

Backend - Appointment Booking

1. POST /api/bookings/add

- **Description:** Add a new booking for a court. Ensures that double-booking for the same time slot is prevented.
- **Method:** POST
- **URL:** /api/bookings/add
- **Request Body:**
 - **courtId** (string, required) - The ID of the court to be booked.
 - **startTime** (string, required) - The start time of the booking in ISO 8601 format.
 - **customerName** (string, required) - The name of the customer booking the court.

Sample Request:

```
json
{
  "courtId": "651234567890abcdef123456",
  "startTime": "2024-10-12T10:00:00Z",
  "customerName": "John Doe"
}
```

Sample Response (Success):

```
json
{
  "message": "Booking created successfully",
  "booking": {
    "_id": "651234567890abcdef123471",
    "court": "651234567890abcdef123456",
    "startTime": "2024-10-12T10:00:00Z",
    "endTime": "2024-10-12T11:00:00Z",
    "customerName": "John Doe",
    "createdAt": "2024-10-10T08:00:00Z"
  }
}
```

Sample Response (Error - Double Booking):

```
json
{
  "error": "Court is already booked for this time slot."
}
```

Response Codes:

- **201 Created:** Booking was successfully created.
 - **400 Bad Request:** Missing or invalid input fields.
 - **409 Conflict:** Double booking attempt for the same court and time slot.
 - **500 Internal Server Error:** Error on the server-side while creating the booking.
-

2. GET /api/bookings/court/

- **Description:** Retrieve all bookings for a specific court by its ID.
- **Method:** GET
- **URL:** /api/bookings/court/:courtId
 - Replace :courtId with the actual ID of the court.

Sample Request:

```
GET /api/bookings/court/651234567890abcdef123456
```

Sample Response:

```
json
{
  "message": "Bookings retrieved successfully",
  "bookings": [
    {
      "_id": "651234567890abcdef123471",
      "startTime": "2024-10-12T10:00:00Z",
      "endTime": "2024-10-12T11:00:00Z",
      "customerName": "John Doe"
    },
    {
      "_id": "651234567890abcdef123472",
      "startTime": "2024-10-12T11:00:00Z",
```

```
        "endTime": "2024-10-12T12:00:00Z",
        "customerName": "Jane Smith"
    }
]
}
```

Response Codes:

- **200 OK:** Successfully retrieved all bookings for the court.
 - **404 Not Found:** The court with the specified ID was not found.
 - **500 Internal Server Error:** Error on the server-side while retrieving bookings.
-

3. GET /api/bookings/centers

- **Description:** Retrieve all centers with their sports and courts populated (excluding bookings).
- **Method:** GET
- **URL:** /api/bookings/centers

Sample Request:

```
GET /api/bookings/centers
```

Sample Response:

```
json
{
  "message": "All centers data retrieved successfully",
  "centers": [
    {
      "_id": "651234567890abcdef123458",
      "name": "Indiranagar Center",
      "location": "Indiranagar, Bangalore",
      "sports": [
        {
          "_id": "651234567890abcdef123457",
          "name": "Badminton",
          "courts": [
            {
```

```
        "_id": "651234567890abcdef123456",
        "name": "Badminton Court 1"
    },
    {
        "_id": "651234567890abcdef123459",
        "name": "Badminton Court 2"
    }
]
},
{
    "_id": "651234567890abcdef123461",
    "name": "Squash",
    "courts": [
        {
            "_id": "651234567890abcdef123460",
            "name": "Squash Court 1"
        },
        {
            "_id": "651234567890abcdef123461",
            "name": "Squash Court 2"
        }
    ]
}
]
},
{
    "_id": "651234567890abcdef123464",
    "name": "Koramangala Center",
    "location": "Koramangala, Bangalore",
    "sports": [
        {
            "_id": "651234567890abcdef123463",
            "name": "Tennis",
            "courts": [
                {
                    "_id": "651234567890abcdef123462",
                    "name": "Tennis Court 1"
                },
            ],
        }
    ]
}
```

```
{
  "centers": [
    {
      "sport": "Tennis",
      "court": "Tennis Court 1",
      "location": "Downtown",
      "status": "Available",
      "booking_id": "651234567890",
      "center_id": "651234567890abcdef123463",
      "name": "Tennis Court 1"
    },
    {
      "sport": "Tennis",
      "court": "Tennis Court 2",
      "location": "Downtown",
      "status": "Available",
      "booking_id": "651234567890",
      "center_id": "651234567890abcdef123463",
      "name": "Tennis Court 2"
    }
  ]
}
```

Response Codes:

- **200 OK:** Successfully retrieved all centers with sports and courts.
- **404 Not Found:** No centers found (if applicable).
- **500 Internal Server Error:** Error on the server-side while retrieving centers.

API Endpoints Summary:

| Endpoint | Method | Description |
|---|-------------------|---|
| <code>/api/bookings/add</code> | <code>POST</code> | Add a new booking (prevents double-booking) |
| <code>/api/bookings/court/:courtId</code> | <code>GET</code> | View all bookings for a specific court |
| <code>/api/bookings/centers</code> | <code>GET</code> | Retrieve all centers with their sports and courts |

Local Usage (EXAMPLE)

```
[POST]localhost:4000/api/bookings/add
[GET]localhost:4000/api/bookings/court/:courtId
[GET]localhost:4000/api/bookings/centers
```