SQL and REST

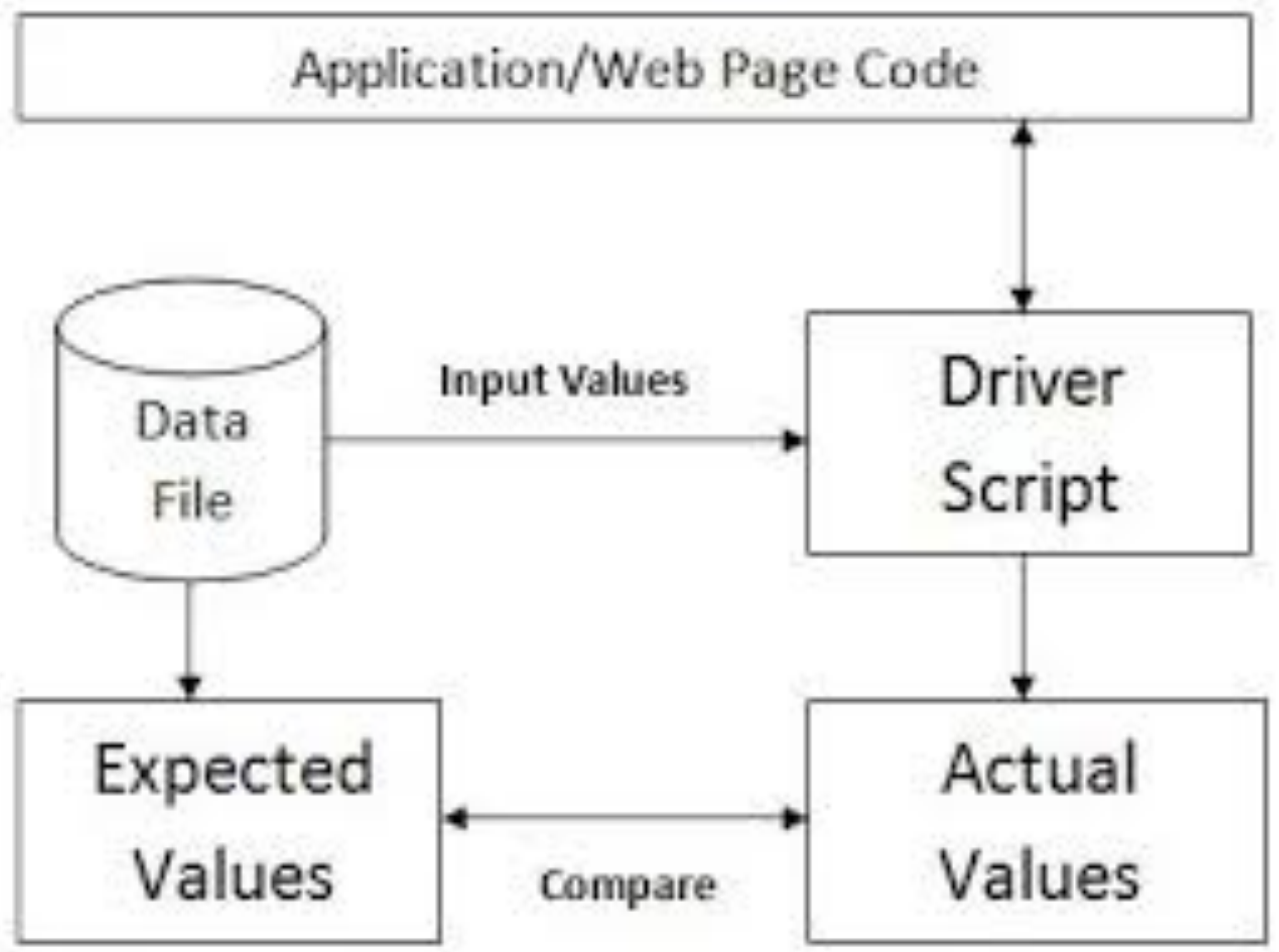
And Everything in between

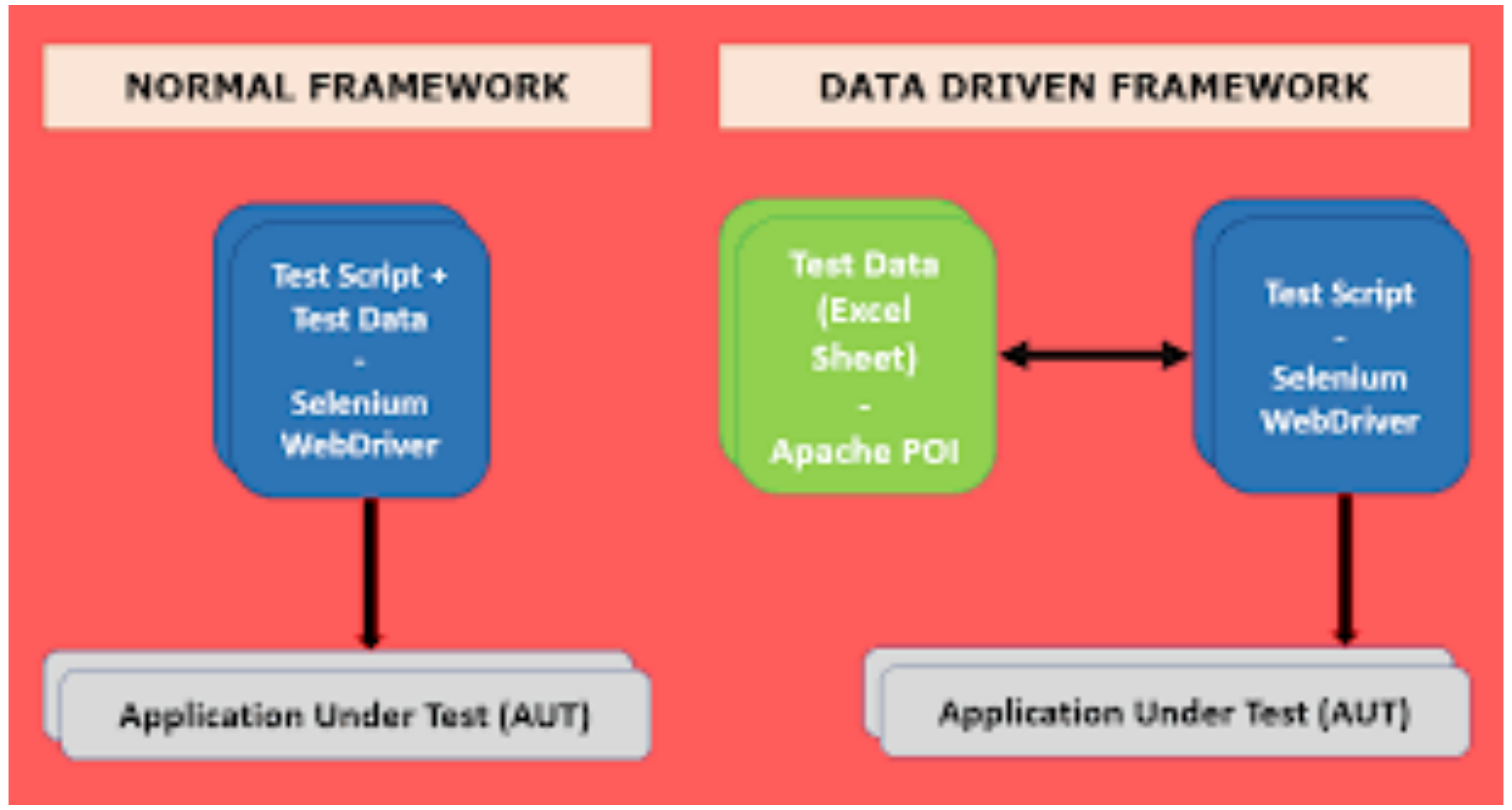
DATA DRIVEN TESTING

WHEN: Whenever a functionality or a module in an app requires testing with multiple sets of data(Parametrization), Multiple inputs then we need to perform data driven testing and automation. These scenarios are one of the things That must be automated.

HOW: Test data is separated from code and stored into external sources: Cucumber Examples table, Excel files, CSV files, Database.

BENEFIT: More organized, Data centralized, Collaboration on test data - it can come from BA, MTs etc




NORMAL FRAMEWORK
DATA DRIVEN FRAMEWORK
Test Script + Test Data - Selenium WebDriver
Test Data (Excel Sheet) - Apache POI
Test Script - Selenium WebDriver
Application Under Test (AUT)
Application Under Test (AUT)

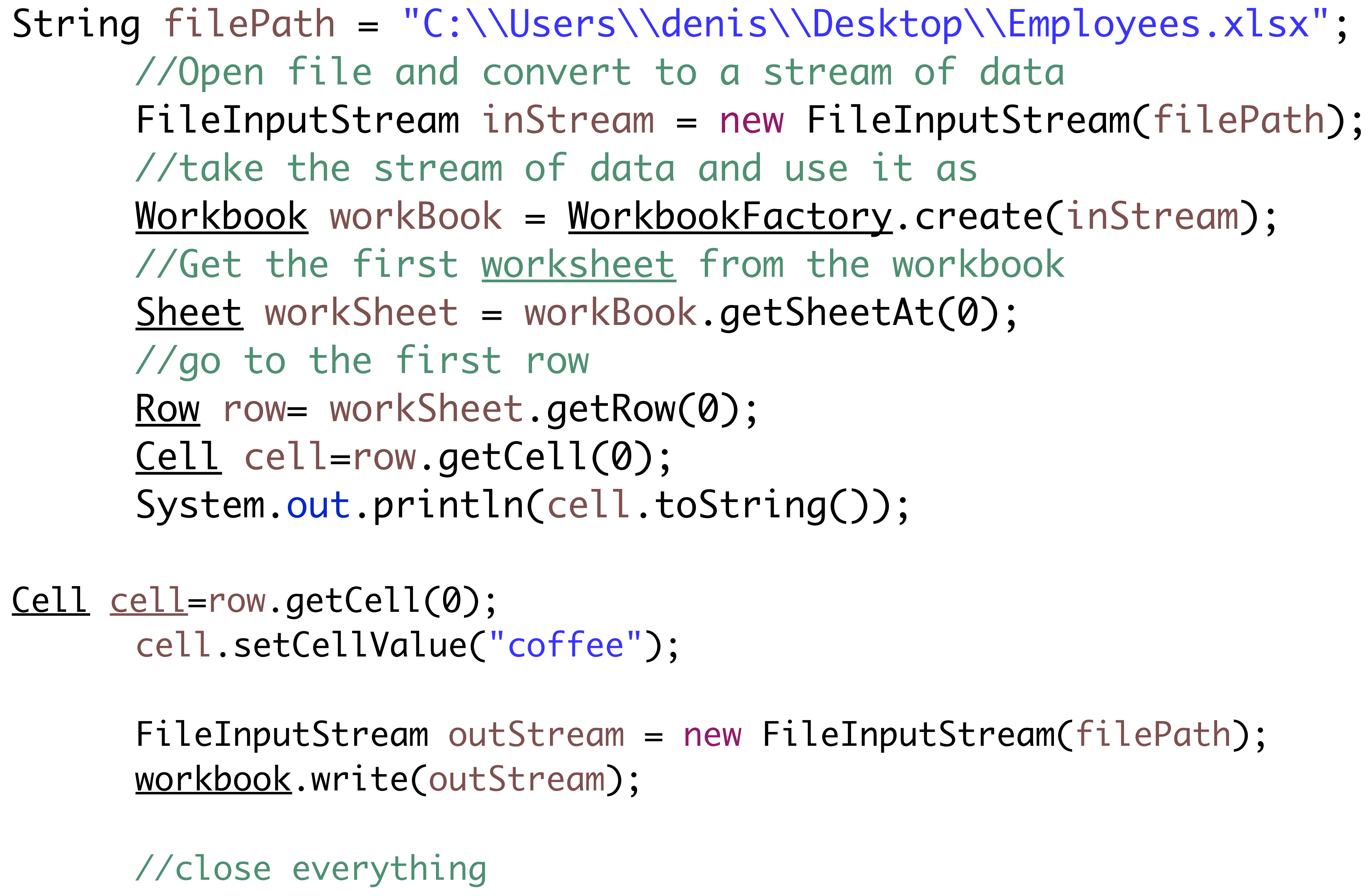
READ DATA FROM EXCEL

- I use Apache POI to read data from excel or any excel related tests.

- FILEINPUTSTREAM > WORKBOOK > WORKSHEET > ROW> CELL

- FILEOUTPUTSTREAM > WORKBOOK.WRITE(OUTSTREAM);

-



```
String filePath = "C:\\Users\\denis\\Desktop\\Employees.xlsx";
        //Open file and convert to a stream of data
        FileInputStream inStream = new FileInputStream(filePath);
        //take the stream of data and use it as
        Workbook workBook = WorkbookFactory.create(inStream);
        //Get the first worksheet from the workbook
        Sheet workSheet = workBook.getSheetAt(0);
        //go to the first row
        Row row= workSheet.getRow(0);
        Cell cell=row.getCell(0);
        System.out.println(cell.toString());

Cell cell=row.getCell(0);
        cell.setCellValue("coffee");

        FileInputStream outStream = new FileInputStream(filePath);
        workbook.write(outStream);

        //close everything
```

WORKBOOK > XSSFWORKBOOK -> XLSX FILES
> HSSFWORKBOOK -> XLS FILES

READ DATA FROM DATABASE

JDBC/JAVA.SQL

DRIVER

CONNECTION > STATEMENT > RESULTSET

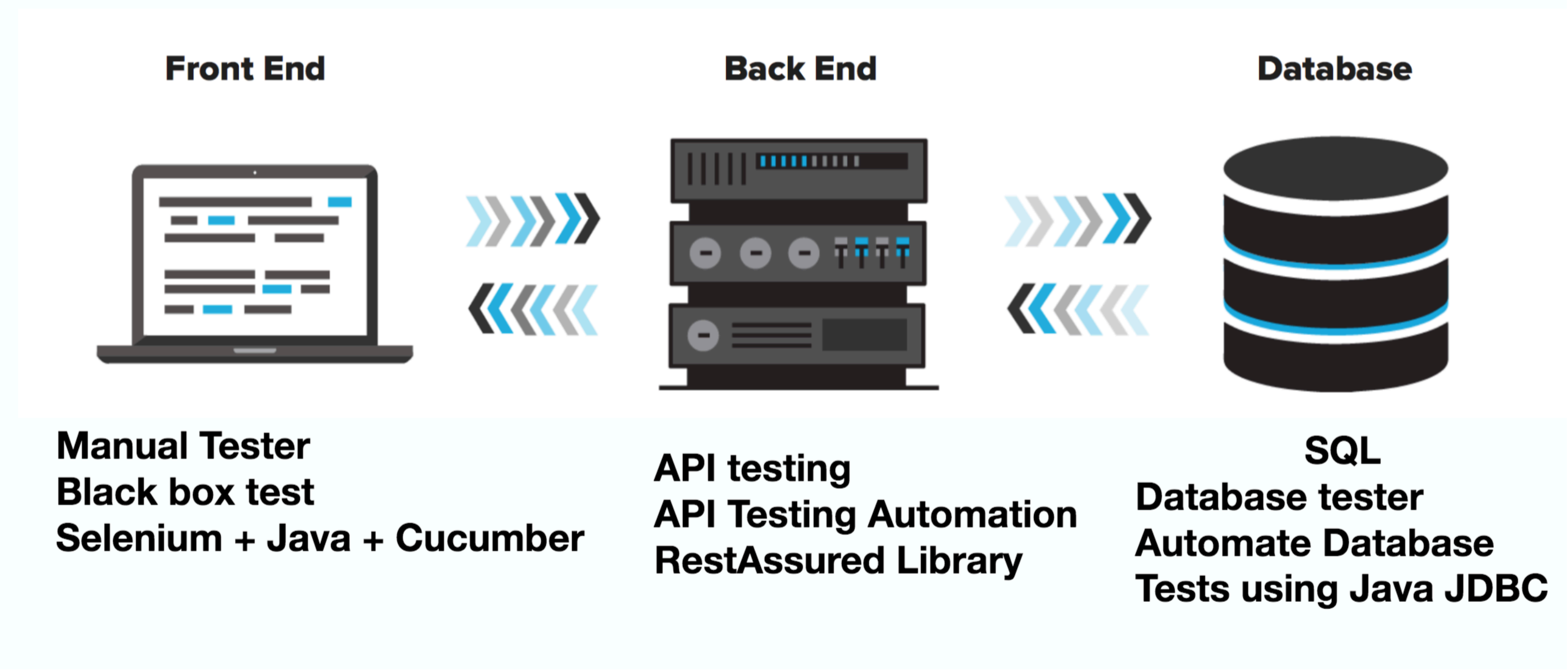
```java
    Connection connection=DriverManager.getConnection(oracleDbUrl, oracleDbUsername, oracleDbPassword);
    //Statement statement=connection.createStatement();
    Statement statement=connection.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_READ_ONLY);
    ResultSet resultSet = statement.executeQuery("select * from countries");

//  while(resultSet.next()) {
//      System.out.println(resultSet.getString(1)+"-"+resultSet.getString("country_name")+"-"+resultSet.getInt("region_id"));
//  }
    //find out how many records in the resultset
    resultSet.last();
    int rowsCount = resultSet.getRow();
    System.out.println("Number of rows:" + rowsCount);

    resultSet.first();
    while(resultSet.next()) {
        System.out.println(resultSet.getString(1)+"-"+resultSet.getString("country_name")+"-"+resultSet.getInt("region_id"));
    }

    resultSet.close();
    statement.close();
    connection.close();
```

Test data is separated from code and stored into external sources: Cucumber Examples table, Excel files, CSV files, Database.

If the amount of data is not that huge, then I use Cucumber Scenario outline with Examples table.

And other times I maintain test data in Excel files, and i use Apacha POI library to read and write data us

If data comes from a database, or I need to do database validation,
I use SQL queries along with JDBC library in java.



Manual Tester
Black box test
Selenium + Java + Cucumber

API testing
API Testing Automation
RestAssured Library

SQL
Database tester
Automate Database
Tests using Java JDBC

I AM EXPERT ON FRONT END TEST AUTOMATION USING JAVA+SELENIUM WEBDRIVER, CUCUMBER, MAVEN, TESTNG, JUNIT
ALSO I AM GOOD AT RESTFUL API TEST AUTOMATION USING POSTMAN, RESTASSURED LIBRARY IN JAVA
I HAVE RICH experience with Database test test automation using
SQL queries along with JDBC library in Java.

Where do we close connection with database in try catch block, in case we will get Exception thrown.
"in catch block" answer didn't satisfy the interviewer.

1) in finally block.
2) I use try-with-resources and it will automatically close

```
try{
    Connection connection=DriverManager.getConnection(url, user, pwd);
    Statement statement=connection.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_READ_ONLY);
    ResultSet resultSet = statement.executeQuery("select * from countries");

}catch(SQLException e) {
    //some code to report the issue
    e.printStackTrace();
}finally{
    try{
        resultSet.close();
        statement.close();
        connection.close();
    }catch(Exception e) {

    }
}
```

2) TRY-WITH-RESOURCES:
Any class or iterface that extends Closable or AutoClosable interfaces
can ONLY be put into try-with-resources block.

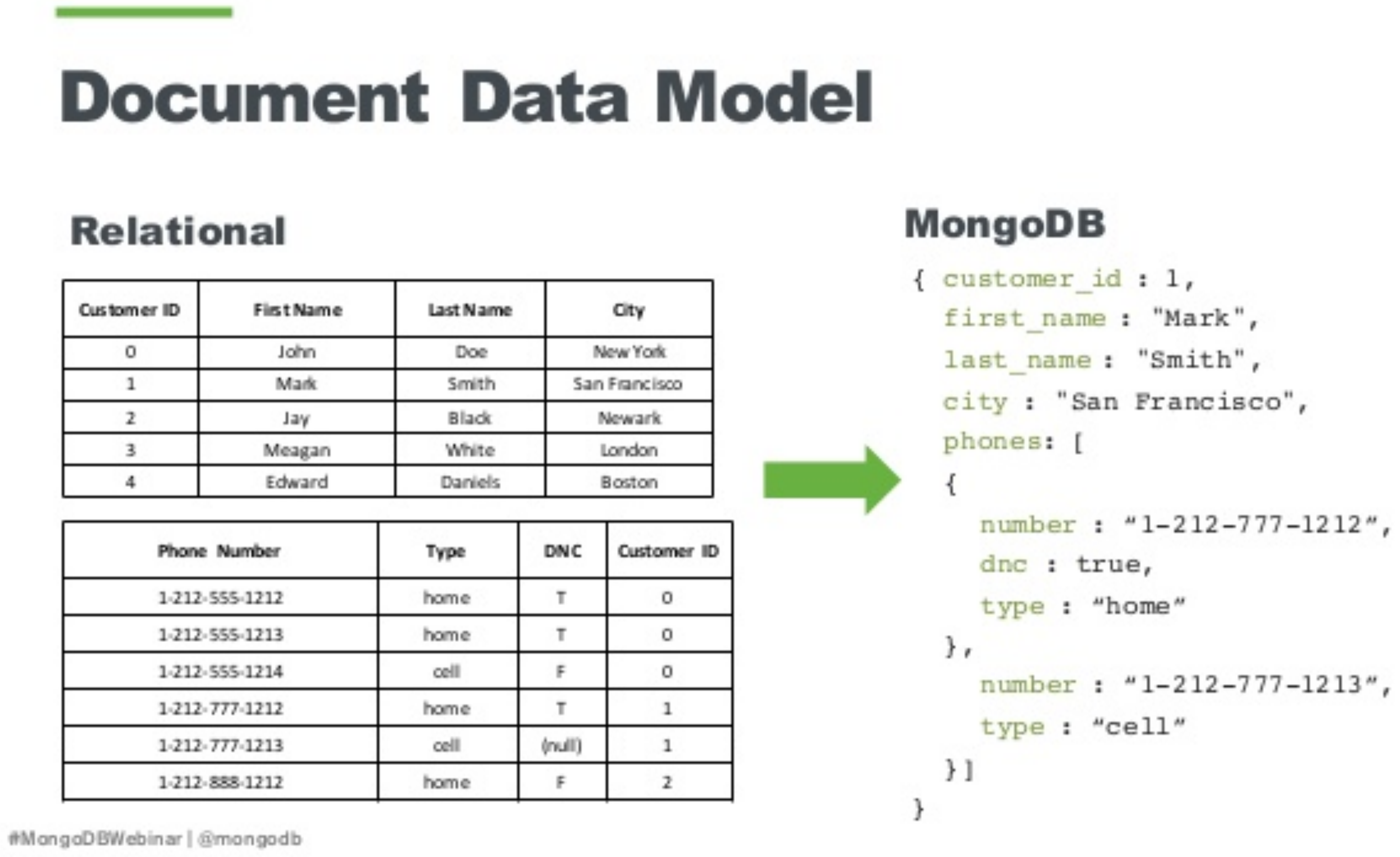
```
try(
    Connection connection=DriverManager.getConnection(url, user, pwd);
    Statement statement=connection.createStatement(ResultSet.TYPE_SCROLL_INSENSITIVE,ResultSet.CONCUR_READ_ONLY);
    ResultSet resultSet = statement.executeQuery("select * from countries"); )
{
    //some code to read data from resultSet
}

}catch(SQLException e) {
    //some code to report the issue
    e.printStackTrace();
}
```

Have you done any backend/database testing?

Yes, I have lots of experience with working with databases.
And I am very comfortable with writing SQL queries.
I have experience with working on Relational Databases like Oracle, MySQl, SQL Server.

Have you worked with non-relational databases?
I don’t have hands on experience but I know that it is like JSON format database and
I have good experience with working with JSON files.
And I am a quick learner

RELATIONAL VS NON-RELATIONAL DATABASE STRUCTURE



Document Data Model

Relational

Customer ID	First Name	Last Name	City
0	John	Doe	New York
1	Mark	Smith	San Francisco
2	Jay	Black	Newark
3	Meagan	White	London
4	Edward	Daniels	Boston

Phone Number	Type	DNC	Customer ID
1-212-555-1212	home	T	0
1-212-555-1213	home	T	0
1-212-555-1214	cell	F	0
1-212-777-1212	home	T	1
1-212-777-1213	cell	(null)	1
1-212-888-1212	home	F	2

MongoDB

```
{ customer_id : 1,
  first_name : "Mark",
  last_name : "Smith",
  city : "San Francisco",
  phones: [
  {
    number : “1-212-777-1212”,
    dnc : true,
    type : “home”
  },
    number : “1-212-777-1213”,
    type : “cell”
  }]
}
```

#MongoDBWebinar | @mongodb

How do you create a table in SQL? What are constraints?

```
CREATE TABLE Computers (
   COMPUTER_ID Number PRIMARY KEY,
   BRAND VARCHAR2(20) NOT NULL,
   TYPE CHAR(1) NULL
);
```

SQL COLUMN CONSTRAINTS:
Primary key, Foreign key, Null, not Null, Unique

DESCRIBE Computers; —> To display table information. Metadata of table

INSERT DATA INTO TABLE

```
INSERT INTO Computers VALUES (123, 'HP', 'L');

SELECT * from Computers;
```

SQL TECHICAL INTERVIEW

Table 1

1	Write a SQL query to find how many records are there in Town table where “TOWN_NAME” has “ell” in them?
2	Write a SQL query to find how many occurrences of each “COUNTRY_CODE” are there in the Town table?
3	Write a SQL query to find the “TOWN_NAME” where there are non-alphanumeric characters?
4	Write a SQL query to select the first 5 records from Town table?

1) COUNT GROUP FUNCTION, WHERE CONDITION, LIKE OPERATOR:

```
SELECT COUNT(*) FROM TOWN WHERE TOWN_NAME LIKE ‘%ell%’;
```

2) COUNT FUNCTION, GROUP BY KEYWORD

```
SELECT COUNTRY_CODE, COUNT(*) FROM TOWN GROUP BY COUNTRY_CODE;
```

3) WHERE CONDITION, LIKE OPERATOR TO FIND NON-ALPHA NUMERICS. LIKE operator with REGULAR EXPRESSION PATTERN. REGEX is used for matching some formats. Like 10 numbers etc

```
SELECT * FROM Computers
WHERE BRAND NOT LIKE ‘%[^a-zA-Z0-9]%';
```

4) WHERE condition with ROWNUM

```
SELECT * FROM TOWN WHERE ROWNUM <= 5;
```

INNER JOIN

WHENEVER WE NEED TO QIUERY DATA FROM MULTIPLE TABLES WE NEED JOINS. AND WHEN WE NEED TO FIND MATCHING RECORDS - WE USE INNER JOINS.

INNER JOIN IS USED WHEN WE NEED TO DISPLAY DATA FROM MULTIPLE TABLES AND IT WILL RETURN MATCHING RECORDS.

1.Write a SQL query to find the salary for the people working in the QA department and the HR Department?

```
SELECT e.EMPID, NAME, SALARY, DEPTID
FROM Employee e JOIN Department d
ON e.empid = d.empid
WHERE DEPTID IN (‘QA’,’HR’);
```

How can you find employee's manager's first and last name?

SELF JOIN

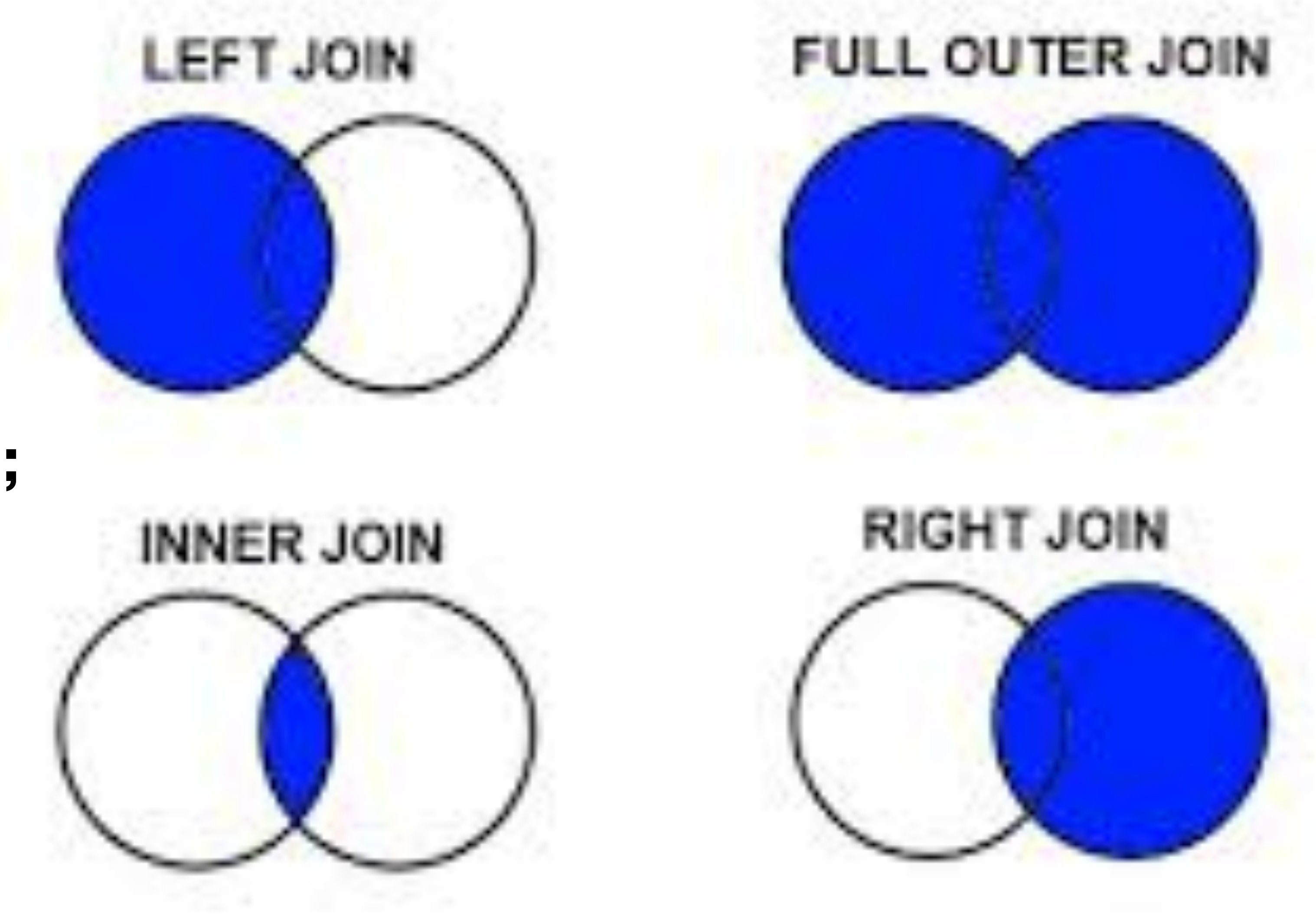
```
SELECT EMP.EMPLOYEE_ID, EMP.FIRST_NAME, MAN.FIRST_NAME, MAN.LAST_NAME
FROM EMPLOYEES EMP JOIN EMPLOYEES MAN
ON EMP.MANAGER_ID = MAN.EMPLOYEE_ID;
```

INNER JOIN VS OUTER JOIN

We need joins whenever we need to query data from multiple tables:

1) inner join returns matching records from both tables According to joining condition
2) outer join returns matching plus non matching data either From right or left tables

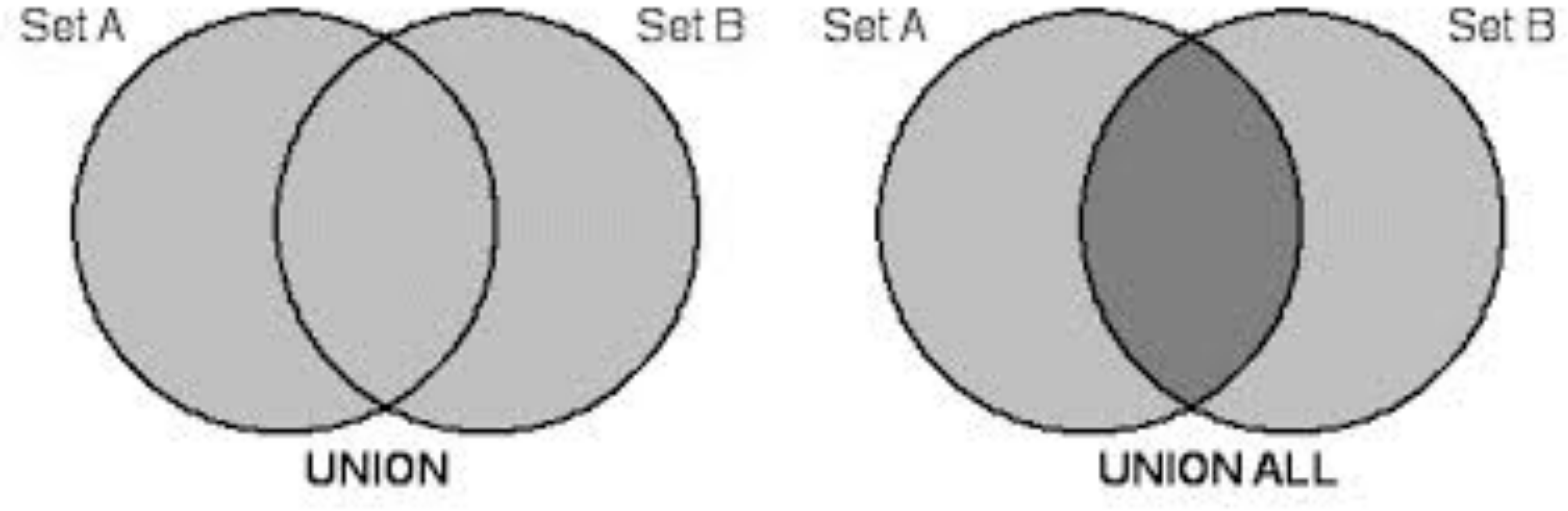
```
SELECT LAST_NAME, DEPARTMENT_NAME
FROM DEPARTMENTS RIGHT OUTER JOIN EMPLOYEES
ON EMPLOYEES.DEPARTMENT_ID = DEPARTMENTS.DEPARTMENT_ID;
```



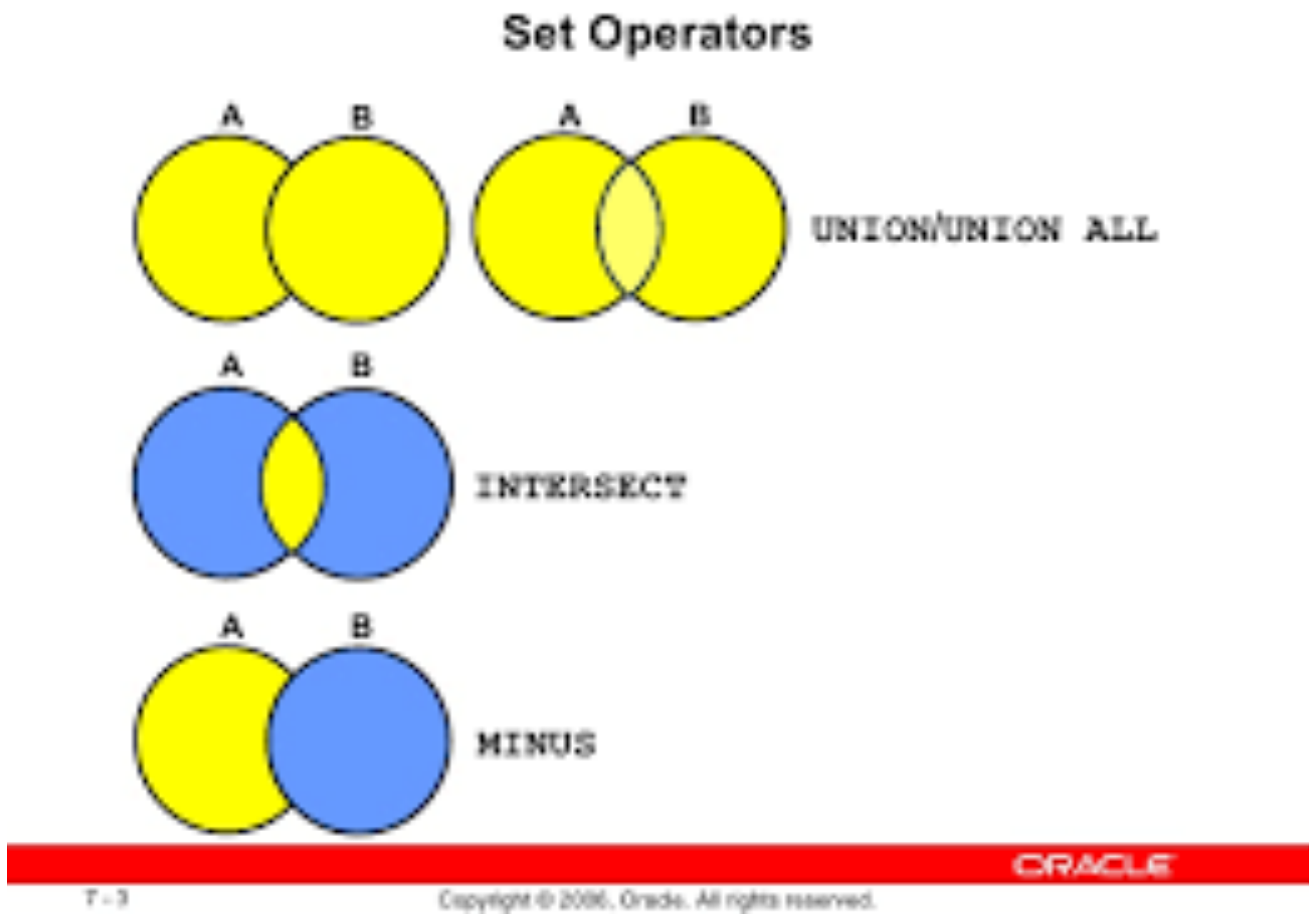
UNION VS UNION ALL

WE USE THESE SET OPERATORS WHEN WE NEED TO DISPLAY DATA FROM 2 QUERIES.

UNION: REMOVES DUPLICATES AND SORTS THE RESULT
UNION ALL: DOES NOT REMOVE DUPLICATES AND DOES NOT SORT RESULTS



INTERSECT VS MINUS



INTERSECT DISPLAYS COMMON DATA IN BOTH QUERIES. REMOVES DUPLICATES and multiple NULLS , SORTS RESULTS

MINUS DISPLAYS DATA minus returns records from first query that is not present in second query.

JOIN VS UNION?

JOINS are used to query data from multiple tables
UNION is used to query data from multiple queries

GROUP BY VS ORDER BY

- GROUP BY is used whenever we work with group functions, it will help to create sub groups within a group

- ORDER BY is used to sort the data in either desc or asc order

HAVING VS WHERE

- HAVING is used whenever our condition includes a GROUP function HAVING MAX(salary) > 9000;

- WHERE is used to filter the results and when we do not use GROUP function

- WHERE employee_id = 234;

DISPLAY 5TH ROW FROM TABLE

- SELECT * from (SELECT EMPLOYEE_ID,FIRST_NAME, ROWNUM AS RN FROM EMPLOYEES) WHERE RN = 5;
- (select * from employees where rownum <=10)
- minus
- (select * from employees where rownum <=9);
- We need to use a correlated subquery.
- Inner query will get columns along with ROW NUMBER. And outer query will look for specific row number

5TH LARGEST SALARY

- select salary from employees e1 where 5= (select count(salary)from employees e2 where e1.salary<=e2.salary);

- //FIX ME

DROP VS TRUNCATE

- BOTH ARE DDL COMMANDS AND CANNOT BE UNDONE
- DROP WILL REMOVE DATA AND TABLE TOGETHER
- TRUNCATE WILL REMOVE ALL DATA BUT NOT TABLE STRUCTURE
- DROP EMPLOYEES;
- TRUNCATE EMPLOYEES;

COMMIT COMMAND

- WHENEVER WE MAKE CHANGES TO DATABASE , WE CAN COMMIT TO SAVE THE CHANGES

DO U KNOW SQL?

- Yes, I am very comfortable with writing SQL Queries and DDL and DML commands. Currently working with Oracle database that is running in AMAZON CLOUD SERVER.

- DDL (Data definition language) : CREATE , ALTER, DROP, TRUNCATE..

- DML(Data manipulation language): SELECT, DELETE, INSERT, UPDATE

What kind of Database testing are you doing?

- I am mostly doing Database validations.

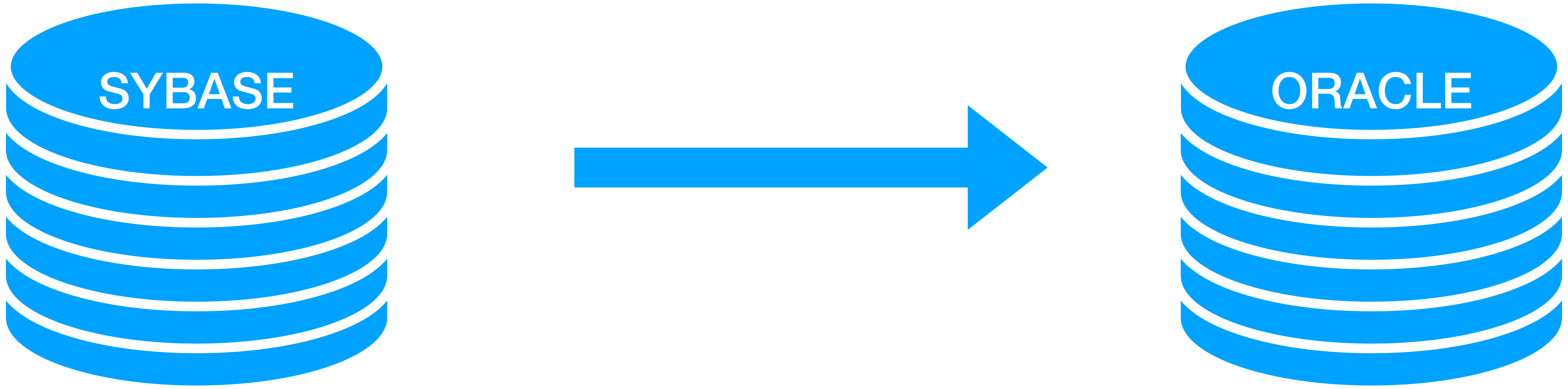
- I make changes or insert data(create loan) in the front end and validate in the database. Data in front end matches the database

- I also make changes using RESTapi and verify that changes are successful in Database as well.

- I also support Database migration process. My code connects to Sybase (legacy database) using JDBC then Connects to Oracle(NEW DB) then compare records to make sure data was migrated successfully

SQL + JDBC —>
JAVA DATA STRUCTURE/LIST OF MAPS

SQL + JDBC —>
JAVA DATA STRUCTURE -> LIST OF MAPS



ASSERT 2 DATA STRUCTURES ARE EQUAL

BATCH JOBS OF APPLICATION

- Some automated code that runs every night time and make some status changes to some data

- COUPON that is valid for 7 days. Batch Job is scanning through each coupon and if coupon hit 7 days it will update the status in database as invalid.

RESTful API

DO YOU HAVE EXPERIENCE WITH RESTFUL API TESTING?

WHERE IN YOUR PROJECT YOU HAVE REST API?
WHY DEVELOPED A REST API?
-> Application in the current project needs to be integrated to other internal and external applications. For the integration our team developed RESTful API , So I am testing it.

PROJECT/APP A ----> PROJECT/APP B IN SAME COMPANY
INTEGRATION WITH REST API

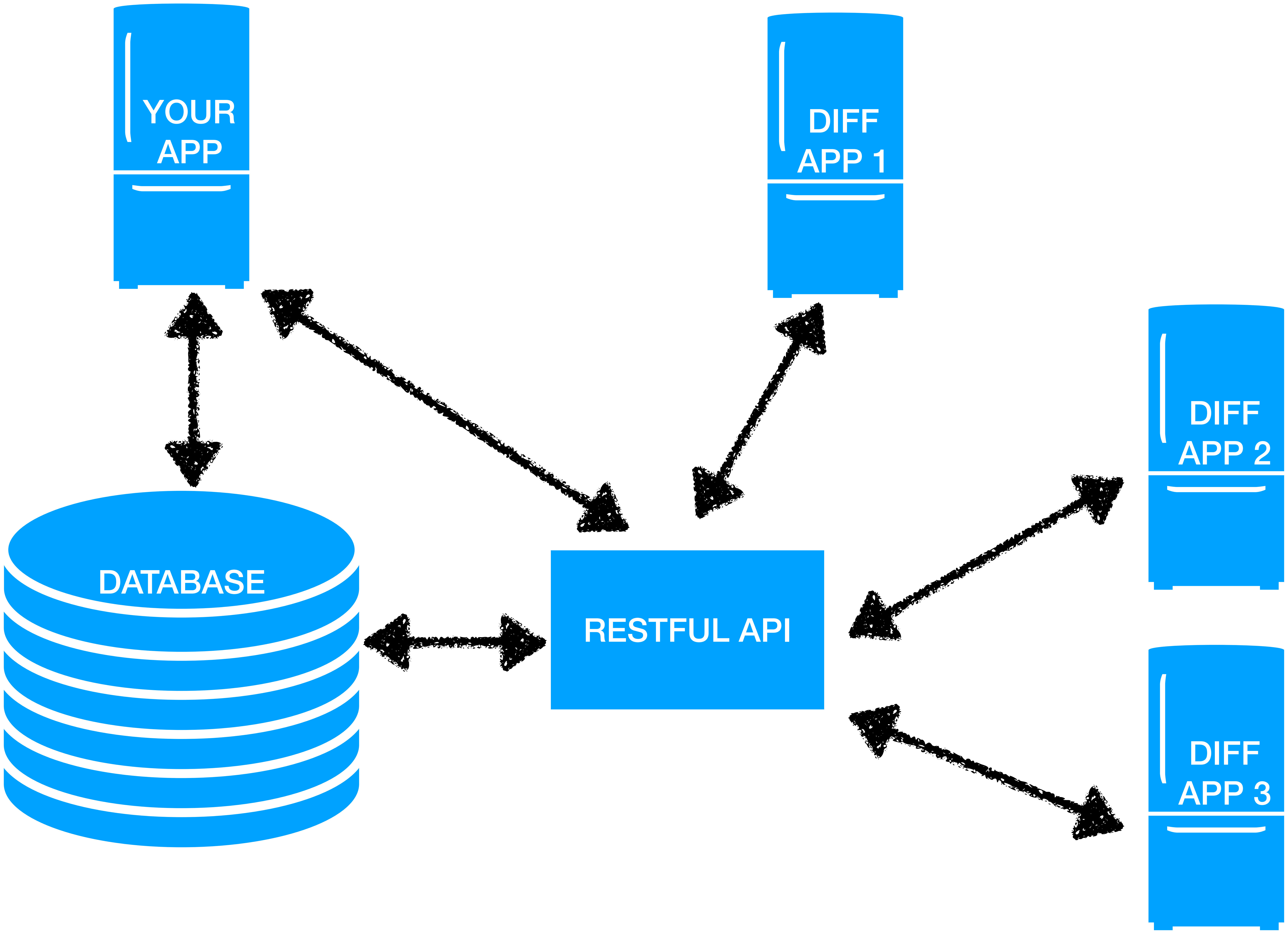
APP A ---> EXTERNAL APPLICATION
INTEGRATION USING REST API
=================================

REST API TESTING:
How do you test rest api?

I verify if each REST API endpoint is working as expected.

I use POSTMAN for manual API testing and use RESTASSURED library in Java for automation.
I send POST,PUT,GET, DETELE type of requests and verify response status code and response body, header.
I also do positive and negative testing of API.
When I do positive testing, I send valid request parameters , valid headers, valid request json body and verify that response status code is 200 successful and Json response body data is also matching the expected.

When I do negative testing, I send invalid request parameters , or invalid headers, or invalid request json body and verify that response status code is not 200 and Json response body contains error message.
=========================


YOUR APP
DIFF APP 1
DIFF APP 2
DATABASE
RESTFUL API
DIFF APP 3

HTTP METHODS/REQUEST TYPES

- GET -> READ , POST -> CREATE, PUT -> UPDATE, DELETE -> DELETE
- POST VS PUT

STATUS CODE

- **2×× Success**

- 200 OK
- 201 Created
- 202 Accepted
- 203 Non-authoritative Information
- 204 No Content

- **4×× Client Error**

- 400 Bad Request
- 401 Unauthorized
- 402 Payment Required
- 403 Forbidden
- 404 Not Found
- 405 Method Not Allowed
- 406 Not Acceptable

- **5×× Server Error**

- 500 Internal Server Error
- 501 Not Implemented
- 502 Bad Gateway
- 503 Service Unavailable
-

HEADERS

- ACCEPT, CONTENT-TYPE —> APPLICATION/JSON, APPLICATION/XML

PARAMETERS

- 2 TYPES:
- PATH PARAMETER(VALUE WILL BE PART OF URL)
- QUERY/REQUEST PARAMETERS (KEY+ VALUE FORMAT)

VALIDATE/CHECK/ASSERT JSON BODY

```
{
    "employee_id": 100,
    "first_name": "Steven",
    "last_name": "King",
    "email": "SKING",
    "phone_number": "515.123.4567",
    "hire_date": "2003-06-17T04:00:00Z",
    "job_id": "AD_PRES",
    "salary": 24000,
    "commission_pct": null,
    "manager_id": null,
    "department_id": 90
}
```

```
1) WHEN().GET(URI).THEN().BODY(“first_name”,EQUALTO(“Steven”));
We used hamcrest matcher that comes with RestAssured library

2)Using JSONPATH:
JsonPath json = WHEN().GET(URI).THENreturn().BODY().JsonPath();
Assert.assertEquals(json.getString(“last_name”),”King”);

3) Using COLLECTIONS. HASHMAP
Response response = WHEN().GET(URI);
Map map = response.body().as(Map.class);

Assert.assertEquals(map.get(“job_id”),”AD_PRES”);

4) Using POJOs.
Public class Employee{…}

Response response = WHEN().GET(URI);
Employee emp = response.body().as(Employee.class);

Assert.assertEquals(Integer.valueOf(emp.getSalary()) , 24000 );
```

SERIALIZATION VS DE-SERIALIZATION

- SERIALIZATION : CONVERT JAVA OBJECT TO JSON
- DE-SERIALIZATION: JSON TO JAVA OBJECT

GSON PARSER for Serialization and Deserialization

- GSON PARSER IS USED TO CONVERT JSON TO JAVA OBJECT OR VISE VERSA.

- 2 USEFUL METHODS:

```
fromJSON, toJSON
```

```
Wrapper value = GSON.fromJSON(response.asString();, Wrapper.class);
String text = value.getTranslatedText();
```

```
given().body(GSON.toJSON(employee)).
when().post(URL).then().statusCode(201);
```

-