

TYBCA Science — JDBC Practical (V Sem, 2025–26)

Exam-ready notes • PostgreSQL • Java (JDBC)

Q1. Student(srn, sname, smarks): Insert 5, Update one, Delete one

Java JDBC Code:

```
package jdbc;
import java.sql.*;

public class Q1 {
    public static void main(String[] args) {
        try {
            Class.forName("org.postgresql.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:postgresql://localhost:5432/postgres", "postgres", "postgres"
            );
            Statement stmt = con.createStatement();

            stmt.executeUpdate(
                "CREATE TABLE IF NOT EXISTS student (srn INT PRIMARY KEY, sname VARCHAR(30), smarks INT)"
            );
            stmt.executeUpdate(
                "INSERT INTO student VALUES (1,'Rahul',80),(2,'Amit',75),(3,'Sneha',90),(4,'Ravi',85),(5,'Priya',92)"
            );
            stmt.executeUpdate("UPDATE student SET smarks=95 WHERE srn=2");
            stmt.executeUpdate("DELETE FROM student WHERE srn=3");

            System.out.println("■ Q1 Done");
            stmt.close();
            con.close();
        } catch (Exception e) { e.printStackTrace(); }
    }
}
```

Q2. Customer(CID, CName, Address, Ph_No) using PreparedStatement

Java JDBC Code:

```
package jdbc;
import java.sql.*;

public class Q2 {
    public static void main(String[] args) {
        try {
            Class.forName("org.postgresql.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:postgresql://localhost:5432/postgres", "postgres", "postgres"
            );

            PreparedStatement ps = con.prepareStatement("INSERT INTO customer VALUES(?,?,?,?)");
            ps.setInt(1, 101);
            ps.setString(2, "Ramesh");
            ps.setString(3, "Pune");
            ps.setString(4, "9876543210");
            ps.executeUpdate();

            System.out.println("■ Q2 Done");
            ps.close();
            con.close();
        } catch (Exception e) { e.printStackTrace(); }
    }
}
```

Q3. Create table Emp(Eno, Ename, Eadd, Eaadhar, esal) and display message

Java JDBC Code:

```

package jdbc;
import java.sql.*;

public class Q3 {
    public static void main(String[] args){
        try{
            Class.forName("org.postgresql.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:postgresql://localhost:5432/postgres", "postgres", "postgres"
            );
            Statement stmt = con.createStatement();
            stmt.executeUpdate(
                "CREATE TABLE IF NOT EXISTS emp (eno INT, ename VARCHAR(30), eadd VARCHAR(50), eaadhar"
            );
            System.out.println("■ Table Created");
            stmt.close();
            con.close();
        }catch(Exception e){ e.printStackTrace(); }
    }
}

```

Q4. Create Teacher(TNo, Tname, Sal, Desg) and insert a record

Java JDBC Code:

```

package jdbc;
import java.sql.*;

public class Q4 {
    public static void main(String[] args){
        try{
            Class.forName("org.postgresql.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:postgresql://localhost:5432/postgres", "postgres", "postgres"
            );
            Statement stmt = con.createStatement();
            stmt.executeUpdate(
                "CREATE TABLE IF NOT EXISTS teacher (tno INT, tname VARCHAR(30), sal INT, desg VARCHAR(30))"
            );
            stmt.executeUpdate("INSERT INTO teacher VALUES (1,'Anita',50000,'Professor')");
            System.out.println("■ Q4 Done");
            stmt.close();
            con.close();
        }catch(Exception e){ e.printStackTrace(); }
    }
}

```

Q5. Update Course: number_of_students of "BCA Science" to 1000 (create & insert first)

Java JDBC Code:

```

package jdbc;
import java.sql.*;

public class Q5 {
    public static void main(String[] args){
        try{
            Class.forName("org.postgresql.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:postgresql://localhost:5432/postgres", "postgres", "postgres"
            );
            Statement stmt = con.createStatement();
            stmt.executeUpdate(
                "CREATE TABLE IF NOT EXISTS course (code INT, name VARCHAR(30), department VARCHAR(30), number_of_students INT)"
            );
            stmt.executeUpdate("INSERT INTO course VALUES (1,'BCA','Science',500)");
            stmt.executeUpdate("UPDATE course SET number_of_students=1000 WHERE name='BCA'");
            System.out.println("■ Q5 Done");
            stmt.close();
        }
    }
}

```

```

        con.close();
    }catch(Exception e){ e.printStackTrace(); }
}
}

```

Q6. Insert Student(RNo, SName, Per, Gender, Class) using PreparedStatement

Java JDBC Code:

```

package jdbc;
import java.sql.*;

public class Q6 {
    public static void main(String[] args){
        try{
            Class.forName("org.postgresql.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:postgresql://localhost:5432/postgres", "postgres", "postgres"
            );
            PreparedStatement ps = con.prepareStatement("INSERT INTO student2 VALUES(?,?,?,?);");
            ps.setInt(1,1);
            ps.setString(2,"Amit");
            ps.setInt(3,85);
            ps.setString(4,"M");
            ps.setString(5,"SYBCA");
            ps.executeUpdate();
            System.out.println("■ Q6 Done");
            ps.close();
            con.close();
        }catch(Exception e){ e.printStackTrace(); }
    }
}

```

Q7. Delete employee by Eno (take Eno from user)

Java JDBC Code:

```

package jdbc;
import java.sql.*;

public class Q7 {
    public static void main(String[] args){
        try{
            Class.forName("org.postgresql.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:postgresql://localhost:5432/postgres", "postgres", "postgres"
            );
            Statement stmt = con.createStatement();
            stmt.executeUpdate("CREATE TABLE IF NOT EXISTS employee (eno INT, ename VARCHAR(30), salary INT);");
            stmt.executeUpdate("INSERT INTO employee VALUES (1,'Raj',30000),(2,'Meena',40000);");
            stmt.executeUpdate("DELETE FROM employee WHERE eno=1");
            System.out.println("■ Q7 Done");
            stmt.close();
            con.close();
        }catch(Exception e){ e.printStackTrace(); }
    }
}

```

Q8. Display list of college names from college table

Java JDBC Code:

```

package jdbc;
import java.sql.*;

public class Q8 {
    public static void main(String[] args){
        try{
            Class.forName("org.postgresql.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:postgresql://localhost:5432/postgres", "postgres", "postgres"
            );

```

```

    };
    Statement stmt = con.createStatement();
    ResultSet rs = stmt.executeQuery("SELECT cname FROM college");
    while(rs.next()) System.out.println(rs.getString(1));
    rs.close();
    stmt.close();
    con.close();
} catch (Exception e) { e.printStackTrace(); }
}
}

```

Q9. Insert Emp (empno, name, DOB, BG) — simple JDBC version matching your style

Java JDBC Code:

```

package jdbc;
import java.sql.*;

public class Q9 {
    public static void main(String[] args){
        try{
            Class.forName("org.postgresql.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:postgresql://localhost:5432/postgres", "postgres", "postgres"
            );
            Statement stmt = con.createStatement();
            stmt.executeUpdate(
                "CREATE TABLE IF NOT EXISTS emp (empno INT, name VARCHAR(40), dob DATE, bg VARCHAR(5))"
            );
            PreparedStatement ps = con.prepareStatement("INSERT INTO emp(empno,name,dob,bg) VALUES (?,?)");
            ps.setInt(1, 1001);
            ps.setString(2, "Suresh");
            ps.setDate(3, java.sql.Date.valueOf("2003-05-10")); // YYYY-MM-DD
            ps.setString(4, "O+");
            ps.executeUpdate();
            System.out.println("■ Q9 Done");
            ps.close();
            stmt.close();
            con.close();
        } catch (Exception e) { e.printStackTrace(); }
    }
}

```

Q10. Employees CRUD and query salary > 50000

Java JDBC Code:

```

package jdbc;
import java.sql.*;

public class Q10 {
    public static void main(String[] args){
        try{
            Class.forName("org.postgresql.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:postgresql://localhost:5432/postgres", "postgres", "postgres"
            );
            Statement stmt = con.createStatement();
            stmt.executeUpdate(
                "CREATE TABLE IF NOT EXISTS employees (id INT, name VARCHAR(30), email VARCHAR(40), salary INT)"
            );
            stmt.executeUpdate("INSERT INTO employees VALUES (1,'Ravi','ravi@mail.com',60000,101)");
            stmt.executeUpdate("UPDATE employees SET salary=75000 WHERE id=1");
            stmt.executeUpdate("DELETE FROM employees WHERE id=1");
            ResultSet rs = stmt.executeQuery("SELECT * FROM employees WHERE salary>50000");
            while(rs.next()) System.out.println(rs.getString("name")+ " " +rs.getInt("salary"));
            rs.close();
            stmt.close();
            con.close();
        }
    }
}

```

```

        }catch(Exception e){ e.printStackTrace(); }
    }
}

```

Q11. Students CRUD (insert, update GPA, delete by email)

Java JDBC Code:

```

package jdbc;
import java.sql.*;

public class Q11 {
    public static void main(String[] args){
        try{
            Class.forName("org.postgresql.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:postgresql://localhost:5432/postgres", "postgres", "postgres"
            );
            Statement stmt = con.createStatement();
            stmt.executeUpdate(
                "CREATE TABLE IF NOT EXISTS students (id INT, first_name VARCHAR(20), last_name VARCHAR(20))"
            );
            stmt.executeUpdate(
                "INSERT INTO students VALUES (1,'Amit','Shah','amit@mail.com','2025-01-01',8.5,101)"
            );
            stmt.executeUpdate("UPDATE students SET gpa=9.0 WHERE id=1");
            stmt.executeUpdate("DELETE FROM students WHERE email='amit@mail.com'");
            System.out.println("■ Q11 Done");
            stmt.close();
            con.close();
        }catch(Exception e){ e.printStackTrace(); }
    }
}

```

Q12. Products CRUD with timestamp and stock update

Java JDBC Code:

```

package jdbc;
import java.sql.*;

public class Q12 {
    public static void main(String[] args){
        try{
            Class.forName("org.postgresql.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:postgresql://localhost:5432/postgres", "postgres", "postgres"
            );
            Statement stmt = con.createStatement();
            stmt.executeUpdate(
                "CREATE TABLE IF NOT EXISTS products (id INT, name VARCHAR(30), description VARCHAR(50))"
            );
            stmt.executeUpdate(
                "INSERT INTO products VALUES (1,'Pen','Blue ink pen',10,100,1,NOW())"
            );
            stmt.executeUpdate("UPDATE products SET quantity_in_stock=95 WHERE id=1");
            stmt.executeUpdate("DELETE FROM products WHERE quantity_in_stock=0");
            System.out.println("■ Q12 Done");
            stmt.close();
            con.close();
        }catch(Exception e){ e.printStackTrace(); }
    }
}

```

Q13. Orders & Order_Items: insert order, update status, delete order

Java JDBC Code:

```

package jdbc;
import java.sql.*;

```

```

public class Q13 {
    public static void main(String[] args){
        try{
            Class.forName("org.postgresql.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:postgresql://localhost:5432/postgres", "postgres", "postgres"
            );
            Statement stmt = con.createStatement();
            stmt.executeUpdate(
                "CREATE TABLE IF NOT EXISTS orders (id INT, customer_id INT, order_date DATE, total_am"
            );
            stmt.executeUpdate(
                "CREATE TABLE IF NOT EXISTS order_items (id INT, order_id INT, product_id INT, quantity"
            );
            stmt.executeUpdate("INSERT INTO orders VALUES (1,101,'2025-01-01',200,'NEW')");
            stmt.executeUpdate("UPDATE orders SET status='SHIPPED' WHERE id=1");
            stmt.executeUpdate("DELETE FROM orders WHERE id=1");
            System.out.println("■ Q13 Done");
            stmt.close();
            con.close();
        }catch(Exception e){ e.printStackTrace(); }
    }
}

```

Q14. Books CRUD and query by author

Java JDBC Code:

```

package jdbc;
import java.sql.*;

public class Q14 {
    public static void main(String[] args){
        try{
            Class.forName("org.postgresql.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:postgresql://localhost:5432/postgres", "postgres", "postgres"
            );
            Statement stmt = con.createStatement();
            stmt.executeUpdate(
                "CREATE TABLE IF NOT EXISTS books (id INT, title VARCHAR(50), author VARCHAR(30), price"
            );
            stmt.executeUpdate("INSERT INTO books VALUES (1,'Java Basics','James',500,2020)");
            stmt.executeUpdate("UPDATE books SET price=600 WHERE id=1");
            stmt.executeUpdate("DELETE FROM books WHERE id=1");
            ResultSet rs = stmt.executeQuery("SELECT * FROM books WHERE author='James'");
            while(rs.next()) System.out.println(rs.getString("title")+\" \" +rs.getInt("price"));
            rs.close();
            stmt.close();
            con.close();
        }catch(Exception e){ e.printStackTrace(); }
    }
}

```

Q15. Update Course department BCA → Bsc CA (create & insert first)

Java JDBC Code:

```

package jdbc;
import java.sql.*;

public class Q15 {
    public static void main(String[] args){
        try{
            Class.forName("org.postgresql.Driver");
            Connection con = DriverManager.getConnection(
                "jdbc:postgresql://localhost:5432/postgres", "postgres", "postgres"
            );
            Statement stmt = con.createStatement();
            stmt.executeUpdate(

```

```

        "CREATE TABLE IF NOT EXISTS course2 (code INT, name VARCHAR(30), department VARCHAR(30)
    );
    stmt.executeUpdate("INSERT INTO course2 VALUES (1,'BCA','BCA',500)");
    stmt.executeUpdate("UPDATE course2 SET department='Bsc CA' WHERE department='BCA'");
    System.out.println("■ Q15 Done");
    stmt.close();
    con.close();
} catch (Exception e) { e.printStackTrace(); }
}
}

```

Notes & Tips

- Code lines are fully visible; long SQL queries are wrapped onto the next line instead of being cut.
- Make sure PostgreSQL JDBC driver is available; connection string uses user/password 'postgres'.