Lists all patients currently admitted, along with all of their attributes:

Format: `list patient`

Example output of a list with 2 patients:

```
You have 2 patient(s) in hospital
1. Mr. A:
    >Time of Admission: 01 JAN 2025
    >Symptoms: Cough
    >Medical History: NA
    >Treatment Status: NA
    >Doctor Assigned: NA

2. Si En:
    >Time of Admission: Tomorrow
    >Symptoms: Eye Fever
    >Medical History: None
    >Treatment Status: NA
    >Doctor Assigned: NA
```

Listing all current doctors: `list doctor`

Lists all doctors currently working, along with all of their attributes:

Format: `list doctor`

Example output of a list with 2 doctors:

```
You have 2 doctor(s) in hospital
1. Pengu:
    >Specialization: Eye Fever Doctor
    >Availability: NA
    >Currently treating: NA

2. Timothy Cheese:
    >Specialization: Knee Pain
    >Availability: NA
    >Currently treating: NA
```

Viewing specified doctor: `doctor`

Views all the attributes of doctors specified in the command:
`doctor` searches for all doctors mentioned by the user.

Format: `doctor <NAME_1>/<NAME_2>/.../<NAME_X>`

Example usage:
`doctor Timothy Cheese/Pengu`

Example output:

```
    Details of specified doctors:
        >Name: Timothy Cheese
            >Specialisation: Knee Pain
            >Availability: NA
            >Currently treating: NA
        >Name: Pengu
```

```
                    >Specialisation: Eye Fever Doctor
                    >Availability: NA
                    >Currently treating: NA
```

Viewing specified patient: `patient`

Views all the attributes of patients specified in the command:
`patient` searches for all patients mentioned by the user.

Format: `patient <NAME_1>/<NAME_2>/.../<NAME_X>`

Example usage:
`patient Mr. A/Si En`

Example output:

```
    Details of specified patients:
        >Name: Mr. A
            >Symptoms: Cough
            >Time of Admission: 01 JAN 2025
            >Medical History: NA
            >Treatment Status: NA
            >Doctor Assigned: NA
        >Name: Si En
            >Symptoms: Eye Fever
            >Time of Admission: Tomorrow
            >Medical History: None
            >Treatment Status: NA
            >Doctor Assigned: NA
```

Registering a new patient: `register`

Registers a new patient with necessary details:

- <NAME>
- <SYMPTOMS>
- <ADMISSION TIME>
- <MEDICAL HISTORY>

Format: `register <NAME> / <SYMPTOMS> / <ADMISSION TIME> / <MEDICAL HISTORY>`

Example usage:

`register John Pork/High Fever/5 Jan 2025 1730/Allergic to cheese`

`register Tim Cheese/Constipation/5:23pm 7 Feb 2026/NA`

Adding a new doctor: `oncall`

Adds a new on-call doctor with necessary details:

- `<NAME>`
- `<SPECIALISATION>`

Format: `oncall <NAME> / <SPECIALISATION>`

Example usage:

`oncall Simon Claw/Cardiologist`

Discharge a patient: `discharge patient <NAME>`

Deletes patient and their corresponding information from the database.

Format: `discharge patient <NAME>`

Example usage:

`discharge patient Lebron James`

Delete a doctor: `delete doctor <NAME>`

Deletes doctor and their corresponding information from the database.

Format: `delete doctor <NAME>`

Example usage:

`delete doctor Michael Jordan`

Update details about Patient or Doctor: `update`

Updates the details about a particular Patient or Doctor in question.

Format: `update <patient / doctor> <NAME> <Details to be updated according to registration format>`

Popular

Display most popular doctor type: `popular doctor type`

Displays the specialisation with most patient visits. Includes patients currently being treated.

Format & Example usage: `popular doctor type`

Display most visited doctor: `popular visited doctor`

Displays the doctor(s) with most patients treated. Includes patients currently being treated.

Format & Example usage: `popular visited doctor`

View patient attributes: `view patient <ATTRIBUTE>`

Displays each patient's specific attribute one by one with formatting.

Available `<ATTRIBUTE>` values:

- `name`
- `symptoms`
- `timestamp`
- `history`
- `treatment`
- `doctor`

Format: `view patient <ATTRIBUTE>`

Example:

- `view patient treatment`

View doctor attributes: `view doctor <ATTRIBUTE>`

Displays each doctor's specific attribute one by one with formatting.

Available `<ATTRIBUTE>` values:

- `name`
- `specialisation`
- `availability`
- `treating`
- `numtreated`

Format: `view doctor <ATTRIBUTE>`

Example:

- `view doctor specialisation`

Hospital Summary Statistics Dashboard: `stats`

Prints an overview of hospital metrics:

- Total number of patients
- Number of patients currently being treated
- Total number of doctors
- Most active doctor (by patient count)
- Average patients per doctor

Format: `stats`

To Exit: `exit`

Exits from the interface

Format: `exit`

Command Summary

- `help` — Show command reference table
- `register <NAME> / <SYMPTOMS> / <ADMISSION TIME> / <MEDICAL HISTORY>` — Register new patient
- `oncall <NAME> / <SPECIALISATION> / <PATIENT NAME>` — Register new doctor
- `patient <NAME>` — View full patient info
- `doctor <NAME>` — View full doctor info
- `list patient` — List all patients with full details
- `list doctor` — List all doctors
- `view patient <ATTRIBUTE>` — View each patient's attribute individually
- `view doctor <ATTRIBUTE>` — View each doctor's attribute individually
- `update patient <NAME> <TREATMENT STATUS> <DOCTOR ASSIGNED>` — Update a patient's info
- `update doctor <NAME> <AVAILABILITY> <PATIENT NAME>` — Update a doctor's info
- `discharge patient <NAME>` — Delete patient from system
- `delete doctor <NAME>` — Delete doctor from system
- `popular doctor type` — Show most common doctor specialisation
- `popular visited doctor` — Show most visited doctor
- `stats` — Show hospital-wide metrics
- `exit` — Exit the program