

জাভায় ডাটাবেজ প্রোগ্রামিং

মো: আবদুল কাদের

জাভা দিয়ে ডাটাবেজ প্রোগ্রামিংয়ের জন্য গত পর্বে চারটি ধাপের কথা উল্লেখ করা হয়েছিল। এর মধ্যে গত পর্বে প্রথম দুটি ধাপ যেমন- ডাটাবেজ তৈরি ও ডাটা সোর্স নেম (DSN) তৈরির পদ্ধতি দেখানো হয়েছে। এ পর্বে পরের দুটি ধাপ যেমন- জাভা প্রোগ্রাম তৈরি ও ডাটাবেজ থেকে ডাটা প্রদর্শন করার পদ্ধতি নিয়ে আলোচনা করা হয়েছে। তবে, প্রোগ্রামটি কাজ করার জন্য চারটি ধাপই সম্পন্ন করতে হবে। জাভা প্রোগ্রাম তৈরির আগে আমাদের কয়েকটি বিষয় সম্পর্কে ধারণা থাকতে হবে।

DriverManager

ড্রাইভার ম্যানেজারের মূল কাজ হলো Jdbc-এর ড্রাইভারগুলোকে ম্যানেজ করা। এটি ড্রাইভার ক্লাসগুলোকে মেমরিতে লোড করে। ব্যবহারকারীর চাহিদামতো পরে তাদেরকে কাস্টমাইজ করা যায়। যেমন- my sql ড্রাইভার লোড করার জন্য কোড-

```
Class.forName("my.sql.Driver");
```

যখন ড্রাইভার ম্যানেজার getConnection মেথডকে কল করে, তখন প্রয়োজনীয় ড্রাইভারকে সে লোড করে।

JdbcOdbcDriver

Jdbc = Java Database Connectivity

Odbc = Open Database Connectivity

জাভা দিয়ে ডাটাবেজের সাথে কানেকশনের জন্য জাভার ডাটাবেজ ড্রাইভার Jdbc এবং মাইক্রোসফটের ড্রাইভার Odbc প্রয়োজন।

Statement

ডাটাবেজ থেকে ডাটা নেয়া/পড়ার জন্য Structured Query Language বা Sql ল্যাঙ্গুয়েজ প্রয়োজন। Statement-এর মাধ্যমে Sql ল্যাঙ্গুয়েজকে রান করানো যায় এবং এর মাধ্যমে ডাটাবেজ থেকে ডাটা সংগ্রহ করা যায়।

ResultSet

এটি Statement-এর মাধ্যমে সংগৃহীত ডাটাকে টেবিল আকারে সংরক্ষণ করে। রেজাল্টসেট কারেন্ট রো-এর ডাটাকে নিয়ে কাজ করার জন্য একটি কার্সর পয়েন্ট মেইনটেইন করে। প্রাথমিকভাবে কার্সরের অবস্থান হয় প্রথম রো-এর প্রথম ডাটাতে। রেজাল্টসেট Next মেথডের মাধ্যমে পরবর্তী রো-তে চলে যায়। যদি রেজাল্টসেট শেষ রো-তে অবস্থান করে, তখন Next মেথড ব্যবহার করলে false রিটার্ন করে। এমতাবস্থায় আবার প্রথম থেকে ডাটা নিয়ে কাজ করার জন্য While মেথড ব্যবহার করা যায়। ডিফল্ট রেজাল্টসেট আপডেট করা যায় না এবং শুধু সামনে অগ্রসর হতে পারে। JDBC 2.0 API-তে এই সীমাবদ্ধতা কাটানো হয়েছে। ফলে এখন রেজাল্টসেটে সংরক্ষিত ডাটাকে আপডেটের পাশাপাশি স্ক্রল করা যায়।

জাভা প্রোগ্রাম তৈরি

নিচের জাভা প্রোগ্রামটি নোটপ্যাডে টাইপ করে Students_Result_Info.java নামে D:\ড্রাইভের java ফোল্ডারে সেভ করতে হবে।

```
import java.sql.*;
public class Students_Result_Info
{
    public static void main (String args[])
    {
        Statement s;
        ResultSet r;
        try
        {
            Class.forName ("sun.jdbc.odbc.JdbcOdbc
Driver");//1
            Connection c = DriverManager.getConnection
("jdbc:odbc:abc","","");//2
            s = c.createStatement(); //3
            r = s.executeQuery("select * from
results");//4
            System.out.print("Roll" + " ");
            System.out.print("English" + " ");
            System.out.println("Math");
            while(r.next())
            {
                System.out.print(r.getString("roll") + " ");
                System.out.print(r.getString("English")+ "
");
                System.out.println(r.getString("Math"));
            }
            s.close(); //5
            c.close(); //6
        }
        catch(Exception e)
        {
            System.out.println("Error"+e);
        }
    }
}
```

কোড বিশ্লেষণ

আমরা আগেই বলেছিলাম, জাভাতে ডাটাবেজ সংক্রান্ত কাজ করার জন্য জাভার sql প্যাকেজ ইম্পোর্ট করতে হবে। প্রথম লাইনের কোড import java.sql.*; এর মাধ্যমে প্যাকেজটিকে ইম্পোর্ট করা হয়েছে।

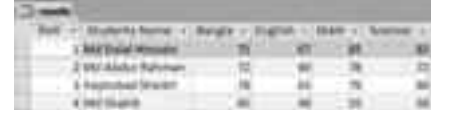
Students_Result_Info ক্লাসের মেইন মেথডে আমরা পর্যায়ক্রমে JdbcOdbcDriver ড্রাইভারকে নিয়ে একটি ডাটা সোর্সের মাধ্যমে ডাটাবেজের সাথে কানেকশন তৈরি করবো। এ জন্য ১নং চিহ্নিত লাইনে প্রথমেই ড্রাইভারকে কল করা হচ্ছে। তারপর ২নং চিহ্নিত লাইনে ডাটাসোর্সের মাধ্যমে ডাটাবেজের সাথে কানেকশন তৈরির জন্য কোড লেখা হয়েছে। এখানে abc হলো ডাটা সোর্স নেম। ডাটা সোর্স নেম কীভাবে তৈরি করা হয়, তা গত পর্বে দেখানো হয়েছে।

মূলত ডাটা সোর্স নেমের মাধ্যমে ডাটাবেজের সাথে কানেকশন তৈরি করা হয়। ডাটা সোর্স নেম (abc) বলে দেয়া ডাটাবেজের টেবিল বা কোয়েরির সাথে কানেকশন তৈরি করবে। ফলে আমরা যখন ড্রাইভারের সাহায্যে উক্ত ডাটা সোর্সকে কল করব, তখন ওই ডাটা সোর্সটি তাকে বলে দেয়া পথ অনুসরণ করে টেবিল বা কোয়েরির সাথে কানেকশন তৈরি

করবে, যাতে আমরা পরে ডাটা পেতে পারি।

৩নং লাইনে একটি স্টেটমেন্ট (Statement) তৈরি করা হয়েছে। আগেই বলা হয়েছে, এর মাধ্যমে আমরা টেবিল থেকে আমাদের পছন্দমতো ডাটা সংগ্রহের জন্য যেকোনো ধরনের কোয়েরি চালাতে পারি। স্টেটমেন্ট ছাড়া কোয়েরি চালানো সম্ভব নয় এবং টেবিল থেকে ডাটা নেয়াও সম্ভব নয়।

স্টেটমেন্টের মাধ্যমে চালানো কোয়েরির মাধ্যমে সংগৃহীত ডাটা টেবিল আকারে মেমরিতে স্টোর করার জন্য ResultSet প্রয়োজন। ৪নং লাইনের মাধ্যমে আমরা কোয়েরির মাধ্যমে প্রাপ্ত ডাটাকে রেজাল্টসেটে রাখা হয়েছে। আমরা Students ডাটাবেজে results নামে টেবিলের সব ডাটাকে মেমরিতে নেয়ার জন্য একটি কোয়েরি (select * from results) লিখেছি। ফলে উক্ত টেবিলের সব ডাটা রেজাল্টসেট তথা মেমরিতে অবস্থান করবে। সেখান থেকে আমাদের প্রয়োজন মারফিক ডাটাগুলোকে আমরা প্রদর্শন করব। Students ডাটাবেজে results টেবিলের ইনপুন দেয়া ডাটাগুলো ছিল।



সংশোধনী : গত পর্বে দেয়া টেবিলের নাম Students Result-এর পরিবর্তে results হিসেবে সংশোধন করে নিন। এজন্য তৈরি করা টেবিলের ওপর রাইট বাটন ক্লিক করে রিনেইম করতে হবে।

আমরা রোল নং অনুযায়ী ইংরেজি ও গণিতে কে কত নম্বর পেয়েছে, সেটা দেখানোর জন্য প্রথমে System.out.print(r.getString("roll")) + " "; এর মাধ্যমে রেজাল্টসেট হতে রোল নং এবং পরবর্তী দুটি লাইনের মাধ্যমে ইংরেজি ও গণিতের মান দেখাব। একটি রো-এর সব ডাটা দেখানোর পর পরবর্তী রো-তে যাওয়ার জন্য next() মেথড ব্যবহার করা হয়েছে। এই মেথডের মাধ্যমে একেবারে শেষ রো পর্যন্ত ডাটাগুলোকে প্রদর্শন করার জন্য while মেথড ব্যবহার করা হয়েছে। শেষ রো-তে আসলে প্রোগ্রামটি false রিটার্ন করবে এবং লুপ থেকে বের হয়ে প্রদর্শনের কাজ সম্পন্ন করবে। কোয়েরি ও কানেকশন বন্ধ করার জন্য ৫ ও ৬ নং লাইনে কোড লেখা হয়েছে। ফলে প্রোগ্রামটির মাধ্যমে মেমরি ব্যবহার করা বন্ধ হয়ে যাবে।

ডাটা প্রদর্শনের পুরো প্রোগ্রামটিকে try মেথডের মধ্যে লেখা হয়েছে। প্রোগ্রামটি চলার সময় যদি কোনো সমস্যা তৈরি হয়, যেমন ডাটাবেজের সাথে কানেকশন তৈরি করতে না পারা বা নির্দিষ্ট ডাটা সোর্স না থাকা ইত্যাদি ক্ষেত্রে প্রোগ্রামটি চলা যাতে বন্ধ না হয় সে ক্ষেত্রে কোনো Exception তৈরি হলে catch-এর মাধ্যমে ওই নির্দিষ্ট সমস্যাটি মেসেজ আকারে প্রদর্শন করবে।

প্রোগ্রাম রান করা

প্রোগ্রামটি রান করার পদ্ধতি অন্যান্য জাভা প্রোগ্রামের মতোই। এ জন্য অবশ্যই আপনার কমপিউটারে Jdk সফটওয়্যার ইনস্টল থাকতে হবে। আমরা সফটওয়্যারটির Jdk1.4 ভার্সন ব্যবহার করেছি। প্রোগ্রামটি রান করার জন্য বরাবরের মতো (বাকি অংশ ৬১ পৃষ্ঠায়)