# PreparedStatement interface

The PreparedStatement interface is a subinterface of Statement. It is used to execute parameterized query.

Let's see the example of parameterized query:

String sql="insert into emp values(?,?,?)";

As you can see, we are passing parameter (?) for the values. Its value will be set by calling the setter methods of PreparedStatement.

### Why use PreparedStatement?

**Improves performance**: The performance of the application will be faster if you use PreparedStatement interface because query is compiled only once.

#### How to get the instance of PreparedStatement?

The prepareStatement() method of Connection interface is used to return the object of PreparedStatement. Syntax:

**public** PreparedStatement prepareStatement(String query)**throws** SQLException{}

### Methods of PreparedStatement interface

The important methods of PreparedStatement interface are given below:

|  |  |
| --- | --- |
| **Method** | **Description** |
| public void setInt(int paramIndex, int value) | sets the integer value to the given parameter index. |
| public void setString(int paramIndex, String value) | sets the String value to the given parameter index. |
| public void setFloat(int paramIndex, float value) | sets the float value to the given parameter index. |
| public void setDouble(int paramIndex, double value) | sets the double value to the given parameter index. |
| public int executeUpdate() | executes the query. It is used for create, drop, insert, update, delete etc. |
| public ResultSet executeQuery() | executes the select query. It returns an instance of ResultSet. |

### Example of PreparedStatement interface that inserts the record

First of all create table as given below:

create table emp(id number(10),name varchar2(50));

Now insert records in this table by the code given below:

**import** java.sql.\*;

**class** InsertPrepared{

**public** **static** **void** main(String args[]){

**try**{

Class.forName("oracle.jdbc.driver.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","oracle");

PreparedStatement stmt=con.prepareStatement("insert into Emp values(?,?)");

stmt.setInt(1,101);*//1 specifies the first parameter in the query*

stmt.setString(2,"Ratan");

**int** i=stmt.executeUpdate();

System.out.println(i+" records inserted");

con.close();

}**catch**(Exception e){ System.out.println(e);}

}

}

### Example of PreparedStatement interface that updates the record

PreparedStatement stmt=con.prepareStatement("update emp set name=? where id=?");

stmt.setString(1,"Sonoo");*//1 specifies the first parameter in the query i.e. name*

stmt.setInt(2,101);

**int** i=stmt.executeUpdate();

System.out.println(i+" records updated");

### Example of PreparedStatement interface that deletes the record

PreparedStatement stmt=con.prepareStatement("delete from emp where id=?");

stmt.setInt(1,101);

**int** i=stmt.executeUpdate();

System.out.println(i+" records deleted");

### Example of PreparedStatement interface that retrieve the records of a table

PreparedStatement stmt=con.prepareStatement("select \* from emp");

ResultSet rs=stmt.executeQuery();

**while**(rs.next()){

System.out.println(rs.getInt(1)+" "+rs.getString(2));

}

### Example of PreparedStatement to insert records until user press n

**import** java.sql.\*;

**import** java.io.\*;

**class** RS{

**public** **static** **void** main(String args[])**throws** Exception{

Class.forName("oracle.jdbc.driver.OracleDriver");

Connection con=DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:xe","system","oracle");

PreparedStatement ps=con.prepareStatement("insert into emp130 values(?,?,?)");

BufferedReader br=**new** BufferedReader(**new** InputStreamReader(System.in));

**do**{

System.out.println("enter id:");

**int** id=Integer.parseInt(br.readLine());

System.out.println("enter name:");

String name=br.readLine();

System.out.println("enter salary:");

**float** salary=Float.parseFloat(br.readLine());

ps.setInt(1,id);

ps.setString(2,name);

ps.setFloat(3,salary);

**int** i=ps.executeUpdate();

System.out.println(i+" records affected");

System.out.println("Do you want to continue: y/n");

String s=br.readLine();

**if**(s.startsWith("n")){

**break**;

}

}**while**(**true**);

con.close();

}}