JDBC - Continuation

- Implement the deletion of records:

```
public void remove(Student student) {
	try {
		PreparedStatement stmt = connection.prepareStatement("delete
from students where id=?");
		stmt.setLong(1, student.getId());
		stmt.execute();
		stmt.close();
	} catch (SQLException e) {
		throw new RuntimeException(e);
	}
}
```

- Modify the class CreateStudent, adding a code to test the insertion method (remember to set a value for the **id**):

```
// removing record
StudentDAO dao = new StudentDAO();
dao.remove(student);
System.out.println("Removed!");
```

- implement a method for updating:

```
public void update(Student student) {
	String sql = "update students set name=?, email=?, address=? where
id=?";

	try {
		PreparedStatement stmt = connection.prepareStatement(sql);
		stmt.setString(1, student.getNome());
		stmt.setString(2, student.getEmail());
		stmt.setString(3, student.getEndereco());
		stmt.setLong(4, student.getId());
		stmt.execute();
		stmt.close();
	} catch (SQLException e) {
		throw new RuntimeException(e);
	}
}
```

- modify the class CreateStudent, adding a code to test the method for updating (remember to provide an **id**):

```
// updating a record
StudentDAO dao = new StudentDAO();
dao.update(student);
System.out.println("Updated!");
```

- implement a method to list all records:

```
public List<Student> getList() { // it imports java.util
		List<Student> students = new ArrayList<Student>(); // it imports java.util

		PreparedStatement stmt;
		try {
			stmt = this.connection.prepareStatement("select * from students order by name");

			ResultSet rs = stmt.executeQuery(); // it imports java.sql

			while (rs.next()) {
				Student student = new Student();
				student.setId(rs.getLong("id"));
				student.setName(rs.getString("name"));
				student.setEmail(rs.getString("email"));
				student.setAddress(rs.getString("address"));

				students.add(student);
			}
			rs.close();
			stmt.close();
		} catch (SQLException e) {
			throw new RuntimeException(e);
		}
		return students;
	}
```

- modify the class CreateStudent, adding a code to test the method `getList()`:

```
		// listing records
		StudentDAO dao = new StudentDAO();
		List<Student> students = dao.getList();
		for ( Student student1 : students){
			System.out.println("Name:	"+student1.getName()+"	Email: "+student1 .getEmail()+" Address: "+ student1.getAddress());
		}
```

- Use the methods we just created to insert, remove, update and list in order to make just tests with the database through the Java application. Using the SQL command `select`, verify in the database if the changes are there.
- **EXERCISE**: Make a CRUD to store your phone contacts:
  - Create a database named contactsdb with a table contacts containing the fields id, name, email, address;
  - Create a Java application and implement the operations of insertion, updating, deletion, searching by name and listing all contacts.