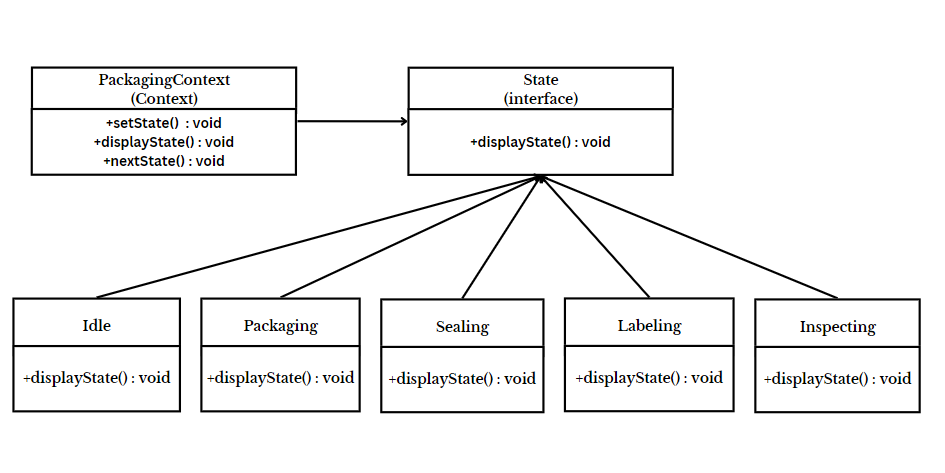
**State Design Pattern**

**Assignment - 2**

Name - Heet Dobariya Roll No. - 22BCP177 Group - G5

* **Program :** Implement state design pattern for Packaging example.
* **UML Diagram :**



* **Code :**

interface State

{

public void displayState();

}

class Idle implements State

{

public void displayState()

{

System.out.println("Waiting for the package to be processed.");

}

}

class Packaging implements State

{

public void displayState()

{

System.out.println("Item is packed.");

}

}

class Sealing implements State

{

public void displayState()

{

System.out.println("Item is sealed.");

}

}

class Labeling implements State

{

public void displayState()

{

System.out.println("Item is labelled.");

}

}

class Inspecting implements State

{

public void displayState()

{

System.out.println("Inspection of the item is done.");

}

}

class PackagingContext

{

private State currentState;

public PackagingContext()

{

currentState = new Idle();

}

public void setState(State state)

{

currentState = state;

}

public void displayState()

{

currentState.displayState();

}

// This method will give the next state

public void nextState() {

if (currentState instanceof Idle) {

System.out.println("Next state will be: Packaging");

}

else if (currentState instanceof Packaging) {

System.out.println("Next state will be: Sealing");

}

else if (currentState instanceof Sealing) {

System.out.println("Next state will be: Labeling");

}

else if (currentState instanceof Labeling) {

System.out.println("Next state will be: Inspecting");

}

else if (currentState instanceof Inspecting) {

System.out.println("Next state will be: Completed");

}

else {

System.out.println("Invalid state");

}

}

}

class PackagingState

{

public static void main(String[] args)

{

PackagingContext pc = new PackagingContext();

pc.displayState();

pc.setState(new Packaging());

pc.displayState();

pc.nextState();

pc.setState(new Sealing());

pc.displayState();

pc.nextState();

pc.setState(new Labeling());

pc.displayState();

pc.nextState();

pc.setState(new Inspecting());

pc.displayState();

}

}

* **Output :**

