

PersistenceLib - Apache NetBeans IDE 12.5

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

190,1/314,0MB

Projects X Services Files

PersistenceLib

- Source Packages
 - utils
 - DataBase.java
 - DataPersistence.java
 - FileManager.java
 - JsonFileManager.java
 - MongoDBManager.java
 - MySQLDBManager.java
- Test Packages
- Libraries
- Test Libraries

Source History

```
1 package utils;
2
3
4 import java.util.ArrayList;
5
6 /**
7  *
8  * @author Camila
9  */
10 public class MongoDBManager implements DataBase{
11
12     @Override
13     public void openConnection(String connectionString) {
14         System.out.println("opening connection --> " + connectionString);
15     }
16
17     @Override
18     public void closeConnection() {
19         System.out.println("closing connection");
20     }
21
22     @Override
23     public void generateConnection(String user, String password, String database) {
24         System.out.println("generating connection " + user + "saving --> " + password + "in databaBase" + database);
25     }
26
27     @Override
28     public void add(String data, String table) {
29         System.out.println("adding in database " + data + "in File --> " + table);
30     }
31 }
```

File Home Share View

dist

unit2 > HW15InterfacesLibraries > PersistenceLib > dist

Name	Type	Size
PersistenceLib	Executable Jar File	

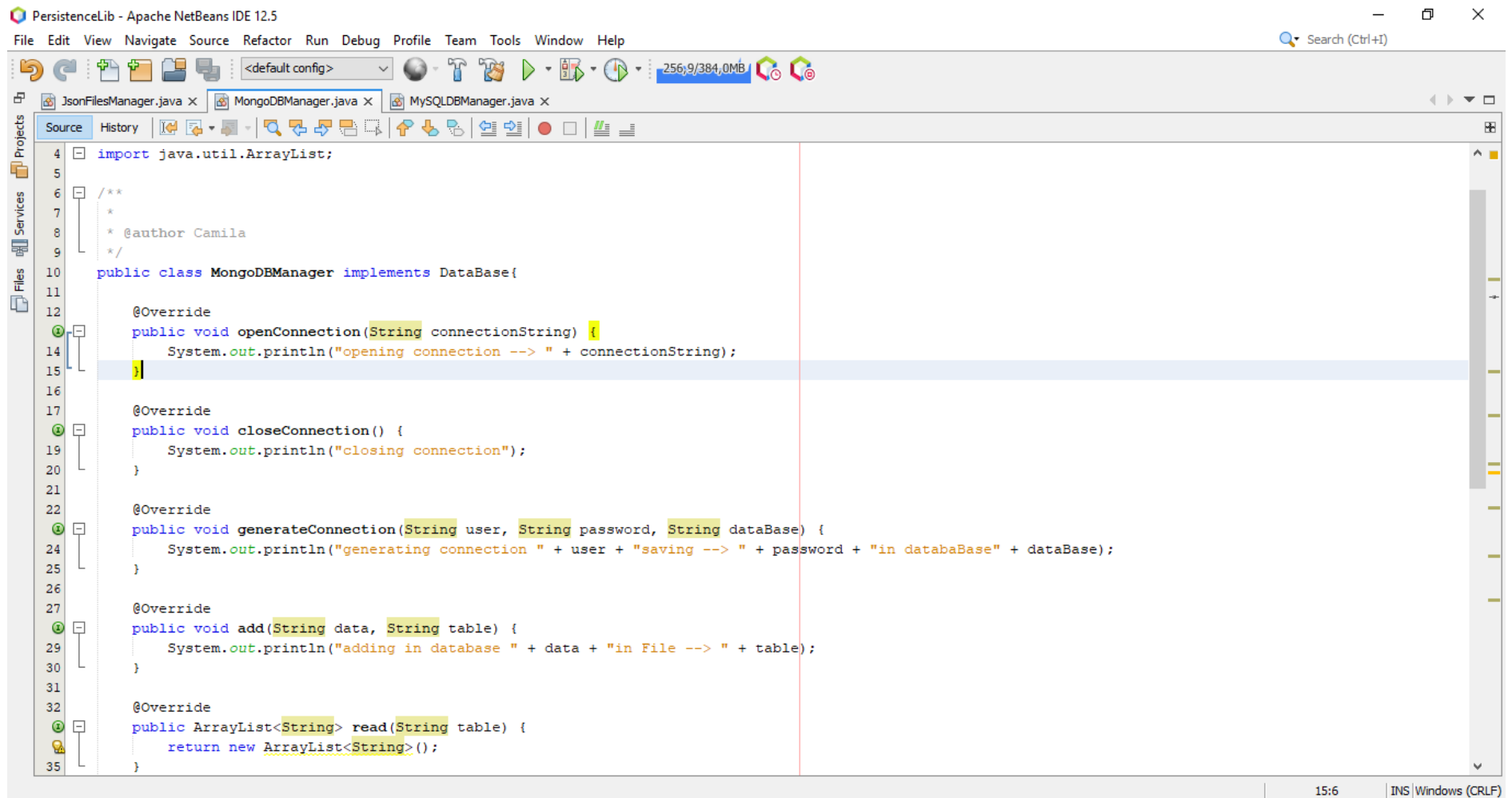
Quick access

- Desktop
- Downloads
- Documents
- Pictures
- Descargas
- Documentos
- II SEMESTRE
- CLASES DE LA UNIDAD 2

1 item

6:4 | INS

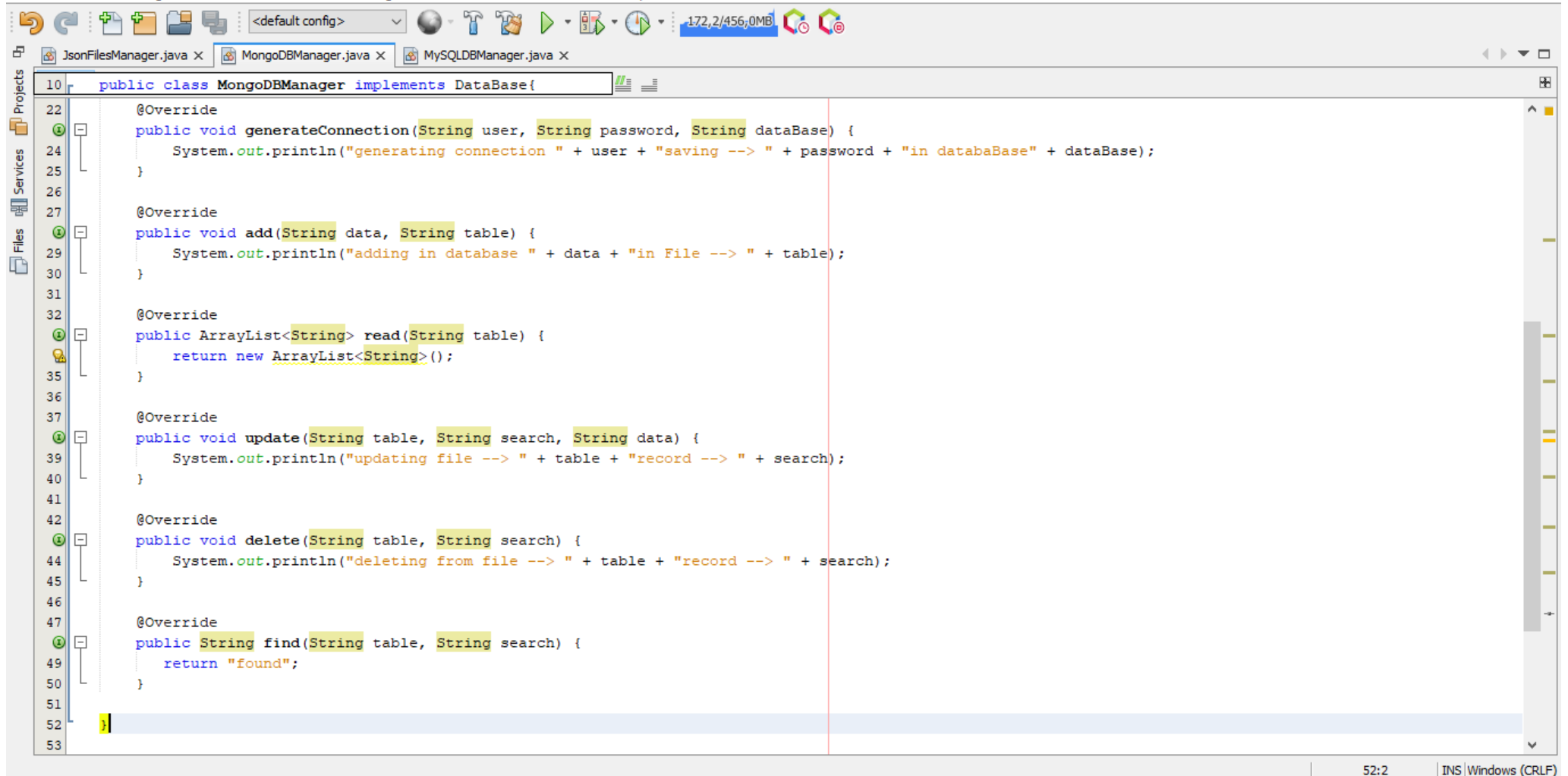
CLASS MONGO:



The screenshot shows the Apache NetBeans IDE 12.5 interface. The title bar reads "PersistenceLib - Apache NetBeans IDE 12.5". The menu bar includes File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, and Help. The toolbar contains various icons for file operations, running, and debugging. The status bar at the bottom shows "15:6" and "INS | Windows (CRLF)".

The "Projects" pane on the left shows a project named "PersistenceLib" with sub-projects "Files", "Services", and "Sources". The "Sources" pane shows three open files: "JsonFileManager.java", "MongoDBManager.java", and "MySQLDBManager.java". The "MongoDBManager.java" file is selected and its source code is displayed in the main editor.

```
4 import java.util.ArrayList;
5
6 /**
7  *
8  * @author Camila
9  */
10 public class MongoDBManager implements DataBase{
11
12     @Override
13     public void openConnection(String connectionString) {
14         System.out.println("opening connection --> " + connectionString);
15     }
16
17     @Override
18     public void closeConnection() {
19         System.out.println("closing connection");
20     }
21
22     @Override
23     public void generateConnection(String user, String password, String dataBase) {
24         System.out.println("generating connection " + user + "saving --> " + password + "in databaBase" + dataBase);
25     }
26
27     @Override
28     public void add(String data, String table) {
29         System.out.println("adding in database " + data + "in File --> " + table);
30     }
31
32     @Override
33     public ArrayList<String> read(String table) {
34         return new ArrayList<String>();
35     }
36 }
```



```
10 public class MongoDBManager implements DataBase{
22     @Override
23     public void generateConnection(String user, String password, String dataBase) {
24         System.out.println("generating connection " + user + "saving --> " + password + "in databaBase" + dataBase);
25     }
26
27     @Override
28     public void add(String data, String table) {
29         System.out.println("adding in database " + data + "in File --> " + table);
30     }
31
32     @Override
33     public ArrayList<String> read(String table) {
34         return new ArrayList<String>();
35     }
36
37     @Override
38     public void update(String table, String search, String data) {
39         System.out.println("updating file --> " + table + "record --> " + search);
40     }
41
42     @Override
43     public void delete(String table, String search) {
44         System.out.println("deleting from file --> " + table + "record --> " + search);
45     }
46
47     @Override
48     public String find(String table, String search) {
49         return "found";
50     }
51
52 }
53
```

52:2 | INS | Windows (CRLF)

CLASS MySQLDBManager:

PersistenceLib - Apache NetBeans IDE 12.5

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

441,3/456,0MB

JsonFileManager.java x MongoDBManager.java x MySQLDBManager.java x

Source History

```
1 package utils;
2
3
4 import java.util.ArrayList;
5
6 /**
7  *
8  * @author Camila
9  */
10 public class MySQLDBManager implements DataBase{
11
12     @Override
13     public void openConnection(String connectionString) {
14         System.out.println("opening connection --> " + connectionString);
15     }
16
17     @Override
18     public void closeConnection() {
19         System.out.println("closing connection");
20     }
21
22     @Override
23     public void generateConnection(String user, String password, String dataBase) {
24         System.out.println("generating connection " + user + "saving --> " + password + "in databaBase" + dataBase);
25     }
26
27     @Override
28     public void add(String data, String table) {
29         System.out.println("adding in database " + data + "in File --> " + table);
30     }
31
32     @Override
```

53:1 | INS | Windows (CRLF)

Source History

```
22  @Override
23  public void generateConnection(String user, String password, String dataBase) {
24      System.out.println("generating connection " + user + "saving --> " + password + "in databaBase" + dataBase);
25  }
26
27  @Override
28  public void add(String data, String table) {
29      System.out.println("adding in database " + data + "in File --> " + table);
30  }
31
32  @Override
33  public ArrayList<String> read(String table) {
34      return new ArrayList<String>();
35  }
36
37  @Override
38  public void update(String table, String search, String data) {
39      System.out.println("updating file --> " + table + "record --> " + search);
40  }
41
42  @Override
43  public void delete(String table, String search) {
44      System.out.println("deleting from file --> " + table + "record --> " + search);
45  }
46
47  @Override
48  public String find(String table, String search) {
49      return "found";
50  }
51
52  }
53
```

53:1 | INS | Windows (CRLF)