**Meeting Room Booking REST API**

* **Langauage : Java , version 8**
* **Framework : JAX-RS,Jersey**
* **DataBase : Berkeley DB (In memory db)**
* **Server Used for Testing : Apache Tomcat 9**
* **Tool to make Requests : POSTMAN**
* **All API are async in nature to support concurrent requests**

GIT Repository URL : <https://github.com/Kangan06/Roombooking>

**Set UP :**

Import the war file in eclipse with all jars.

Set up tomcat

Or

Deploy the war on tomcat

**Data Structure :**

**MeetingRoom:**

**RoomId** : (Long) id for meeting room,

**Name** : (String) meeting room name,

**Type** : (int) type of room (4 or 8) i.e. 4 seater or 8 seater,

**Floor** : (int) which floor room is available,

**Book** :

**bookingId**: (Long) Refrence id for booking,

**roomId** : (Long) Id of room booked,

**starttime** : (Date) date time of start of booking,

**endtime** : (Date) date time of end of booking,

**employeeId** : (Int) id of employee who booked the room

**Employee** :

**employeeId** : (Int) id of employee,

**First\_name** : (String) first name of employee,

**Last\_name** : (String) last name of employee.

**Team**: (String) team of employee

API:

**Assuming the server is running on localhost:8080**

* **To initialize the sample data if want to**
  + GET request
    - <http://localhost:8080/TestApp/restservices/TestApp/createData>
    - This will create an initial data with 3 employee, 4 Rooms and 2 bookings
    - Rooms with following data:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Name | Type | Building | Floor |
| 1 | Room1 | 4 | A | 2 |
| 2 | Room2 | 4 | A | 3 |
| 3 | Room3 | 8 | B | 2 |
| 4 | Room4 | 8 | B | 3 |

* + - Employee with following data

|  |  |  |  |
| --- | --- | --- | --- |
| Employee Id | First name | Last name | Team |
| 1 | John | Smith | Database |
| 2 | Alex | Warner | Testing |
| 3 | David | Warner | Database |

* + - Bookings with following data:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| BookingId | Room Id | StartTime | Endtime | EmployeeId |
| 1 | 2 | 2020-06-22 15:00:00 | 2020-06-22 16:00:00", | 1 |
| 2 | 3 | 2020-06-23 15:00:00"," | 2020-06-23 16:00:00 | 1 |

* **To ADD a meeting room :**
  + <http://localhost:8080/TestApp/restservices/TestApp/addRoom>
  + POST request with Payload :
    - id: <id>
    - name: <name>
    - type:<type>
    - building:<building>
    - floor:<floor>
  + A screenshot of a cell phone

    Description automatically generated
* **To get rooms by name:**
  + <http://localhost:8080/TestApp/restservices/TestApp/getRoomByName>
  + POST request with payload :
    - name : <room name>

A screenshot of a cell phone

Description automatically generated

* **To get all rooms by type:**
  + <http://localhost:8080/TestApp/restservices/TestApp/getRoomByType>
  + POST request with Payload
    - type:<typeA screenshot of a cell phone

      Description automatically generated
* **To get all rooms by building**
  + <http://localhost:8080/TestApp/restservices/TestApp/getRoomByBuilding>
  + POST request with Payload
    - building : <building name>

A screenshot of a social media post

Description automatically generated

* **To get all rooms by floor**
  + <http://localhost:8080/TestApp/restservices/TestApp/getRoomByFloor>
  + POST request with payload
    - floor : <floor no>

A screenshot of a social media post

Description automatically generated

* **To Make a booking**
  + <http://localhost:8080/TestApp/restservices/TestApp/makeBooking>
  + POST request with payload
    - id : <booking id>
    - roomId : <room id>
    - starttime : <start date and time in format of 2020-06-22 15:00:00>
    - endtime : <end date and time in format of 2020-06-22 15:00:00>
    - employeeId : <id of employee making booking>

if we try to book an already booked room . Tt will give you message that room is already booked. For e.g :

**if we try to book room with id 2 from time 2020-06-22 15:00:00 to time 2020-06-22 16:00:00**

**booking will be unsuccessful as we have already made the same booking will initializing the data.**

A screenshot of a cell phone

Description automatically generated

For successful booking reference if will be returned;.

A screenshot of a cell phone

Description automatically generated

* **To cancel a booking**
  + <http://localhost:8080/TestApp/restservices/TestApp/cancelBooking>
  + POST request with payload
    - id : <booking id>

A screenshot of a cell phone

Description automatically generated

* **To get booking by id :**
  + <http://localhost:8080/TestApp/restservices/TestApp/getBooking>
  + POST request with Payload
    - id : <booking id>

A screenshot of a cell phone

Description automatically generated

* **To add employee** 
  + http://localhost:8080/TestApp/restservices/TestApp/addEmployee
    - POST request with payload
      * Id: <id>
      * FirstName:<first name>
      * LastName:<last name>
      * team:<team>

A screenshot of a cell phone

Description automatically generated