

Name : Rizqullah Maziyah Isnaeni
NIM : 1202201389
Class : SI44INT

1. Add Hotel

```
# add hotel endpoint
@app.route('/hotels', methods=['POST'])
def add_hotel():
    data = request.get_json()
    name = data['name']
    city = data['city']
    rating = data['rating']
    price = data['price']
    query = "INSERT INTO hotels (name, city, rating, price) VALUES (%s, %s, %s, %s)"
    values = (name, city, rating, price)
    cursor.execute(query, values)
    db.commit()
    return jsonify({'message': 'Hotel added successfully'})
```

http://localhost:5000/hotels

Save



POST

http://localhost:5000/hotels

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings

Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON

Beautify

```
1  {
2    "name": "Margo Hotel",
3    "city": "Depok",
4    "rating": 4.4,
5    "price": 700000
6  }
```

Body Cookies Headers (5) Test Results

200 OK 29 ms 209 B Save Response

Pretty

Raw

Preview

Visualize

JSON



```
1  {
2    "message": "Hotel added successfully"
3  }
```

```
mysql> select * from hotels;
```

hotel_id	name	city	rating	price
1	Margo Hotel	Depok	4.4	700000.00
2	ASTON Tropicana	Bandung	4.6	750000.00
3	Grand Asrilia	Bandung	4.5	650000.00
4	Favehotel	Depok	4.4	395000.00

```
4 rows in set (0.00 sec)
```

The data will be input in the MySQL database

2. Get Hotel

```
# get hotels endpoint
@app.route('/hotels', methods=['GET'])
def get_hotels():
    query = "SELECT * FROM hotels ORDER BY city, rating;"
    cursor.execute(query)
    result = cursor.fetchall()
    hotels = []
    for hotel in result:
        hotel_data = {}
        hotel_data['name'] = hotel[1]
        hotel_data['city'] = hotel[2]
        hotel_data['price'] = hotel[4]
        hotel_data['rating'] = hotel[3]
        hotels.append(hotel_data)
    return jsonify({'hotels': hotels})
```

http://localhost:5000/hotels

Save

Send

GET

http://localhost:5000/hotels

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings

Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON

Beautify

Body Cookies Headers (5) Test Results

200 OK 16 ms 647 B Save Response

Pretty Raw Preview Visualize JSON

```
1
2  "hotels": [
3    {
4      "city": "Bandung",
5      "name": "Grand Asrilia",
6      "price": "650000.00",
7      "rating": 4.5
8    },
9    {
10     "city": "Bandung",
11     "name": "ASTON Tropicana",
12     "price": "750000.00",
13     "rating": 4.6
```

```

16         "city": "Depok",
17         "name": "Margo Hotel",
18         "price": "700000.00",
19         "rating": 4.4
20     },
21     {
22         "city": "Depok",
23         "name": "Favehotel",
24         "price": "395000.00",
25         "rating": 4.4
26     }
27 ]
28 }

```

3. Update Hotel

```

# update hotel endpoint
@app.route('/hotels/<int:hotel_id>', methods=['PUT'])
def update_hotel(hotel_id):
    data = request.get_json()
    name = data.get('name')
    city = data.get('city')
    rating = data.get('rating')
    price = data.get('price')
    query = "UPDATE hotels SET "
    sets = []
    values = []
    if name:
        sets.append("name = %s")
        values.append(name)
    if city:
        sets.append("city = %s")
        values.append(city)
    if rating:
        sets.append("rating = %s")
        values.append(rating)
    if price:
        sets.append("price = %s")
        values.append(price)
    query += ", ".join(sets)
    query += " WHERE hotel_id = %s"
    values.append(hotel_id)
    cursor.execute(query, values)
    db.commit()
    return jsonify({'message': 'Hotel updated successfully'})

```

PUT ▼ http://127.0.0.1:5000/hotels/1 Send ▼

Params Authorization Headers (8) **Body** ● Pre-request Script Tests Settings Cookies

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL **JSON** ▼ Beautify

```

1 {
2   ... "rating": 4.6
3 }

```

Body Cookies Headers (5) Test Results 200 OK 104 ms 211 B Save Response ▼

Pretty Raw **Preview** Visualize

```
{ "message": "Hotel updated successfully" }
```

hotel_id	name	city	rating	price
1	Margo Hotel	Depok	4.6	700000.00
2	ASTON Tropicana	Bandung	4.6	750000.00
3	Grand Asrilia	Bandung	4.5	650000.00
4	Favehotel	Depok	4.4	395000.00

4 rows in set (0.00 sec)

Change the rating of Margo Hotel from 4.4 to 4.6

4. Delete Hotel

```
# delete hotel endpoint
@app.route('/hotels/<int:hotel_id>', methods=['DELETE'])
def delete_hotel(hotel_id):
    query = "DELETE FROM hotels WHERE hotel_id = %s"
    values = (hotel_id,)
    cursor.execute(query, values)
    db.commit()
    return jsonify({'message': 'Hotel deleted successfully'})
```

DELETE ▼ http://127.0.0.1:5000/hotels/4 Send ▼

Params Authorization Headers (8) **Body** ● Pre-request Script Tests Settings Cookies

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL **JSON** ▼ Beautify

1

Body Cookies Headers (5) Test Results 200 OK 31 ms 211 B Save Response ▼

Pretty Raw **Preview** Visualize

```
{ "message": "Hotel deleted successfully" }
```

hotel_id	name	city	rating	price
1	Margo Hotel	Depok	4.6	700000.00
2	ASTON Tropicana	Bandung	4.6	750000.00
3	Grand Asrilia	Bandung	4.5	650000.00

3 rows in set (0.00 sec)

5. Add Booking

```
@app.route('/booking', methods=['POST'])
def add_booking():
    data = request.get_json()
    hotel_id = data['hotel_id']
    name = data['name']
    email = data['email']
    phone = data['phone']
    checkin_date = data['checkin_date']
    checkout_date = data['checkout_date']
    query = "INSERT INTO bookings (hotel_id, name, email, phone, checkin_date, checkout_date) VALUES (%s, %s, %s, %s, %s, %s)"
    values = (hotel_id, name, email, phone, checkin_date, checkout_date)
    cursor.execute(query, values)
    db.commit()
    booking_id = cursor.lastrowid
    return jsonify({'message': f'Booking added successfully with booking id: {booking_id}'})
```

POST http://127.0.0.1:5000/bookings Send

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings Cookies

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL **JSON** Beautify

```
1 {
2   "hotel_name": "Grand Asrilia",
3   "city": "Bandung",
4   "name": "Pertiwi Adiwarna",
5   "email": "pertiwi@gmail.com",
6   "phone": "08xxxxxxx",
7   "checkin_date": "2023-05-01",
8   "checkout_date": "2023-05-05"
9 }
```

Body Cookies Headers (5) Test Results 200 OK 57 ms 230 B Save Response

Pretty Raw Preview Visualize

```
{ "message": "Booking added successfully with booking id: 1" }
```

booking_id	hotel_id	name	email	phone	checkin_date	checkout_date
1	3	Pertiwi Adiwarna	pertiwi@gmail.com	08xxxxxxx	2023-05-01	2023-05-05

1 row in set (0.00 sec)

6. Get Booking

```
# get booking API
@app.route('/bookings/<int:booking_id>', methods=['GET'])
def get_booking(booking_id):
    query = "SELECT * FROM bookings WHERE booking_id = %s"
    values = (booking_id,)
    cursor.execute(query, values)
    result = cursor.fetchone()
    if not result:
        return jsonify({'message': 'Booking not found'})
    booking = {}
    booking['booking_id'] = result[0]
    booking['hotel_id'] = result[1]
    booking['name'] = result[2]
    booking['email'] = result[3]
    booking['phone'] = result[4]
    booking['checkin_date'] = result[5].strftime('%Y-%m-%d')
    booking['checkout_date'] = result[6].strftime('%Y-%m-%d')
    return jsonify({'booking': booking})
```

GET ▼ http://127.0.0.1:5000/bookings/1 Send ▼

Params Authorization Headers (8) **Body** ● Pre-request Script Tests Settings Cookies

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL **JSON** ▼ Beautify

1 5

Body Cookies Headers (5) Test Results 200 OK 29 ms 390 B Save Response ▼

Pretty Raw Preview Visualize **JSON** ▼ ≡

```
1  {
2    "booking": {
3      "booking_id": 1,
4      "checkin_date": "2023-05-01",
5      "checkout_date": "2023-05-05",
6      "email": "pertiwi@gmail.com",
7      "hotel_id": 3,
8      "name": "Pertiwi Adiwarna",
9      "phone": "08xxxxxxxx"
10   }
11 }
```