

TERMINAL APPLICATION PRESENTATION

Jeevan Ng

PLANT WATERING SCHEDULE TERMINAL APPLICATION

- Add Plant to Schedule
- Remove Plant from Schedule
- View Plants that need Watering
- Mark Plant as Watered
- Update the Amount of Water Needed for a Plant
- View Entire Plant List
- Exit program

IMPORTANT ASPECTS OF THE PROGRAM

- How does the program decide which display which plants need to be watered today?
- What is the logic?

```
if plant["Frequency"].lower() == "weekly" and difference days  
>= 7 or plant["Frequency"].lower() == "fortnightly" and  
difference days >= 14 or plant["Frequency"].lower() ==  
"monthly" and difference days >= 30:
```

ERROR HANDLING

- Raising Exceptions when the user enters an input that we do not want them to.
- Trying to predict what wrong inputs the user may enter and accounting for that.

```
except ValueError as e:
```

DEVELOPMENT AND BUILD PROCESS

- Keeping it simple
- Using Pseudocode to construct the logic
- Styling

ETHICAL ISSUES

- Terminal application does not have great accessibility, especially those with visual impairments.
- Security in regards to how the data is encrypted
- Privacy and collection of personal information

FAVOURITE PARTS

- Overall style of the program
- The simplicity
- The logic to display which plants need to be watered today