**Certainly! The provided Java code is part of a simple Hospital Management System, specifically focusing on the Doctor class. Let's go through the code line by line with explanations:**

package HospitalManagementSystem;

**This line declares the package name in which the Doctor class resides.**

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.Scanner;

**These import statements bring in the necessary classes from the java.sql package for working with JDBC (Java Database Connectivity) and the Scanner class for user input.**

public class Doctor {

**This line starts the definition of the Doctor class.**

private Connection connection;

**This line declares a private member variable connection of type Connection, which represents a database connection.**

public Doctor(Connection connection){

this.connection = connection;

}

**This is the constructor of the Doctor class. It takes a Connection object as a parameter and initializes the connection member variable with the provided connection when creating an instance of the Doctor class.**

public void viewDoctors(){

**This line starts the definition of the viewDoctors method, which is used to display information about all doctors in the database.**

String query = "select \* from doctors";

try{

PreparedStatement preparedStatement = connection.prepareStatement(query);

ResultSet resultSet = preparedStatement.executeQuery();

System.out.println("Doctors: ");

System.out.println("+----------------+-------------------------+-----------------------+");

System.out.println("| Doctor ID | Name | Specialization |");

System.out.println("+----------------+-------------------------+-----------------------+");

**These lines set up a SQL SELECT query to retrieve all doctor records from the database. The query is then executed, and the results are stored in a ResultSet. The subsequent lines print a table header for the doctor information.**

while(resultSet.next()){

int id = resultSet.getInt("id");

String name = resultSet.getString("name");

String specialization = resultSet.getString("specialization");

System.out.printf("| %-14s | %-23s | %-21s |\n",id,name,specialization);

System.out.println("+---------------+-------------------------+-----------------------+");

}

}catch (SQLException e){

e.printStackTrace();

}

}

**In this block, a while(resultSet.next()) loop iterates through the result set and prints each doctor's information in a tabular format using System.out.printf. Any SQL exception is caught and printed in the catch block.**

public boolean getDoctorById(int id){

**This line starts the definition of the getDoctorById method, which is used to check if a doctor with a specific ID exists in the database.**

String query = "SELECT \* FROM doctors WHERE id = ?";

try{

PreparedStatement preparedStatement = connection.prepareStatement(query);

preparedStatement.setInt(1,id);

ResultSet resultSet = preparedStatement.executeQuery();

if(resultSet.next()) {

return true;

}else{

return false;

}

}catch(SQLException e){

e.printStackTrace();

}

return false;

}

}

**In this block, a SQL SELECT query is prepared and executed to retrieve a doctor record based on the specified ID. If a record is found in the result set, the method returns true; otherwise, it returns false. Any SQL exception is caught and printed in the catch block. If an exception occurs, the method also returns false.**