J2EE

JAVA 2 ENTERPRISED EDITION

Pre-requisites

- Core Java
- SQL
- Web Technologies

TECHNOLOGIES INVOLVED IN THIS J2EE

- JDBC (Java Database Connectivity)
- Servlets
- JSP (Java Server Page)

PUNITH B

J2SE VS J2EE

- **J2SE (Java Standard Edition)**
  - ➢ The core java or J2EE contains only the core fundamentals of JAVA.
  - ➢ Less library content.
  - ➢ Using Standard edition, we can build only standalone application.

- **J2EE (Java Enterprise Edition)**
  - ➢ J2EE is the enterprise edition which is used to desing web applications.
  - ➢ More library content.
  - ➢ Using Enterprise edition, we can build real time and dynamic web application.

- **SERVER:** It is a system which is used to perform that provides a service for the application for accepting client request and providing the client response.

  Ex: MySQL Server(Database), Tomcat Server(Application Server) etc..

- **HOST**: It is a platform which is used to run the server.
  - ➢ **Local Host:** The server will be limited to that particular system.
  - ➢ **Remote Host**: The server will be in the remote place and it is used to connect to that remote server.

- **Port Number:** It is a way to identify the specific server or to connect to a specific server.

  Ex: Oracle DB server – 1521 , MySQL – 3306, Apache Tomcat - 8080

- **URL (Uniform Resource Locator):** It specifies the address of a particular resource.

In JDBC, the URL pattern is :

main_protocol:sub_protocol://localhost/IP_address:portnumber/database_name?user=username&password=password;

Jdbc:mysql://localhost:3306/dbname?user=root&password=tiger;

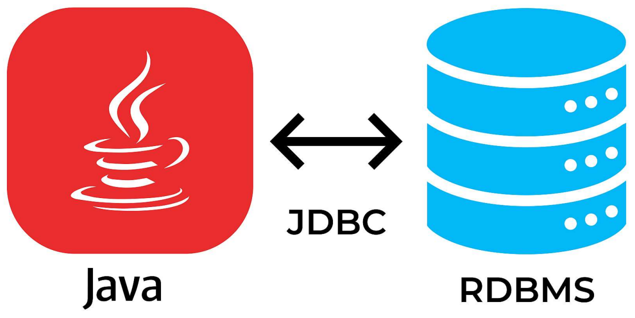
API (APPLICATION PROGRAMMING INTERFACE)

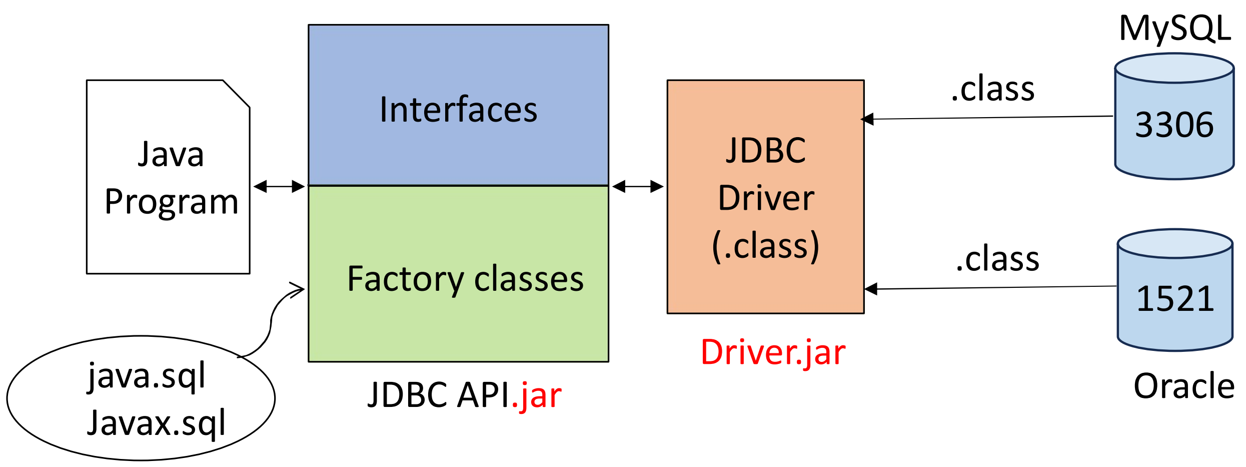
- API is used to establish connection between one application and another application.
- The backbone of API is Abstraction.
- The result of the API is loose coupling.

Ex: JDBC API, Apache Poi, Jexcel etc..

JDBC API

JDBC API is used to establish connection between Java and Database.



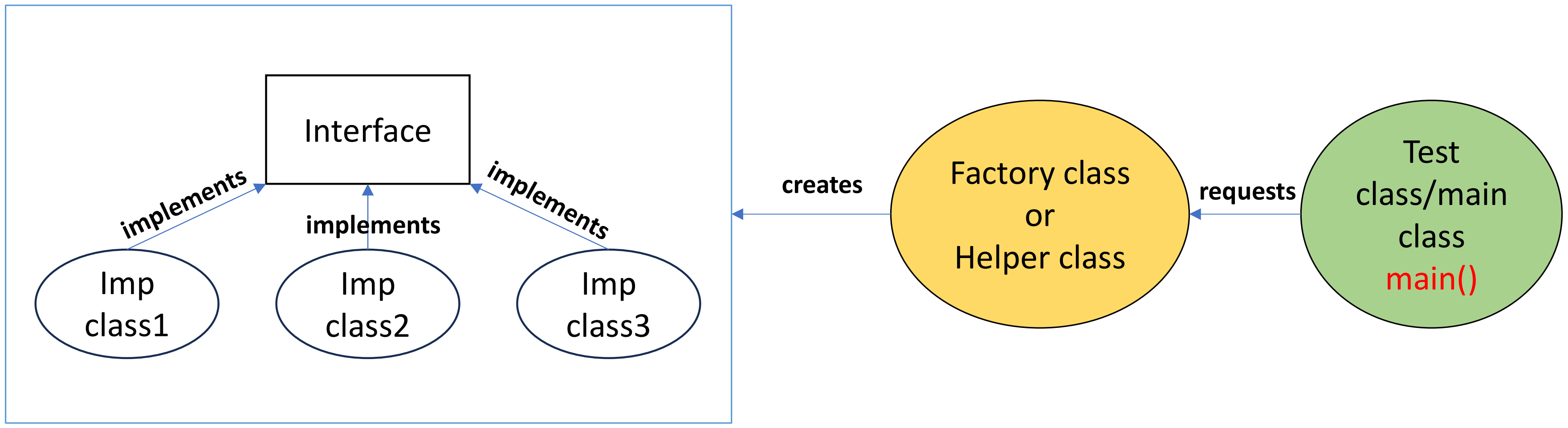


- JDBC API is given by Sun Microsystems and is available in JRE library in the form of jar file.
- JDBC API consists of two packages. They are
  - java.sql
  - javax.sql
- JDBC API consists of Interfaces and Factory classes.
- Implementation classes will be given by respective database servers or venders.
- These implementation classes given by database servers are known as JDBC Drivers.

PUNITH B

FACTORY DESIGN PATTERN

- Factory is used to create multiple objects of same type.
- Factory design pattern consists of three types of logics:
  - ➢ Implementation Logic (Interfaces and implementation classes)
  - ➢ Object Creational Logic (Factory Class)
  - ➢ Customer Utilization Logic (Test class or main class).



PUNITH B

```java
//Implementation Logic
public interface IPayment
{
   void doPayment();
}

public class UPI implements Ipayment
{
   @Override
   public void doPayment() {
   System.out.println("Payment successful through UPI");
   }
}

public class DebitCard implements Ipayment
{
   @Override
   public void doPayment() {
   System.out.println("Payment successful using Debit Card");
   }
}
```

```
public class CreditCard implements Ipayment
{
   @Override
   public void doPayment() {
   System.out.println("Payment successful using Credit Card");
   }
}


//Object Creational Logic
public class PaymentMode //Factory or helper class
{
  public static IPayment payment(String in)
  //factory or helper method
  {
    if(in.equalsIgnoreCase("UPI")){
    return new UPI();
    }
    else if(in.equalsIgnoreCase("Credit")){
    return new CreditCard();
    }
    else if(in.equalsIgnoreCase("Debit")){
    return new DebitCard();
    }
    else{
    System.err.println("No such payment option available");
     return null;
    }
  }
}
```

```java
//Customer Utilization Logic
public class Test {
  public static void main(String[] args) {
    Scanner sc=new Scanner(System.in);
    System.out.println("Enter the Payment mode:");
    String payment_mode=sc.next();
    IPayment ip=PaymentMode.payment(payment_mode);
    if(ip!=null)
    {
      ip.doPayment();
    }
  }
}
```

Output:

```
Enter the Payment mode:
credit
Payment successful using Credit Card

Enter the Payment mode:
cash
No such payment option available
```

PUNITH B