PATIENTSKY DEVELOPMENT TEST

Purpose: To write a program that can list down the available times for a meeting for several participants

General Overview of the design:

Designed the Service using Spring boot Gradle structure. The Patient sky service is designed to collect the patient Data by reading the Json file and stores the data as key value pair in map in which the key will be calendars id which is the unique factor among the patients and the common factor among appointment, Timeslot entity of the Json file to keep the data in sync.

Currently the service is taking the hardcoded input value. May be in the future this could be enhanced to take the real time input by exposing this functionality as a rest endpoint. After collecting the patient data and inputs, service returns the available time slots to the particular calendar Id.

Service implementation

Service is implemented using below layers

Json Files: Test Patients data are placed in the resource folder of the project.

Main class: Entry point to the service. Creates a bean to the Controller class and invokes the appointmentHandler method of controller class.

- System.exit: Used to terminate the application based on the exit code.
  If exitcode is 0 then Application terminated without any error.
  If exitcode is 1 then Application not ended successfully.

Model Class: Patient data is mapped to the java object.

- **Appointments**:- Model class that is used to map the sub entity Appointment of the patient data Json file
  - @JsonProperty : Annotation is used to match the property instance of the Json file. Since in java it is best practice to use camel casing instead of snake casing.
- **TimeSlots :** Model class that is used to map the sub entity timeslots of the patient data Json file
- **TimeSlotTypes :** Model class that is used to map the sub entity timeslotstypes of the patient data Json file
- **CalendarsData :** This is the wrapper object that is used to hold the Appointments, Timeslots, TimeslotTypes entity together on an object.

Controller Class: Handles the application. The main objective of the class is to collect the patient information and pass it to the service class.

- **getPatientDetails:** This method reads the Json file which is placed under the "src/main/resources" and maps it to CalendarsData entity which holds the patients details like Appointments, Timeslots, TimeslotTypes. This method returns the List of CalendarsData which contains the details of one or more patient.

- **Inputs :** The application takes 4 real time inputs
  - **List<UUID> calendarIds: -** Calendar id's to be searched.
  - **Duration :-** Duration of the meeting
  - **START and END TIME :-** START AND END TIME OF THE MEETING. Basically, represents the interval of the meeting.
  - **timeSlotTypeId:-** this can be either null or can take any specific timeslottypeid.

- **findAvailableTime :** This is a service class method where the logic to fetch available time slot is being implemented.

Service Class: Business logic is implemented. The main objective of the class is to implement the logic to return the available time slot.

- **findAvailableTime :** This iterates through the input calendarIds to checks if the patientDataMap has CalendarsData for the particular calendars id's.
  - If no then displays the appropriate message on the console.
  - If yes then it go ahead with the execution and gets the available time slots.

- **getSlotsWithTimeSlotTypeId :** This checks if given duration and optional timeslotTypeId matches with any of the Timeslots of patient Data.
- **compareTimeInterval :** This checks if the given interval matches or comes within any of the allocated time slot in patients data.

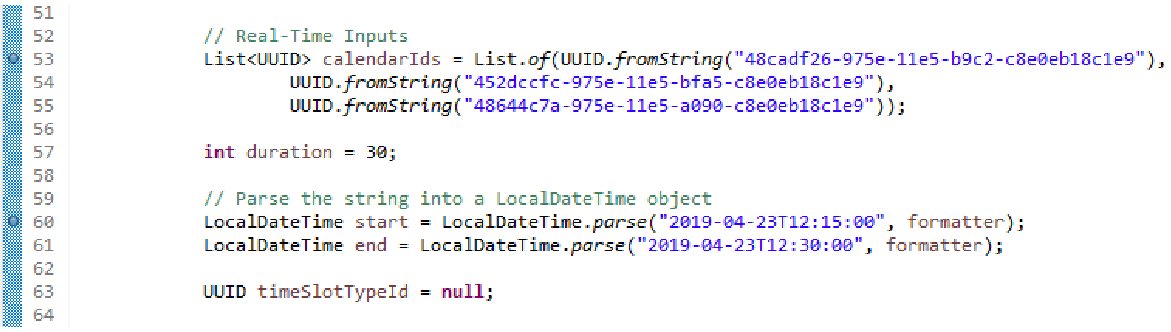
Test Classes : Wrote unit test cases for service and controller class using junits.

Application RUN Instruction :-

Step 1:- Import the project from deepthi-7/PatientSky_Repo: PatientSky coding assessment repository (github.com) to he suitable IDE like STS.

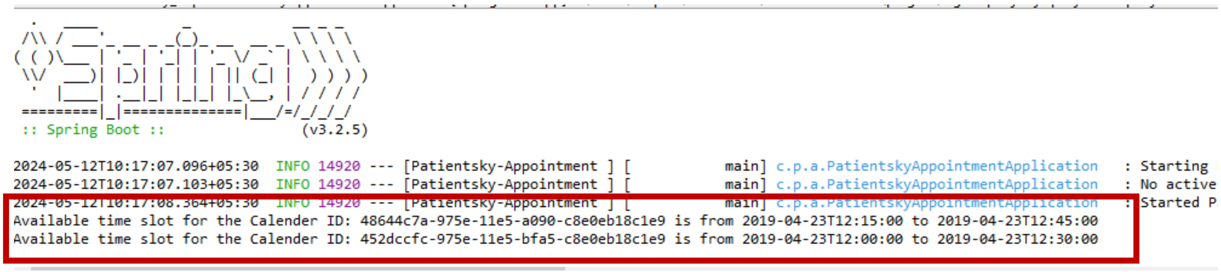
Step 2 :- Provide input to the application by making changes at L53-L63

PatientSky_Repo/src/main/java/com/patientsky/appointment/controller/AppointmentController.java at main · deepthi-7/PatientSky_Repo (github.com)



```
51
52              // Real-Time Inputs
53              List<UUID> calendarIds = List.of(UUID.fromString("48cadf26-975e-11e5-b9c2-c8e0eb18c1e9"),
54                      UUID.fromString("452dccfc-975e-11e5-bfa5-c8e0eb18c1e9"),
55                      UUID.fromString("48644c7a-975e-11e5-a090-c8e0eb18c1e9"));
56
57              int duration = 30;
58
59              // Parse the string into a LocalDateTime object
60              LocalDateTime start = LocalDateTime.parse("2019-04-23T12:15:00", formatter);
61              LocalDateTime end = LocalDateTime.parse("2019-04-23T12:30:00", formatter);
62
63              UUID timeSlotTypeId = null;
64
```

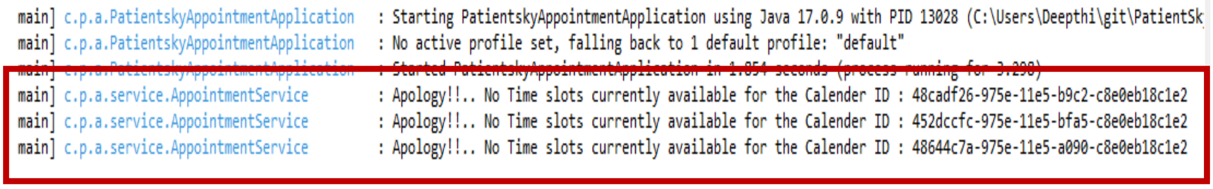
Step 3 :- Run the project as the Spring boot Application



```
  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/
 :: Spring Boot ::                (v3.2.5)

2024-05-12T10:17:07.096+05:30  INFO 14920 --- [Patientsky-Appointment ] [           main] c.p.a.PatientskyAppointmentApplication   : Starting
2024-05-12T10:17:07.103+05:30  INFO 14920 --- [Patientsky-Appointment ] [           main] c.p.a.PatientskyAppointmentApplication   : No active
2024-05-12T10:17:08.364+05:30  INFO 14920 --- [Patientsky-Appointment ] [           main] c.p.a.PatientskyAppointmentApplication   : Started P
Available time slot for the Calender ID: 48644c7a-975e-11e5-a090-c8e0eb18c1e9 is from 2019-04-23T12:15:00 to 2019-04-23T12:45:00
Available time slot for the Calender ID: 452dccfc-975e-11e5-bfa5-c8e0eb18c1e9 is from 2019-04-23T12:00:00 to 2019-04-23T12:30:00
```

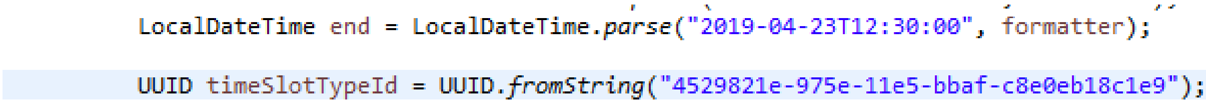
Different Test scenario :-

- **If the Given Calendars Id doesn't match with any of the patient data.**

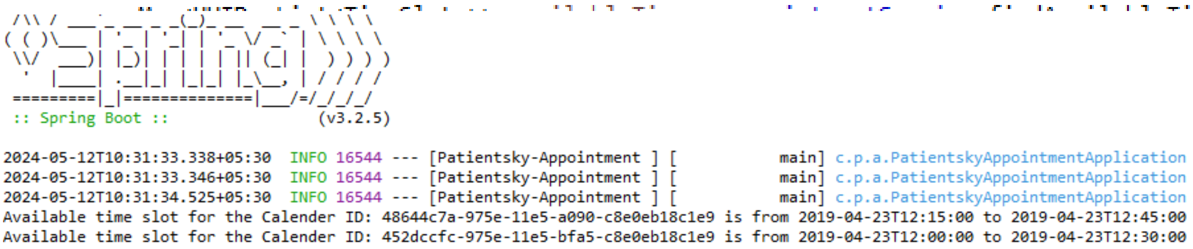


```
main] c.p.a.PatientskyAppointmentApplication    : Starting PatientskyAppointmentApplication using Java 17.0.9 with PID 13028 (C:\Users\Deepthi\git\PatientSk
main] c.p.a.PatientskyAppointmentApplication    : No active profile set, falling back to 1 default profile: "default"
main] c.p.a.PatientskyAppointmentApplication    : Started PatientskyAppointmentApplication in 1.054 seconds (process running for 3.290)
main] c.p.a.service.AppointmentService          : Apology!!.. No Time slots currently available for the Calender ID : 48cadf26-975e-11e5-b9c2-c8e0eb18c1e2
main] c.p.a.service.AppointmentService          : Apology!!.. No Time slots currently available for the Calender ID : 452dccfc-975e-11e5-bfa5-c8e0eb18c1e2
main] c.p.a.service.AppointmentService          : Apology!!.. No Time slots currently available for the Calender ID : 48644c7a-975e-11e5-a090-c8e0eb18c1e2
```

- **If the timeslottypeId id value is given**

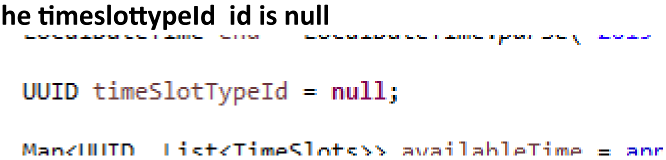


```
        LocalDateTime end = LocalDateTime.parse("2019-04-23T12:30:00", formatter);

        UUID timeSlotTypeId = UUID.fromString("4529821e-975e-11e5-bbaf-c8e0eb18c1e9");
```

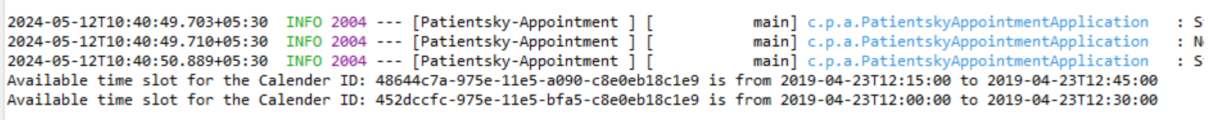


```
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/
 :: Spring Boot ::                (v3.2.5)

2024-05-12T10:31:33.338+05:30  INFO 16544 --- [Patientsky-Appointment ] [           main] c.p.a.PatientskyAppointmentApplication
2024-05-12T10:31:33.346+05:30  INFO 16544 --- [Patientsky-Appointment ] [           main] c.p.a.PatientskyAppointmentApplication
2024-05-12T10:31:34.525+05:30  INFO 16544 --- [Patientsky-Appointment ] [           main] c.p.a.PatientskyAppointmentApplication
Available time slot for the Calender ID: 48644c7a-975e-11e5-a090-c8e0eb18c1e9 is from 2019-04-23T12:15:00 to 2019-04-23T12:45:00
Available time slot for the Calender ID: 452dccfc-975e-11e5-bfa5-c8e0eb18c1e9 is from 2019-04-23T12:00:00 to 2019-04-23T12:30:00
```

- **If the timeslottypeId id is null**



```
    UUID timeSlotTypeId = null;
```



```
2024-05-12T10:40:49.703+05:30  INFO 2004 --- [Patientsky-Appointment ] [           main] c.p.a.PatientskyAppointmentApplication   : S
2024-05-12T10:40:49.710+05:30  INFO 2004 --- [Patientsky-Appointment ] [           main] c.p.a.PatientskyAppointmentApplication   : N
2024-05-12T10:40:50.889+05:30  INFO 2004 --- [Patientsky-Appointment ] [           main] c.p.a.PatientskyAppointmentApplication   : S
Available time slot for the Calender ID: 48644c7a-975e-11e5-a090-c8e0eb18c1e9 is from 2019-04-23T12:15:00 to 2019-04-23T12:45:00
Available time slot for the Calender ID: 452dccfc-975e-11e5-bfa5-c8e0eb18c1e9 is from 2019-04-23T12:00:00 to 2019-04-23T12:30:00
```