

Modifying DB App

This document explains about modifying database application such as changing prompts, UI controls etc.

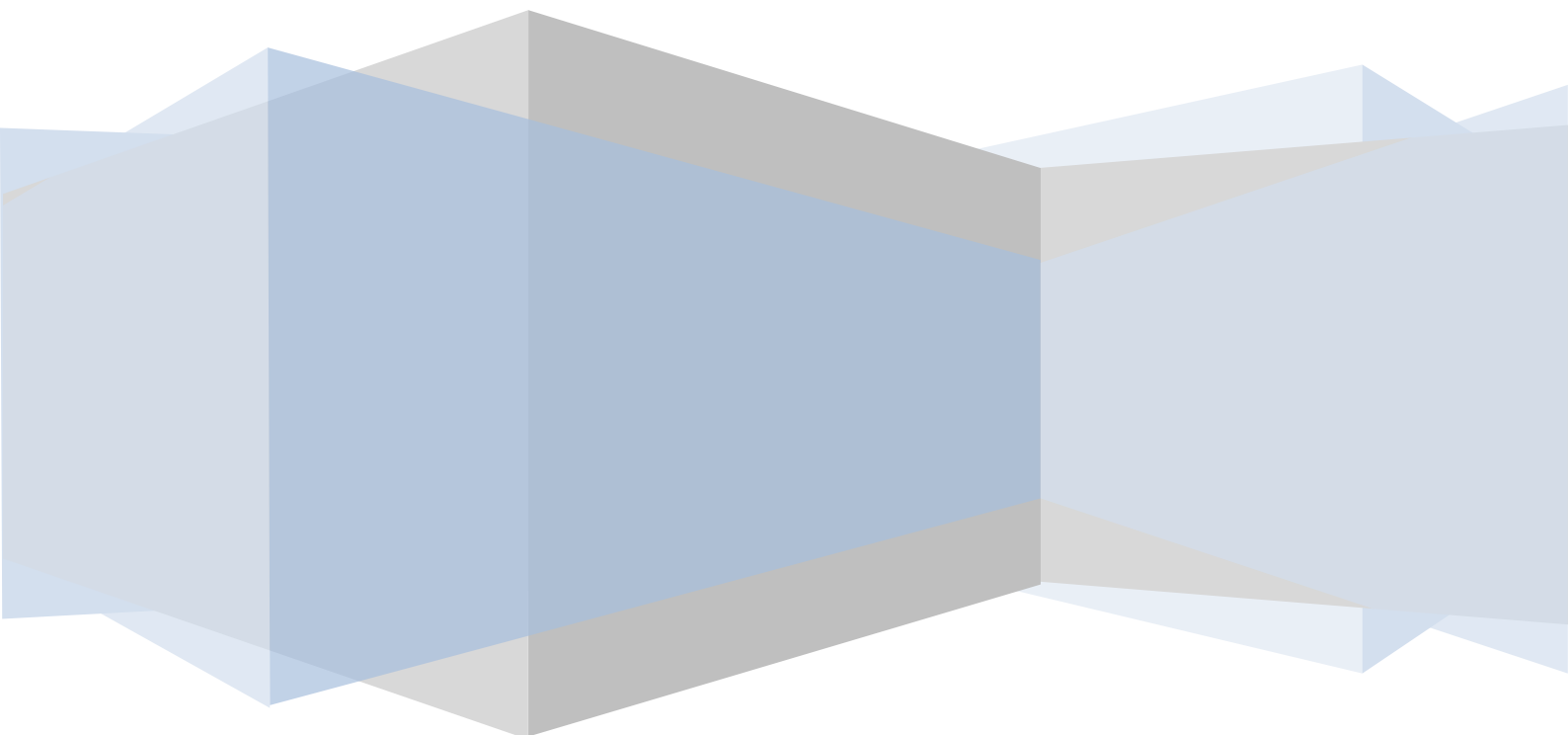


Table of Contents

1. Modifying the Application.....	3
1.1 Adding Referential Integrity	3
1.2 Changing Variable Prompt	8
1.3 Changing UI Controls	16
1.3.1 Changing Control	16
1.4 Formatting the Form	19
1.5 Adding Row Condition.....	23
1.6 Adding Rules	31
1.7 Adding Dynamic Menu	55

1. Modifying the Application

Modifying the application includes adding and modifying rules, changing prompts, removing back button from the web page, adding referential integrity and also hiding controls in form with conditions etc.

1.1 Adding Referential Integrity

Referential integrity can be added using *DB Schema Manager* interface in database interfaces.

For example: In *customer* table set *AccountID* as a foreign key which is a primary key in *account* table.

1. Go to *Main Menu -> Database Interfaces -> DB Schema Manager* and click *Add/Show* button of *Table Schema* panel.

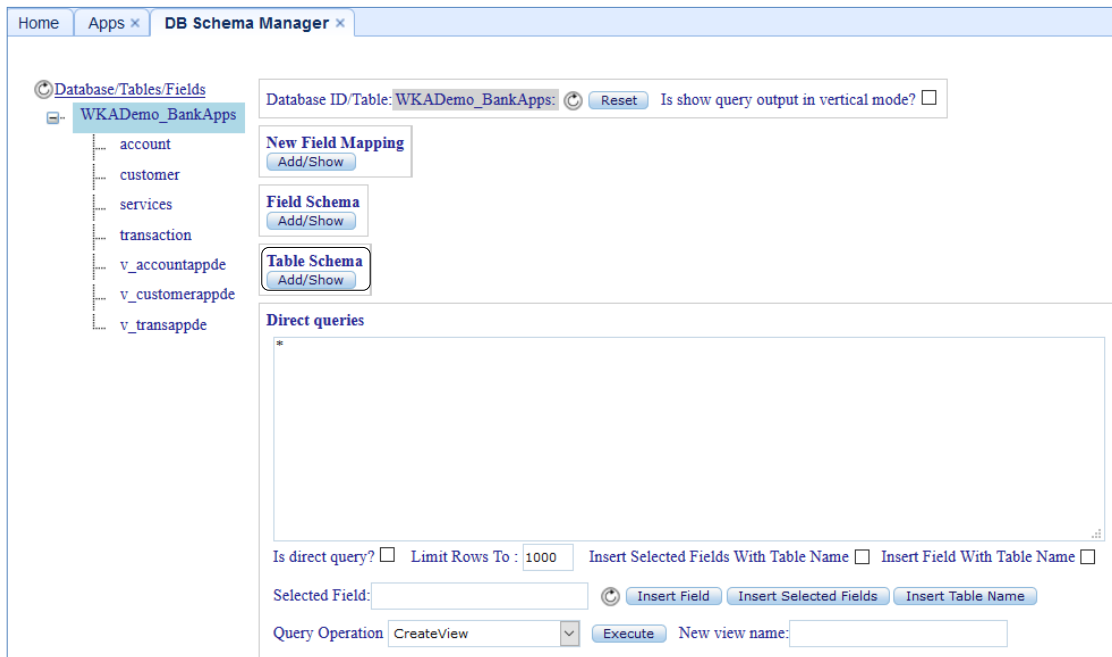


Figure 1: DB Schema Manager Interface

2. After clicking *Add/Show*, the following screen is displayed.

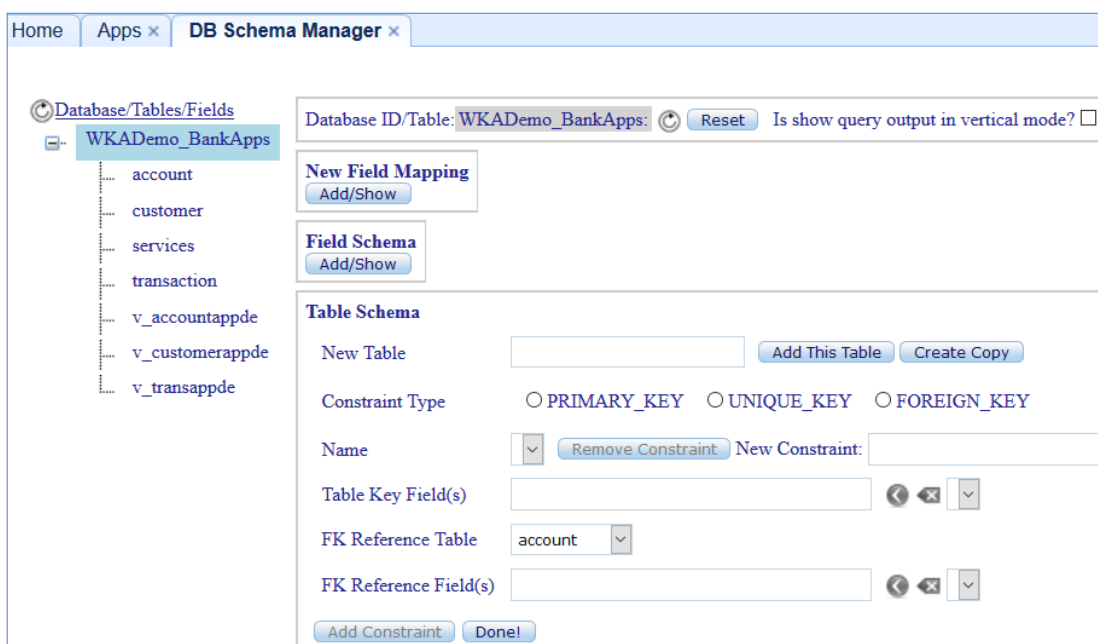


Figure 2: Table Schema Panel

3. Select the *Constraint Type* as *Foreign Key* and enter the constraint name (Example: *FK_CustomerAccount*) in *New Constraint* field.

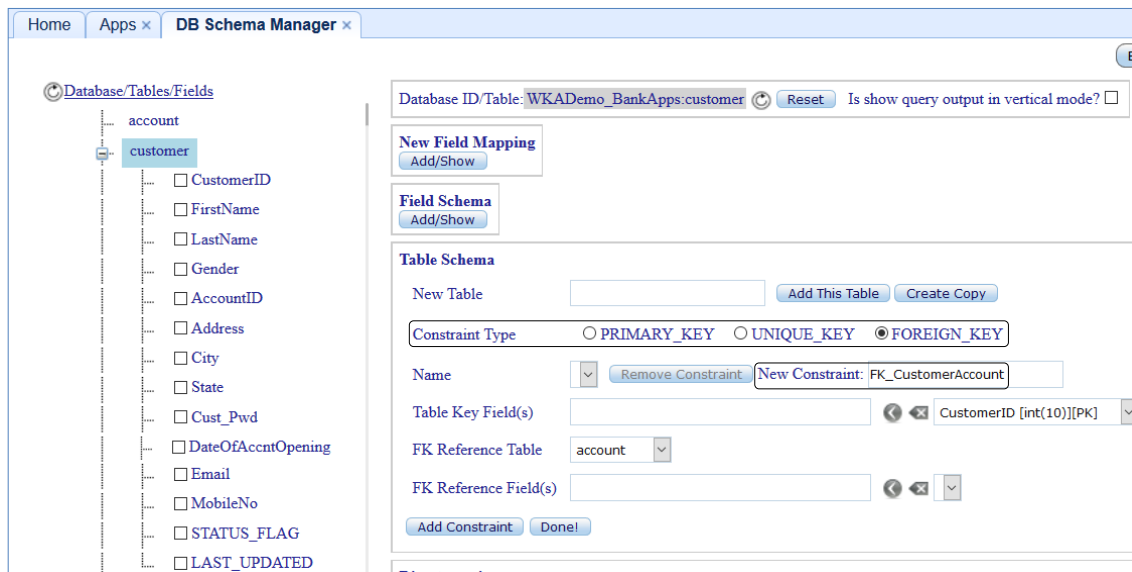


Figure 3: Selecting Foreign Key Constraint and Entering Constraint Name

4. Select from *Table Key Field(s)* dropdown menu, the field (*AccountID [int(10)]*) to which foreign key needs to be set.

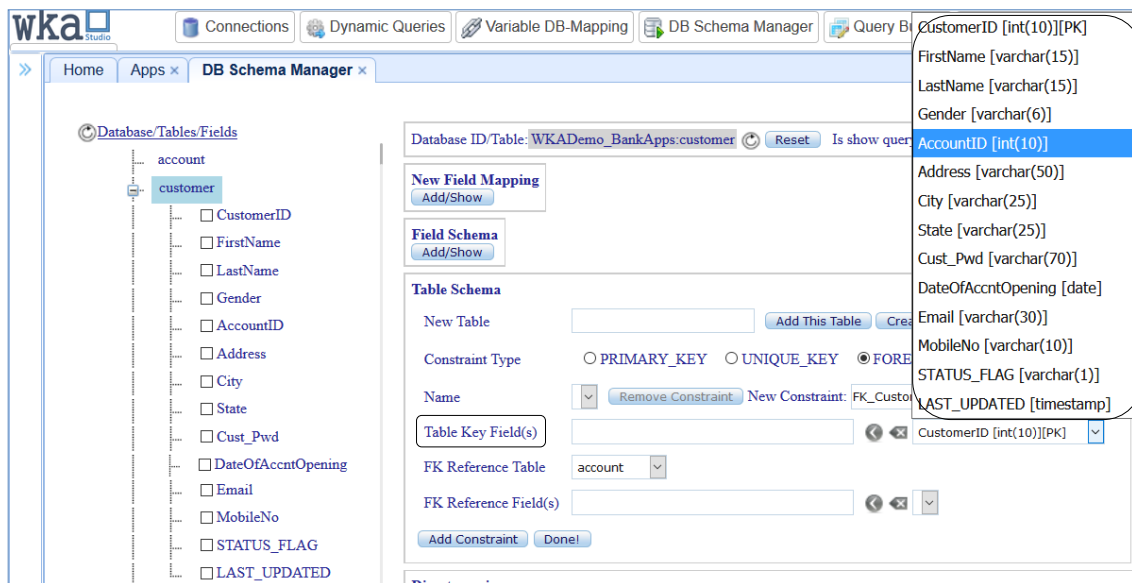


Figure 4: Selecting Table Key Field

5. To move the selected field in *Table Key Field(s)*, click the button .

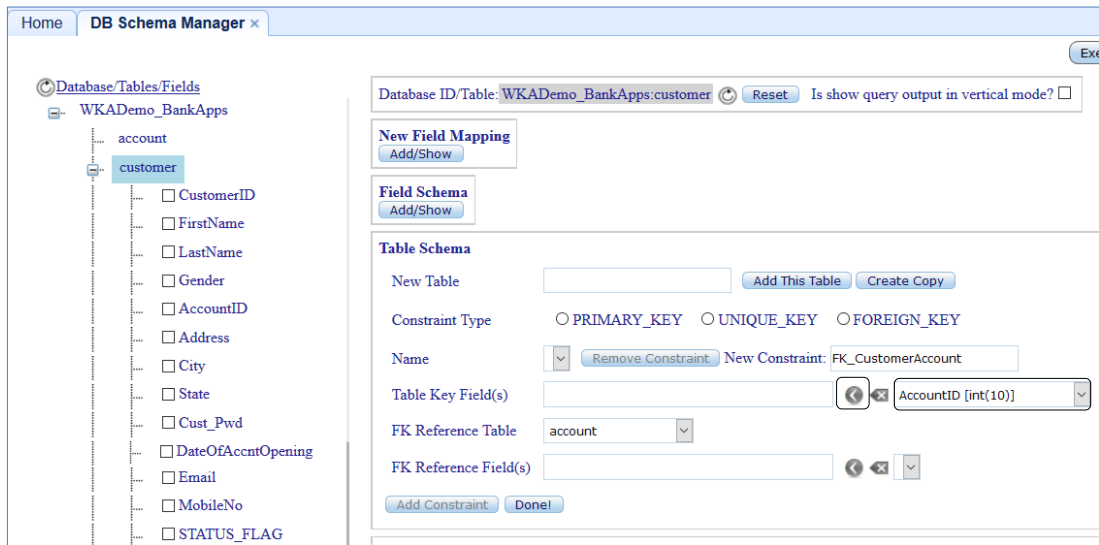


Figure 5: Moving Field Name

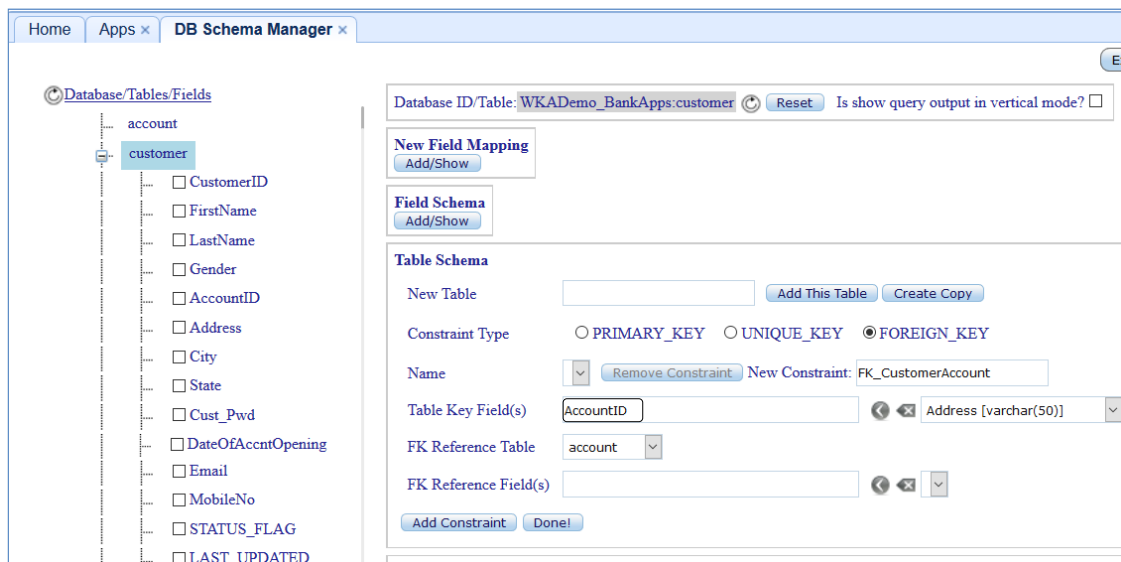


Figure 6: Field Name Moved

6. Select the table (*account*) from *FK Reference Table* dropdown menu.

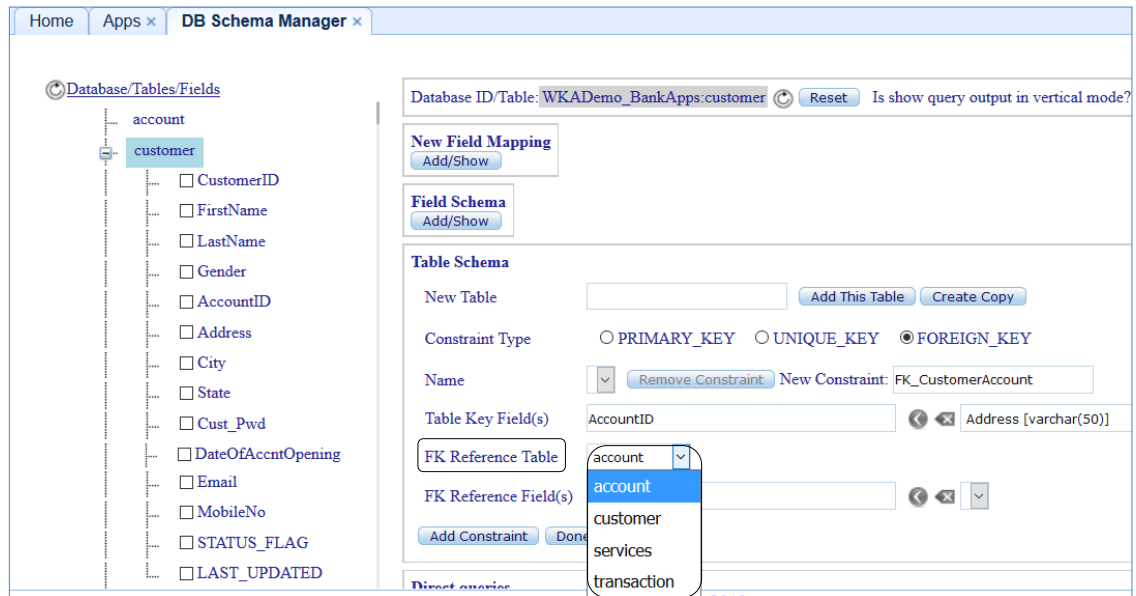


Figure 7: Selecting the Table

7. Select foreign key field (*AccountID[int(10)][PK]*) from *FK Reference Field* dropdown menu.

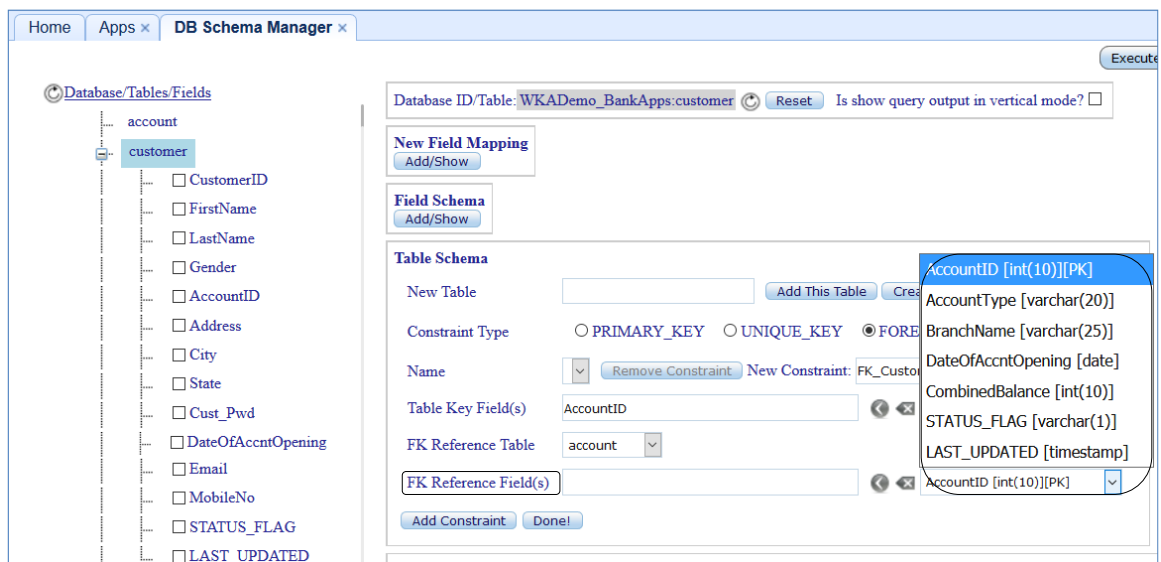



Figure 8: Selecting Foreign Key Field

8. To move the selected field in *FK Reference Key Field(s)*, click the button  and click *Add Constraint*.

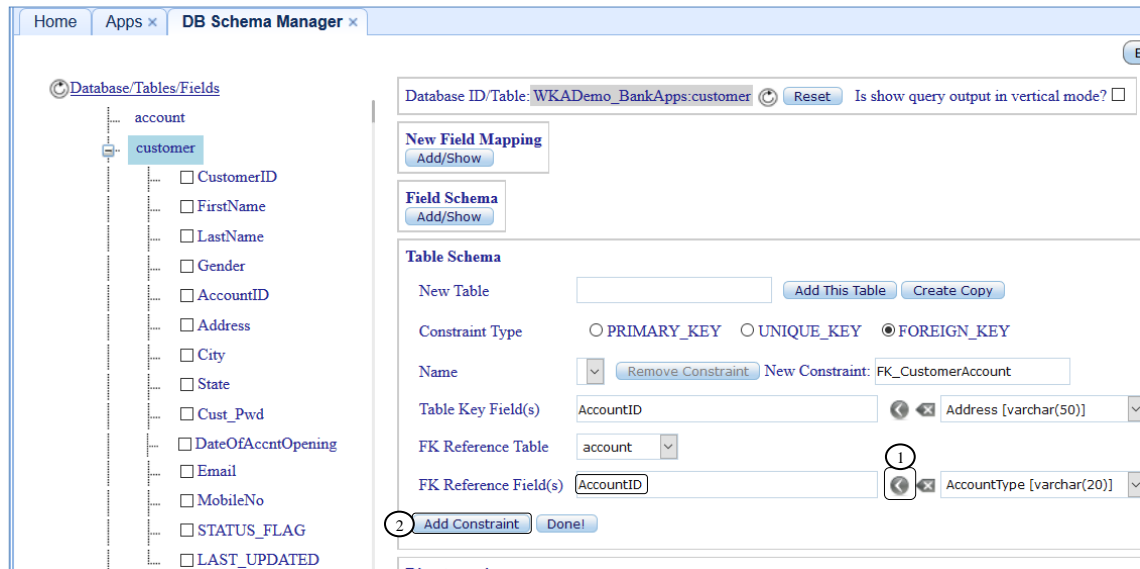


Figure 9: Moving Foreign Key Field

9. After clicking *Add Constraint*, foreign key constraint is added and the following message is displayed. Also, the constraint name entered (*FK_CustomerAccount*) gets added in *Name* dropdown menu.

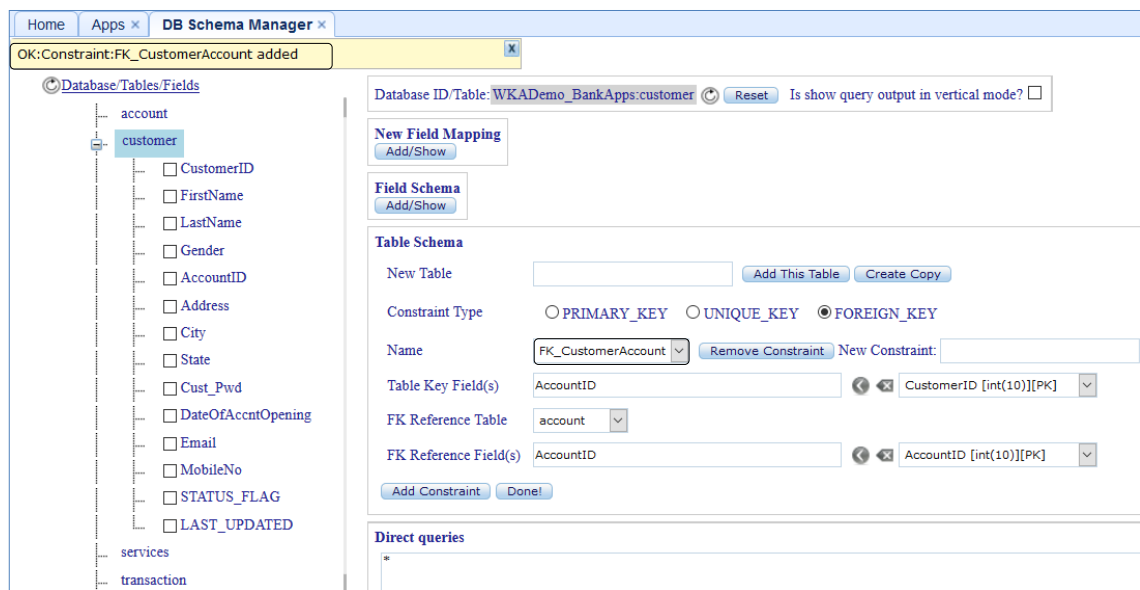


Figure 10: Foreign Key Constraint Added

1.2 Changing Variable Prompt

As per the requirement, variable prompt can be changed. As shown in figure below prompt for variable *CustomerID* is *Enter Customer ID*.

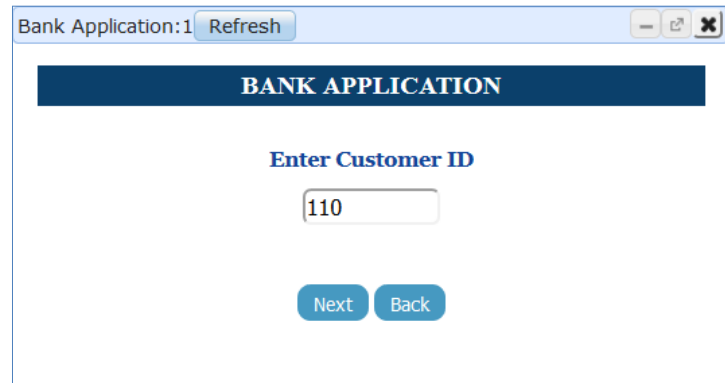


Figure 11: Prompt For Variable Customer ID

To change prompt for *Customer ID*:

1. Go to *Main Menu -> Domain Vocabulary -> Global Variables*. Select *CustomerID* variable from the list of variables in variable group *Customer* and click *HTML Web Page Options 1* from the list of options.

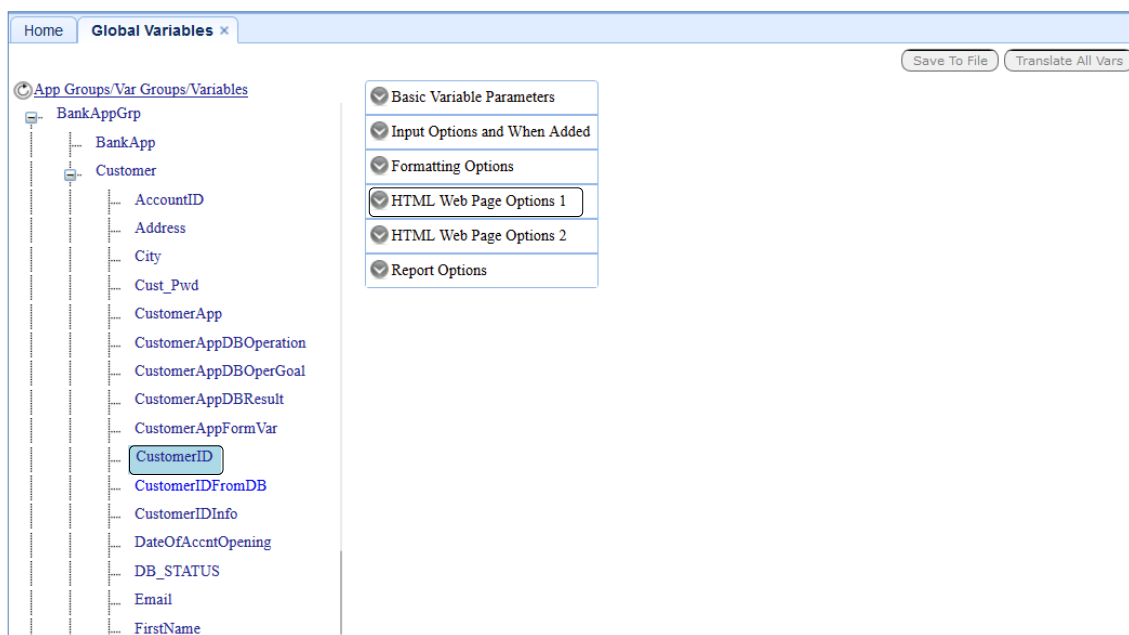


Figure 12: Global Variables Interface

2. After clicking *HTML Web Page Options 1*, the default prompt for *CustomerID* can be seen (*Enter Customer ID*).

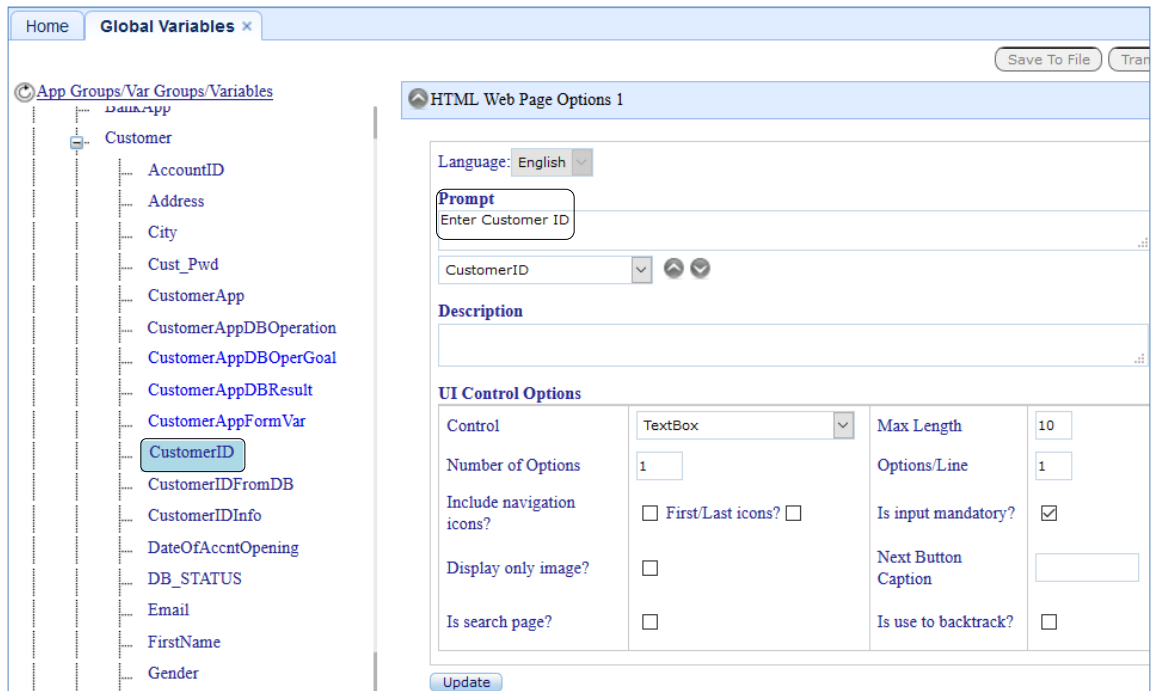


Figure 13: Default Prompt for Variable CustomerID

3. Edit the prompt as *Customer ID*.

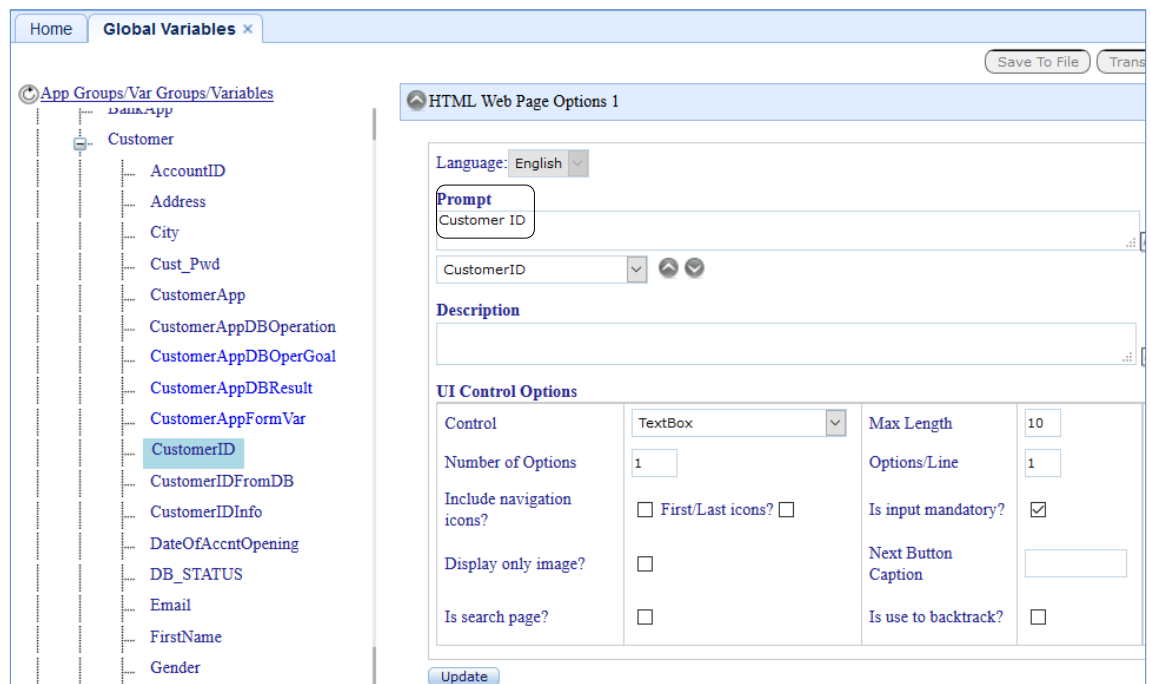


Figure 14: Editing Variable Prompt for CustomerID

4. To update the input options of variable, click *Update*.

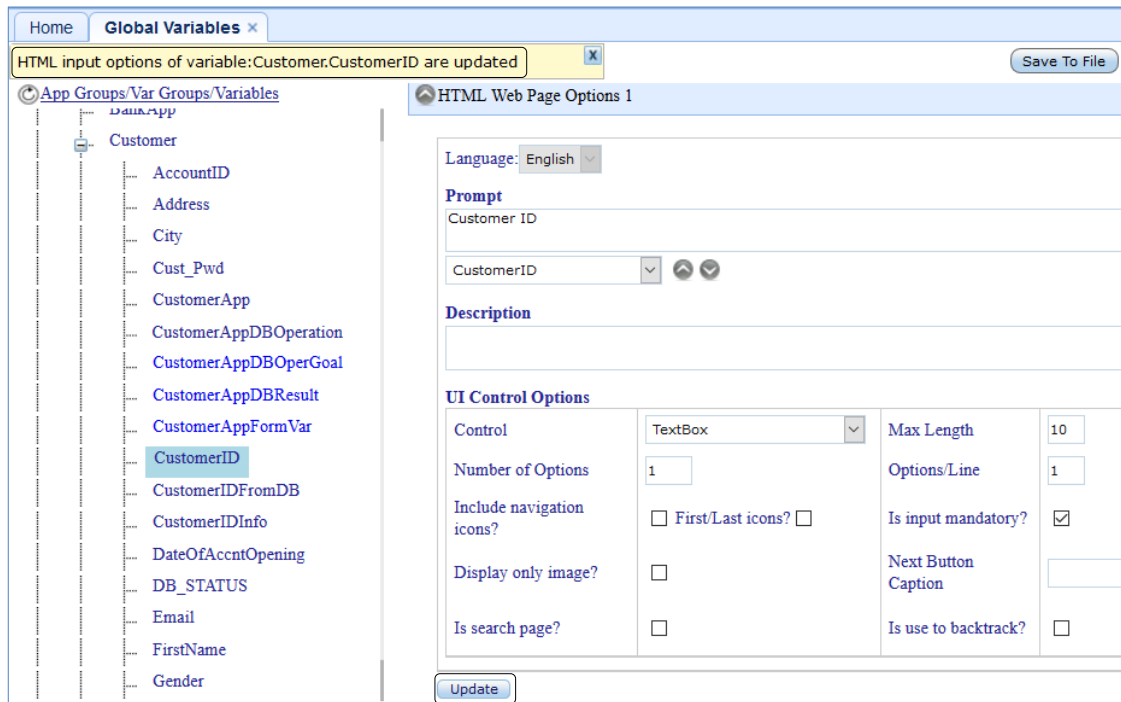


Figure 15: Updating the Variable

- To view the changes on web page, first a web page is created for variable *CustomerID*. To create a web page, go to *Main Menu -> Core Engine Configs -> Expert Systems*. Select *Customer* variable group from *Variables and Values* dropdown menu.

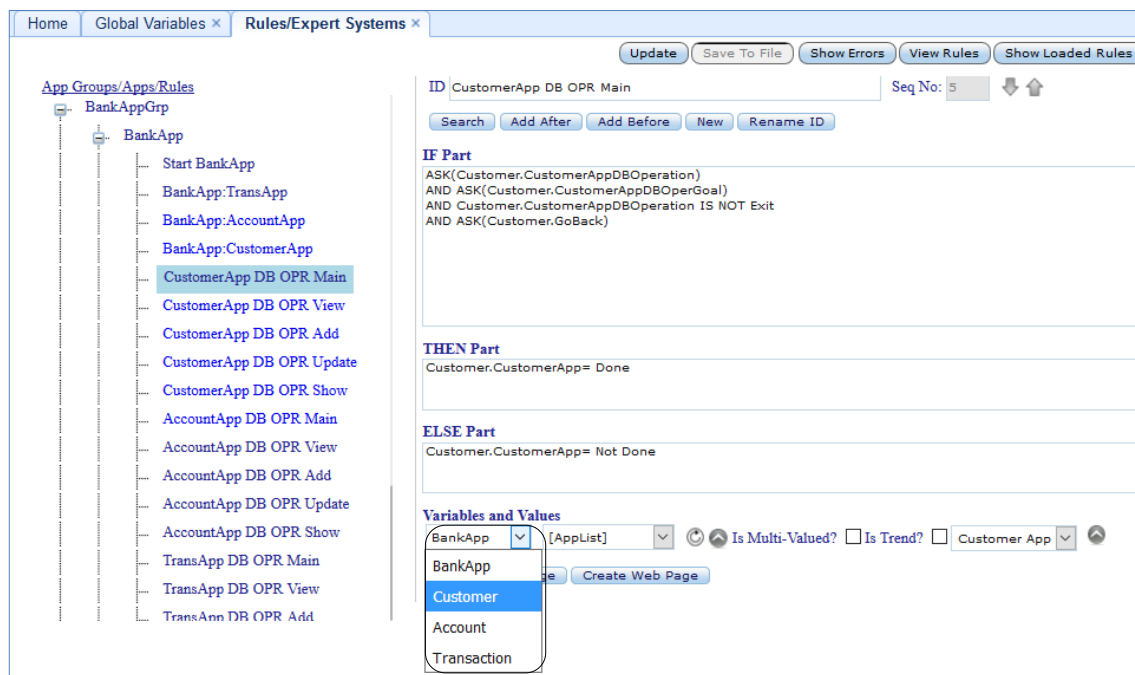


Figure 16: Selecting Variable Group

- Select the variable *CustomerID* from *Variables and Values* dropdown menu and click *Create Web Page*. The following message is displayed.

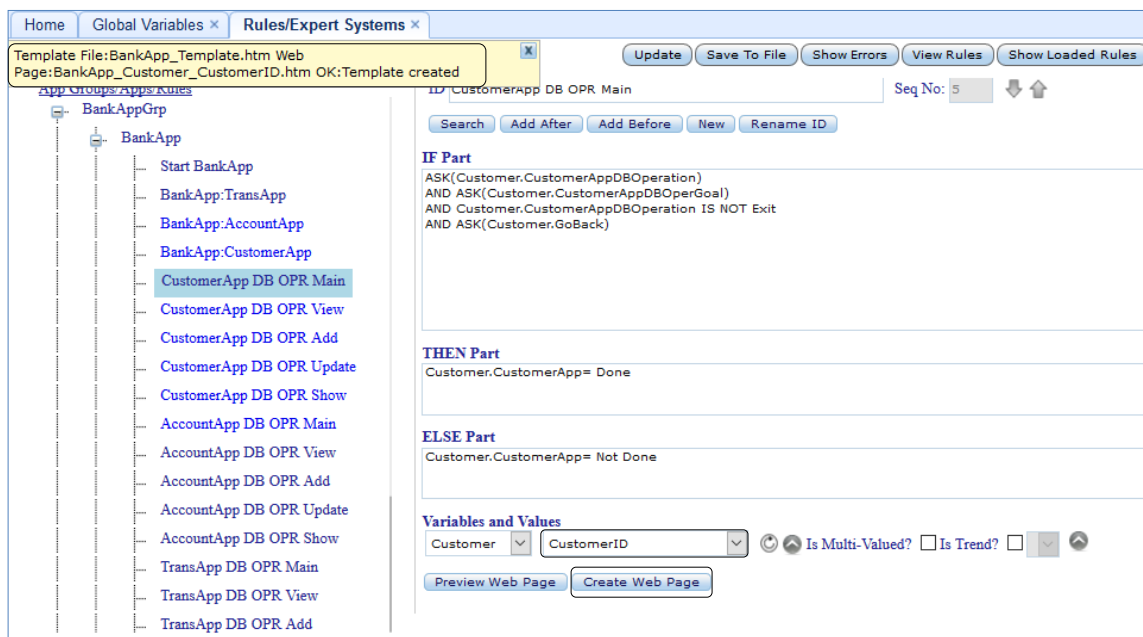


Figure 17: Creating Web Page

- To preview the changes, click *Preview Web Page*.

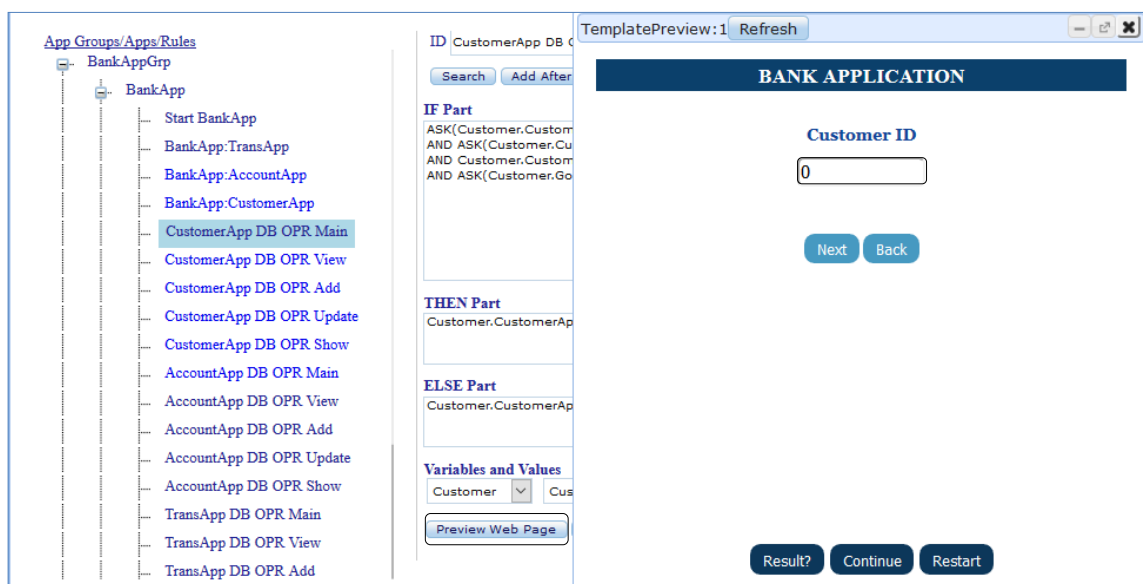


Figure 18: Previewing Web Page

- The variable *CustomerID* is present in form therefore need to recreate the form in which that variable is used otherwise changes will not be reflected in the form. To recreate the form, go to *Main Menu -> Forms and Reports -> Form Designer* and click the option *Select Form/Report* from the list of options. Select the form created (*FormCustomerApp*).

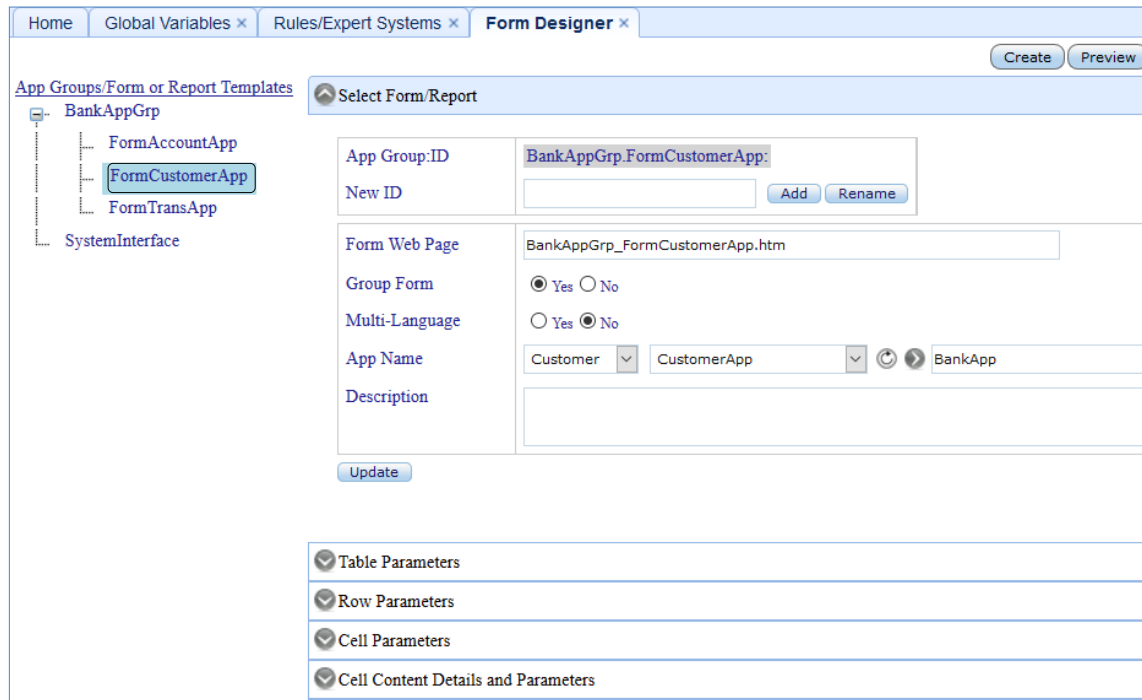


Figure 19: Selecting Form

7. To recreate the form, click *Create* and the following message is displayed.

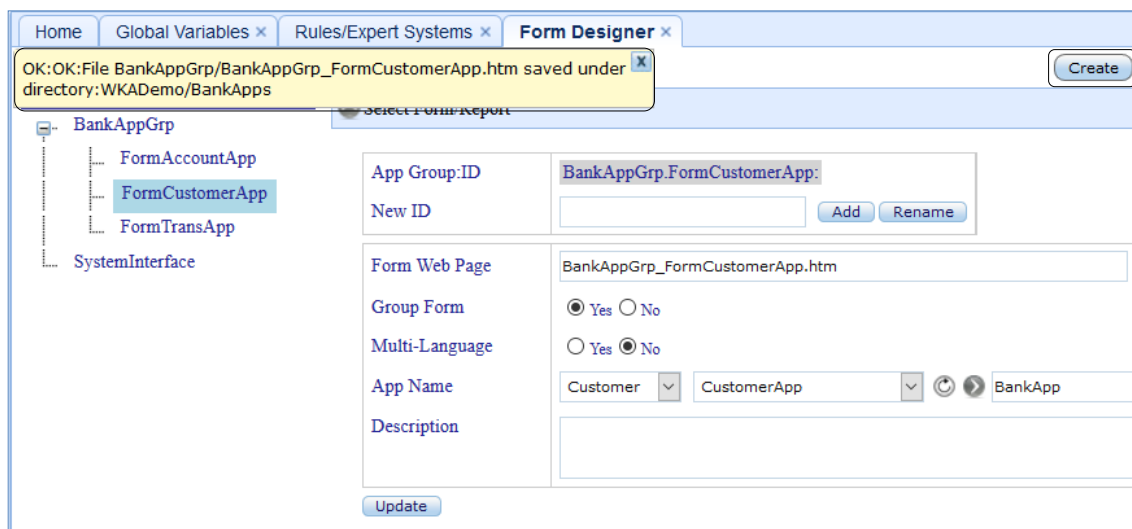


Figure 20: Form Recreated

8. After recreating the form, to view the changes run the application. To run the application, go to *Main Menu -> Apps, APIs and Docs -> Apps* and can view that the prompt has been changed for variable *CustomerID*.

The screenshot shows a dark blue header with the text 'BANK APPLICATION'. Below the header, the text 'Customer ID' is displayed in bold. Underneath, there is a text input field containing the number '110'. At the bottom of the form, there are two blue buttons: 'Next' and 'Back'.

Figure 21: Variable Prompt Changed

9. To view the changes in form, click *Next* in above screen.

The screenshot shows a dark blue header with the text 'BANK APPLICATION'. Below the header, the text 'Customer Data' is displayed in bold. The form contains several fields: 'Customer ID' (110), 'Enter First Name' (Rishi), 'Enter Last Name' (Verma), 'Select Gender' (radio buttons for Female and Male, with Male selected), 'Enter Account ID' (10), 'Enter Address' (Parvati Heritage Bandra), 'Select City' (Mumbai), 'Select State' (radio buttons for Maharashtra, Karnataka, and Gujrat, with Maharashtra selected), 'Enter Cust_Pwd' (masked with dots), and 'Enter Date Of Acct Opening' (02/01/2016). At the bottom, there are three blue buttons: 'Result?', 'Continue', and 'Restart'.

Figure 22: Variable Prompt Changed in Form

10. Similarly, to change the prompts of all variables for *Customer* variable group follow the same process as explained in section above and all variable prompts are changed as shown in screen shots below.

BANK APPLICATION

Customer Data

Customer ID	110
First Name	Rishi
Last Name	Verma
Gender	<input type="radio"/> Female <input checked="" type="radio"/> Male
Account ID	10
Address	Parvati Heritage Bandra
City	Mumbai
State	<input checked="" type="radio"/> Maharashtra <input type="radio"/> Karnataka <input type="radio"/> Gujrat
Password	••••••••
Date Of A/C Opening	02/01/2016

Result? Continue Restart

Figure 23: All Variable Prompts Changed for Customer

BANK APPLICATION

Gender	<input type="radio"/> Female <input checked="" type="radio"/> Male
Account ID	10
Address	Parvati Heritage Bandra
City	Mumbai
State	<input checked="" type="radio"/> Maharashtra <input type="radio"/> Karnataka <input type="radio"/> Gujrat
Password	••••••••
Date Of A/C Opening	02/01/2016
Email	rishi12@gmail.com
Mobile No	9865234589
CustomerIDInfo	Press 'Back' button to view another record

Next Back

Result? Continue Restart

Figure 24: All Variable Prompts Changed for Customer

Note: Similarly, to change the prompts of all variables for Account and Transaction variable group follow the same process as explained in section 1.2 above and all variable prompts are changed as shown in screen shots below.

BANK APPLICATION

Account ID

1

Next Back

Figure 25: Prompt Changed for Account

BANK APPLICATION	
Account Data	
Account ID	1
Account Type	<input checked="" type="radio"/> Savings Account <input type="radio"/> Current Account <input type="radio"/> Recurring Deposit <input type="radio"/> Fix Deposit
Branch Name	Hinjewadi
Date Of A/C Opening	23/07/2011
Combined Balance	60500
AccountIDInfo	Press 'Back' button to view another record
<input type="button" value="Next"/> <input type="button" value="Back"/>	
<input type="button" value="Result?"/> <input type="button" value="Continue"/> <input type="button" value="Restart"/>	

Figure 26: All Prompts Changed for Account

BANK APPLICATION	
Bank Application:1 <input type="button" value="Refresh"/>	
Account Trans ID	
1	
<input type="button" value="Next"/> <input type="button" value="Back"/>	

Figure 27: Prompt Changed for Transaction

BANK APPLICATION	
Transaction Data	
Account Trans ID	1
Customer ID	102
Account ID	5
Transaction Type	<input checked="" type="radio"/> Debit <input type="radio"/> Credit
Date Of Transaction	12/05/2015
Amount	5000
Mode	<input checked="" type="radio"/> Cash Withdrawal <input type="radio"/> Cheque
AccountTransIDInfo	Press 'Back' button to view another record
<input type="button" value="Next"/> <input type="button" value="Back"/>	
<input type="button" value="Result?"/> <input type="button" value="Continue"/> <input type="button" value="Restart"/>	

Figure 28: All Prompts Changed for Transaction

1.3 Changing UI Controls

HTML or UI controls for the variable can be changed. It includes changing control, control size, maximum length of characters, number of options per line etc.

1.3.1 Changing Control

Controls can be changed for variable. For example: In the screen shot below *State* variable has *RadioBoxList* control which can be changed to *ComboList* control. To change the control:

1. Go to *Main Menu -> Domain Vocabulary ->Global Variables* and select *HTML Web Page Options 1* from the list of options. Select *State* variable from variable group *Customer*.

The screenshot shows a web application interface with a breadcrumb trail: Home > Global Variables > HTML Web Page Options 1. On the left, a tree view under 'App Groups/Var Groups/Variables' lists various variables, with 'State' selected. The main panel displays the configuration for 'State' under 'HTML Web Page Options 1'. It includes a 'Language' dropdown set to 'English', a 'Prompt' field containing 'State', and a 'State' dropdown menu with navigation arrows. Below this is a 'Description' field. The 'UI Control Options' section contains a table with the following data:

Control	RadioBoxList	Max Length
Number of Options	1	Options/Line
Include navigation icons?	<input type="checkbox"/> First/Last icons? <input type="checkbox"/>	Is input mandatory?
Display only image?	<input type="checkbox"/>	Next Button Caption
Is search page?	<input type="checkbox"/>	Is use to backtrack?

An 'Update' button is located at the bottom of the configuration panel.

Figure 29: HTML Web Page Options 1 Interface

2. To change the control to *ComboList*, select from *Control* dropdown menu, *ComboList*.

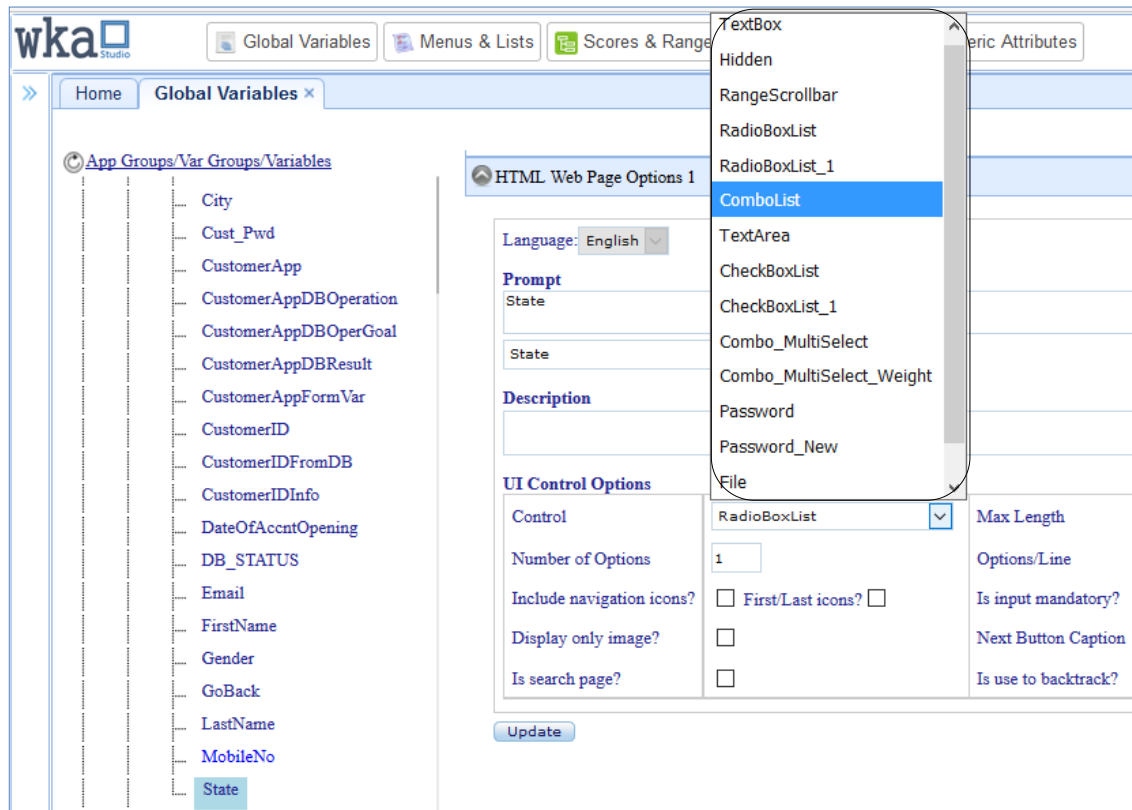


Figure 30: Selecting Control

3. To update the input options of variable, click *Update* and the following message is displayed.

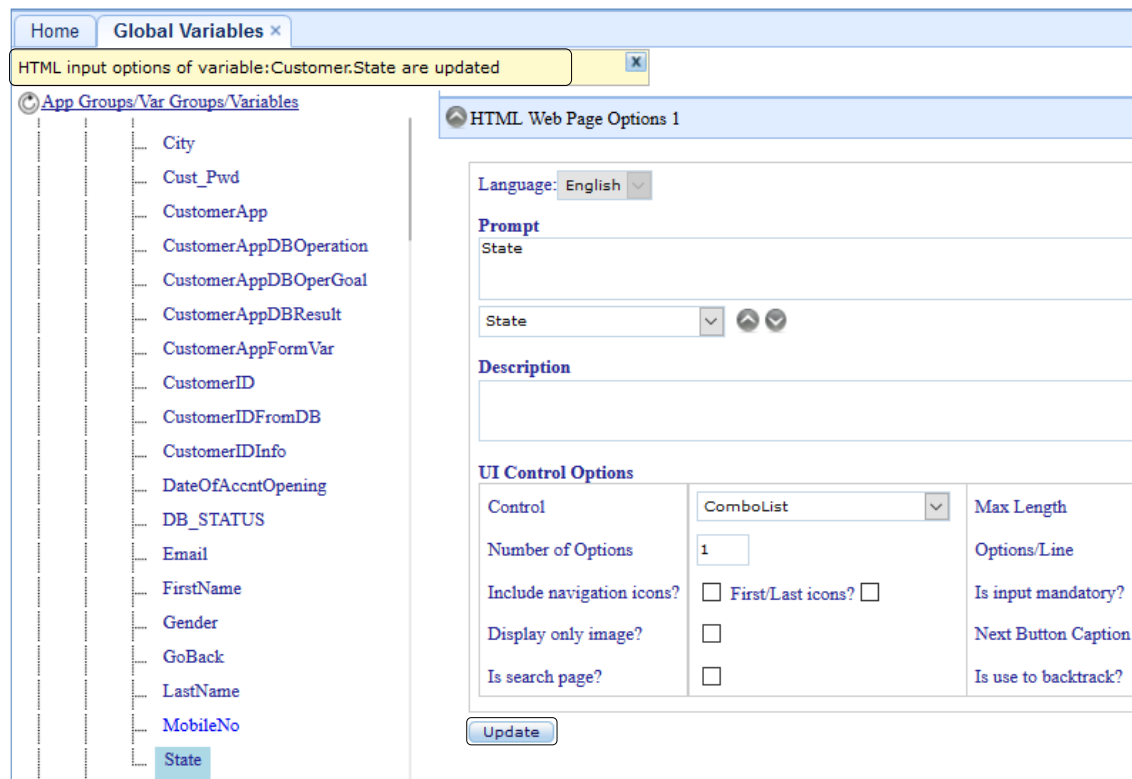


Figure 31: Updating Variable

4. Recreate the form for variable *State* from *Forms and Reports* interface as explained in section 1.2 above (Refer Figure 19 and Figure 20).
5. After recreating the form, to view the changes run the application. To run the application, go to *Main Menu -> Apps, APIs and Docs -> Apps* and can view that the control has been changed for variable *State*.

The screenshot shows a web form titled "BANK APPLICATION" with a section for "Customer Data". The fields are as follows:

Customer ID	110
First Name	Rishi
Last Name	Verma
Gender	<input type="radio"/> Female <input checked="" type="radio"/> Male
Account ID	10
Address	Parvati Heritage Bandra
City	Mumbai
State	Maharashtra
Password	••••••••
Date Of A/C Opening	02/01/2016
Email	rishi12@gmail.com
Mobile No	9865234589

At the bottom of the form are three buttons: "Result?", "Continue", and "Restart".

Figure 32: Control for State Changed

6. Similarly, to change the control of *Account Type* variable to *ComboList* for *Account App* follow the same process as explained in section 1.3 above and control is changed as shown in screen shots below.

The screenshot shows a web form titled "BANK APPLICATION" with a section for "Account Data". The fields are as follows:

Account ID	1
Account Type	Savings Account
Branch Name	Hinjewadi
Date Of A/C Opening	23/07/2011
Combined Balance	60500
AccountIDInfo	Press 'Back' button to view another record

Below the "AccountIDInfo" field are two buttons: "Next" and "Back". At the bottom of the form are three buttons: "Result?", "Continue", and "Restart".

Figure 33: Control for Account Type Changed

1.4 Formatting the Form

Formatting of the form including alignment of the variables and labels, font size, font type, font colour, heading size, text alignment, table properties etc. can be set. For Example: In the screen shot below, label for form is *Customer Data* and its alignment can be set to center.

Customer Data	
Customer ID	110
First Name	Rishi
Last Name	Verma
Gender	<input type="radio"/> Female <input checked="" type="radio"/> Male
Account ID	10
Address	Parvati Heritage Bandra
City	Mumbai
State	Maharashtra
Password	••••••••
Date Of A/C Opening	02/01/2016
Email	rishi12@gmail.com
Mobile No	9865234589

Figure 34: Aligning the Label to Center

To align the label *Customer Data* to center, follow the following steps:

1. Go to *Main Menu ->Forms and Reports->Form Designer*. Select the form *FormCustomerApp* and click *Cell Content Details and Parameters* from the list of options.

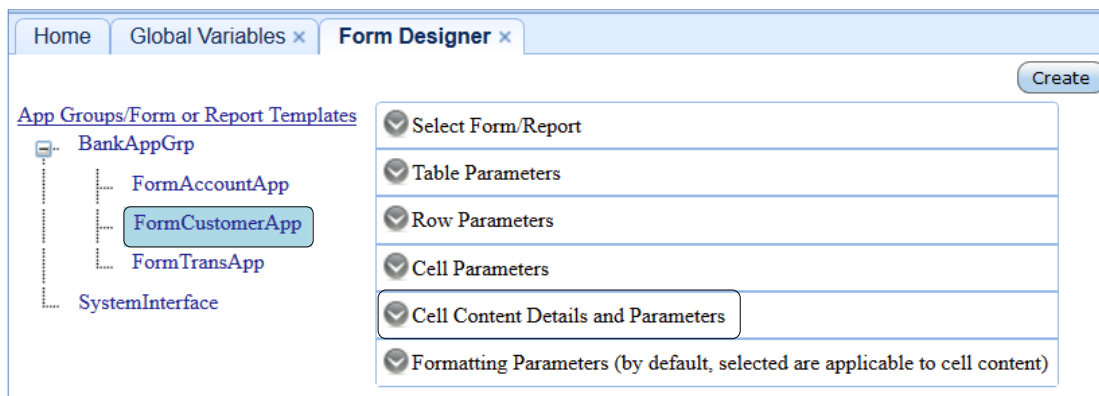


Figure 35: Selecting the Form

2. Select the table, row and element (column) to modify from *Table: Row: Element* dropdown menu. Since the label *Customer Data* which needs to be changed is in first row and first column in the form (Refer Figure 34 above) so select Table as T1 (if there are multiple tables), Row as 1 and Element as 1.

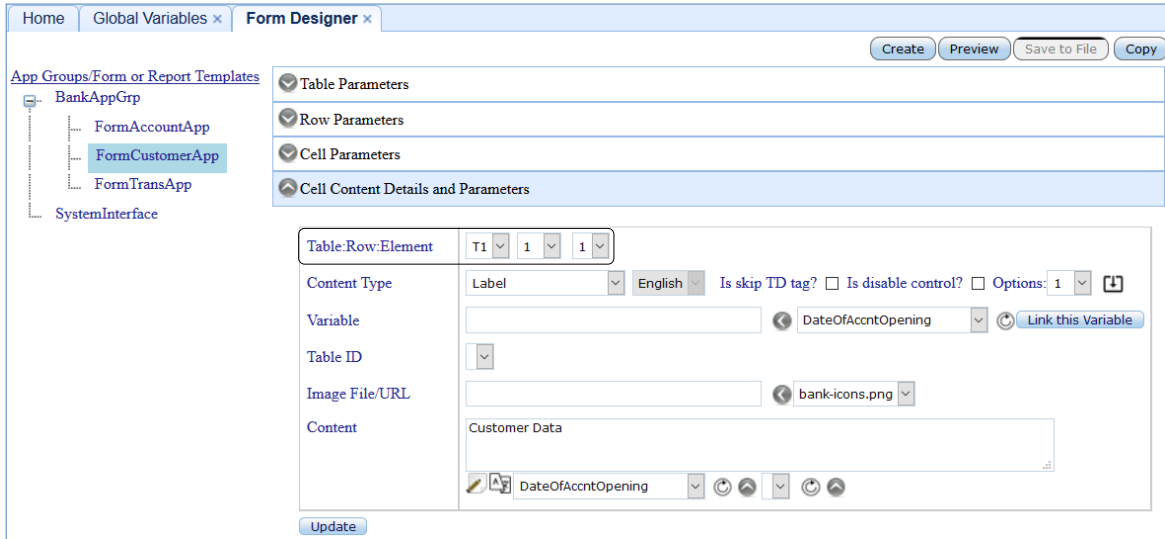


Figure 36: Selecting Table, Row & Column to Modify

- To align the label to center, click *Formatting Parameters* (by default, selected are applicable to cell content) option from the list of options.

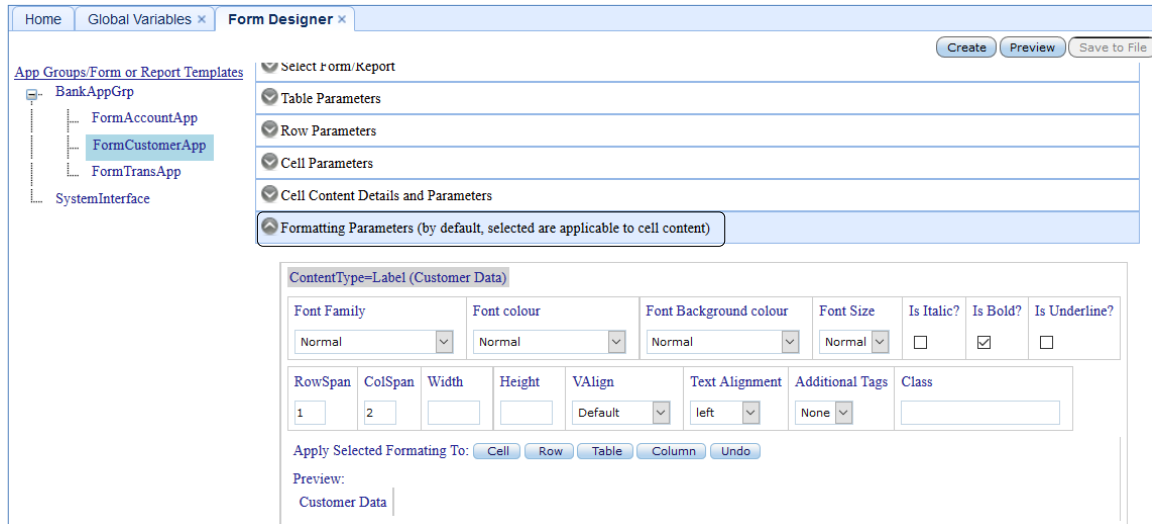


Figure 37: Formatting Parameters Panel (by default, selected are applicable to cell content)

- Set *Text Alignment* to *Center* from *Text Alignment* dropdown menu.

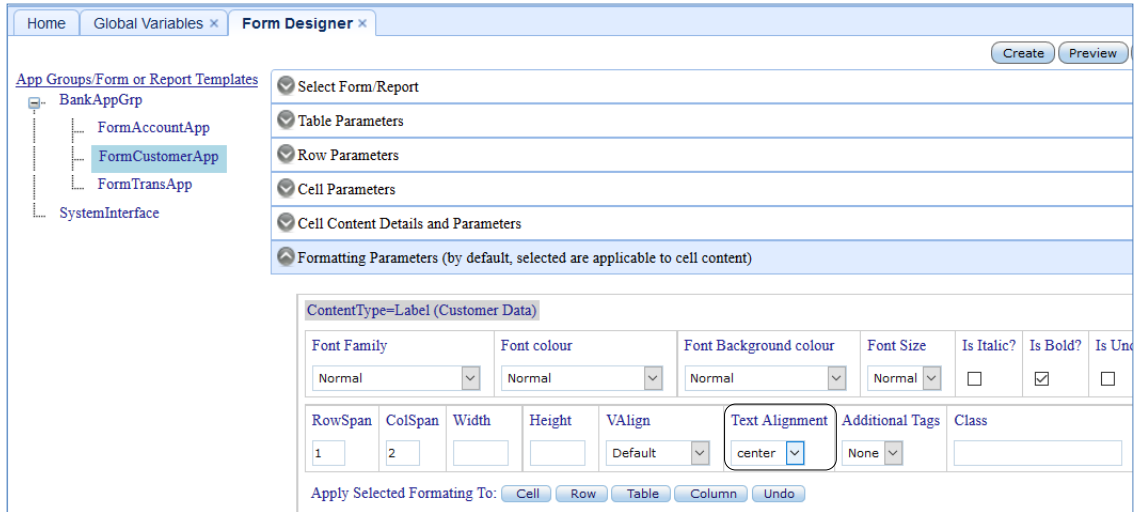


Figure 38: Set Text Alignment to Center

5. To update the changes done, click *Update* and the following message is displayed.

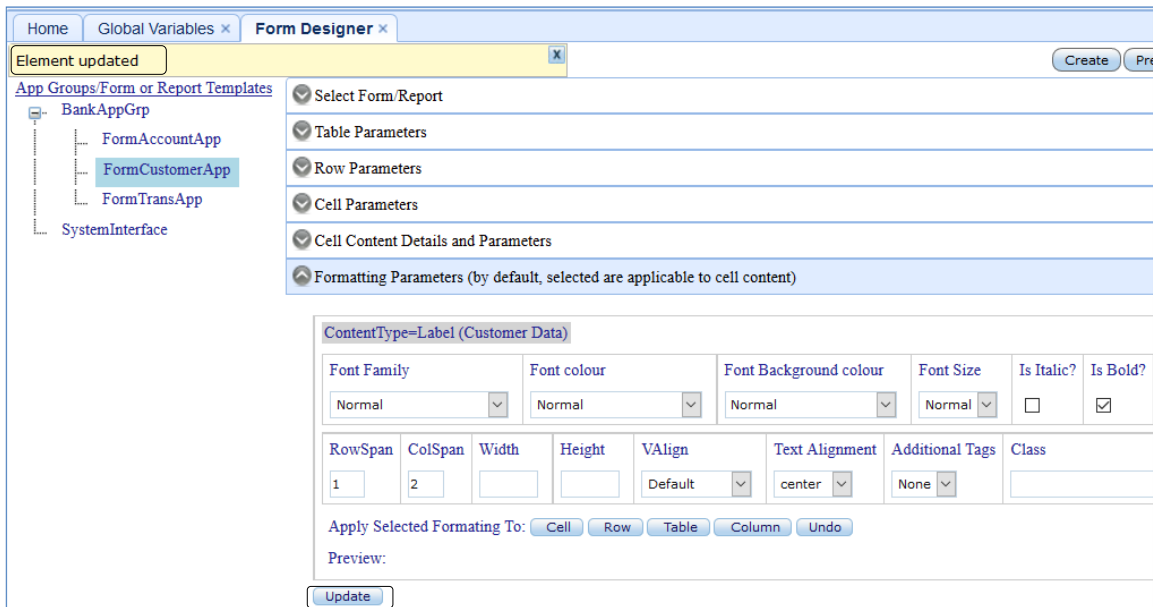


Figure 39: Updating the Form

6. To create the form, click *Create* and the following message is displayed.

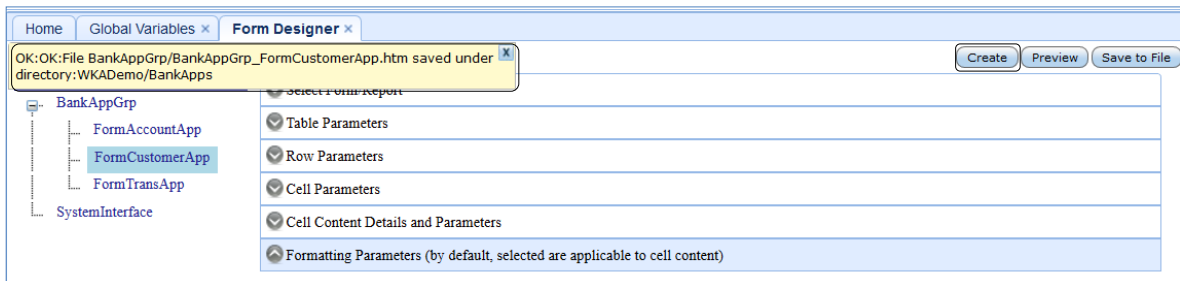


Figure 40: Creating the Form

7. To view the changes, run the application. To run the application, go to *Main Menu -> Apps, APIs and Docs -> Apps* and can view that the label is set at center position.

The screenshot shows a web form titled "BANK APPLICATION" with a section for "Customer Data". The form fields and their values are as follows:

Customer Data	
Customer ID	110
First Name	Rishi
Last Name	Verma
Gender	<input type="radio"/> Female <input checked="" type="radio"/> Male
Account ID	10
Address	Parvati Heritage Bandra
City	Mumbai
State	Maharashtra
Password	••••••••
Date Of A/C Opening	02/01/2016
Email	rishi12@gmail.com
Mobile No	9865234589

Figure 41: Label Aligned to Center

2. Similarly, to change the alignment of label for *FormAccountApp* and *FormTransApp*, follow the same process as explained in section 1.4 above and alignment of the label (Account Data and Transaction Data) is changed as shown in screen shots below.

The screenshot shows a web form titled "BANK APPLICATION" with a section for "Account Data". The form fields and their values are as follows:

Account Data	
Account ID	1
Account Type	Savings Account
Branch Name	Hinjewadi
Date Of A/C Opening	23/07/2011
Combined Balance	60500
AccountIDInfo	Press 'Back' button to view another record

Next Back

Figure 42: Label Aligned to Center for FormAccountApp

BANK APPLICATION	
Transaction Data	
Acct Trans ID	1
Customer ID	102
Account ID	5
Transaction Type	<input checked="" type="radio"/> Debit <input type="radio"/> Credit
Date Of Transaction	12/05/2015
Amount	5000
Mode	<input checked="" type="radio"/> Cash Withdrawal <input type="radio"/> Cheque
AcctTransIDInfo	Press 'Back' button to view another record
<input type="button" value="Next"/> <input type="button" value="Back"/>	

Figure 43: Label Aligned to Center for FormTransApp

1.5 Adding Row Condition

This option allows hiding or showing particular row or column at specific condition. For example: In *FormCustomerApp* form there is no need to display the password when *View* DB Operation is selected, it must be visible when *Add* or *Update* operation is selected. (Refer Document *DB App Creation*). To hide or show the particular row or column, first the variable (*IsAskPassword*) needs to be created.

- To create variable *IsAskPassword*, go to *Main Menu -> Domain Vocabulary -> Global Variables* and enter the variable name *IsAskPassword* in *New Variable* field. Click *Add* to create a new variable. After the variable is created, choose *Type* as *Input*, *Data Type* as *Text* and *Default Value* as *No* from *Type*, *Data Type* and *Default Value* dropdown menu.

Figure 44: Creating a Variable and Setting the Data Type

- To update the variables created, click *Update* and the following message is displayed.

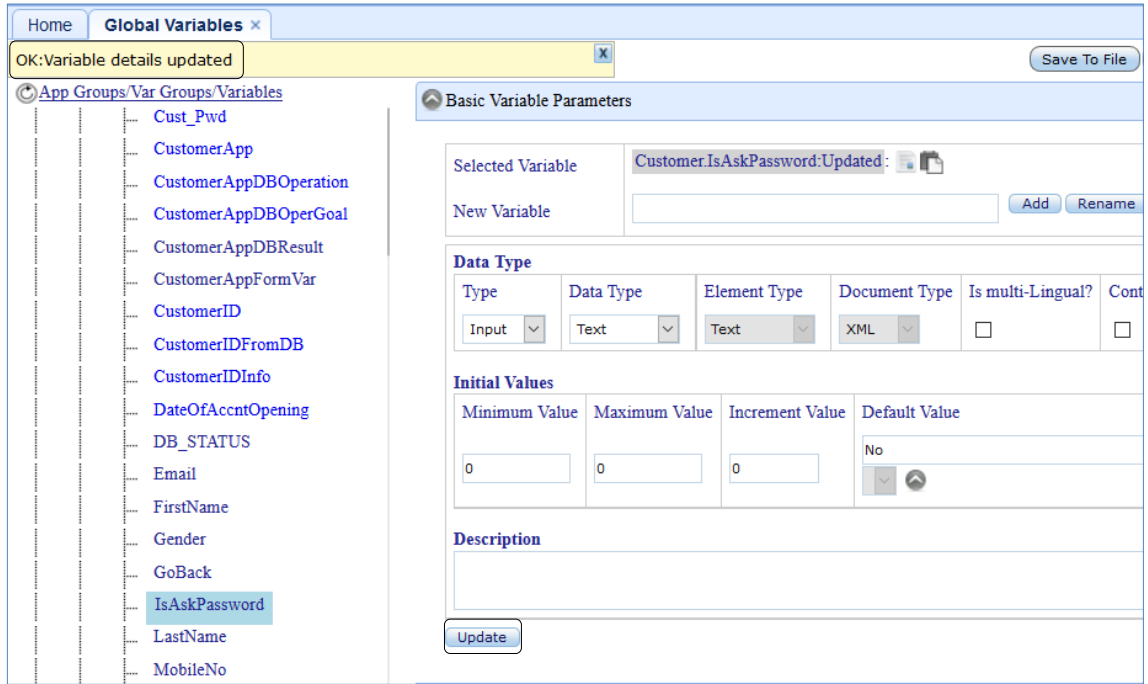


Figure 45: Updating Variables

- To hide the row, go to *Main Menu -> Forms and Reports -> Form Designer*. Click *Row Parameters* option from the list of options and select *FormCustomerApp* form.

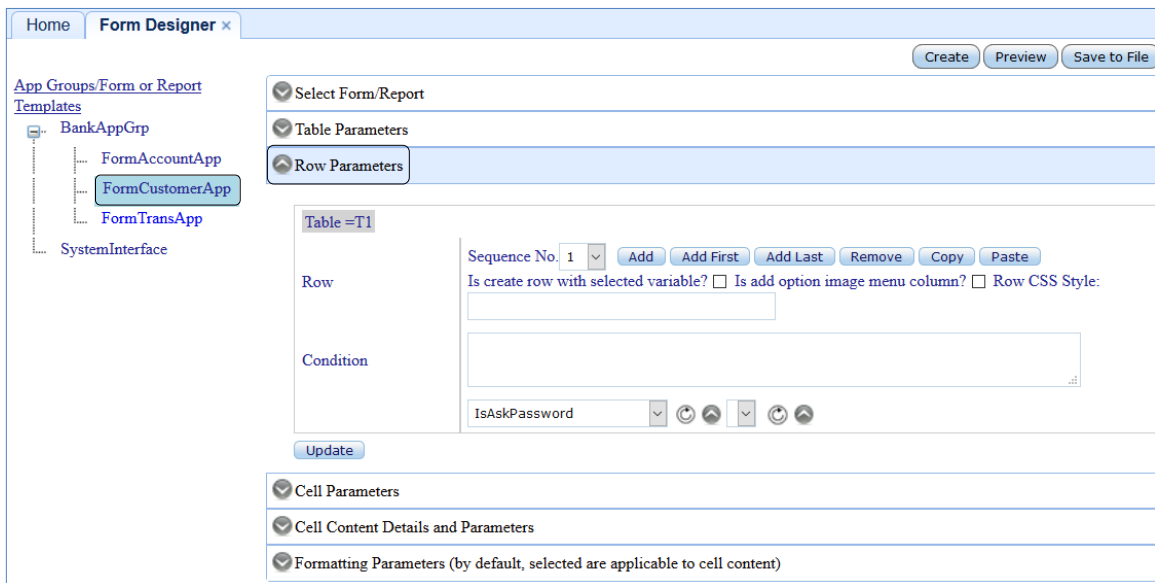


Figure 46: Row Parameters Panel

- To hide the row number, select row number from *Sequence No.* dropdown menu.

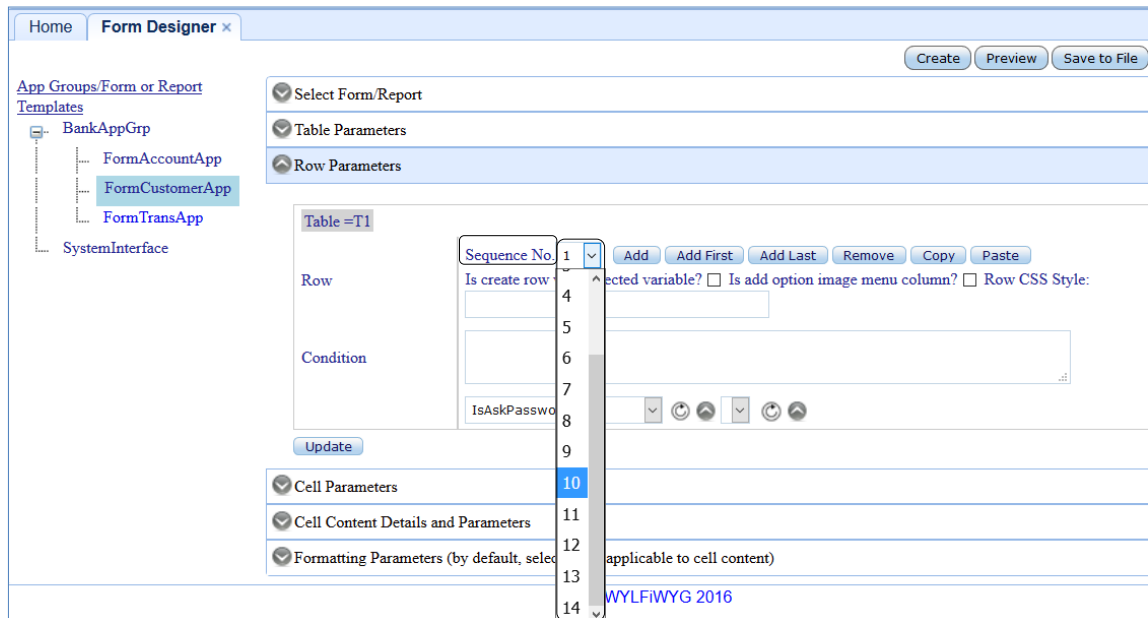


Figure 47: Selecting Row Number

5. Select *IsAskPassword* variable from the variable list.

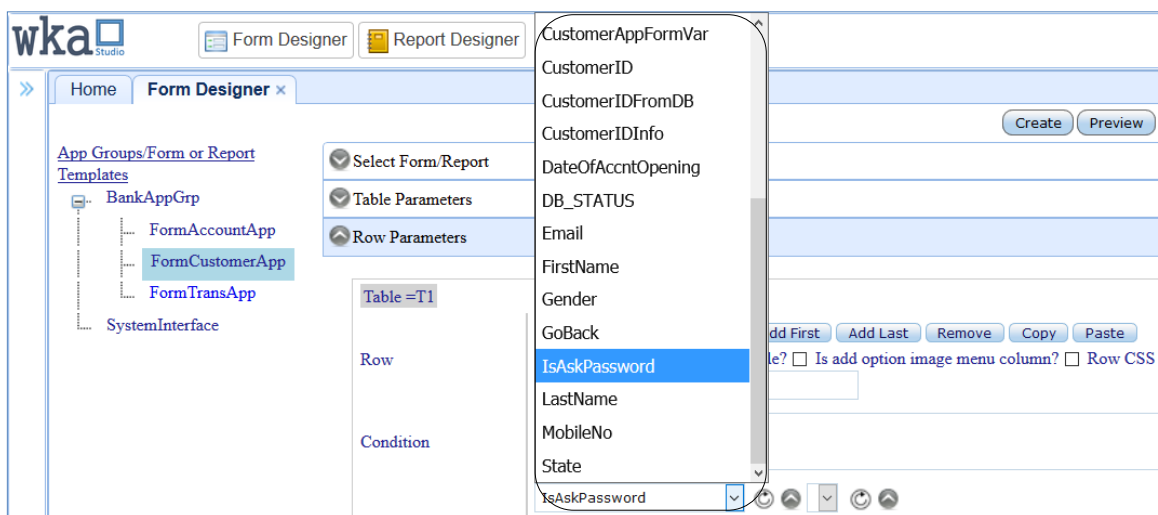



Figure 48: Selecting Variable from Variable List

6. To insert the variable created in *Condition* field, click the button  and enter condition *IsAskPassword== 'Yes'* in *Condition* box.

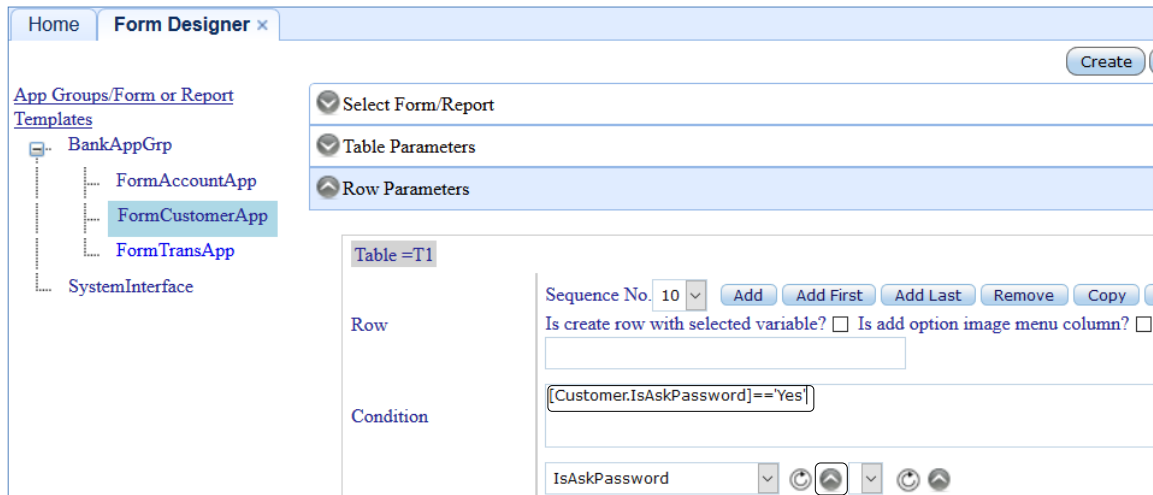


Figure 49: Setting Condition for Row Selected

7. To update the row, click *Update* and the following message is displayed.

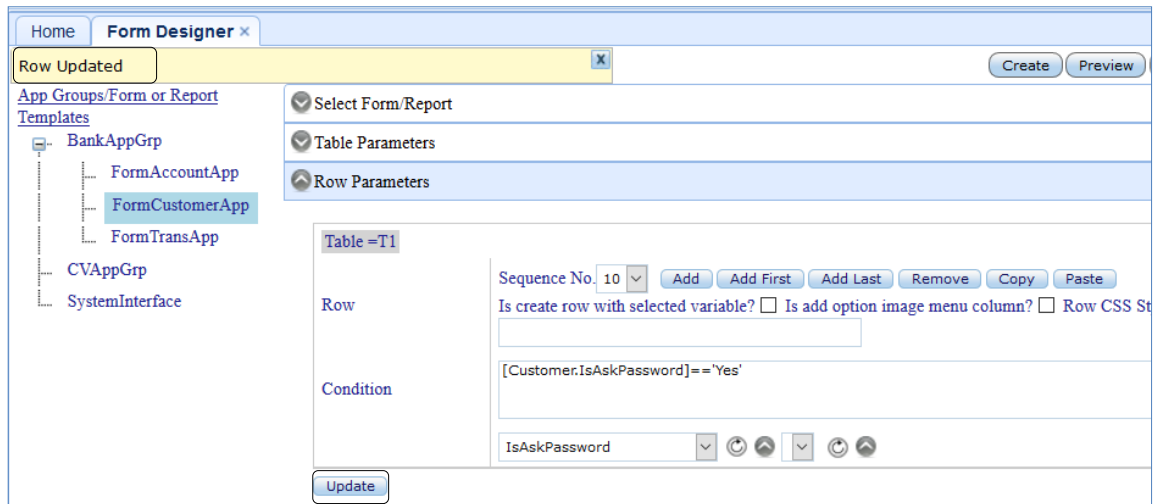


Figure 50: Updating the Row

8. To recreate the form, click *Create*.

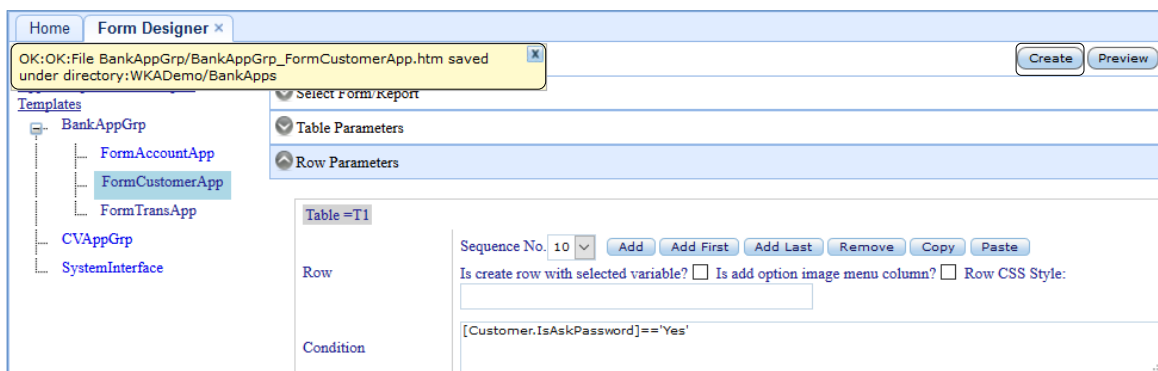


Figure 51: Recreating the Form

9. To preview the form, click *Preview* to see condition applied on row=10.

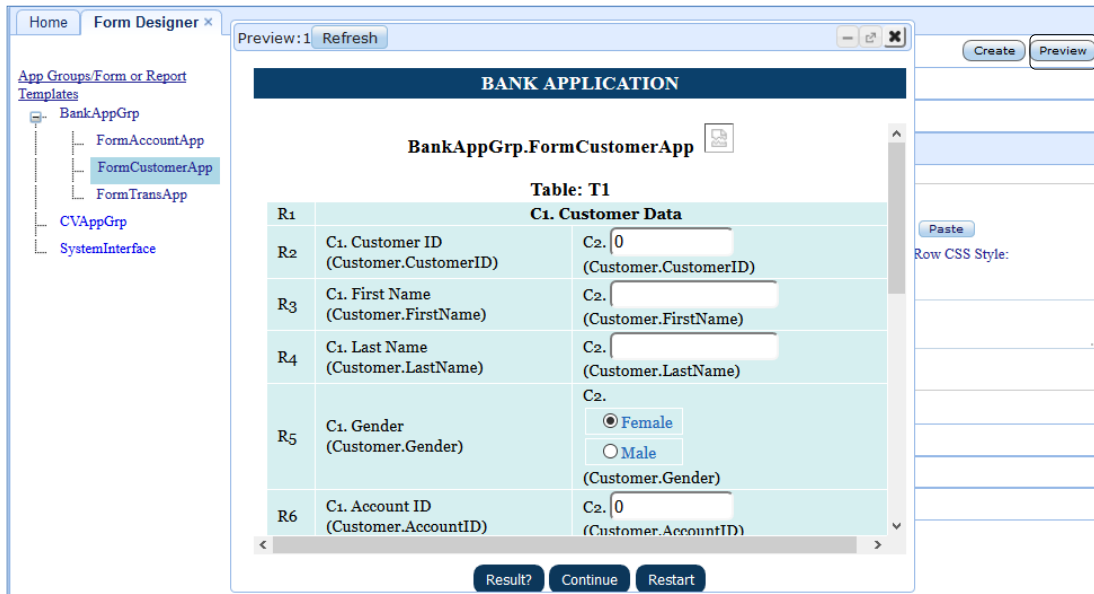


Figure 52: Previewing the Form

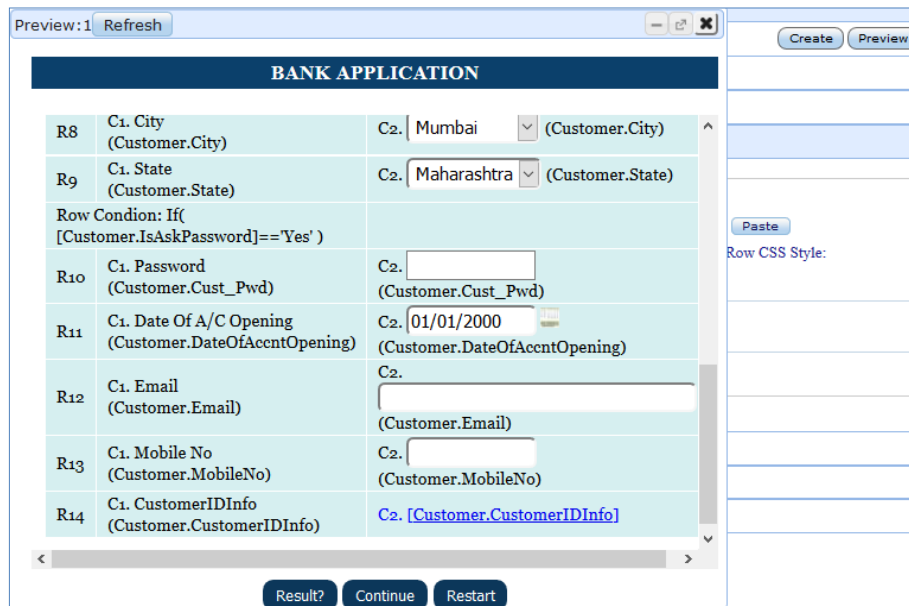


Figure 53: Previewing the Form

10. Since the password for form should be visible in *Add* and *Update DB Operation* so the condition *IsAskPassword:=Yes* must be assigned in *Add* and *Update* rule. To add the condition in *Add* and *Update* rule, go to *Main Menu -> Core Engine Configs -> Expert Systems*, select the rule *CustomerApp DB OPR Add* and *CustomerApp DB OPR Update* as shown in screen shots below:

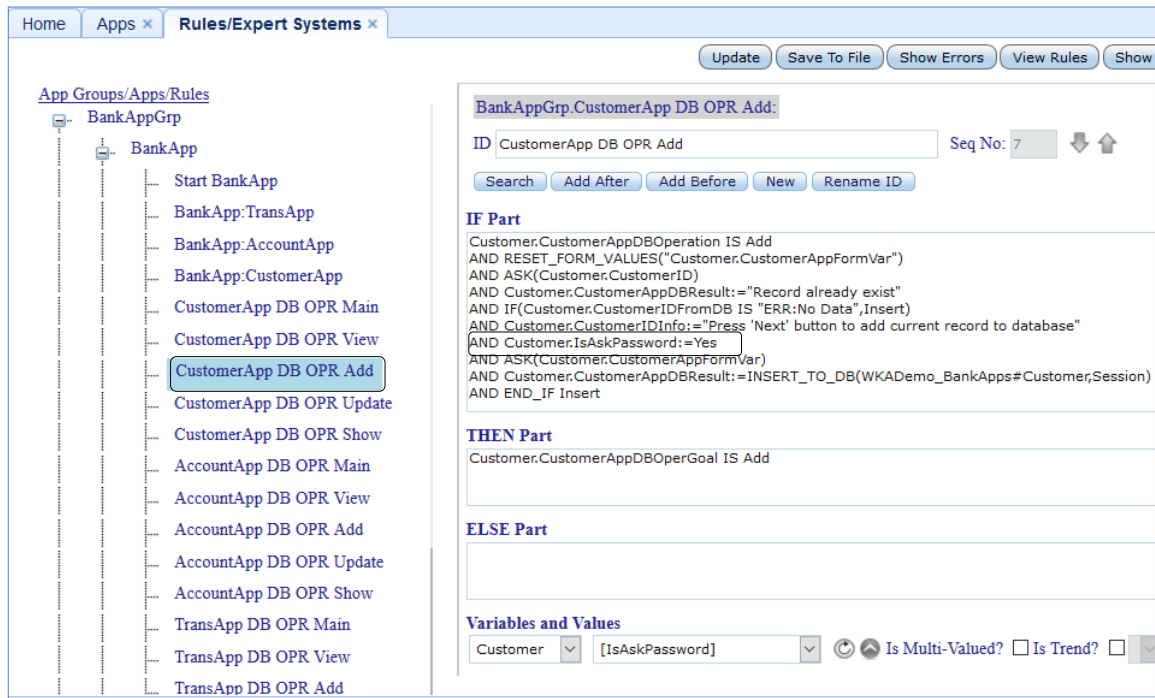


Figure 54: Adding the Condition in Add Rule

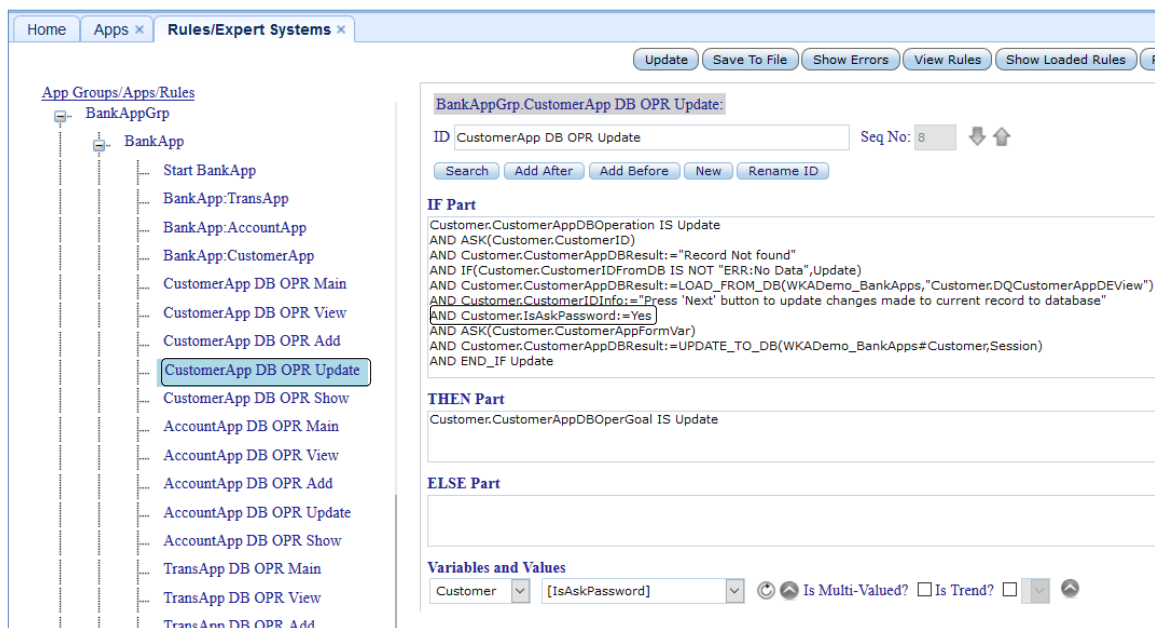


Figure 55: Adding the Condition in Update Rule

- To view the changes for *CustomerApp*, run the application. To run the application, go to *Main Menu -> Apps, APIs and Docs -> Apps* and select *View* operation from *DB Operation* dropdown menu and click *Next*.

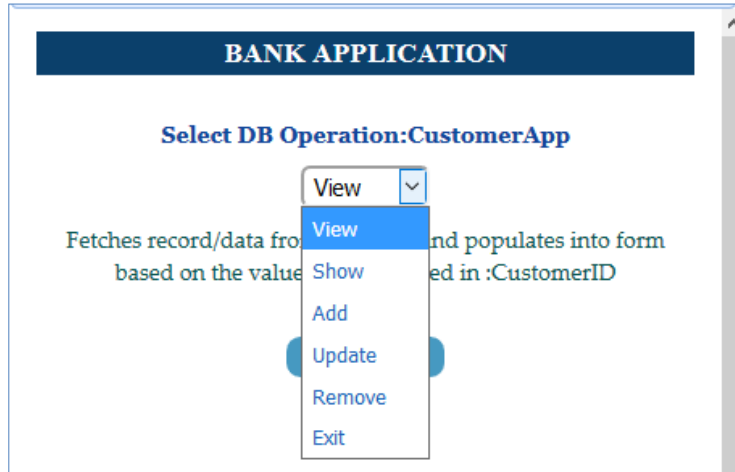


Figure 56: Selecting DB Operation

9. Enter the desired *Customer ID* and click *Next*.

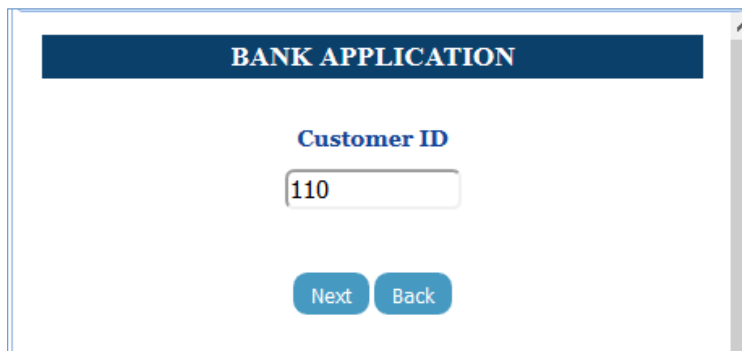


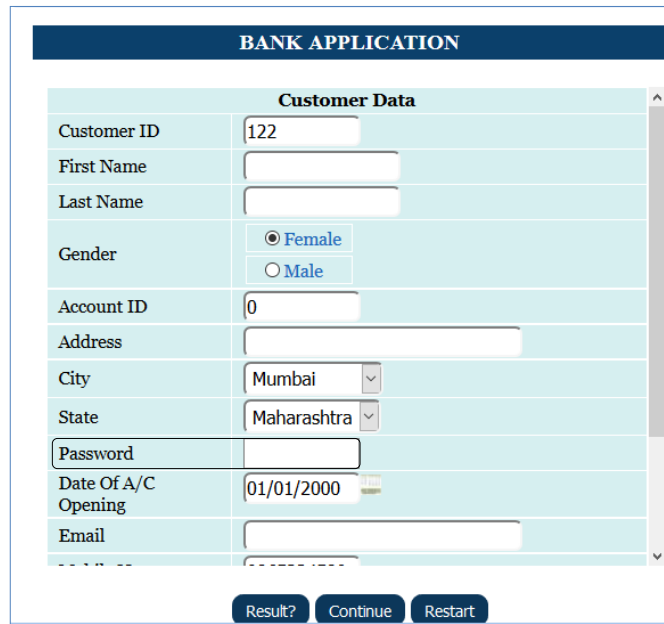
Figure 57: Entering Customer ID

10. The record with customer id 110 is displayed and the row Password is hidden.



Figure 58: Row Password Hidden

11. To view the changes for *Add DB Operation*, select *Add* operation from *DB Operation* dropdown menu as shown in Figure No.56. The row *Password* is seen.



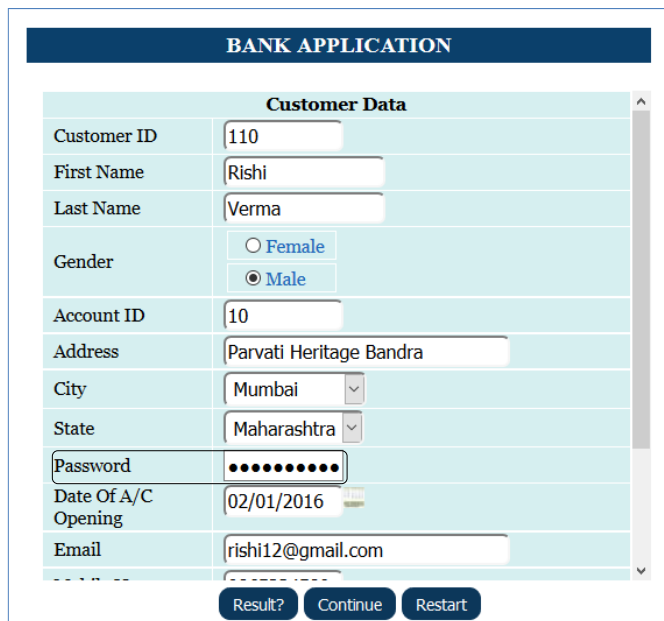
The screenshot shows a web form titled "BANK APPLICATION" with a "Customer Data" section. The fields are as follows:

Customer Data	
Customer ID	122
First Name	
Last Name	
Gender	<input checked="" type="radio"/> Female <input type="radio"/> Male
Account ID	0
Address	
City	Mumbai
State	Maharashtra
Password	
Date Of A/C Opening	01/01/2000
Email	

Buttons at the bottom: Result?, Continue, Restart.

Figure 59: Row Password

12. To view the changes for *Update DB Operation*, select *Update* operation from *DB Operation* dropdown menu as shown in Figure No.56. The row *Password* is seen.



The screenshot shows a web form titled "BANK APPLICATION" with a "Customer Data" section. The fields are as follows:

Customer Data	
Customer ID	110
First Name	Rishi
Last Name	Verma
Gender	<input type="radio"/> Female <input checked="" type="radio"/> Male
Account ID	10
Address	Parvati Heritage Bandra
City	Mumbai
State	Maharashtra
Password	••••••••
Date Of A/C Opening	02/01/2016
Email	rishi12@gmail.com

Buttons at the bottom: Result?, Continue, Restart.

Figure 60: Row Password

13. Similarly, in *Transaction App*, transaction must be done on the basis of *AccountID* not *AccntTransID* therefore hide the *Customer ID* and *AccntTransID* row. Follow the same steps as explained above to hide the row.

BANK APPLICATION	
Transaction Data	
Account ID	5
Transaction Type	<input checked="" type="radio"/> Debit <input type="radio"/> Credit
Date Of Transaction	12/05/2015
Amount	5000
Mode	<input checked="" type="radio"/> Cash Withdrawal <input type="radio"/> Cheque
AccntTransIDInfo	Press 'Back' button to view another record
<input type="button" value="Next"/> <input type="button" value="Back"/>	

Figure 61: Row AcctTransID and CustomerID Hidden

1.6 Adding Rules

Adding rules allows creating separate rules for debit and credit transaction. To fetch *CombinedBalance* (account table) based on the *AccountID* for debit and credit transaction, a dynamic query needs to be created.

1. To create dynamic query, go to *Main Menu -> Database Interfaces -> Dynamic Queries*.

The screenshot shows the 'Dynamic Queries' interface. On the left is a tree view under 'App Groups/Var Groups/Dyn.Queries' with 'BankAppGrp' selected. The right panel contains the following fields:

- Dyn Query ID:** [Empty field]
- New ID:** [Empty field] with buttons 'Add New', 'Restore', and 'Rename'.
- Purpose:** 'GeneralQry' (dropdown) with checkbox 'Is reload database menu when update/refresh?'.
- Selected Database:** 'TEXT_FILE' (dropdown).
- Field Name/s:** [Large empty text area]
- Skip distinct clause?:** [Unchecked checkbox]
- View Name:** [Empty field] with 'New Table:' label and 'Create Table' button.

Figure 62: Dynamic Queries Interface

2. Select variable group for which dynamic query needs to be created. Enter dynamic query ID (Example: *DQBalance*) in *New ID* field. Since single value needs to be retrieved at run time so select query purpose from *Purpose* dropdown menu as *DynamicField*.

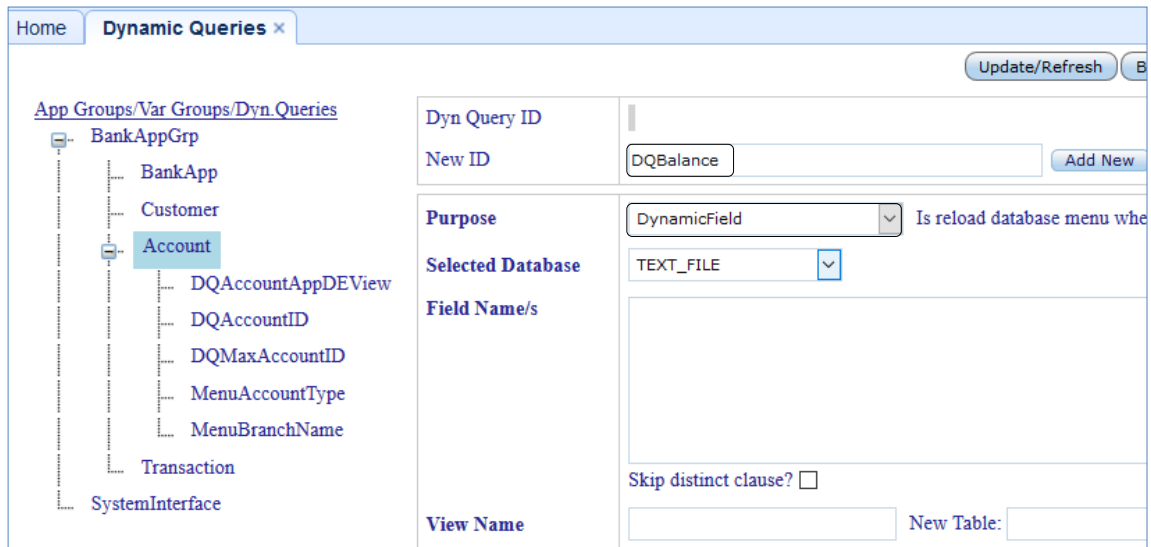


Figure 63: Entering Dynamic Query ID and Selecting Purpose

3. Select database (*WKADemo_BankApps*) from *Selected Database* dropdown menu from which data is to be fetched and click *Add New*.

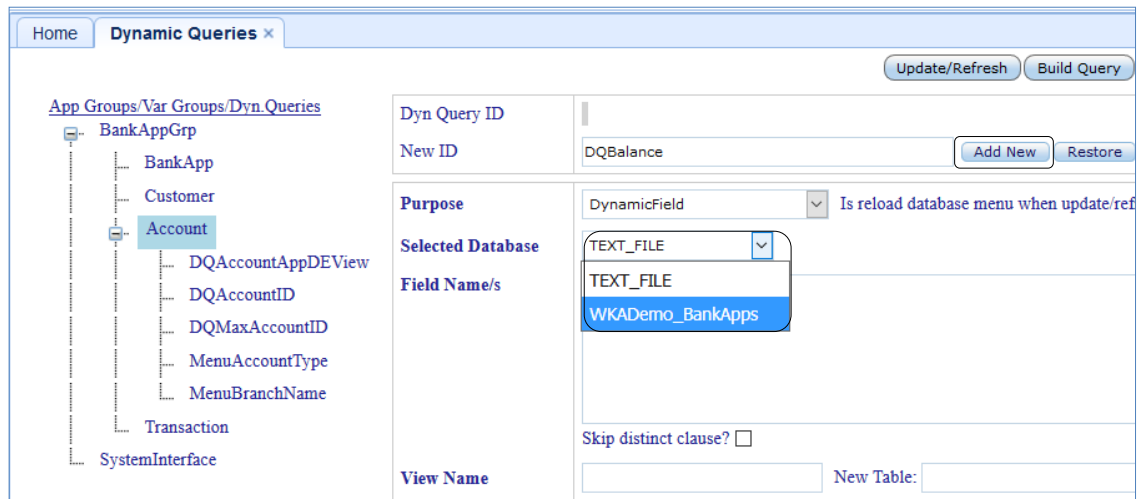


Figure 64: Selecting Database

4. After clicking *Add New*, the following popup is displayed.

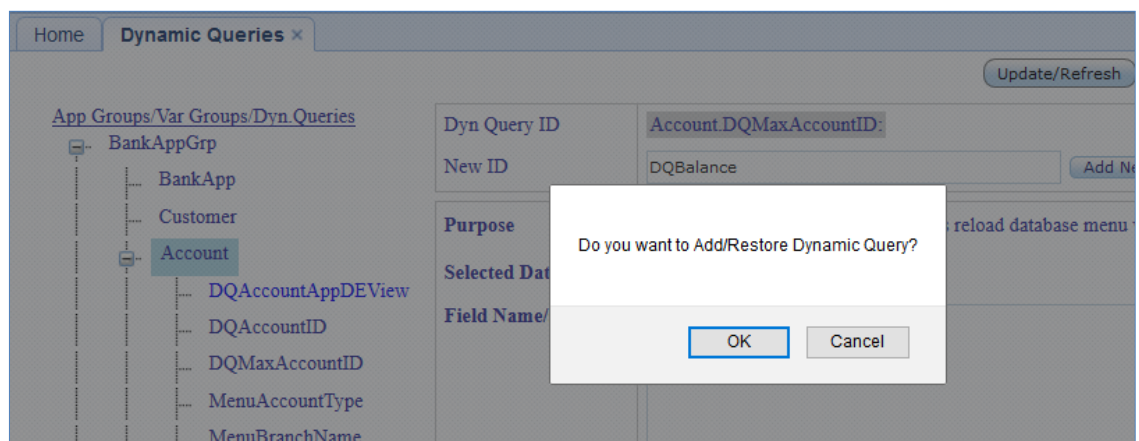


Figure 65: Adding Dynamic Query

- Click *Ok* and query is added into the selected variable group. To fetch *CombinedBalance* from *Account* table, click *Build Query*.

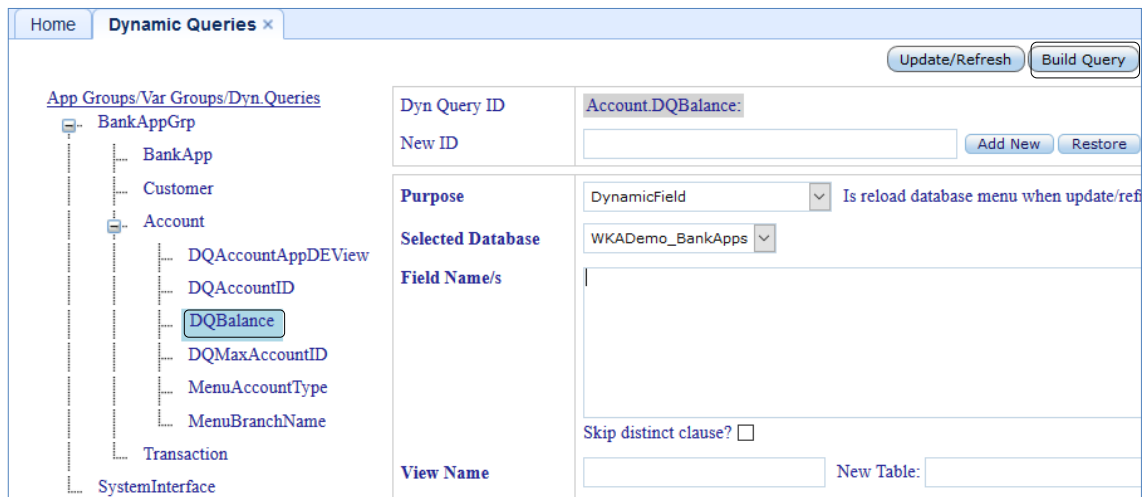


Figure 66: Query Added

- After clicking *Build Query*, the *Query Builder* interface is opened.

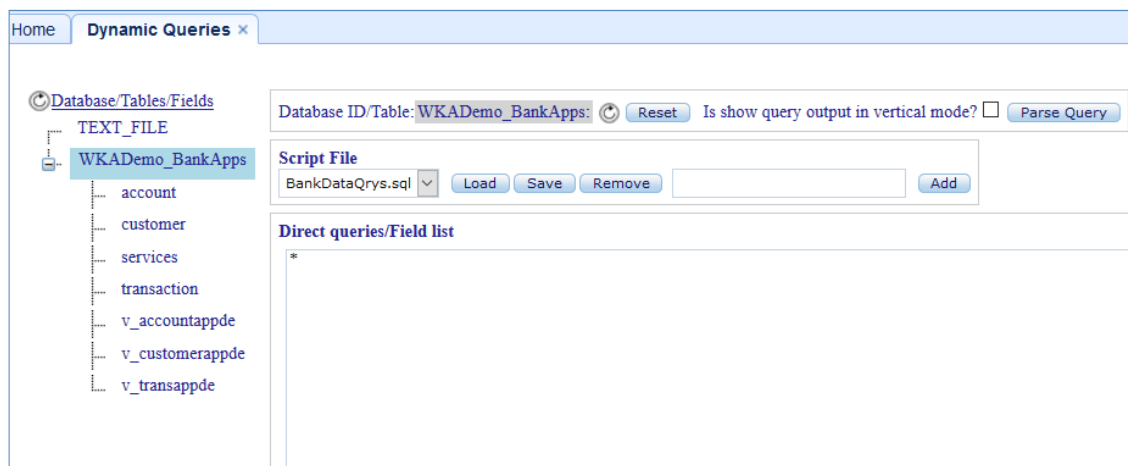


Figure 67: Query Builder Interface

- Select the *account* table from *WKADemo_BankApps* database and select the field *CombinedBalance* from *account* table. When *CombinedBalance* is selected, it gets populated in the *Selected Field* text box.

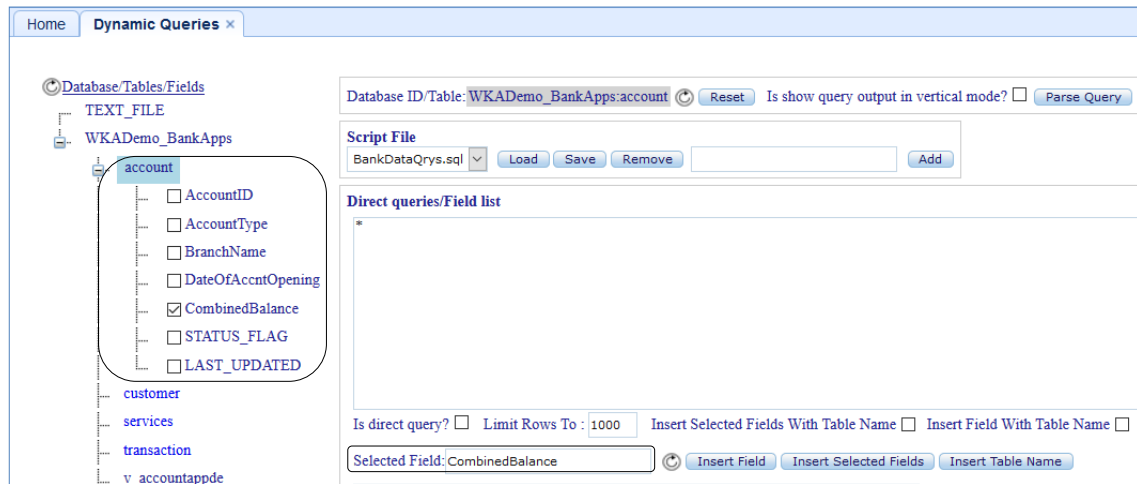


Figure 68: Selecting Table and Field

- To insert the field in *Direct queries/Field list*, click *Insert Field* and click *OK* to go back to *Dynamic Queries* interface.

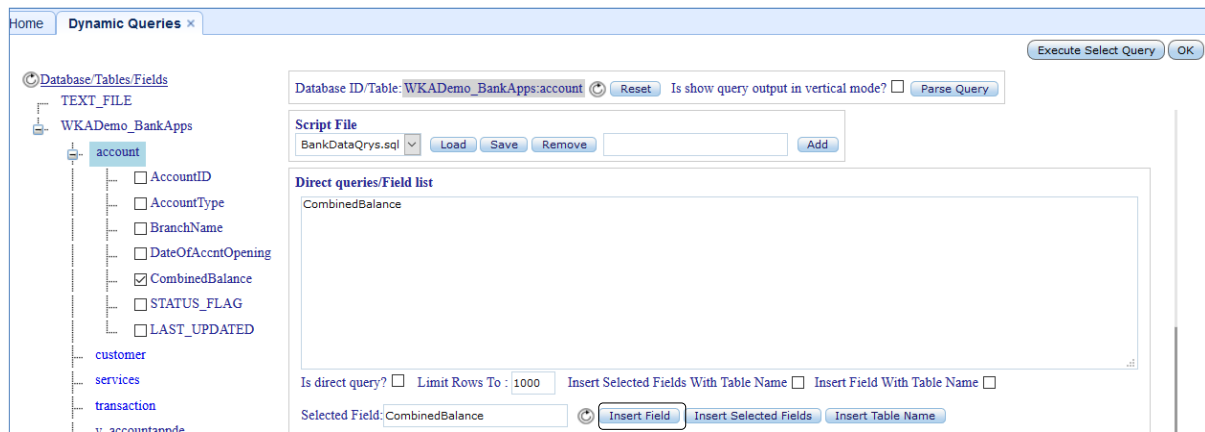


Figure 69: Inserting Field Name

- After clicking *Ok*, field and table name is populated in *Field Name/s* and *View Name* field of the *Dynamic Query* interface.

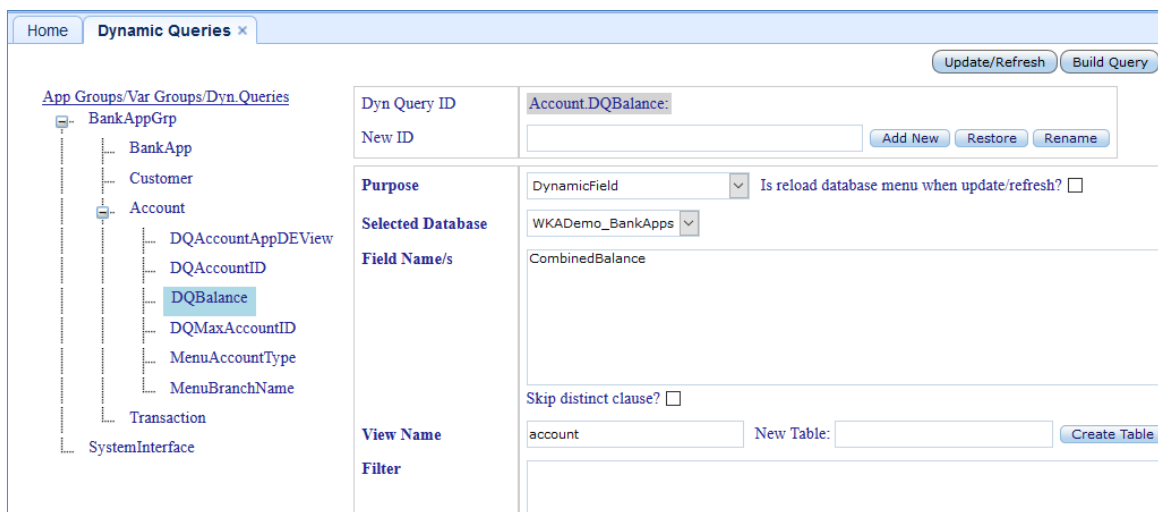


Figure 70: Dynamic Query Interface

10. *AccountID* from *Transaction* variable group is asked at the time of transaction. Therefore, select *AccountID* variable and *Transaction* variable group from *Variable* dropdown menu to fetch *CombinedBalance*.

The screenshot shows the 'Dynamic Queries' configuration window. On the left, a tree view shows the hierarchy: App Groups/Var Groups/Dyn.Queries > BankAppGrp > Account > DQBalance. The 'DQBalance' node is selected. The main configuration area on the right has the following fields:

- Dyn Query ID:** Account.DQBalance
- New ID:** (empty)
- Purpose:** DynamicField
- Selected Database:** WKADemo_BankApps
- Field Name/s:** CombinedBalance
- View Name:** account
- Filter:** (empty)
- Variable:** Transaction (group) and AccountID (variable)

Figure 71: Selecting Variable Group and Variable

11. To insert field into *Filter* field, click the button .

The screenshot shows the 'Dynamic Queries' configuration window with the 'Filter' field populated. The configuration is identical to Figure 71, but the 'Filter' field now contains the text: `AccountID=[Transaction.AccountID]`. The 'Insert field' button (a square with an upward-pointing arrow) is visible next to the variable dropdown.

Figure 72: Inserting Field

12. To update the query, click *Update/Refresh*.

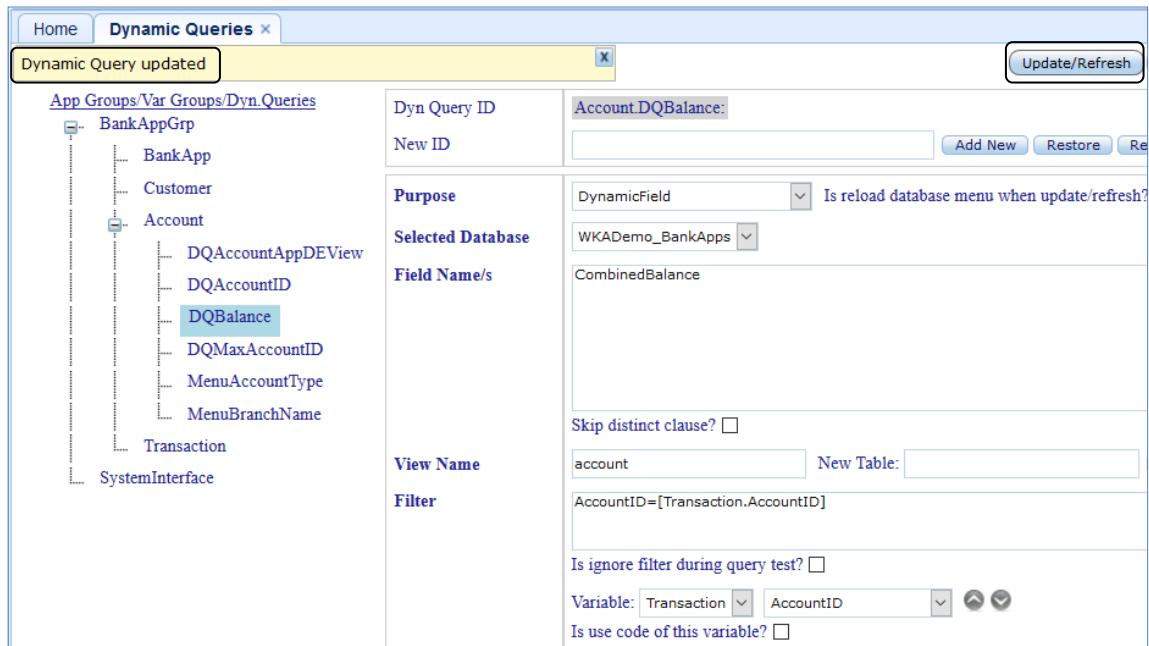


Figure 73: Updating the Query

- To link *Dynamic Query* with a variable, create a variable *Balance* in *Transaction Variable Group*. Follow the same steps as variable created for *IsAskPassword* as explained in section 1.5 above. To update the variable details, click *Update* button.

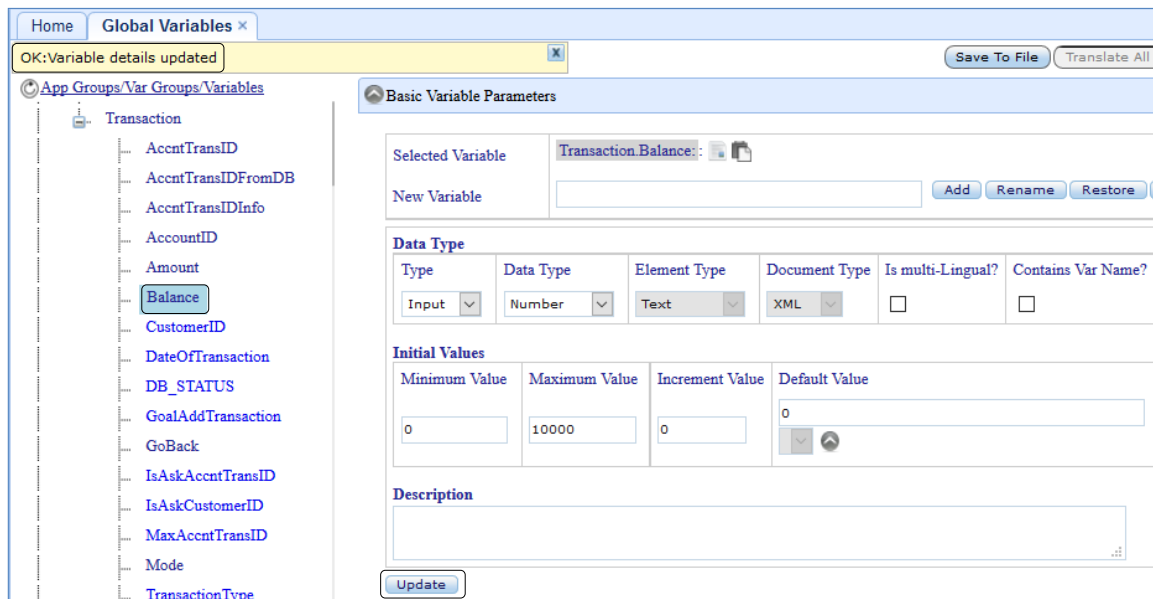


Figure 74: Creating and Updating the Variable

- Click *Input Options and When Added* option from the list of options in *Global Variables* interface.

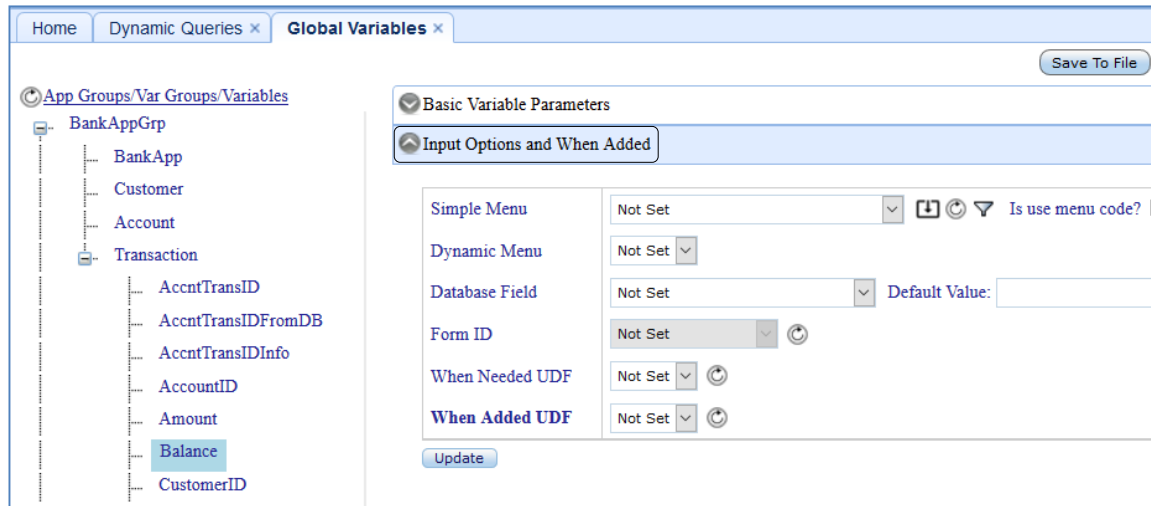


Figure 75: Input Options and When Added Interface

15. Select dynamic query created (*Account.DQBalance*) from *Database Field* dropdown menu.

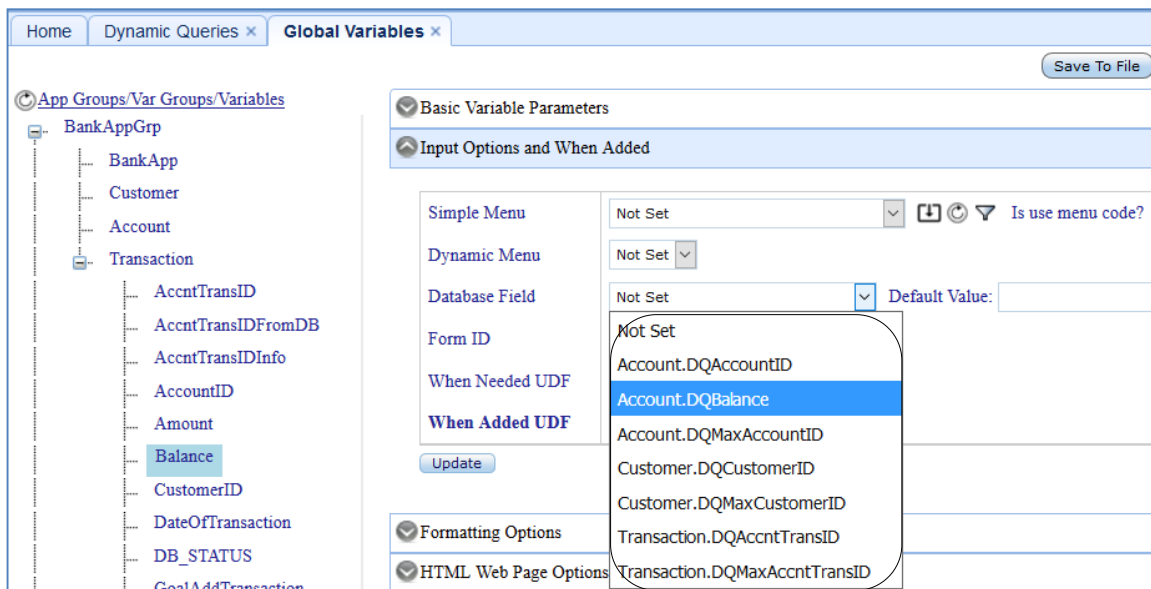


Figure 76: Selecting Dynamic query

16. To update the variable details, click *Update*.

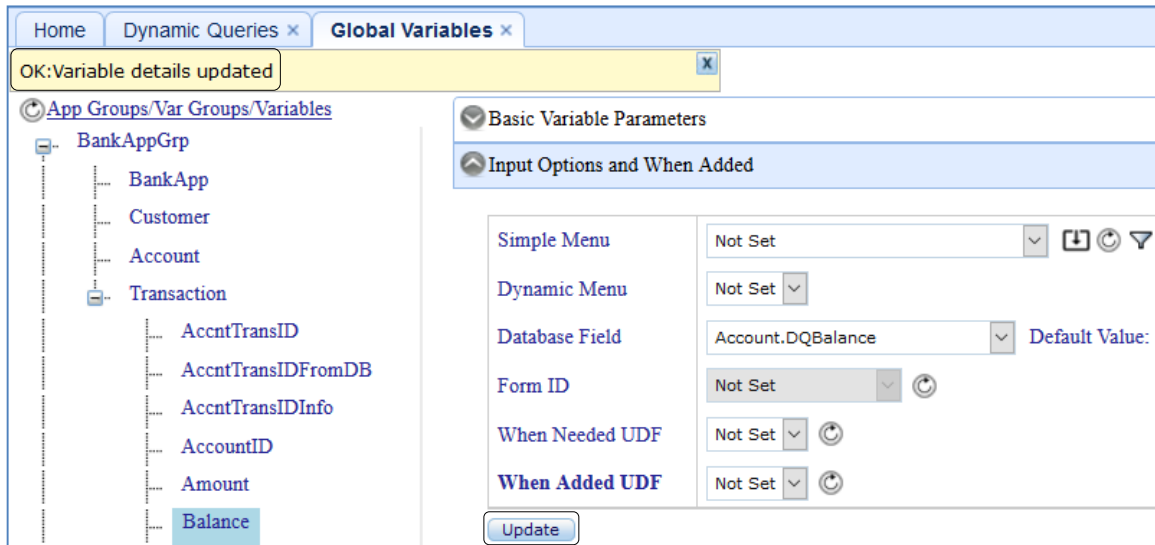


Figure 77: Updating the Variable

17. Create one goal variable *GoalAddTransaction* (used as a sub goal in debit and credit rule) whose *Type* is *Goal* & *Data Type* is *Text* and click *Update*.

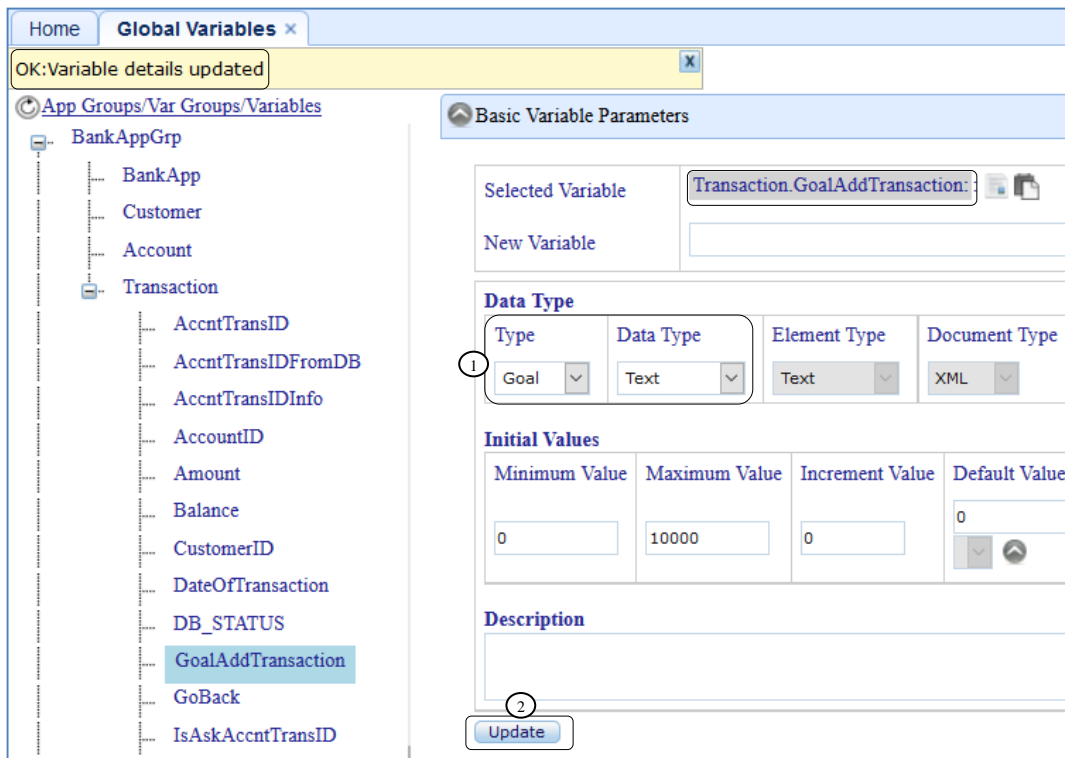


Figure 78: Adding Goal Variable

18. After a variable detail is updated, make changes in *Add* rule of transaction, ask *AccountID* in *Add* rule before the *Form* is asked and update the rule.

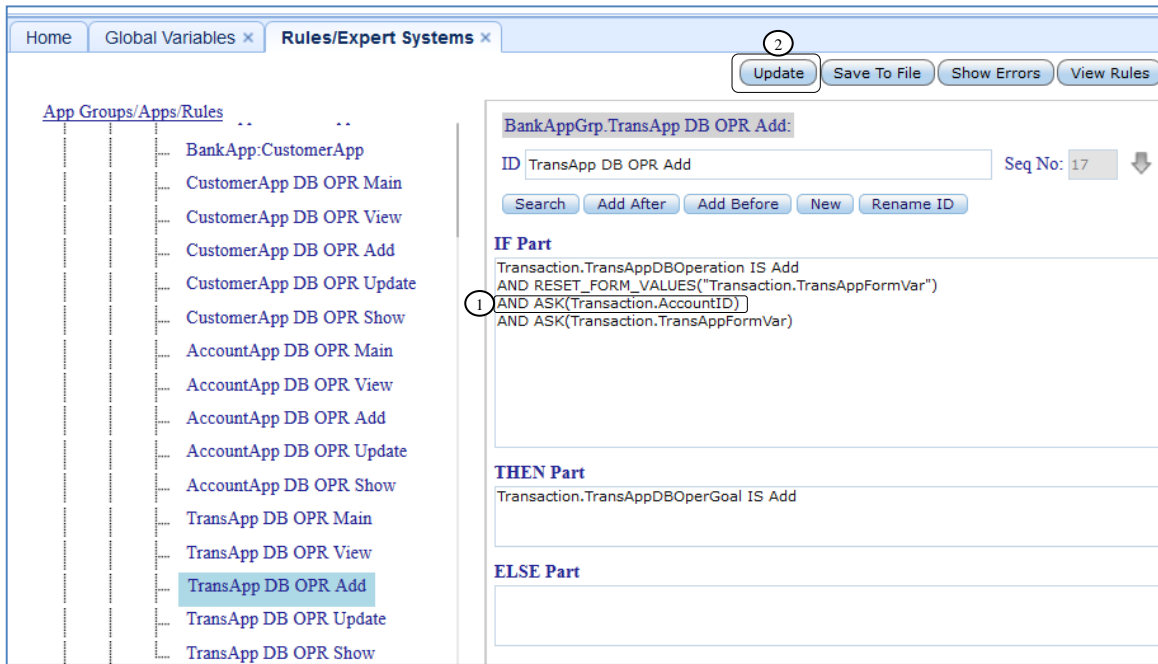


Figure 79: Modifying and Updating Rules

19. Set the current date of transaction as *NOW* (“”). *NOW* function returns the current date.

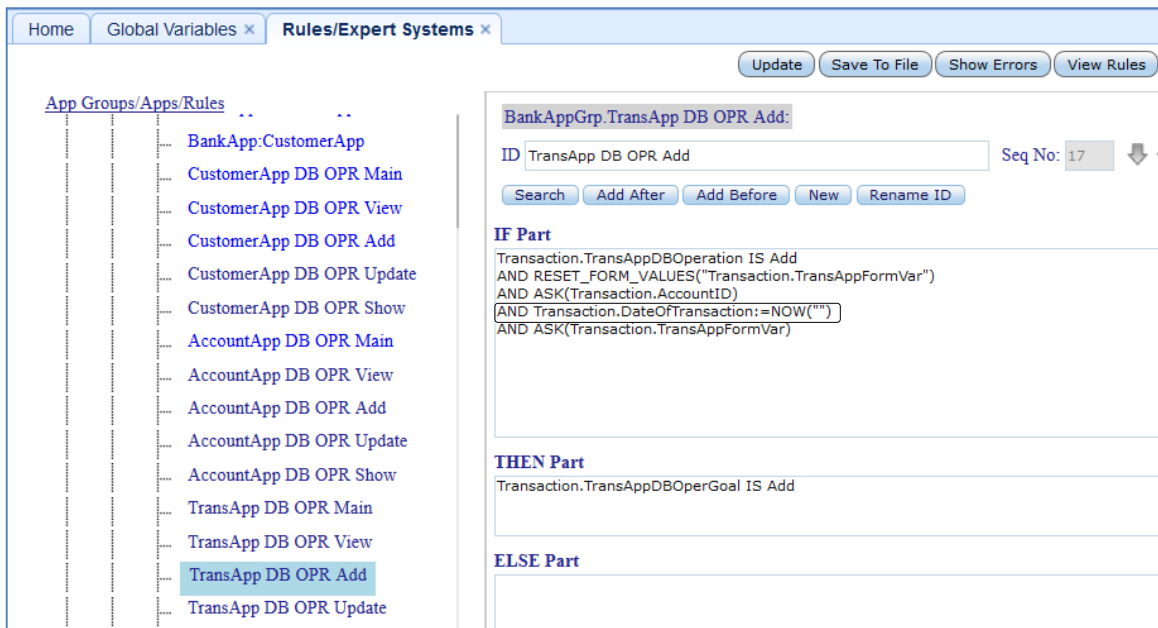


Figure 80: Setting the Current Date

20. Ask the goal variable (*Transaction.GoalAddTransaction*) in *Add* rule in *IF* part. Select variable from *Variables* and *Values* dropdown menu.

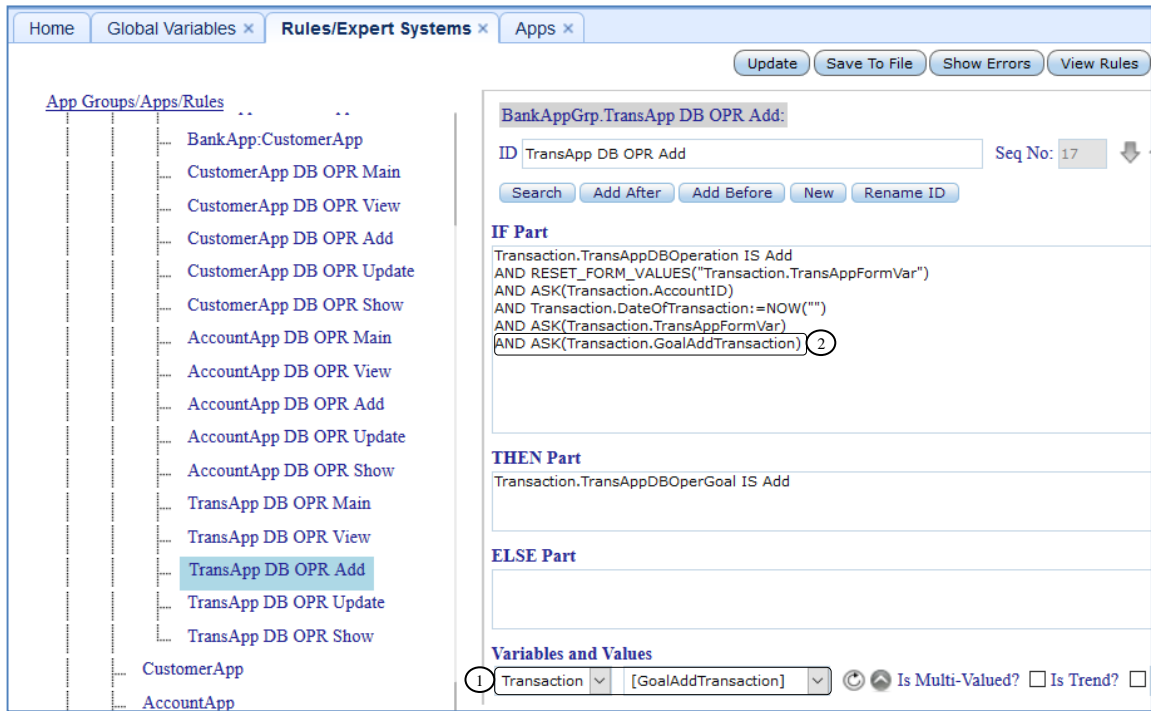


Figure 81: Selecting Variable from Variables and Values

18. Create rules for debit and credit transactions. Enter name of rule (*TransApp Debit*) in *ID* field. Click *Add After* which adds the new rule (*TransApp Debit*) after selected rule i.e. here rule for debit is added after *TransApp DB OPR Add* rule.

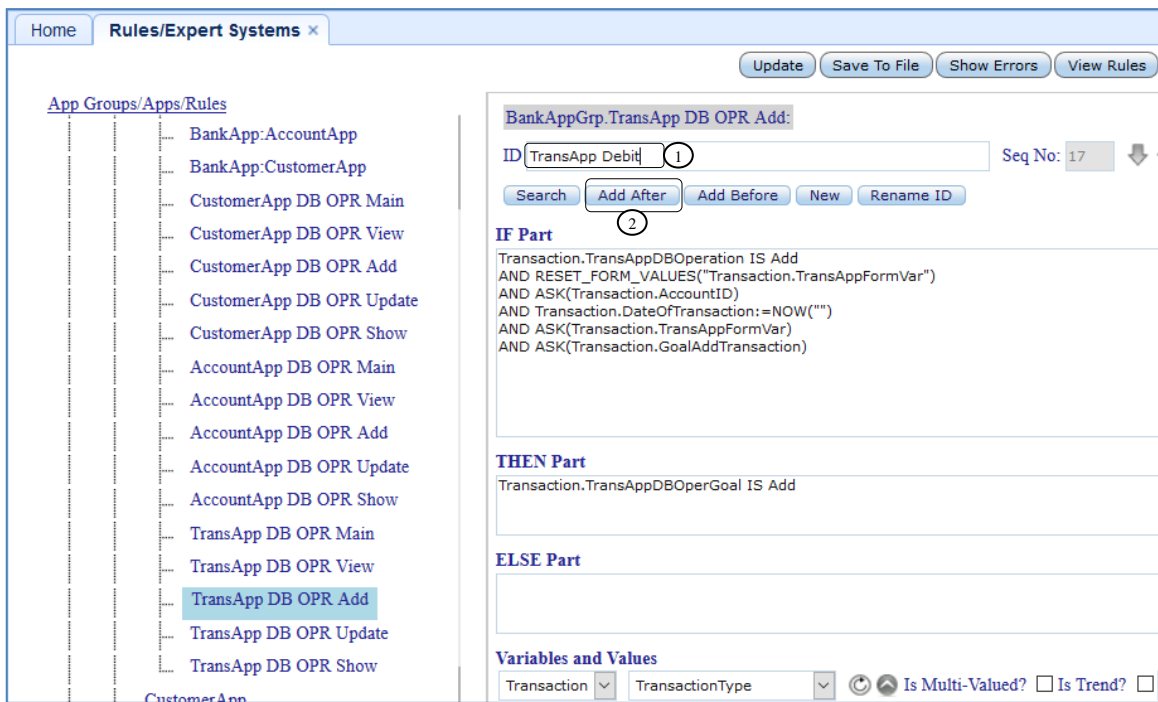


Figure 82: Entering New Rule

19. After clicking *Add After*, the following screen shows popup which carries previous rule.

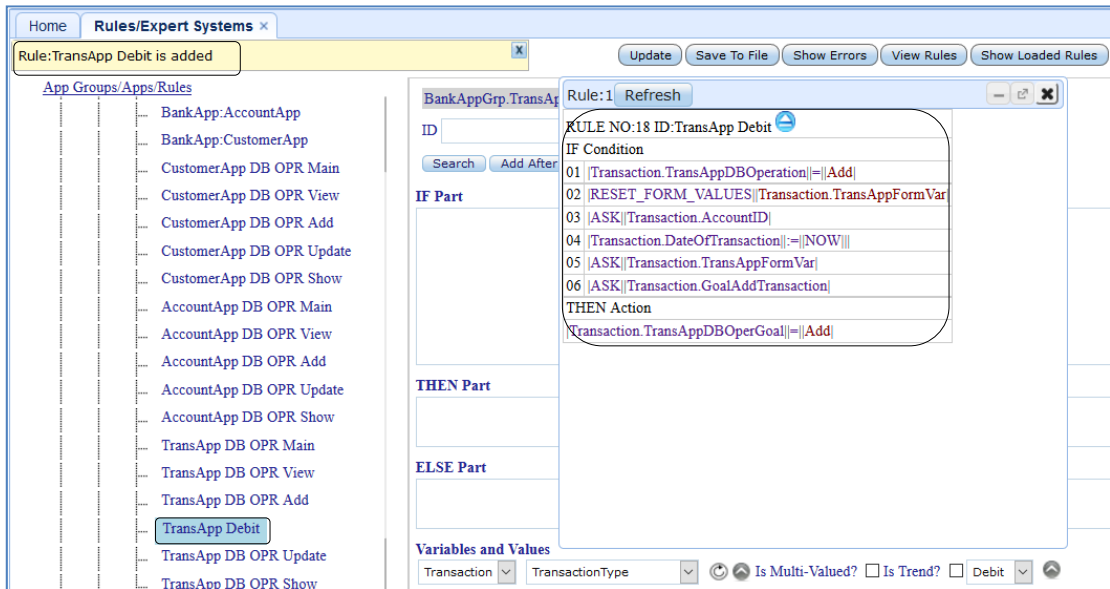


Figure 83: Entering New Rule

18. To clear IF, THEN and ELSE part of the rule, click *New*.

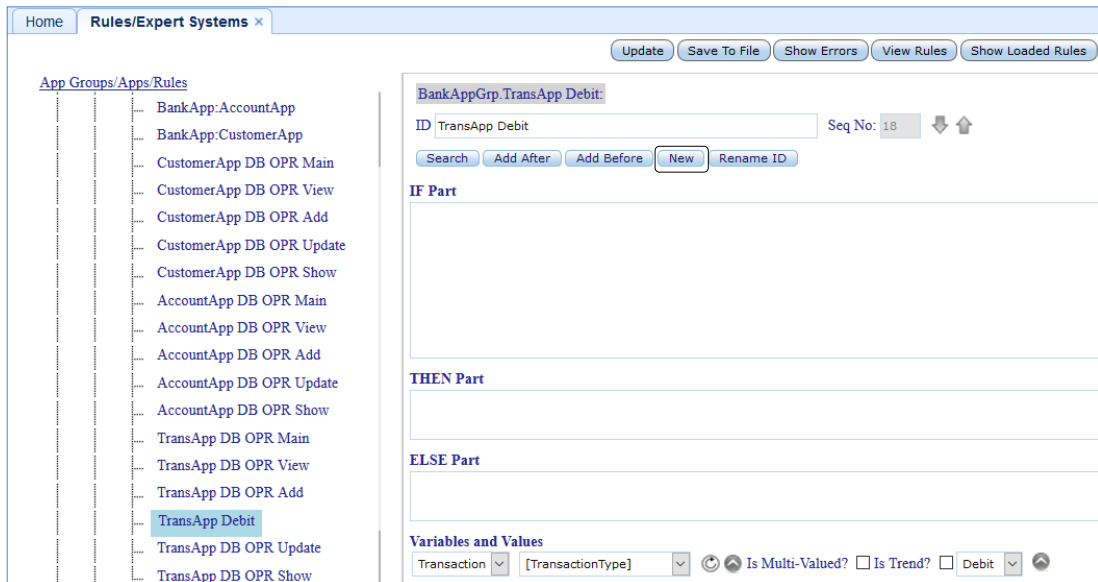


Figure 84: Clearing IF, THEN and ELSE Part

19. After creating rule (*TransAppDebit*), write a rule for debit in *IF* Part and specify *GoalAddTransaction* goal variable in THEN Part of the rule.

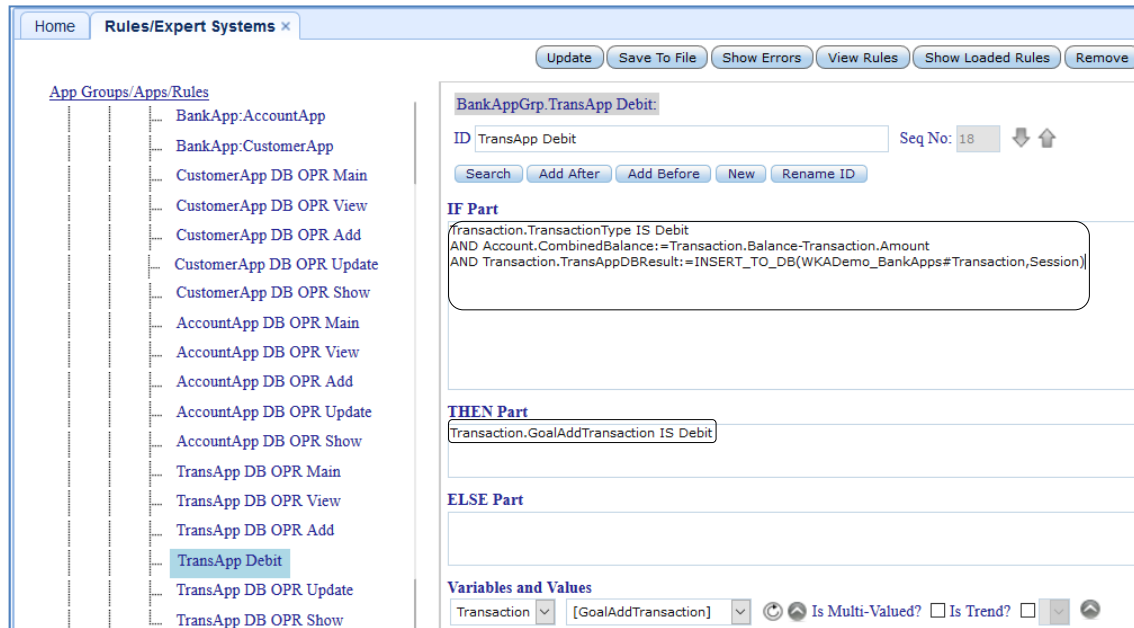


Figure 85: Entering Rule and Goal Variable

20. After debit transaction record is inserted into *Transaction* table, assign *AccountID* field of *Account* table in *Add* rule.

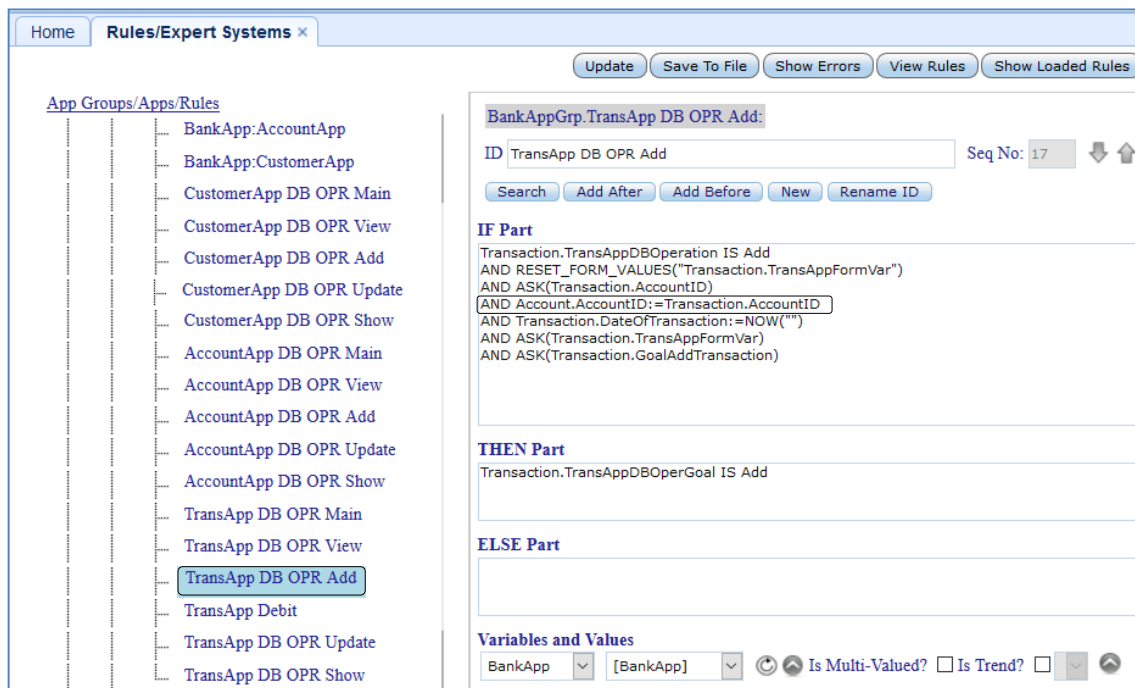


Figure 86: Assigning AccountID Field

21. To update the rule, click *Update* and the following message with update screen is displayed.

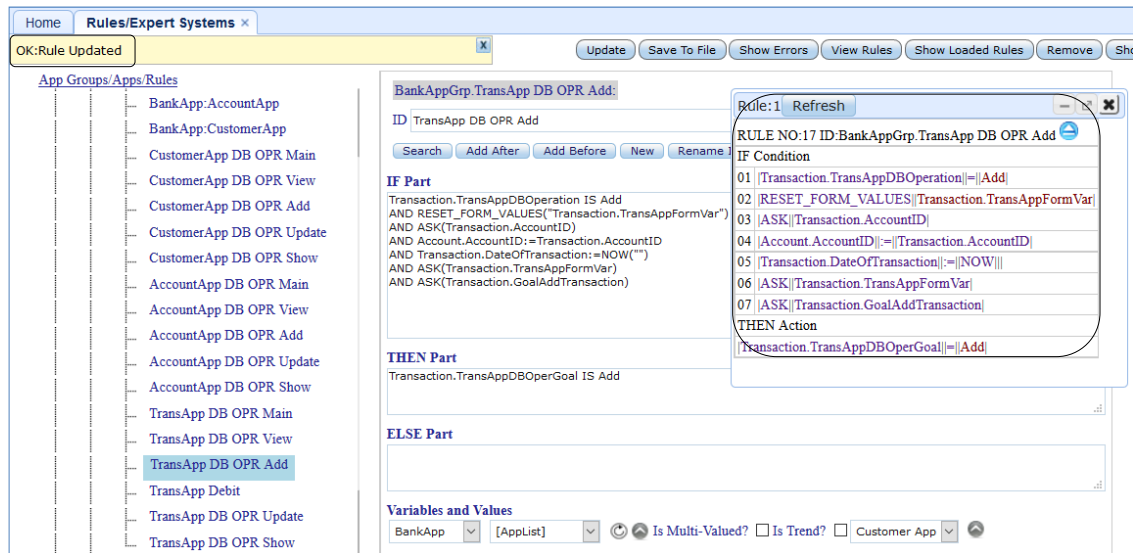


Figure 87: Updating the Rule

22. When debit transaction record is inserted into *transaction* table, *account* table is also updated at the same time so add *Update* function for *account* table and update the rule.

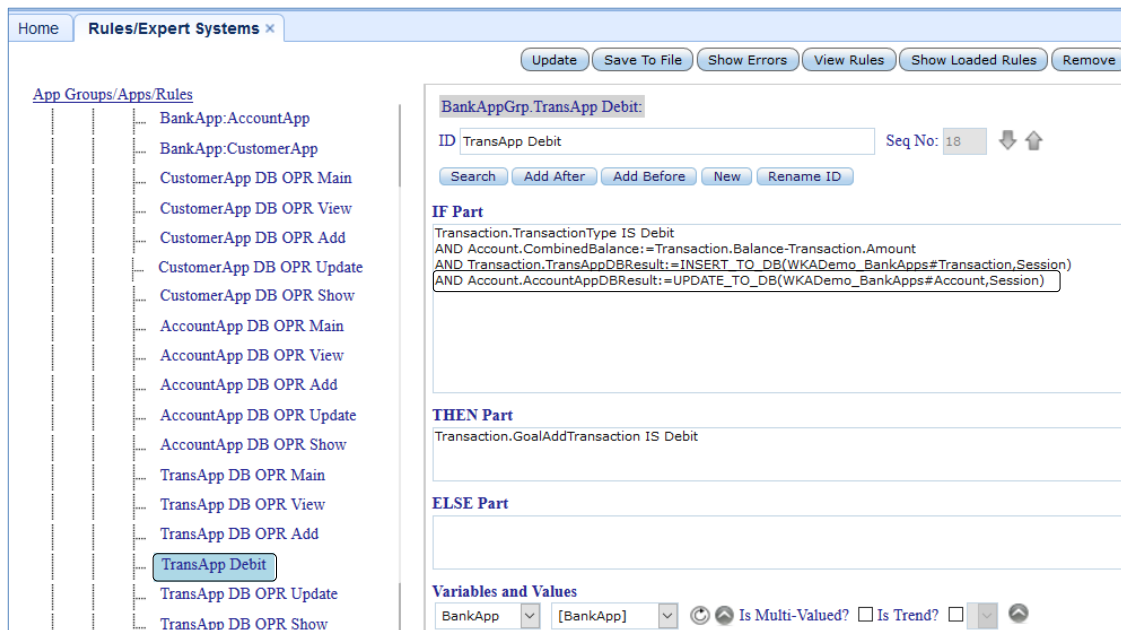


Figure 88: Editing Rule for TransApp Debit

23. To view the changes for *Trans App*, run the application. To run the application, go to *Main Menu* -> *Apps, APIs and Docs* -> *Apps* and select app (*Trans App*) from the list of apps and click *Next*.

BANK APPLICATION

Select app

Customer App

Account App

Trans App

Next Back

Figure 89: Selecting App

24. Select *Add* DB Operation from DB Operation dropdown menu.

BANK APPLICATION

Select DB Operation:TransApp

Fetches record/data from da the value of ke

opulates into form based on :AccntTransID

View

View

Show

Add

Update

Remove

Exit

Figure 90: Selecting DB Operation

25. Enter the desired *Account ID* in *Account ID* field and click *Next*.

BANK APPLICATION

Account ID

5

Next Back

Figure 91: Entering Account ID

26. Enter transaction details and select *Transaction Type* as *Debit*. Click *Next*.

BANK APPLICATION	
Transaction Data	
Account ID	5
Transaction Type	<input checked="" type="radio"/> Debit <input type="radio"/> Credit
Date Of Transaction	12/05/2017
Amount	5000
Mode	<input checked="" type="radio"/> Cash Withdrawal <input type="radio"/> Cheque
AccntTransIDInfo	Press 'Back' button to view another record
<input type="button" value="Next"/> <input type="button" value="Back"/>	

Figure 92: Entering Transaction Details

27. The message *DB Operation Result: OK: 1* is shown which indicates that the record is added. Select DB Operation as *Main DB operation menu* and click *Next*.

BANK APPLICATION	
DB Operation Result:OK:1	
What you want to do?	
<input checked="" type="radio"/> Main DB operation menu <input type="radio"/> Exit DB operations	
Click on 'Back' button to go back to Main DB Menu	
<input type="button" value="Next"/> <input type="button" value="Back"/>	

Figure 93: DB Operation Status

28. After clicking *Next*, session report is displayed.

BANK APPLICATION	
ApplicationDirectory	BankAppGrp
WelcomeText	
AppList	Trans App
Transaction DB Operation	Add
AccountID	5
AccountID	5
MaxAcntTransID	22
AcntTransID	22
DateOfTransaction	12/05/2017
TransactionType	Debit
Amount	5000
Mode	Cash Withdrawal
TransAppFormVar	Complete
Balance	87000
CombinedBalance	82000
Transaction DB Result	OK:1
<input type="button" value="Result?"/> <input type="button" value="Continue"/> <input type="button" value="Restart"/>	

Figure 94: Session Report

BANK APPLICATION	
MaxAcctTransID	22
AcctTransID	22
DateOfTransaction	12/05/2017
TransactionType	Debit
Amount	5000
Mode	Cash Withdrawal
TransAppFormVar	Complete
Balance	87000
CombinedBalance	82000
Transaction DB Result	OK:1
Account DB Result	OK:1
GoalAddTransaction	Debit
Transaction DB Operation Goal:TransApp	Add
GoBack	Transaction.TransAppDBOperation
TransApp	Done
SubAppGoal	Trans App
BankApp	Done

Figure 95: Session Report

As shown in above session report, balance for Account ID=5 is 87000 which is debited by Amount 5000 so *CombinedBalance* is 82000.

29. Similarly, follow the same steps to add rule for credit as explained above for adding rule for debit.

The screenshot shows the 'Rules/Expert Systems' window. On the left, a tree view lists various rules under 'App Groups/Apps/Rules', with 'TransApp Credit' selected. The main area displays the configuration for 'BankAppGrp.TransApp Credit' (Seq No: 19). The 'IF Part' contains a complex logical expression: `Transaction.TransactionType IS Credit AND Account.CombinedBalance:=Transaction.Balance+Transaction.Amount AND Transaction.TransAppDBResult:=INSERT_TO_DB(WKADemo_BankApps#Transaction_Session) AND Account.AccountAppDBResult:=UPDATE_TO_DB(WKADemo_BankApps#Account_Session)`. The 'THEN Part' contains the action: `Transaction.GoalAddTransaction IS Credit`. The 'Variables and Values' section shows a dropdown for 'Transaction' and a text field for '[MaxAcctTransID]'.

Figure 96: Adding Rule for Credit

30. To update the rule, click *Update* and the following message with update screen is displayed.

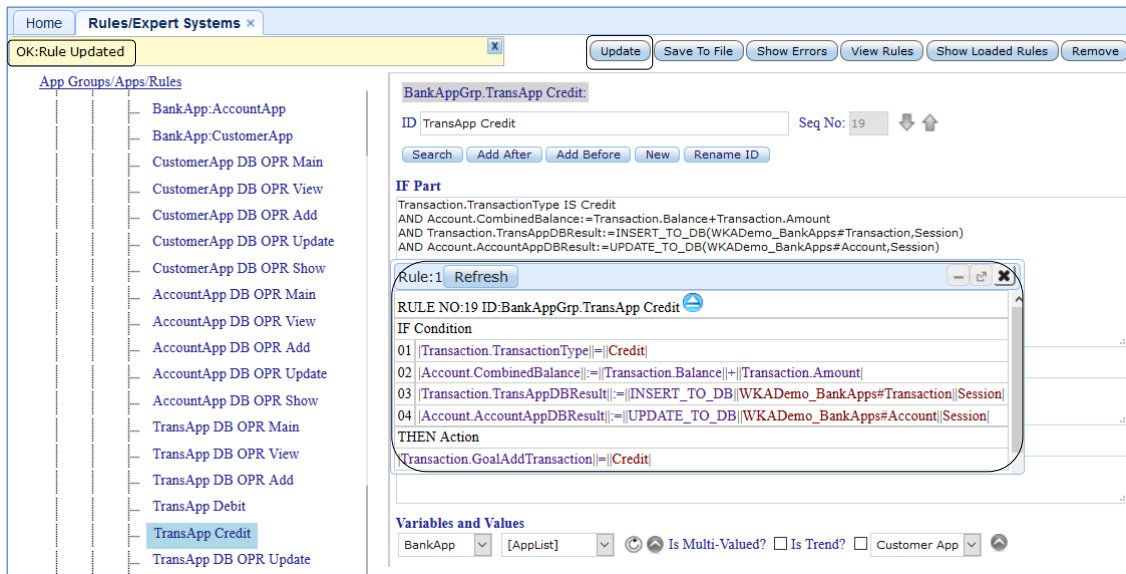


Figure 97: Updating the Rule

31. To view the changes, run the application. Select app (*Trans App*) from the list of apps and click *Next*.

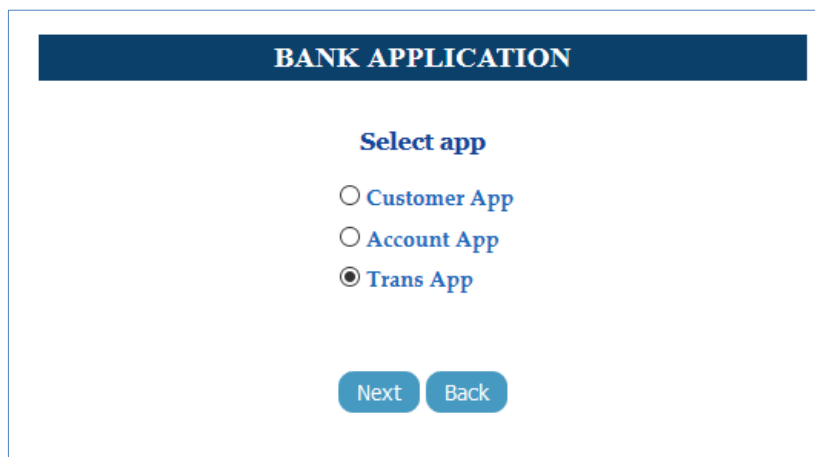


Figure 98: Selecting App

32. Select *Add* operation from *Select DB Operation* dropdown menu and click *Next*.

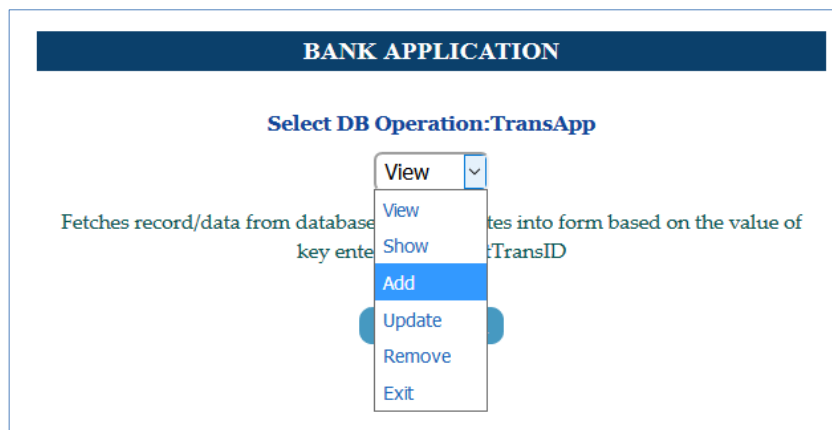


Figure 99: Selecting DB Operation

33. Enter account ID in *Account ID* field and click *Next*.

The screenshot shows a form titled "BANK APPLICATION" with a dark blue header. Below the header, the text "Account ID" is displayed above a text input field containing the number "5". At the bottom of the form, there are two buttons: "Next" and "Back".

Figure 100: Entering Account ID

34. Enter transaction details and select *Transaction Type* as *Credit*. Click *Next*.

The screenshot shows a form titled "BANK APPLICATION" with a dark blue header. Below the header, there is a table with the following data:

Transaction Data	
Account ID	5
Transaction Type	<input type="radio"/> Debit <input checked="" type="radio"/> Credit
Date Of Transaction	12/05/2017
Amount	2000
Mode	<input type="radio"/> Cash Withdrawal <input checked="" type="radio"/> Cheque
AccntTransIDInfo	Press 'Back' button to view another record

At the bottom of the form, there are two buttons: "Next" and "Back".

Figure 101: Selecting Transaction Type

35. Select DB Operation as *Main DB operation menu* and click *Next*.

The screenshot shows a form titled "BANK APPLICATION" with a dark blue header. Below the header, the text "DB Operation Result:OK:1" is displayed, followed by "What you want to do?". There are two radio button options: "Main DB operation menu" (which is selected) and "Exit DB operations". Below these options, the text "Click on 'Back' button to go back to Main DB Menu" is displayed. At the bottom of the form, there are two buttons: "Next" and "Back".

Figure 102: DB Operation Status

36. After clicking *Next*, session report is displayed.

BANK APPLICATION	
ApplicationDirectory	BankAppGrp
WelcomeText	
AppList	Trans App
Transaction DB Operation	Add
AccountID	5
AccountID	5
MaxAcctTransID	32
AcctTransID	32
DateOfTransaction	12/05/2017
TransactionType	Credit
Amount	2000
Mode	Cheque
TransAppFormVar	Complete
Balance	82000
CombinedBalance	84000
Transaction DB Result	OK:1

Figure 103: Session Report

As shown in above session report, balance for Account ID: 5 is 82000 which is credited by Amount 2000 so *CombinedBalance* is 84000.

37. For Debit transaction, account balance should not be zero. Minimum balance should be Rs.500 so add condition in *Debit* rule.

The screenshot shows a web-based interface for configuring rules. On the left, a tree view lists various application rules, with 'TransApp Debit' selected. The main area displays the configuration for this rule:

- Rule Name:** BankAppGrp.TransApp Debit
- ID:** TransApp Debit
- Seq No:** 18
- IF Part:**

```

Transaction.TransactionType IS Debit
AND Account.CombinedBalance:!=Transaction.Balance-Transaction.Amount
AND Transaction.TransAppDBResult:="Minimum balance should be 500Rs."
(AND IF(Account.CombinedBalance>=500,DebitIF) )
AND Transaction.TransAppDBResult:=INSERT_TO_DB(WKADemo_BankApps#Transaction,Session)
AND Account.AccountAppDBResult:=UPDATE_TO_DB(WKADemo_BankApps#Account,Session)
AND END_IF DebitIF

```
- THEN Part:** Transaction.GoalAddTransaction IS Debit
- ELSE Part:** (Empty)
- Variables and Values:** Transaction [AcctTransID] (Is Multi-Valued? Is Trend?)

Figure 104: Adding Rule for Debit Transaction

38. To update the rule, click *Update*.

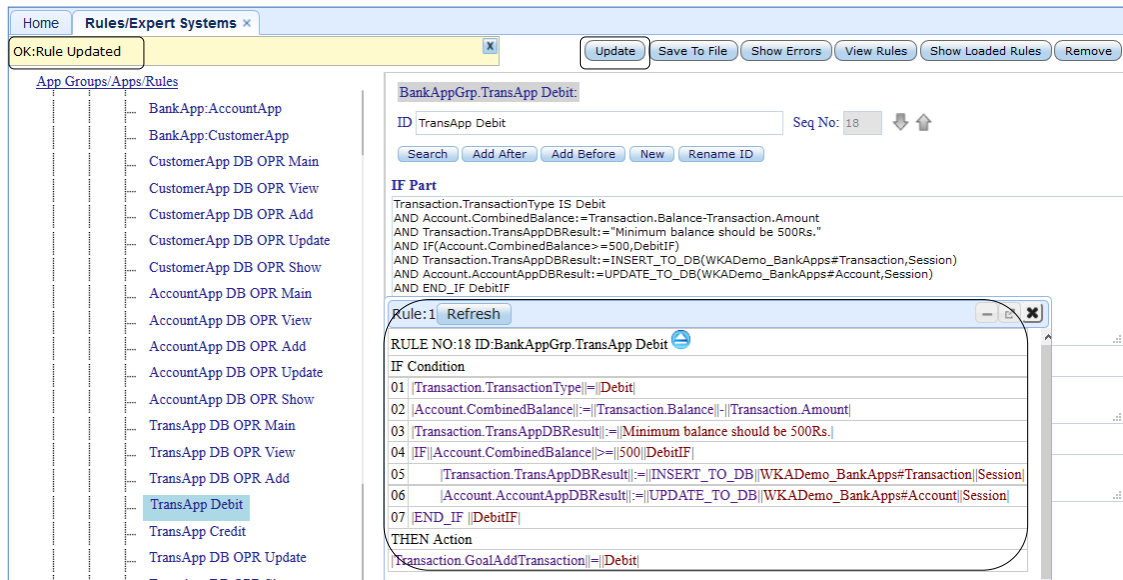


Figure 105: Updating the Rule

39. To view the changes, run the application. Select app as *Trans App* from the list of apps.

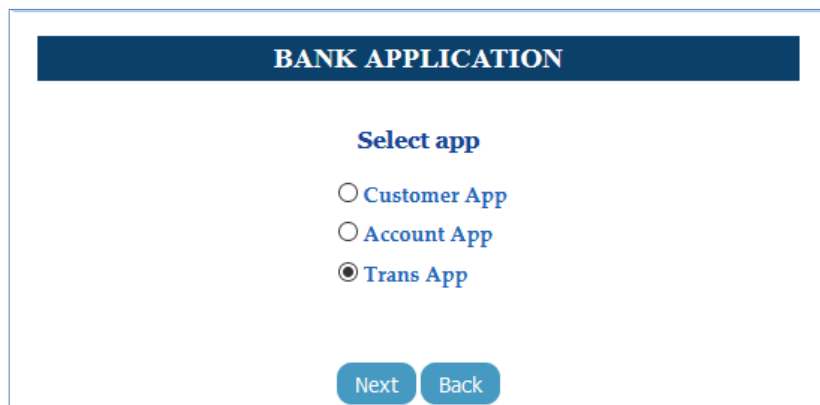


Figure 106: Selecting App

40. Select *Add* operation from *Select DB Operation* dropdown menu and click *Next*.

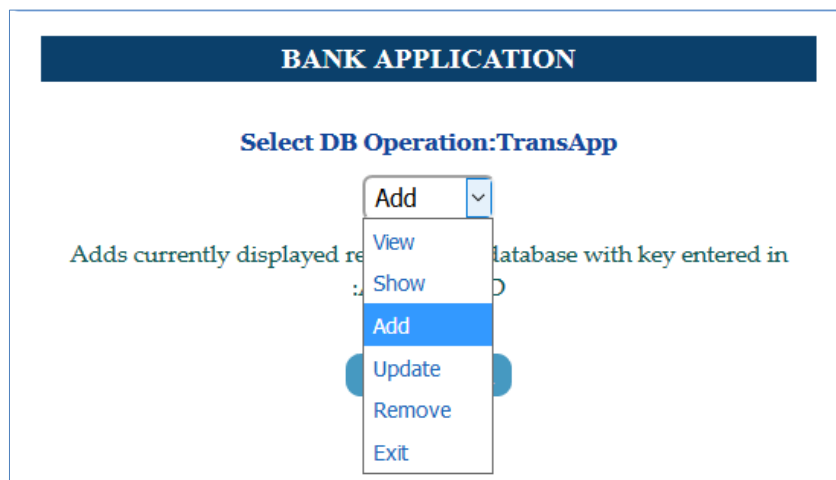


Figure 107: Selecting App

41. Enter account ID in *Account ID* field and click *Next*.

The screenshot shows a dark blue header with the text "BANK APPLICATION". Below the header, the text "Account ID" is displayed in blue. Underneath, there is a white input field containing the number "6". At the bottom of the form, there are two blue buttons: "Next" and "Back".

Figure 108: Entering Account ID

42. Enter transaction details and select *Transaction Type* as *Debit*. Click *Next*.

The screenshot shows a dark blue header with the text "BANK APPLICATION". Below the header, the text "Transaction Data" is displayed in bold. The form contains several fields: "Account ID" with the value "6", "Transaction Type" with radio buttons for "Debit" (selected) and "Credit", "Date Of Transaction" with the value "12/05/2017" and a calendar icon, "Amount" with the value "6000", and "Mode" with radio buttons for "Cash Withdrawal" (selected) and "Cheque". At the bottom, there is a field labeled "AccntTransIDInfo" which is empty. Below the form are two blue buttons: "Next" and "Back".

Figure 109: Entering Transaction Details

43. Select DB Operation as *Main DB operation menu* and click *Next*.

The screenshot shows a dark blue header with the text "BANK APPLICATION". Below the header, the text "DB Operation Result: Minimum balance should be 500Rs. What you want to do?" is displayed in bold. Below this, there are two radio button options: "Main DB operation menu" (selected) and "Exit DB operations". Below the options, there is a line of text: "Click on 'Back' button to go back to Main DB Menu". At the bottom, there are two blue buttons: "Next" and "Back".

Figure 110: DB Operation Status

44. To make the changes in show Rule (*TransApp DB OPR Show*), *AccountID* (This ID is for account table) is added and the filter of dynamic query called in *SHOW_REPORT* function is changed to *TransInfo.AccountID* as shown in the screen shots below.

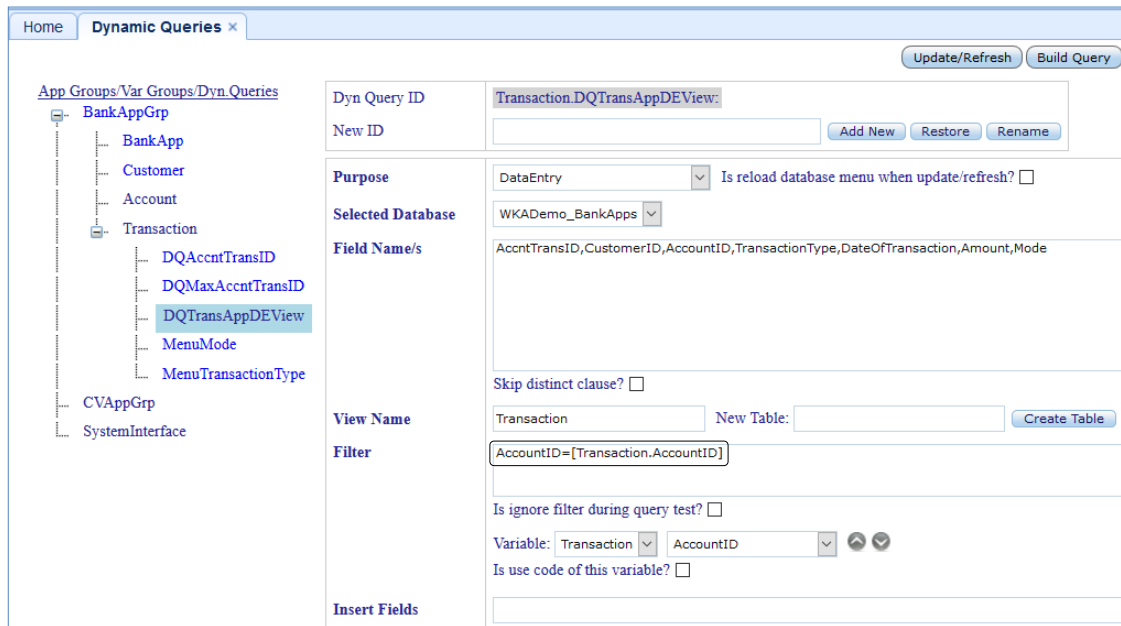


Figure 111: Dynamic Query Interface

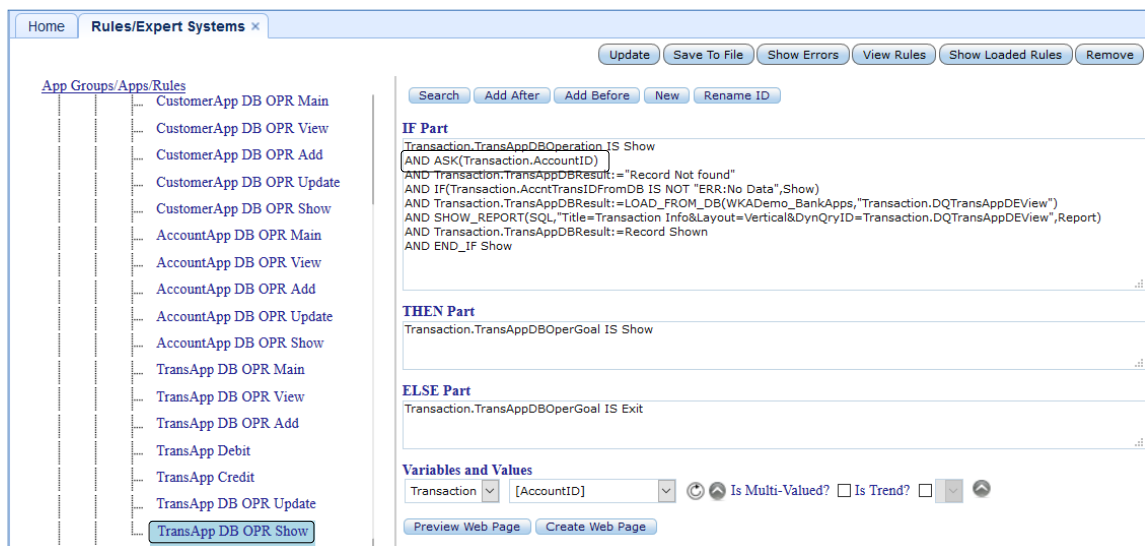


Figure 112: Making Changes in Transaction Show Rule

45. To update the rule, click *Update*.

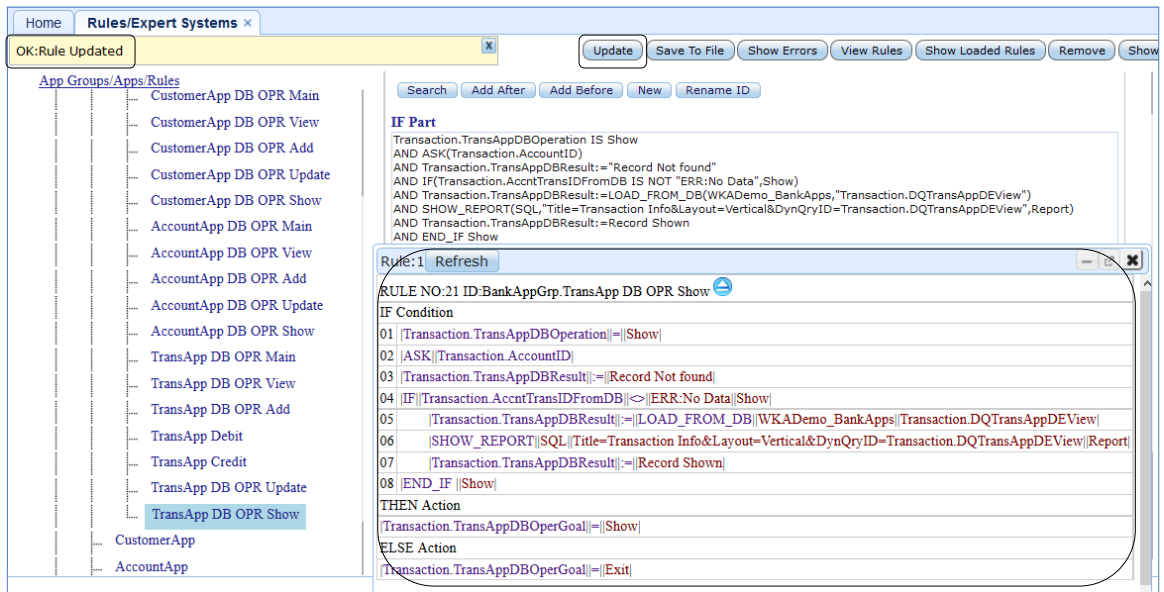


Figure 113: Updating the Show Rule

46. To view the changes, run the application, select *Show* from *DB Operation* dropdown menu.

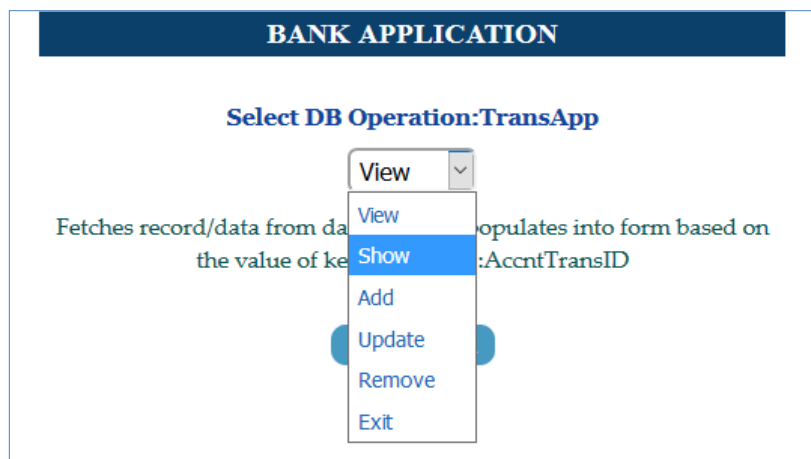


Figure 114: Selecting DB Operation

47. Enter *AccountID: 9* (enter ID whose record is present in the *transaction* table) and click *Next*.

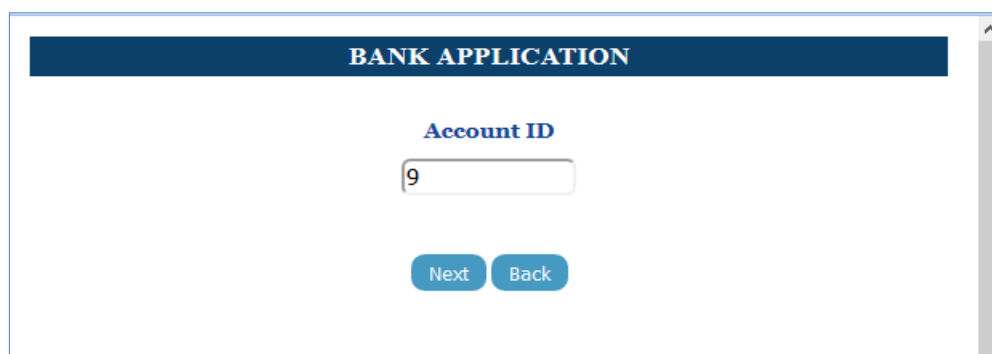


Figure 115: Entering Account ID

48. After clicking *Next*, session Report is displayed.

Transaction Info	
AcctTransID	36
CustomerID	104
AccountID	9
TransactionType	Debit
DateOfTransaction	15/05/2017
Amount	2000
Mode	Cash Withdrawal
#2	
AcctTransID	37
CustomerID	104
AccountID	9
TransactionType	Credit
DateOfTransaction	15/05/2017
Amount	4000
Mode	Cheque

Result? Continue Restart

Figure 116: Session Report for Transaction

AcctTransID	36
CustomerID	104
AccountID	9
TransactionType	Debit
DateOfTransaction	15/05/2017
Amount	2000
Mode	Cash Withdrawal
#2	
AcctTransID	37
CustomerID	104
AccountID	9
TransactionType	Credit
DateOfTransaction	15/05/2017
Amount	4000
Mode	Cheque

Next Back

Figure 117: Session Report for Transaction

49. Since record is available for *AccountID: 9* so the message shown is *DB Operation Result: Record Shown*. Select *Main DB operation menu* and click *Next*.

BANK APPLICATION

DB Operation Result: Record Shown
What you want to do?

Main DB operation menu
 Exit DB operations

Click on 'Back' button to go back to Main DB Menu

Figure 118: DB Operation Status

50. After clicking *Next*, session report for selected *AccountID* is shown.

BANK APPLICATION

ApplicationDirectory	BankAppGrp
WelcomeText	
AppList	Trans App
Transaction DB Operation	Show
AccountID	9
Transaction DB Result	Record Shown
AccntTransID	37
CustomerID	104
TransactionType	Credit
DateOfTransaction	15/05/2017
Amount	4000
Mode	Cheque
Transaction DB Operation Goal:TransApp	Show
GoBack	Transaction.TransAppDBOperation
TransApp	Done
SubAnnGoal	Trans Ann

Figure 119: Session Report

As shown in above report, for AccountID: 9 the required transaction information is shown.

1.7 Adding Dynamic Menu

The dynamic menu holds the value at *run-time*. Values are retrieved from database or from static menu based on filter. When the possible values of the variable are to be retrieved and populated from database at run-time, dynamic menu is linked to variable. For example, if a variable takes list of cities in a selected State, the value of variable depends upon State selected. A dynamic menu is assigned in such a case which retrieves list of cities by sending state name/code as filter. To add dynamic menu:

1. Go to *Main Menu* -> *Domain Vocabulary* -> *Menus & Lists*. Click *MenuCity* menu from *Customer* variable groups. By default, *Menu Parameters* option is shown.

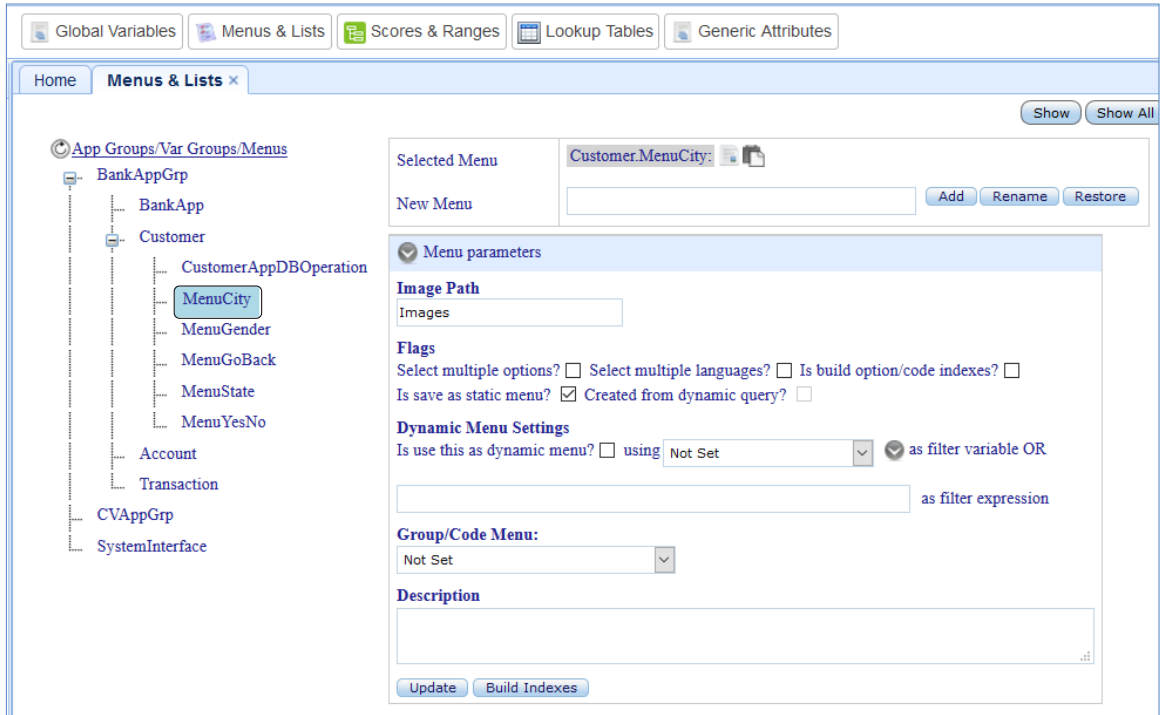


Figure 120: Menus & Lists Interface

- To use *MenuCity* as a dynamic menu, check the option *Is use this as dynamic menu?*

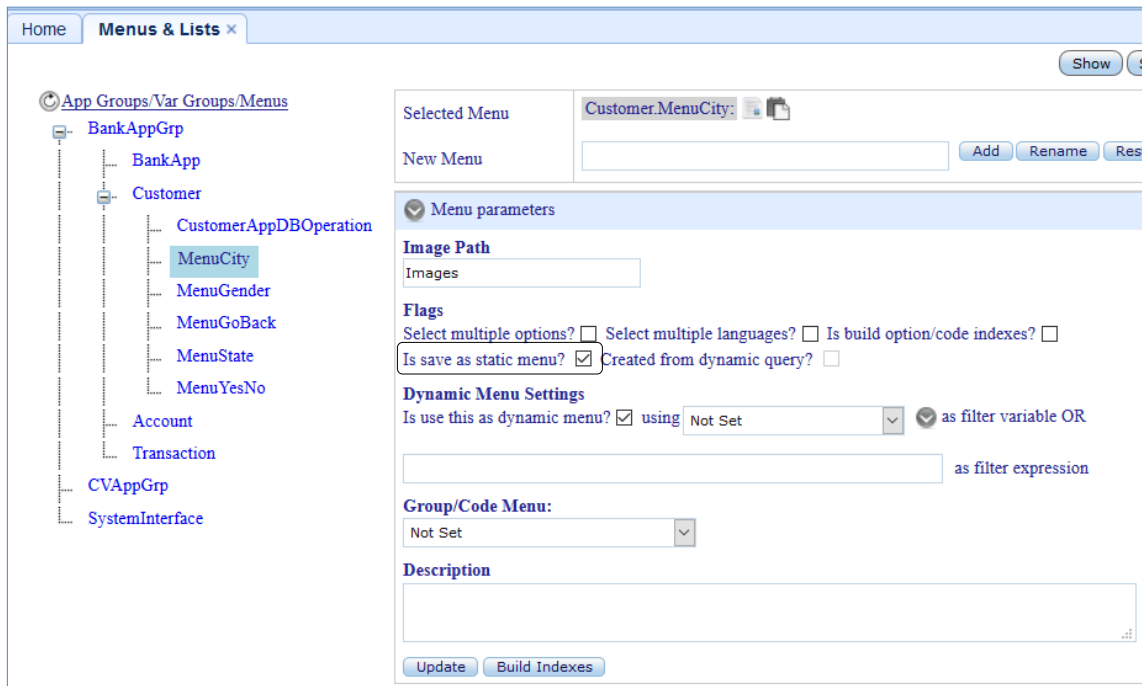


Figure 121: Menus & Lists Interface

- Select filter variable (*State*) from dropdown menu whose value is to be used as a filter at run time.

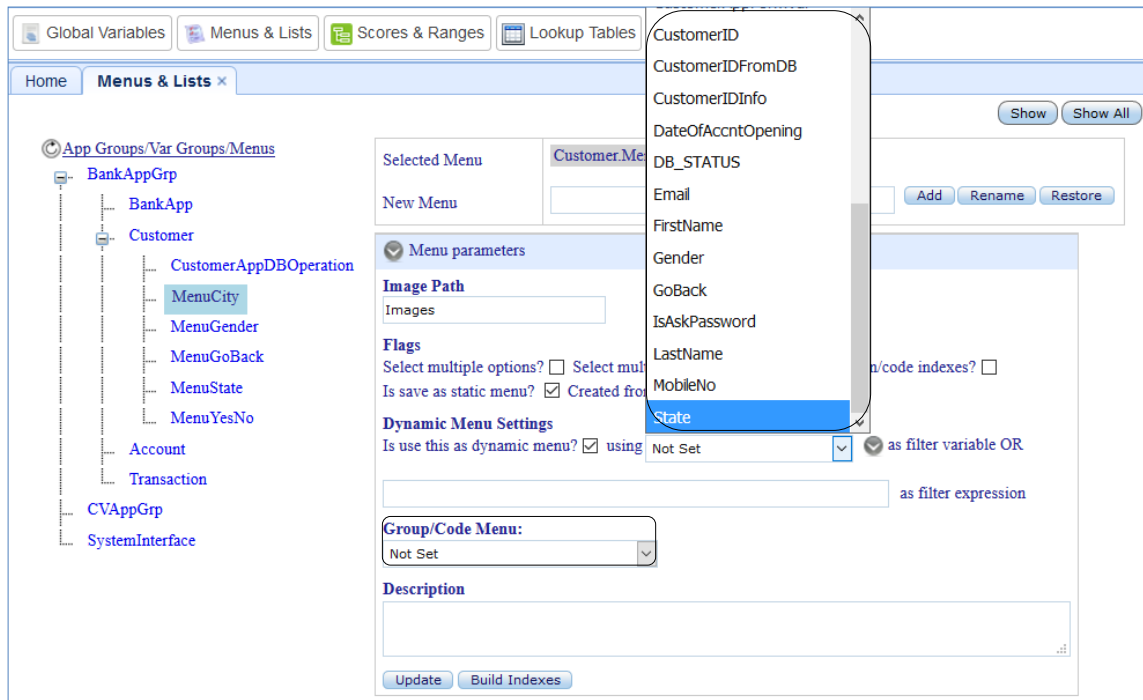


Figure 122: Selecting Filter Variable

4. Select group/code menu options (*Customer.MenuState*) from *Group/Code Menu* dropdown menu to set groups/classes or codes of current menu options.

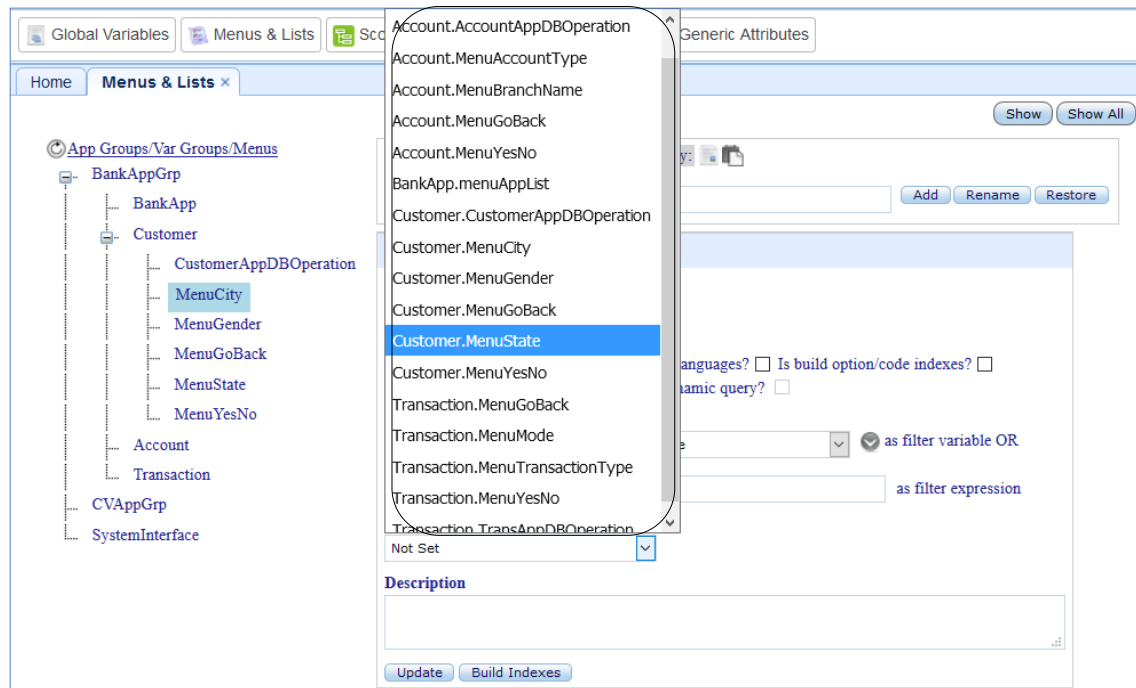


Figure 123: Selecting Group/Code Menu

5. To update the menu, click *Update*.

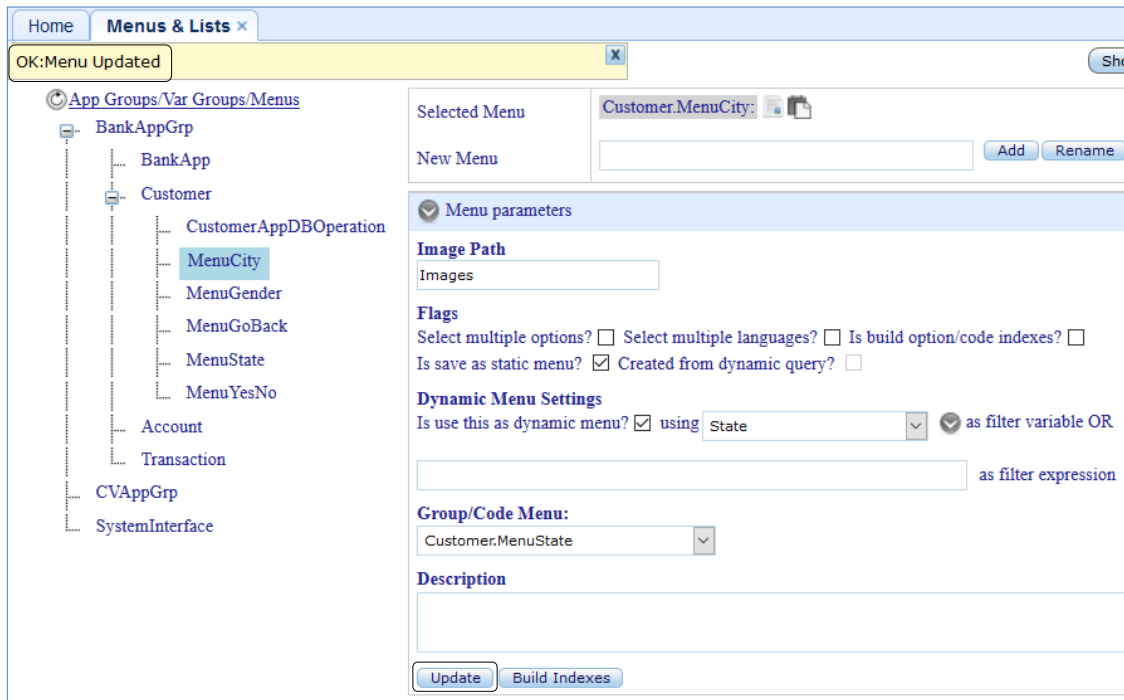


Figure 124: Updating the Menu

- Click *Menu Parameters* option and *Set Menu Parameters* option is shown. The menu options for which the filter variable is set is populated in *Code/Class/Group* dropdown menu. For Example: Select *Mumbai* from *Selected Option* dropdown menu for which the code is to be set. By default, *Code/Class/Group* is set as 1.

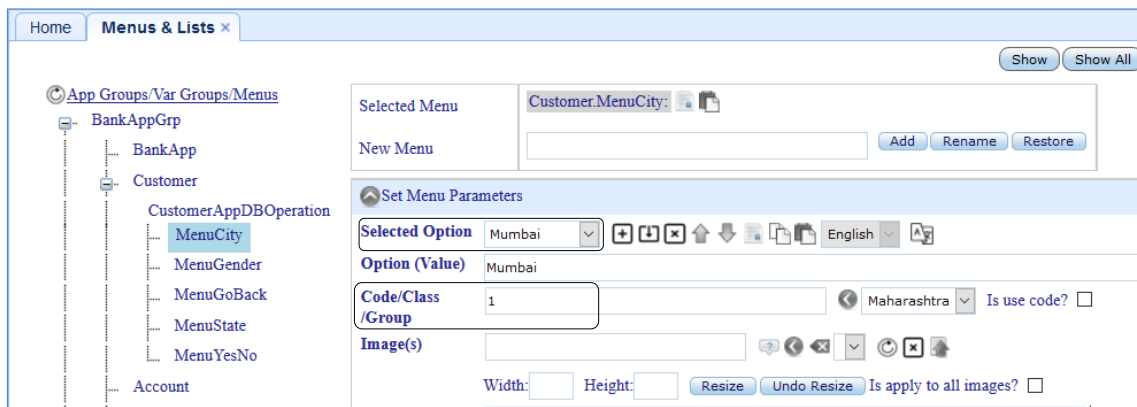


Figure 125: Set Menu Parameters Interface

- Select *Maharashtra* from *Code/Class/Group* dropdown menu based on which the code is displayed.

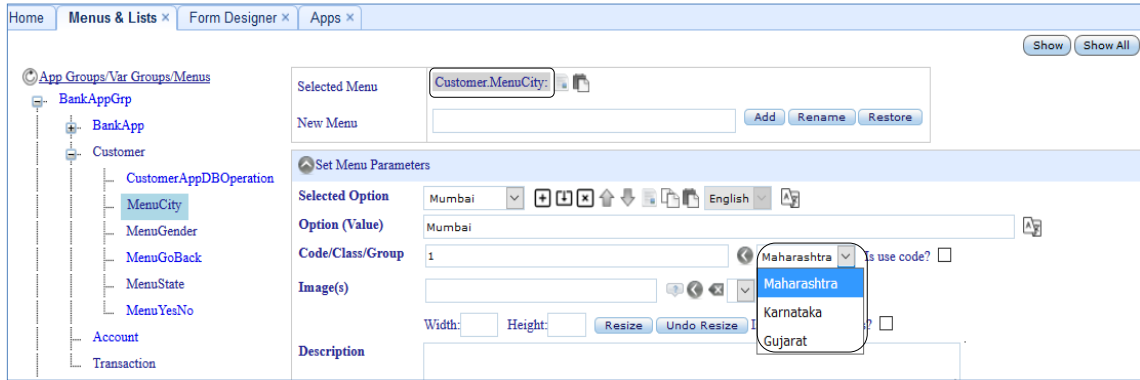


Figure 126: Selecting State

- To set the code for the city selected (*Mumbai*), check the option *Is use code?*

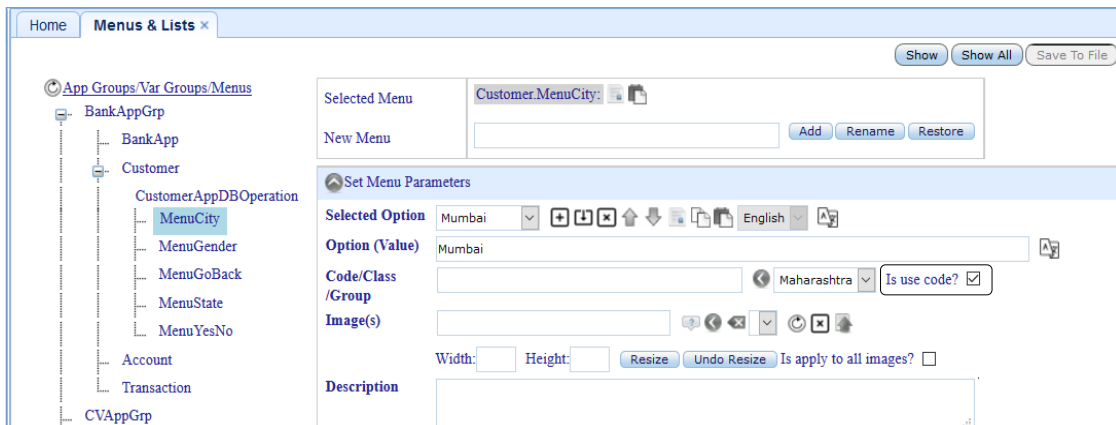



Figure 127: Checking the Option

- To insert the code (*MH*) of selected state in *Code/Class/Group* field, click the button .

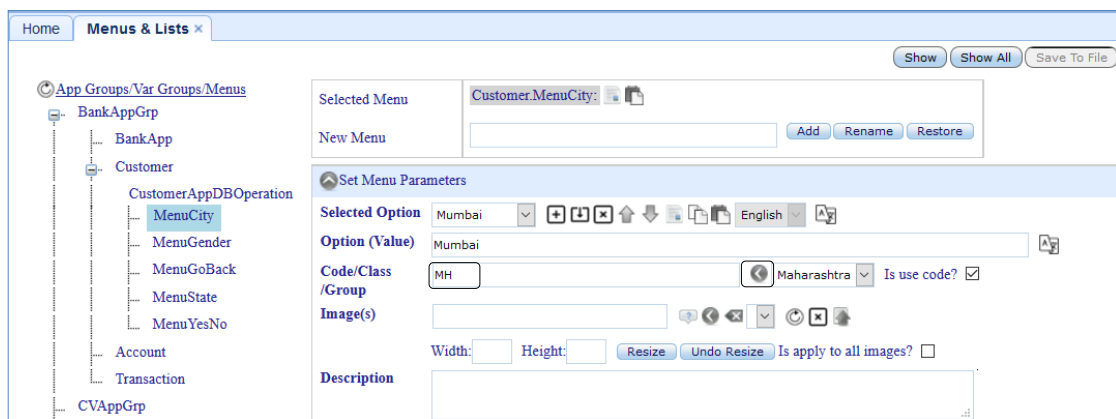


Figure 128: Inserting the Code

- To update the details, click the button  and the message *Ok* is displayed.

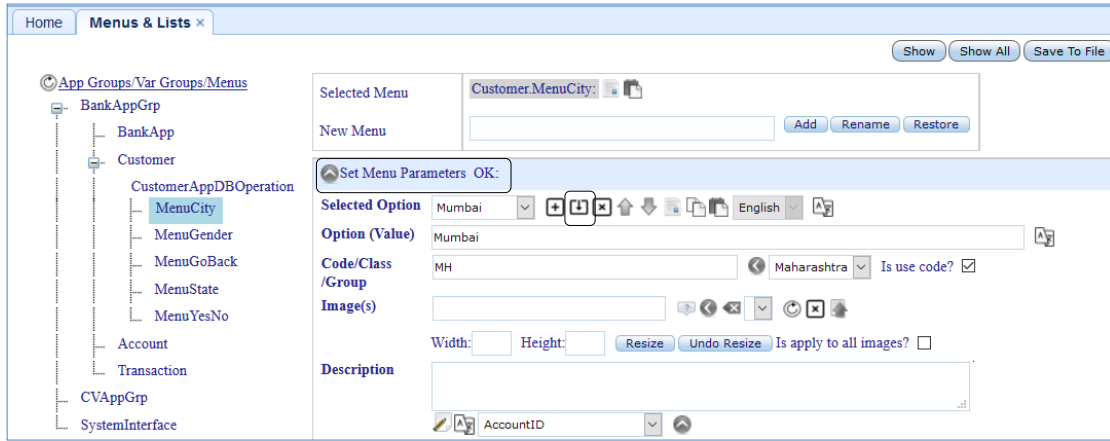


Figure 129: Updating Menu Option

- Similarly, follow the same steps as explained above to insert the code for city *Bangalore* and State *Karnataka*.

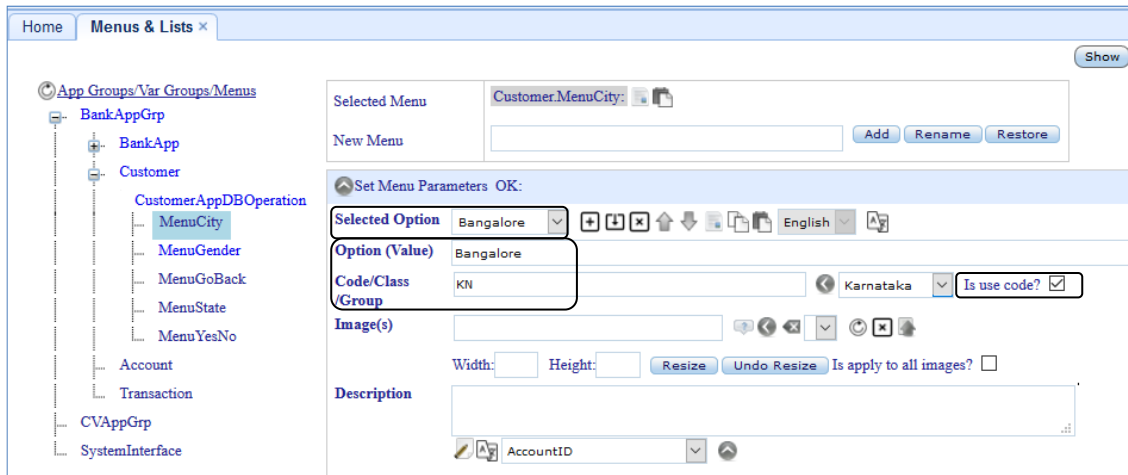


Figure 130: Inserting the Code for Bangalore

- To view the list of codes, click *Show* and the following screen is displayed.

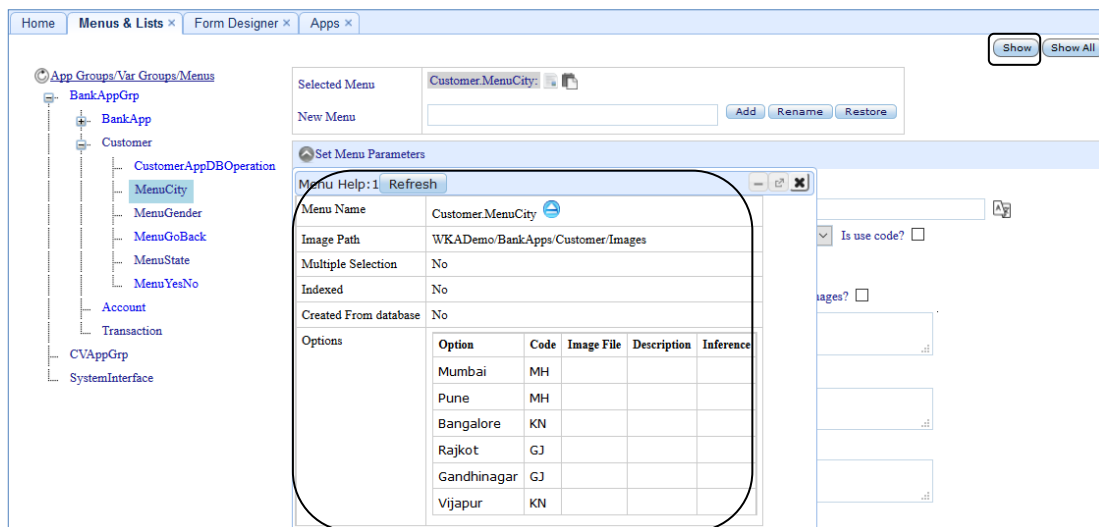


Figure 131: Viewing the List of Codes

13. The variable *Customer* is present in form therefore need to recreate the form in which that variable is used otherwise changes will not be reflected in the form. To recreate the form, go to *Main Menu -> Forms and Reports -> Form Designer* and click the option *Select Form/Report* from the list of options. Select the form created (*FormCustomerApp*) and select *Create*, the following message is displayed.

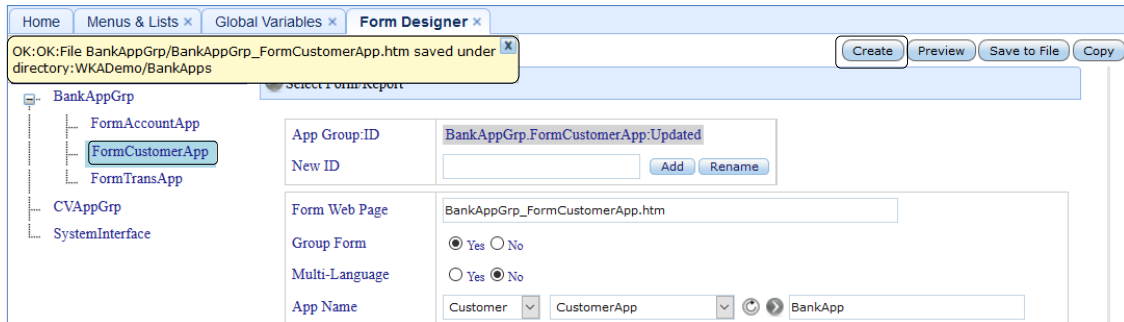


Figure 132: Recreating the Form

14. To view the changes, run the application. Select app (*Customer App*) from the list of apps and click *Next*.

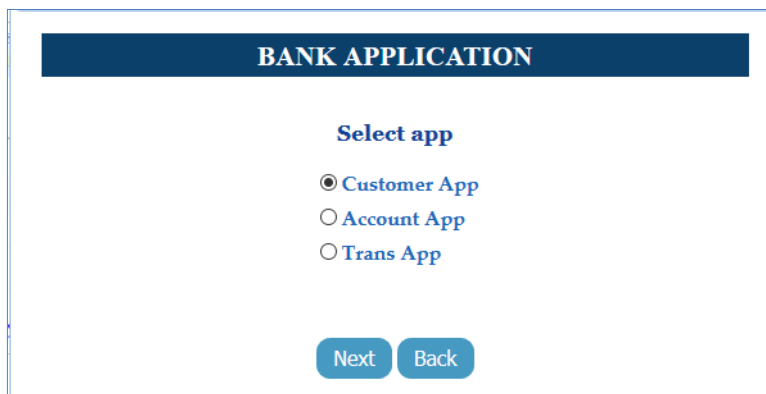


Figure 133: Selecting App

15. Select *Add* operation from *DB Operation* dropdown menu and click *Next*.

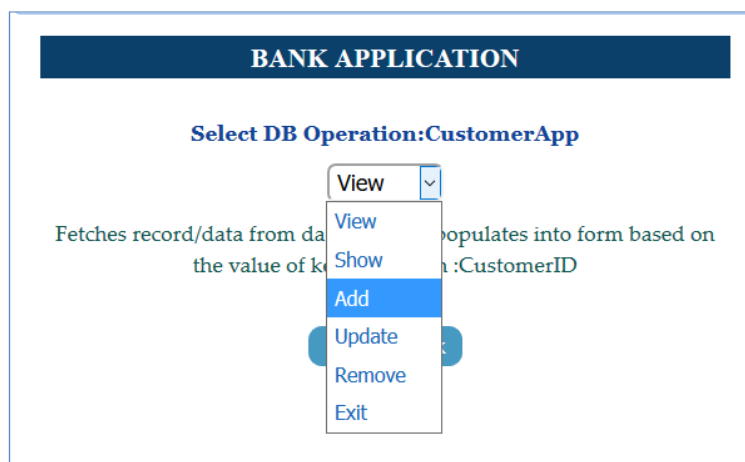
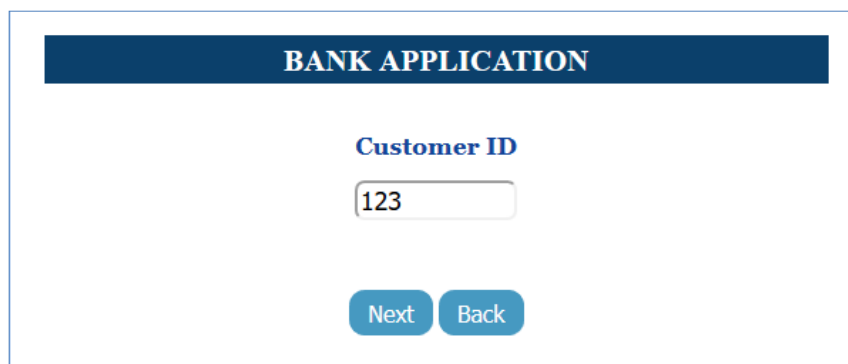


Figure 134: Selecting DB Operation

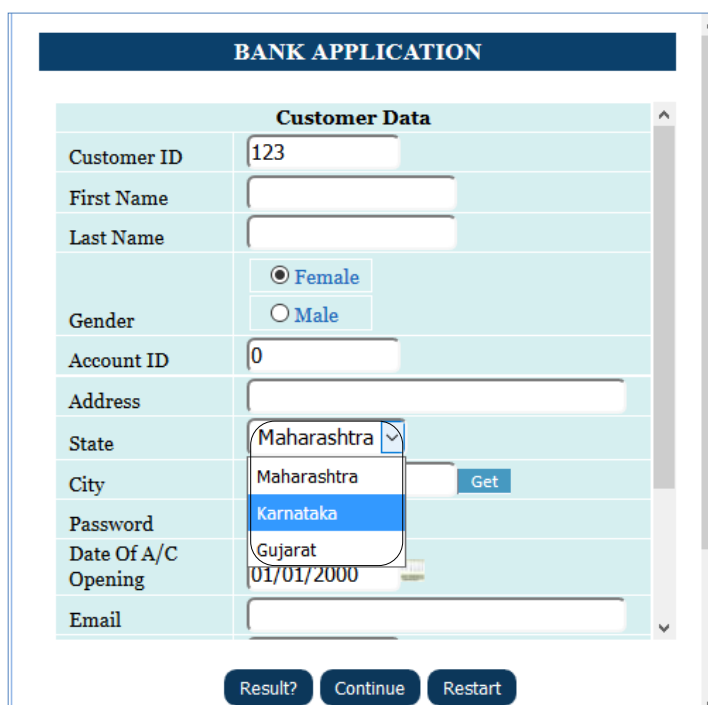
16. Enter the desired *Customer ID* and click *Next*.



The screenshot shows a web form titled "BANK APPLICATION". Below the title is a section labeled "Customer ID". There is a text input field containing the number "123". Below the input field are two buttons: "Next" and "Back".

Figure 135: Entering Customer ID

17. After clicking *Next*, *Customer Data* form is displayed. Select state (*Karnataka*) from *State* dropdown menu.



The screenshot shows a web form titled "BANK APPLICATION". Below the title is a section labeled "Customer Data". The form contains several fields: "Customer ID" (123), "First Name", "Last Name", "Gender" (radio buttons for Female and Male, with Female selected), "Account ID" (0), "Address", "State" (dropdown menu open showing Maharashtra, Karnataka, Gujarat), "City" (with a "Get" button), "Password", "Date Of A/C Opening" (01/01/2000), and "Email". At the bottom of the form are three buttons: "Result?", "Continue", and "Restart".

Figure 136: Selecting State

18. To view the cities of the state selected, click *Get*.

BANK APPLICATION

Customer Data

Customer ID: 123

First Name:

Last Name:

Gender: Female Male

Account ID: 0

Address:

State: Karnataka

City: Mumbai

Dropdown menu (City): Bangalore (selected), OK

Figure 137: Viewing the Cities

19. Select the desired city from dropdown menu.

BANK APPLICATION

Customer Data

Customer ID: 123

First Name:

Last Name:

Gender: Female Male

Account ID: 0

Address:

State: Karnataka

City: Mumbai

Dropdown menu (City): Bangalore (selected), Bangalore, Vijapur

Figure 138: Selecting City

20. Click *OK* and city of the selected state is shown.

Customer Data	
Customer ID	123
First Name	
Last Name	
Gender	<input checked="" type="radio"/> Female <input type="radio"/> Male
Account ID	0
Address	
State	Karnataka
City	Vijapur Get
Password	
Date Of A/C Opening	15/05/2017
Email	

Result? Continue Restart

Figure 139: Selecting City

21. Similarly, select State *Maharashtra*. Click *Get* and select the desired city from dropdown menu.

Customer Data	
Customer ID	123
First Name	
Last Name	
Gender	<input checked="" type="radio"/> Female <input type="radio"/> Male
Account ID	0
Address	
State	Maharashtra
City	Bangalore Get
City (Dropdown)	Mumbai Mumbai Pune
Password	
Date Of A/C Opening	

Figure 140: Selecting City

22. Select state Gujarat. Click *Get* and select the desired city from dropdown menu.

BANK APPLICATION	
Customer Data	
Customer ID	123
First Name	
Last Name	
Gender	<input checked="" type="radio"/> Female <input type="radio"/> Male
Account ID	0
Address	
State	Gujarat
City	Mumbai Get
Password	
Date Of A/C Opening	
	<div style="border: 1px solid black; padding: 5px;"><input type="text" value="Rajkot"/> Rajkot Gandhinagar</div>

Figure 141: Selecting City