Project Report-Kevin Matthew Tanuwijaya-2502036514

What is it?

Reminder bot on discord which reminds if you have any tasks to do for today or upcoming

Command lists of it

!commands

It shows all the lists of commands that exist

!help

Used to specifically help people to make a reminder

!ReminderDay1 until !ReminderDay31

Used to make a reminder for a specific day that day

!ReminderDaily1 until !ReminderDaily5

Used to make a daily reminder, upto5 things you can remind

!reminderSunday until !reminderSaturday

Make a weekly reminder that is stored for that day of the week

!ShowEventsToday

Show all the reminders you have for today

!ShowReminders

Show all the reminders you made for the month

!ShowRemindersThisWeek

Shows todays reminder and all the other reminders coming up in 6 days

!ShowDailyReminders

-used to show all the daily reminders that you made

!ShowRemindersRemaining

Used to show all the remaining reminders you have that month

!CheckTime

Used to show the time to the seconds and show the day today

!reset

-a command used to make all the list from day 1-31

UML-Diagram

TurbotCommands
RemindSentence: String[]
RemindWeek: String[]
RemindDaily: String[]

RemindAndCalen
WeekCHK: String
Hours: integer
Minutes: integer
Seconds: integer
Day: integer
WeekNum: integer
get_Week( ): void
FindHour( ): void
DayNum( ): void
CheckWeek( ): String

How does it work?

For the program to make and store a message in a date, we use the !reminder*number* command to store it in the day itself.

It will look at the number for the date and will store it on the date itself

The command array is made by storing all the arguments made in a text message given in the discord chat

```
//until reminder day 31 all used to make an item on that day
}else if (Command[0].equalsIgnoreCase(CommandCall+"ReminderDay1")) {
    String X=Command[1];
    RemindSentence[1]=X;

}else if (Command[0].equalsIgnoreCase(CommandCall+"ReminderDay2")) {
    String X=Command[1];
    RemindSentence[2]=X;

}else if (Command[0].equalsIgnoreCase(CommandCall+"ReminderDay3")) {
    String X=Command[1];
    RemindSentence[3]=X;

}else if (Command[0].equalsIgnoreCase(CommandCall+"ReminderDay4")) {
    String X=Command[1];
    RemindSentence[4]=X;

}else if (Command[0].equalsIgnoreCase(CommandCall+"ReminderDay5")) {
    String X=Command[1];
    RemindSentence[5]=X;

}else if (Command[0].equalsIgnoreCase(CommandCall+"ReminderDay6")) {
```

For showing the events that day, we can check on the current day and what day of the week it is, then the system will get the array and will get which reminder we have today

```
//used to show what reminders exist today
}else if (Command[0].equalsIgnoreCase(CommandCall+"ShowEventsToday")) {

    //get the day and use it to search immediately on the list
    RNC.DayNum();
    int Tod=RNC.Day;

    String X=RemindSentence[Tod];
    event.getChannel().sendMessage("**Reminders:**").queue();
    event.getChannel().sendMessage(X).queue();

    //shows the weekly reminder
    event.getChannel().sendMessage("**Weekly Reminders:**").queue();
    RNC.get_Week();
    int WeekN=RNC.WeekNum;
    String Y=RemindWeek[WeekN];
    event.getChannel().sendMessage(Y).queue();
```

To show all reminders, a for loop is made to check all the reminders made that month.

```
//show all the reminders that exist
}else if (Command[0].equalsIgnoreCase(CommandCall+"Showreminders")) {

    //for loop for all the list in existence for specific day reminders
    event.getChannel().sendMessage("Reminders:").queue();
    for (int i=0;i<31;i++) {

        String X=RemindSentence[i];
        if (X!=null) {
            event.getChannel().sendMessage("Day "+i+" "+X).queue();
        }

    }
```

There are also those who can search what remaining reminders we have for the month.

```
//used to show the remaining reminders this month
}else if (Command[0].equalsIgnoreCase(CommandCall+"ShowremindersRemaining")) {

    //for loop for all the list in existence for specific day reminders
    event.getChannel().sendMessage("Reminders:").queue();
    RNC.DayNum();
    for (int i=RNC.Day;i<31;i++) {

        String X=RemindSentence[i];
        if (X!=null) {
            event.getChannel().sendMessage("Day "+i+" "+X).queue();
        }

    }
```

A for loop will also go through the array for this to show 7 days of upcoming events, including the day itself.

```
//show all reminders for upcoming week
}else if (Command[0].equalsIgnoreCase(CommandCall+"ShowremindersThisweek")) {
    RNC.DayNum();
    int day=RNC.Day;
    int week=day+7;

    for (int i=day;i<week;i++) {

        String X=RemindSentence[i];
        if (X!=null) {
            event.getChannel().sendMessage("Day "+i+ " "+X).queue();
        }
    }
```

A for loop is used to clear it by giving each value a null or empty value for all the indexes

```
//used to reset the day list
}else if(Command[0].equalsIgnoreCase(CommandCall+"Reset")){
    for(int i=0;i<33;i++) {
        RemindSentence[i]=null;
        event.getChannel().sendMessage("The list has been reset").queue();
    }
```

The program evidence