```
Yesterday's session : ED-RULES-API
-----------------------------------
-> ED-RULES-API is used to execute business rules for the plan

-> Every plan will have some business rules and those business rules will decide
citizen is eligible for plan or not.

-> ED-RULES-API will take "citizen data" as input and provide "plan data" as output.

Note: Citizens who got approved for the plan will get benefit amount from govt.
-------------------------------------------------------------------------

-> In yesterday's session we have implemented business rules in java program which is
not recommended.

-> If want to add/remove/change any rule then it will become difficult. Maintenence
cost will be increased for rules add/remove/modify.

-> To ovecome these problems, in realtime people will use 'RULE Engines'.

-> By using 'RULE ENGINES' we can seperate java logic and business rules.

-> Below are the famous 'Rule Engines' available in market

                1) OPA (Commercial)
                2) JBOSS DROOLS (Open Source)


-----------------------------------------------------------------

jboss DRL syntax
-------------------------------------------------------------------

rule  "rule-name-1"
        when
                //rule logic
        then
                //action
end

rule "rule-name-2"
        when
                //rule logic
        then
                //action
end

==================================================================
Sample discount application
==================================================================
@Data
public class Product {

        private String type;
        private int discount;
}
----------------------rules.drl------------------------------------
package com.rule;

import com.ashok.model.Product
```

```
rule "Offer for Diamond"
        when
                productObject: Product(type=="diamond")
        then
                productObject.setDiscount(10);
        end

rule  "Offer for Gold"
         when
                productObject: Product(type=="gold")
         then
                productObject.setDiscount(5);
         end

rule  "Offer for Silver"
         when
                productObject: Product(type=="silver")
         then
                productObject.setDiscount(4);
         end
-----------------------RulesTest.java--------------------------------
public class DroolsTest {

        public static void main(String[] args) throws Exception {
                DroolsTest test = new DroolsTest();
                test.executeDrools();
        }

        public void executeDrools() throws DroolsParserException, IOException {
                Product product = new Product();
                product.setType("silver");

                String ruleFile = "/com/rule/Rules.drl";
                InputStream resourceAsStream =
getClass().getResourceAsStream(ruleFile);
                Reader reader = new InputStreamReader(resourceAsStream);

                PackageBuilder packageBuilder = new PackageBuilder();
                packageBuilder.addPackageFromDrl(reader);

                org.drools.core.rule.Package rulesPackage =
packageBuilder.getPackage();
                RuleBase ruleBase = RuleBaseFactory.newRuleBase();
                ruleBase.addPackage(rulesPackage);

                WorkingMemory workingMemory = ruleBase.newStatefulSession();
                workingMemory.insert(product);
                workingMemory.fireAllRules();

                System.out.println("The discount for the product " + product.getType()
+ " is " + product.getDiscount());
        }
}
------------------------------------------------------------------------
```