**Lab Exercise 3- Exploring RHS (Then) Syntax in Drools**

**Objective:**

* To create and experiment with various RHS actions in Drools rules, including modifying facts, inserting new facts, and calling external methods.

**Prerequisites:**

* A basic understanding of Drools and Java.
* Completion of previous lab exercises on Drools, especially the one focused on the LHS (When) syntax.

**Step 1: Set Up the Project**

Use the existing project setup from previous exercises or set up a new Drools project using Maven.

**Step 2: Define a New Java Model**

Create a Customer Class:

In the src/main/java/com/example/model directory, create a Customer.java file with the following content:

package com.example.model;

public class Customer {

private String name;

private int loyaltyPoints;

private String membership;

public Customer(String name, int loyaltyPoints, String membership) {

this.name = name;

this.loyaltyPoints = loyaltyPoints;

this.membership = membership;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public int getLoyaltyPoints() {

return loyaltyPoints;

}

public void setLoyaltyPoints(int loyaltyPoints) {

this.loyaltyPoints = loyaltyPoints;

}

public String getMembership() {

return membership;

}

public void setMembership(String membership) {

this.membership = membership;

}

@Override

public String toString() {

return "Customer{name='" + name + "', loyaltyPoints=" + loyaltyPoints + ", membership='" + membership + "'}";

}

}

**Step 3: Create DRL Rules to Explore RHS Syntax**

Create the Rule File:

In the src/main/resources directory, create a file named customerRules.drl with the following content:

package com.example.rules

rule "Upgrade to Gold Membership"

when

$customer : Customer(loyaltyPoints > 1000, membership == "Silver")

then

$customer.setMembership("Gold");

System.out.println("Upgraded to Gold Membership: " + $customer);

update($customer);

end

rule "Add Loyalty Points"

when

$customer : Customer(membership == "Gold")

then

$customer.setLoyaltyPoints($customer.getLoyaltyPoints() + 100);

System.out.println("Added 100 loyalty points to Gold Member: " + $customer);

update($customer);

end

rule "Insert a New Customer"

when

$customer : Customer(loyaltyPoints < 500)

then

Customer newCustomer = new Customer("NewCustomer", 300, "Bronze");

System.out.println("Inserting new customer: " + newCustomer);

insert(newCustomer);

end

rule "Call an External Method"

when

$customer : Customer(membership == "Bronze")

then

sendPromotionEmail($customer);

end

function void sendPromotionEmail(Customer customer) {

System.out.println("Sending promotion email to: " + customer.getName());

}

**Explanation of the Rules:**

* Upgrade to Gold Membership: Upgrades a customer from Silver to Gold membership if they have more than 1000 loyalty points.
* Add Loyalty Points: Adds 100 loyalty points to customers who are Gold members.
* Insert a New Customer: Inserts a new customer with a Bronze membership if an existing customer has fewer than 500 loyalty points.
* Call an External Method: Calls an external method to send a promotional email to customers with a Bronze membership.

**Step 4: Update the Main Application**

Modify the DroolsTest Class:

Update the DroolsTest.java file to include the new Customer class:

package com.example;

import com.example.model.Customer;

import org.kie.api.KieServices;

import org.kie.api.runtime.KieContainer;

import org.kie.api.runtime.KieSession;

public class DroolsTest {

public static void main(String[] args) {

// Load the knowledge base

KieServices ks = KieServices.Factory.get();

KieContainer kContainer = ks.getKieClasspathContainer();

KieSession kSession = kContainer.newKieSession("ksession-rules");

// Create sample customers

Customer customer1 = new Customer("Alice", 1200, "Silver");

Customer customer2 = new Customer("Bob", 800, "Gold");

Customer customer3 = new Customer("Charlie", 300, "Bronze");

// Insert the customers into the session

kSession.insert(customer1);

kSession.insert(customer2);

kSession.insert(customer3);

// Fire all rules

kSession.fireAllRules();

// Dispose the session

kSession.dispose();

}

}

**Step 5: Run the Application**

**Compile and Run:**

Compile and run the DroolsTest class as described in previous exercises.

**Expected Output:**

The console should display output similar to the following:

Upgraded to Gold Membership: Customer{name='Alice', loyaltyPoints=1200, membership='Gold'}

Added 100 loyalty points to Gold Member: Customer{name='Alice', loyaltyPoints=1300, membership='Gold'}

Added 100 loyalty points to Gold Member: Customer{name='Bob', loyaltyPoints=900, membership='Gold'}

Inserting new customer: Customer{name='NewCustomer', loyaltyPoints=300, membership='Bronze'}

Sending promotion email to: Charlie

Sending promotion email to: NewCustomer