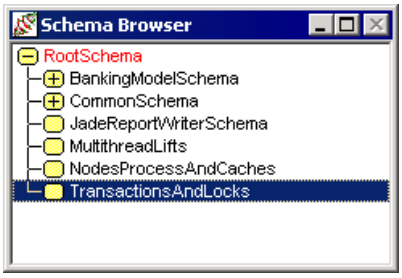
Practical Exercise 8

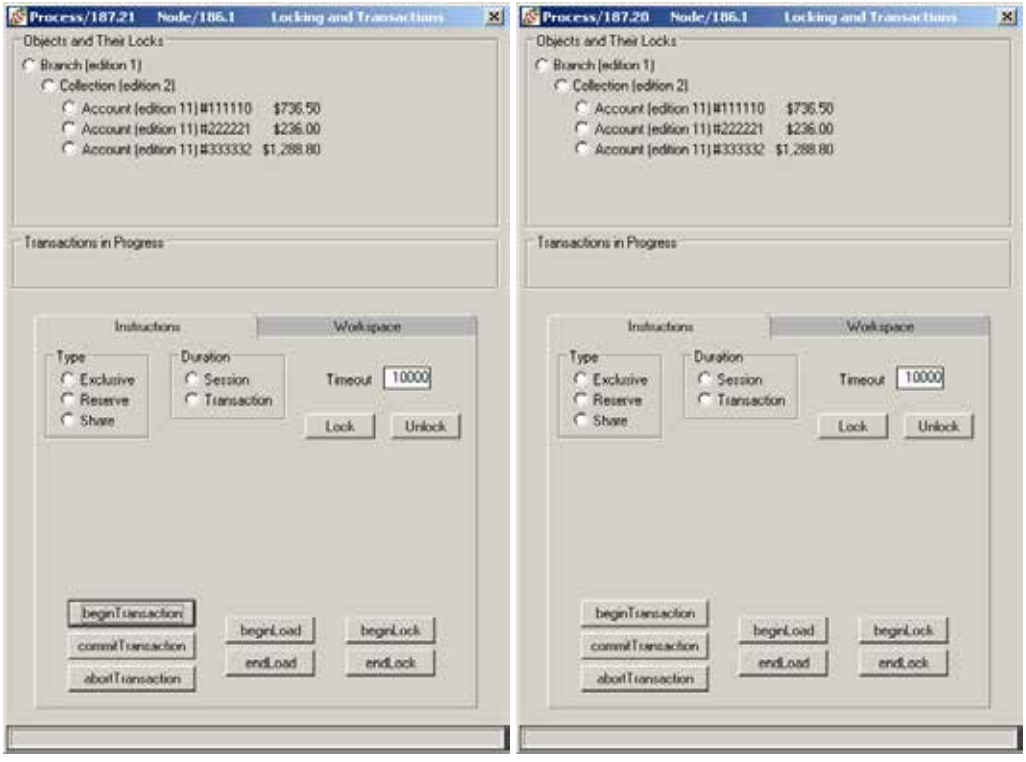
If you have not completed all previous exercises, refer to the JADE Developers Course, Practical Exercise 8 for details of schema files you can load at this point.

Running the Transactions and Locking Application

1. Switch to the Schema Browser by clicking the  button.
2. Select the **TransactionsAndLocks** schema as shown.



3. Right click on the  run application button. Two copies of the **TransactionsAndLocks** application start.



4. Arrange the two windows as shown above.

Lock Types

Try the following situations using the **Instructions** tab.

5. Use both applications and try to acquire a shared lock on the same object. Now do the same for a reserved lock. What happens? Try setting an exclusive on the same object. What type of locks can you set if an object already has a shared lock? What about a reserved or exclusive lock?

6. Summarize your results in the following table.

	Shared	Reserved	Exclusive
Shared	Both shared lock and entered transaction state	Both shared lock and entered transaction state	Account object was locked by first process and
Reserved	Both shared lock and entered transaction state	Account object was locked by first process and	Account object was locked by first process and
Exclusive	Account object was locked by first process and	Account object was locked by first process and	Account object was locked by first process and

Read Transactions

7. Select the **Example 3 – Read Transactions (1)** method from the combo box on the **Workspace** tab. Execute the method and take note of which objects are being locked.

8. Now select the **Example 4 – Read Transactions (2)** method from the combo box. Execute the method. Which objects are being locked now and why?

   In example 3 only the collection is being locked, while on example 4 each account of the collection is being locked. This happens because the example 4 method is accessing a property of each of the account objects.

9. Can you run the **Example 4 – Read Transactions (2)** method in both processes at the same time?

   Yes, unlike example 3.

Update Transactions

10. Select the **Example 1 – Update Transactions (1)** method from the combo box on the **Workspace** tab. Execute the method. Which objects are being locked and why?

    The collection has a shared lock while each of the accounts have an exclusive lock.

11. Can you run this method in both processes at the same time?

    No.

Collection Methods

12. Select the **Example 6 – Collection Methods** method from the combo box on the **Workspace** tab. Execute the method and note which objects are being locked.

    No object is locked.

Iterator Methods

13. Select the **Example 7 – Iterators (1)** method from the combo box on the **Workspace** tab. Execute the method and note which objects are being locked.

    Collection and accounts are locked.

14. Now select the **Example 8 – Iterators (2)** method and execute it. Which objects are being locked now and why?

    Only the collection is locked. Both methods produce the same output, however example 7 access a property of each account

15. Can you run the two methods at the same time?

    Yes

Lock Durations

16. On the **Instructions** tab, select an account object and lock it with a shared transaction lock. Click the **beginTransaction** button followed by the **commitTransaction** button. What happened?

    A shared lock was applied, the transaction began and once commited, the lock was removed.

17. Now select **Session** duration and repeat the transaction. What happened this time?

    A shared lock was applied, the transaction began but the lock was kept after the transaction was commited.

18. Try using the **abortTransaction** button instead of the **commitTransaction** button. Is there a difference in behavior?

    No.

19. Repeat the steps using **beginLoad** and **endLoad**. What did you observe?

    Even the shared transaction lock wasn't removed after endload.

20. Repeat the steps using **beginLock** and **endLock**. What did you observe?

    The same behaviour as with beginLoad and endLoad.

Deadlock

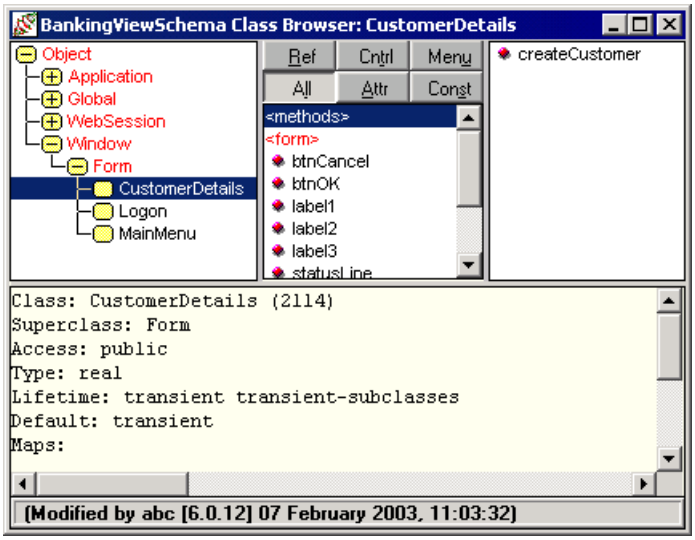
21. On the left side, lock the first account with a reserve lock. Have the second account selected. On the right side, lock the second account with a reserve lock. Have the first account selected. On the left side, click the Lock button, then quickly click the Lock button on the right side. What happens?

    The second process couldn't access the first account due to being

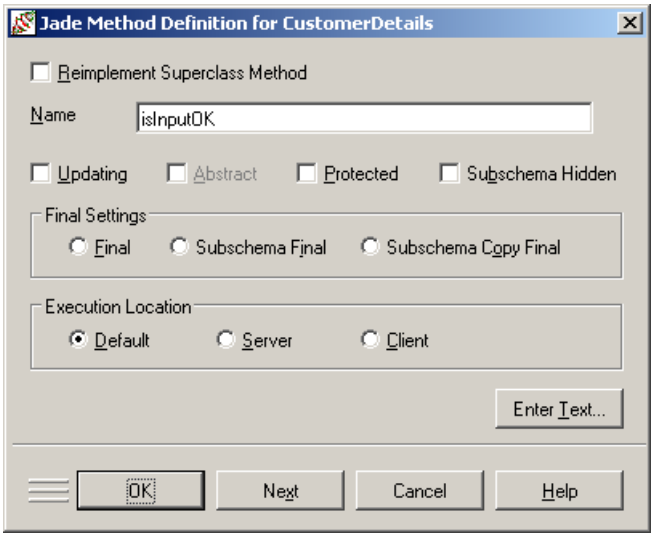
22. Close both copies of the **TransactionsAndLocks** application.

Code an isInputOK Method

23. In the BankingViewSchema, select the CustomerDetails class.



24. Select the **Methods | New Jade Method …** menu command. Complete the **Jade Method Definition** dialog as shown and click the **OK** button.



25. Code the method as shown below:

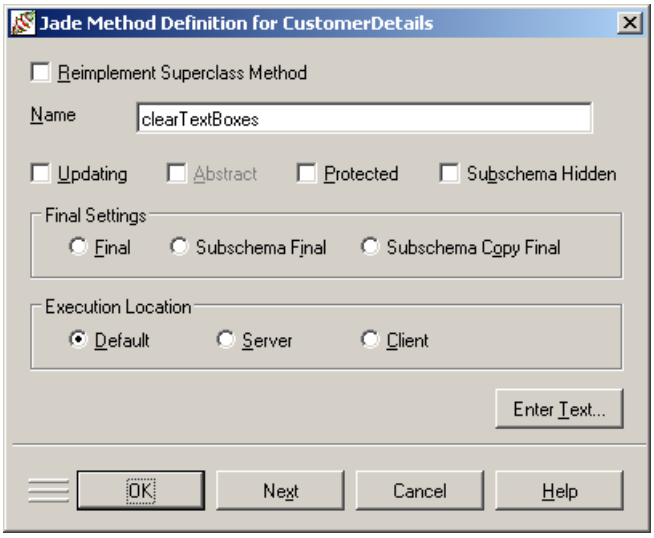
```
isInputOK() : Boolean;

vars
    ok : Boolean;
begin
    ok := true;
    if txtLastName.text = null then
        txtLastName.setFocus();
        statusLine.caption := 'Please enter a last name';
        ok := false;
    elseif txtFirstNames.text = null then
        txtFirstNames.setFocus();
        statusLine.caption := 'Please enter first names';
        ok := false;
    elseif txtAddress.text = null then
        txtAddress.setFocus();
        statusLine.caption := 'Please enter an address';
        ok := false;
    endif;
    return ok;
end;
```

26. Press the F8 key to compile the method. You should see a message **"Compilation complete – no errors".**

Code a clearTextBoxes Method

27. Select the **Methods | New Jade Method …** menu command. Complete the **Jade Method Definition** dialog as shown and click the **OK** button



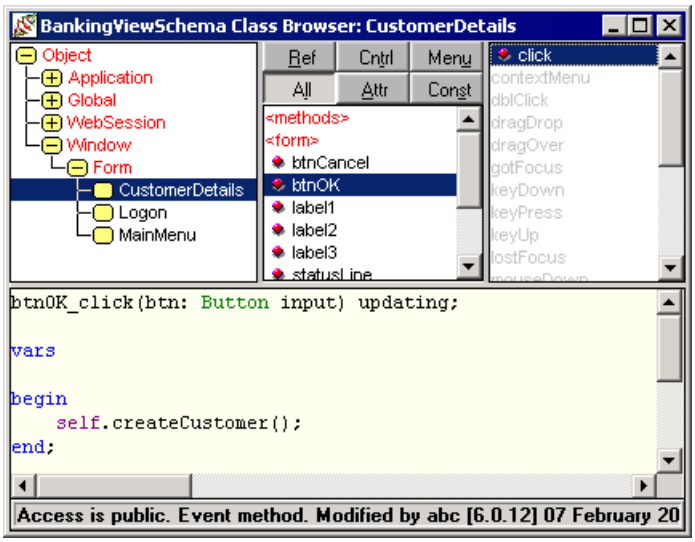
28. Code the method as shown below:

```
clearTextBoxes();
begin
    txtAddress.text := null;
    txtFirstNames.text := null;
    txtLastName.text := null;
end;
```

29. Press the F8 key to compile the method. You should see a message **"Compilation complete – no errors".**

Modify the btnOK_click Method

30. Select **btnOK** in the middle pane of the Class Browser, and select **click** in the right pane of the Class Browser as shown below.



31. Code the method as shown below:

```
btnOK_click(btn : Button input) updating;

begin
    if not self.isInputOK() then
```

```
            return;
        endif;

        self.createCustomer();
        self.clearTextBoxes();
        statusLine.caption := 'Customer added';
        txtLastName.setFocus();
    end;
```

32. Press the F8 key to compile the method. You should see a message **"Compilation complete – no errors".**

33. Open the app with your shortcut and try add in a new Customer with only a last name and first name. What happens?

    It asks for an address.