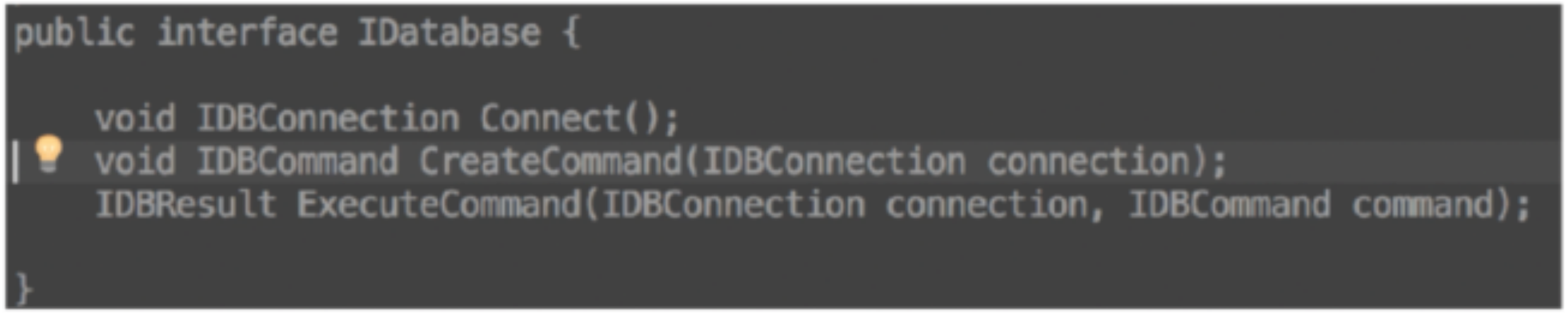
THE FACTORY PATTERN

SO - WIRING IT ALL UP..

1. CREATE THE INTERFACE

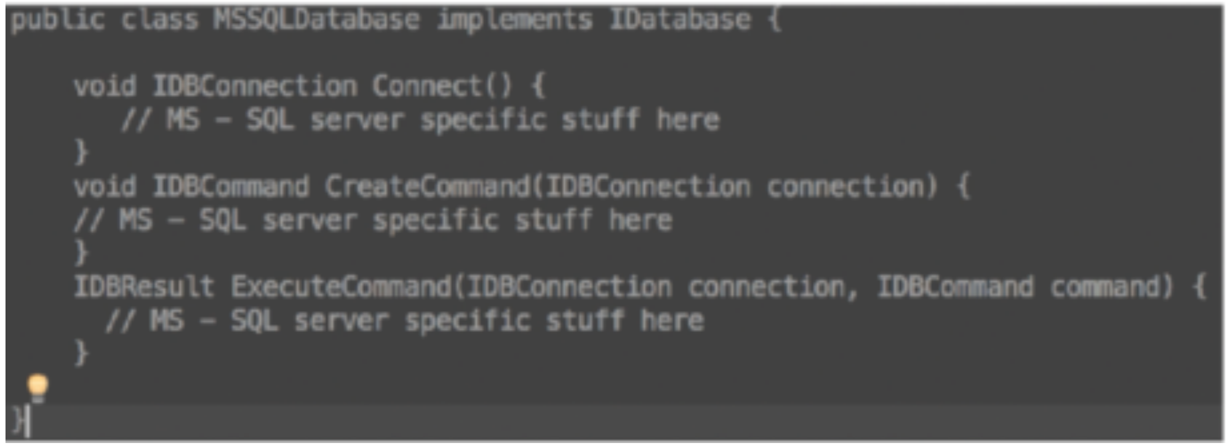


```
public interface IDatabase {

    void IDBConnection Connect();
    void IDBCommand CreateCommand(IDBConnection connection);
    IDBResult ExecuteCommand(IDBConnection connection, IDBCommand command);

}
```

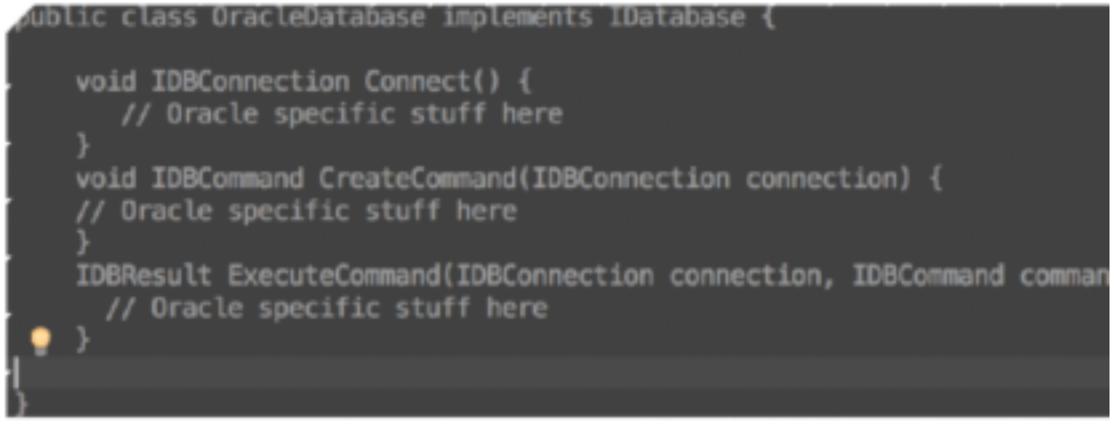
2. IMPLEMENT THE INTERFACE TWICE, ONCE EACH FOR MS-SQL AND ORACLE



```
public class MSSQLDatabase implements IDatabase {

    void IDBConnection Connect() {
       // MS - SQL server specific stuff here
    }
    void IDBCommand CreateCommand(IDBConnection connection) {
    // MS - SQL server specific stuff here
    }
    IDBResult ExecuteCommand(IDBConnection connection, IDBCommand command) {
      // MS - SQL server specific stuff here
    }

}
```

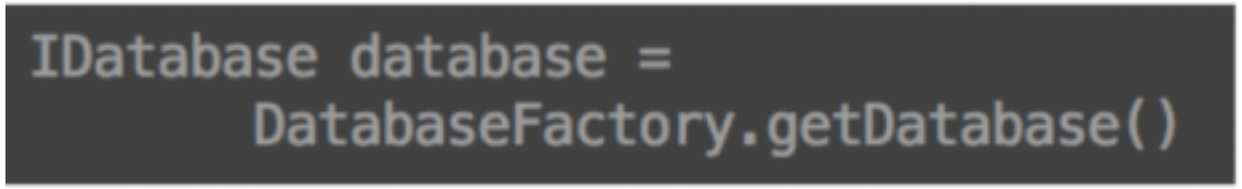


```
public class OracleDatabase implements IDatabase {

    void IDBConnection Connect() {
       // Oracle specific stuff here
    }
    void IDBCommand CreateCommand(IDBConnection connection) {
    // Oracle specific stuff here
    }
    IDBResult ExecuteCommand(IDBConnection connection, IDBCommand comman
      // Oracle specific stuff here
    }

}
```

3. CREATE A MEMBER VARIABLE OF TYPE IDATABASE AND USE IT THROUGHOUT THE CODE



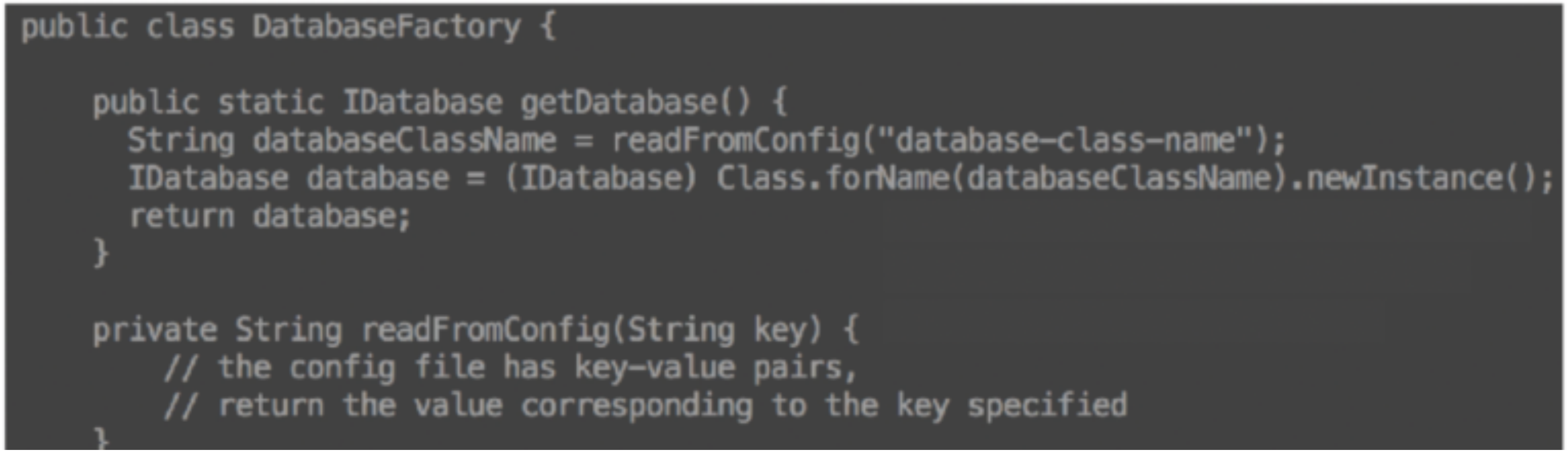
```
IDatabase database =
       DatabaseFactory.getDatabase()
```

2 POINTS TO NOTE HERE

THE MEMBER VARIABLE NEEDS TO BE OF THE INTERFACE TYPE

THE INSTANTIATION IS USING A STATIC METHOD ON THE FACTORY CLASS

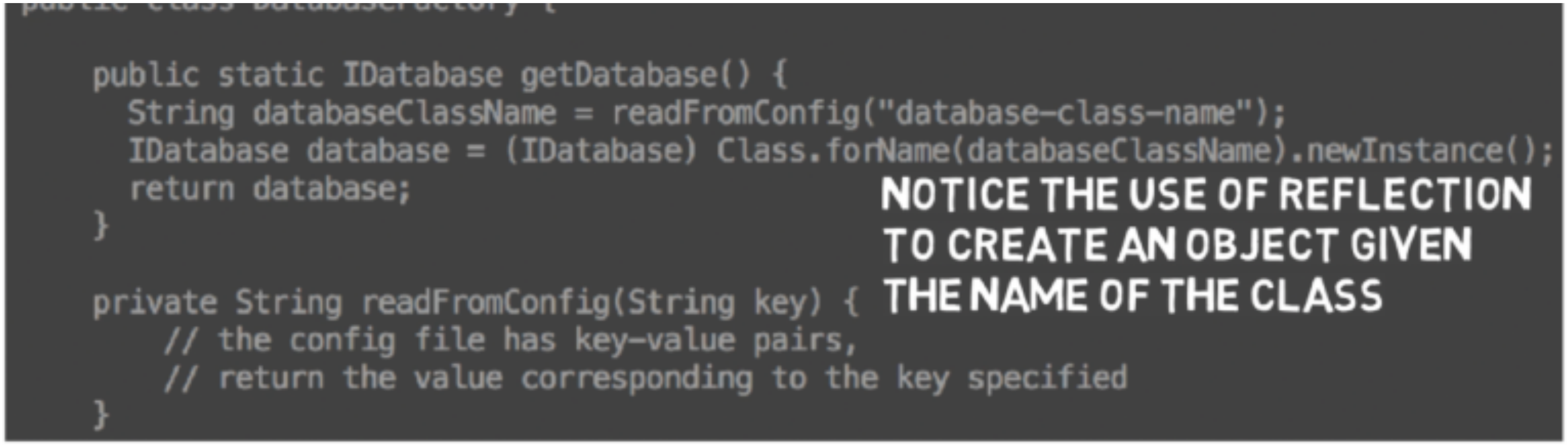
4. IN THE FACTORY CLASS, HAVE A STATIC METHOD THAT USES REFLECTION TO CREATE THE RIGHT OBJECT



```
public class DatabaseFactory {

    public static IDatabase getDatabase() {
      String databaseClassName = readFromConfig("database-class-name");
      IDatabase database = (IDatabase) Class.forName(databaseClassName).newInstance();
      return database;
    }

    private String readFromConfig(String key) {
        // the config file has key-value pairs,
        // return the value corresponding to the key specified
    }
```



```
public class DatabaseFactory {

    public static IDatabase getDatabase() {
      String databaseClassName = readFromConfig("database-class-name");
      IDatabase database = (IDatabase) Class.forName(databaseClassName).newInstance();
      return database;
    }

    private String readFromConfig(String key) {
        // the config file has key-value pairs,
        // return the value corresponding to the key specified
    }
```

NOTICE THE USE OF REFLECTION TO CREATE AN OBJECT GIVEN THE NAME OF THE CLASS

5. SPECIFY THE DATABASE CLASS TO BE USED IN THE CONFIG FILE

LATER, WHEN THE SWITCHOVER HAPPENS, SIMPLY EDIT THIS STRING TO CONTAIN THE NAME OF THE ORACLE DATABASE

```
database-class-name="MSSQLDatabase"
```

THAT'S IT! THAT IS THE FACTORY PATTERN