



HYDERABAD INSTITUTE OF ARTS, SCIENCE, AND TECHNOLOGY

Database Systems – Lab 12 Manual

Instructor: Miss Ayesha Eman

Date: 30/09/2025

Topic: Working with JSON Data in SQL for Modern Applications

1. Introduction

JSON (JavaScript Object Notation) is a lightweight, human-readable data format used to represent structured and semi-structured data. It is widely used in APIs, web applications, IoT systems, and NoSQL databases. JSON supports key-value pairs, arrays, and nested objects, making it flexible for storing dynamic data.

Modern relational databases such as MySQL, PostgreSQL, and SQL Server now include native JSON support, enabling developers to combine the reliability of relational databases with the flexibility of semi-structured storage.

2. Learning Outcomes

By the end of this lab, students will be able to:

- Understand how relational databases handle JSON data.
- Perform SQL queries on JSON fields including extraction and filtering.
- Model real-world scenarios like IoT, user profiles, and e-commerce with JSON in SQL.
- Apply advanced SQL operations to semi-structured data.

3. Creating Tables with JSON Columns

Example: Creating a table with a JSON column for user profiles.

```
CREATE TABLE users (  
  id INT AUTO_INCREMENT PRIMARY KEY,  
  name VARCHAR(50),
```

```
    profile JSON
);
```

4. Inserting JSON Data

```
INSERT INTO users (name, profile) VALUES
('Zohaib', '{"age": 22, "city": "Karachi", "skills": ["SQL", "Python", "Machine Learning"],
"preferences": {"theme": "dark", "language": "English"}}'),
('Bilal', '{"age": 23, "city": "Lahore", "skills": ["Java", "Spring", "Docker"], "preferences":
{"theme": "light", "language": "Urdu"}}');
```

5. Querying JSON Data

Examples of querying JSON fields:

1. Extract a nested field:

```
SELECT name, JSON_EXTRACT(profile, '$.city') AS city FROM users;
```

2. Extract from arrays:

```
SELECT name, JSON_EXTRACT(profile, '$.skills[0]') AS first_skill FROM users;
```

3. Filter using JSON keys:

```
SELECT name FROM users WHERE JSON_EXTRACT(profile, '$.preferences.theme') = 'dark';
```

6. Updating JSON Data

Example of updating JSON data:

```
UPDATE users
SET profile = JSON_SET(profile, '$.skills[3]', 'Kubernetes')
WHERE name = 'Bilal';
```

7. Industry Use Case – IoT Logs

```
CREATE TABLE iot_logs (
    id INT AUTO_INCREMENT PRIMARY KEY,
    device_id VARCHAR(20),
    data JSON
);
```

```
INSERT INTO iot_logs (device_id, data) VALUES
('dev101', '{"temp": 29.5, "humidity": 60, "status": "active"}'),
('dev102', '{"temp": 34.2, "humidity": 45, "status": "inactive"}');
```

```
SELECT device_id, JSON_EXTRACT(data, '$.temp') AS temperature
FROM iot_logs
WHERE JSON_EXTRACT(data, '$.status') = 'active';
```

8. Lab Tasks for Students

Task 1: Create a table 'orders' with a JSON column to store product details and buyer info.

Task 2: Insert at least 3 sample orders with different buyers and multiple items.

Task 3: Write SQL queries to:

- Display all buyers from Karachi.
- Show all items from an order where price > 5000.
- Extract the second item from each order.

Task 4: Update one order JSON to add a new item.

Task 5: Query orders where buyer city = 'Lahore'.

9. Conclusion

This lab introduces modern SQL features with JSON data, bridging structured and semi-structured storage. Students gain hands-on practice relevant to real-world applications in IoT, e-commerce, and AI systems. By completing these tasks, students acquire skills directly applicable to industry projects and advanced database systems.