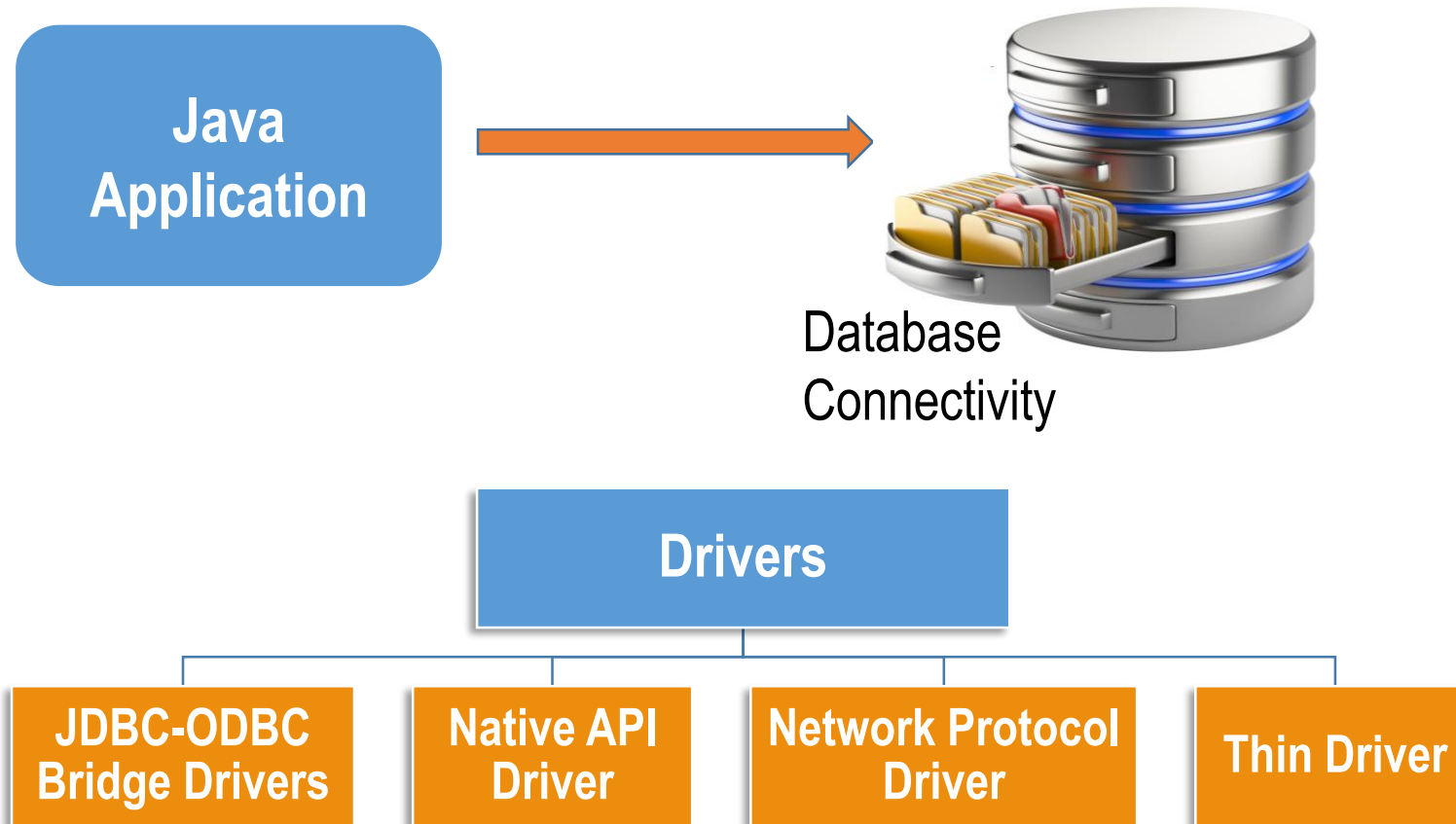


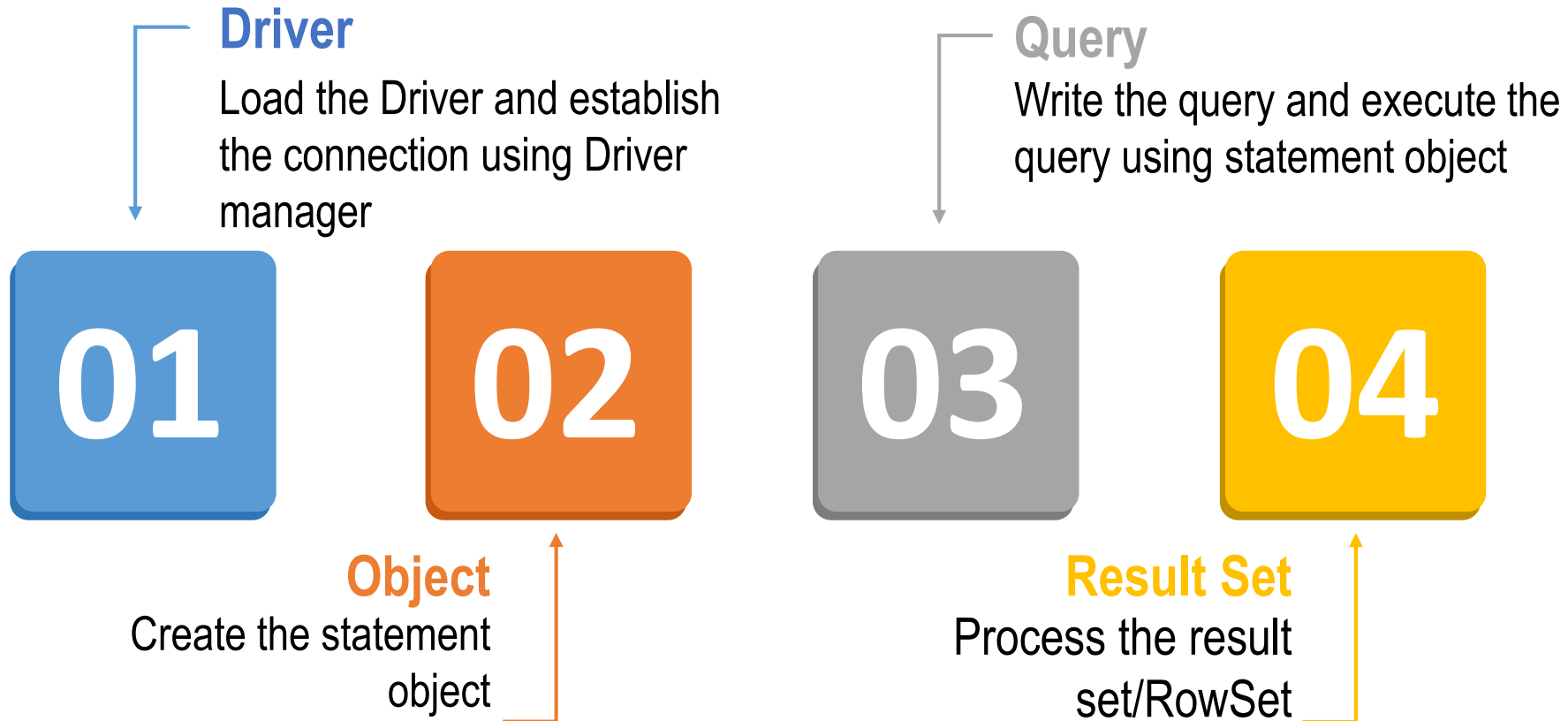
SPRING JDBC INTRODUCTION

JDBC IN JAVA APPLICATIONS



Thin driver is the most **preferred driver**

JDBC STEPS



LIMITATIONS OF JDBC

JDBC - Limitations

- ✓ Need to deal with many databases, tables and large number of users
- ✓ Accessed over the internet/web
- ✓ Remote RDBMS
- ✓ Not preferred for production environments

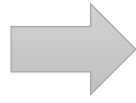
PROBLEMS WITH JDBC

JDBC - Problems

- ✓ Demands writing lots of code
- ✓ Handle exceptions explicitly for the database logic
- ✓ Transaction management needs to be done by the developer

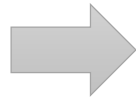
SPRING DATA ACCESS

DAO Support



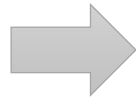
Easy to work with data access technologies

Transaction Management



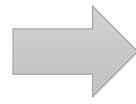
Implemented in both declarative and programmatic approaches

JDBC Connectivity



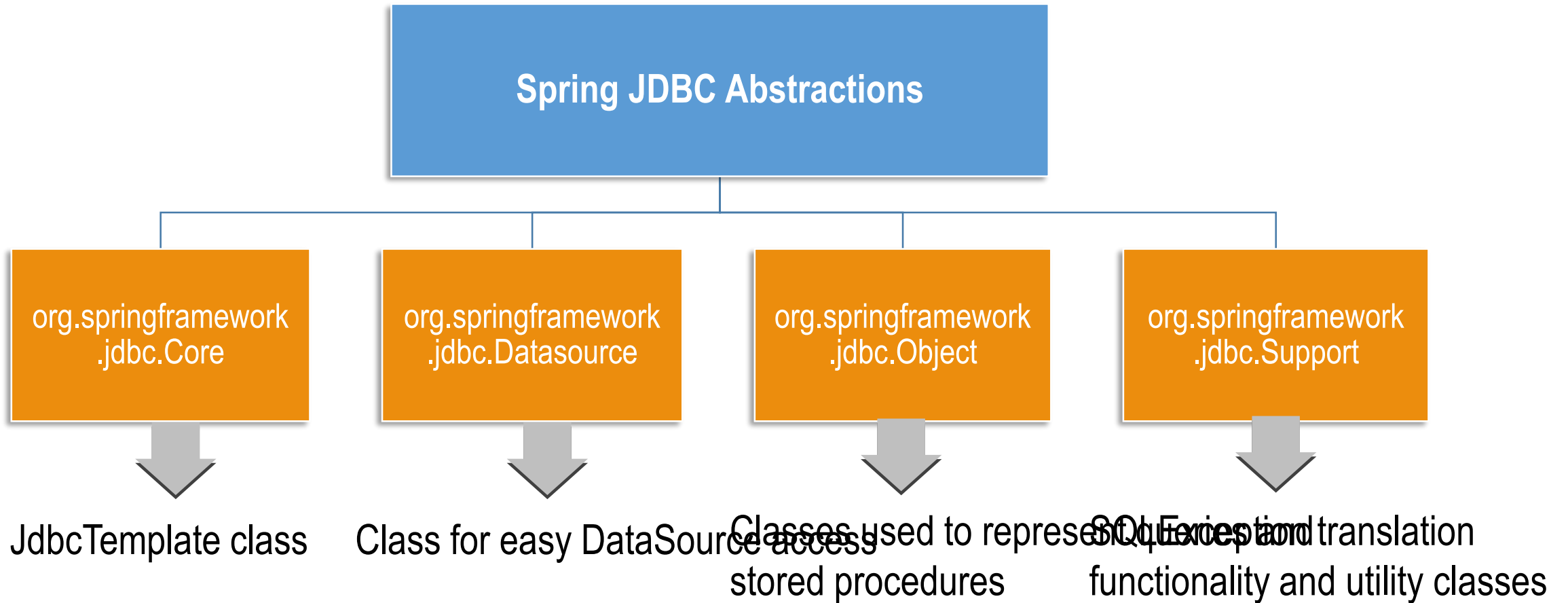
Database operations using JDBC API

Handle Exceptions



Occurs in data access logic

SPRING JDBC ABSTRACTIONS

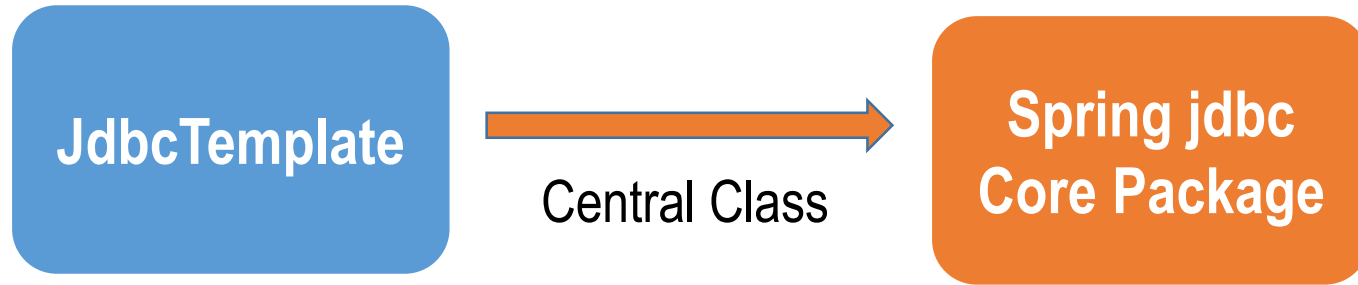


SPRING JDBC - ROLES

Action	Spring	Developer
Define connection parameters		✓
Open the connection	✓	
Specify the SQL statement		✓
Declare parameters and provide parameter values		✓
Prepare and execute the statement	✓	
Set up the loop to iterate through the results (if any)	✓	
Do the work for each iteration		✓
Process any exception	✓	
Handle transactions	✓	
Close the connection, statement and <u>resultset</u>	✓	

SPRING JDBC TEMPLATE

SPRING JDBCTEMPLATE CLASS

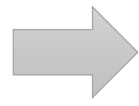


Spring JdbcTemplate

- ✓ Common errors can be avoided
- ✓ Lot of boiler plate code can be avoided
- ✓ Handles the SQL exception and uses specific exception classes under `org.springframework.dao`
- ✓ CRUD operations are easily done

SPRING JDBCTEMPLATE CLASS METHODS

Update



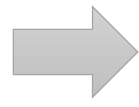
Used to insert, update and delete records



```
public int update(String query)
```

```
public int update(String query,  
Object... args)
```

Execute

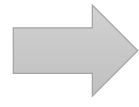


Executes DDL query

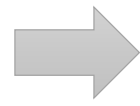


```
public void execute(String query)
```

Query



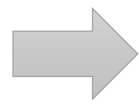
Retrieves records from database using either Rowmapper or ResultsetExtractor interface



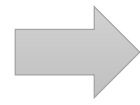
```
public T query(String sql,  
ResultSetExtractor rse)
```

```
public List query(String sql,  
RowMapper rse)
```

QueryForObject



Executes a query for a result object and given static SQL



```
public T queryForObject(String query,  
Class requiredType)
```