



Notes projet (PersonneDao) est constitué de trois parties essentielles :

1. **Partie Metier** : la casse **Personne.java**
2. **La partie DAO** : L'interface **Dao.java** et la Classe **PersonneDao.java**
3. **La partie Bdd** : la classe **DbConnection.java** (le pattern singleton java)

-----\*\*\*\*\*-----

## 1. Partie Metier : la casse **Personne.java**

```
package metier;
public class Personne {

    private int NNI;
    private String nom;
    private String prenom;

    public Personne() {
        super();
        // TODO Auto-generated constructor stub
    }
    public Personne(int nNI, String nom, String prenom) {
        super();
        NNI = nNI;
        this.nom = nom;
        this.prenom = prenom;
    }

    public int getNNI() {
        return NNI;
    }

    public void setNNI(int nNI) {
        NNI = nNI;
    }
}
```

```

    public String getNom() {
        return nom;
    }
    public void setNom(String nom) {
        this.nom = nom;
    }
    public String getPrenom() {
        return prenom;
    }
    public void setPrenom(String prenom) {
        this.prenom = prenom;
    }

    @Override
    public String toString() {
        return "Personne [NNI=" + NNI + ", nom=" + nom + ", prenom=" +
prenom + "];"
    }
}

```

### La partie DAO : L'interface `Dao.java` et la Classe `PersonneDao.java`

```

package dao;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.List;
import java.util.Optional;
import com.mysql.jdbc.PreparedStatement;
import jdbc.DbConnection;
import metier.Personne;

public class PersonneDao implements Dao<Personne> {
    private Connection conn;

    public PersonneDao() {
        try {
            conn = DbConnection.getInstance().getConnection();
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }

    @Override
    public Personne get(long id) {
        Personne personne = null;
        String requet = "select * from personne where NNI = " + (int) id;
    }
}

```

```

        System.out.println(requet);
        try {
            Statement pstmt = conn.createStatement();
            ResultSet rs = pstmt.executeQuery(requet);
            while (rs.next()) {
                int nni = rs.getInt(1);
                String nom = rs.getString(2);
                String prenom = rs.getString(3);
                System.out.println("NNI: " + nni + " nom:" + nom + "
prenom:" + prenom);
                personne = new Personne(nni, nom, prenom);
            }
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        return personne;
    }

    @Override
    public List<Personne> getAll() {
        // TODO Auto-generated method stub
        return null;
    }

    @Override
    public void save(Personne t) {
        // TODO Auto-generated method stub
    }

    @Override
    public void update(Personne t, String[] params) {
        // TODO Auto-generated method stub
    }

    @Override
    public void delete(Personne t) {
        // TODO Auto-generated method stub
    }

    public static void main(String[] args) {
        PersonneDao pdao = new PersonneDao();

        pdao.get(121212);
    }
}

```

**La partie Bdd** : la classe `DbConnection.java` (le pattern singleton java)

```

package jdbc;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class DbConnection {
    private static DbConnection instance;
    private Connection connection;
    private String url = "jdbc:mysql://localhost:3306/gestionpersonne";
    private String username = "root";
    private String password = "";

    private DbConnection() throws SQLException {
        try {
            Class.forName("com.mysql.jdbc.Driver");
            this.connection = DriverManager.getConnection(url,
username, password);
            System.out.println("Connection etablie");
        } catch (ClassNotFoundException ex) {
            System.out.println("Something is wrong with the DB
connection String : " + ex.getMessage());
        }
    }

    public Connection getConnection() {
        return connection;
    }

    public static DbConnection getInstance() throws SQLException {
        if (instance == null) {
            instance = new DbConnection();
        } else if (instance.getConnection().isClosed()) {
            instance = new DbConnection();
        }
        return instance;
    }

    public static void main(String[] args) throws SQLException {
        DbConnection.getInstance().getConnection();
    }
}

```

**Appliquer le même concept sur votre projet  
(GestionNotes).**

## Partie SGBD

1. SE connecter au serveur MySQL
2. Créer une base de données
3. Utiliser la base de données
4. Créer les tables
5. Insérer des données dans les tables
6. Description des tables
7. Exécuter des requêtes

```
Setting environment for using XAMPP
Dell@DESKTOP-D8D9ALB c:\xampp
# mysql -u root -p
Enter password: ****
Welcome to the MariaDB monitor.  Com
Your MariaDB connection id is 11
Server version: 10.4.19-MariaDB mari
Copyright (c) 2000, 2018, Oracle, Ma
Type 'help;' or '\h' for help. Type
MariaDB [(none)]>
```

```
MariaDB [(none)]> create database gestionPersonne;
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [(none)]> use gestionpersonne;
Database changed
MariaDB [gestionpersonne]>
```

```
MariaDB [gestionpersonne]> create table personne (
  -> nni int primary key,
  -> nom varchar(50),
  -> prenom varchar(50)
  -> );
Query OK, 0 rows affected (0.047 sec)
```

```
MariaDB [gestionpersonne]> desc personne;
```

| Field  | Type        | Null | Key | Default | Extra |
|--------|-------------|------|-----|---------|-------|
| nni    | int(11)     | NO   | PRI | NULL    |       |
| nom    | varchar(50) | YES  |     | NULL    |       |
| prenom | varchar(50) | YES  |     | NULL    |       |

```

3 rows in set (0.021 sec)
```

```
MariaDB [gestionpersonne]> insert into personne values(121212, "ahmed", "sidi");
Query OK, 1 row affected (0.013 sec)
```

```
MariaDB [gestionpersonne]> select * from personne;
```

| nni    | nom   | prenom |
|--------|-------|--------|
| 121212 | ahmed | sidi   |

```

1 row in set (0.000 sec)
```

```
MariaDB [gestionpersonne]> insert into personne values(131313, "baba", "dah");
Query OK, 1 row affected (0.008 sec)
```

```
MariaDB [gestionpersonne]> select * from personne;
```

```
+-----+-----+-----+
| nni    | nom   | prenom |
+-----+-----+-----+
| 121212 | ahmed | sidi   |
| 131313 | baba  | dah    |
+-----+-----+-----+
2 rows in set (0.000 sec)
```

-----\*\*\*\*-----

Fin