Module 12: Architecture of a Spring Application

Exercise 12.1 – Bank Application Dependency Injection

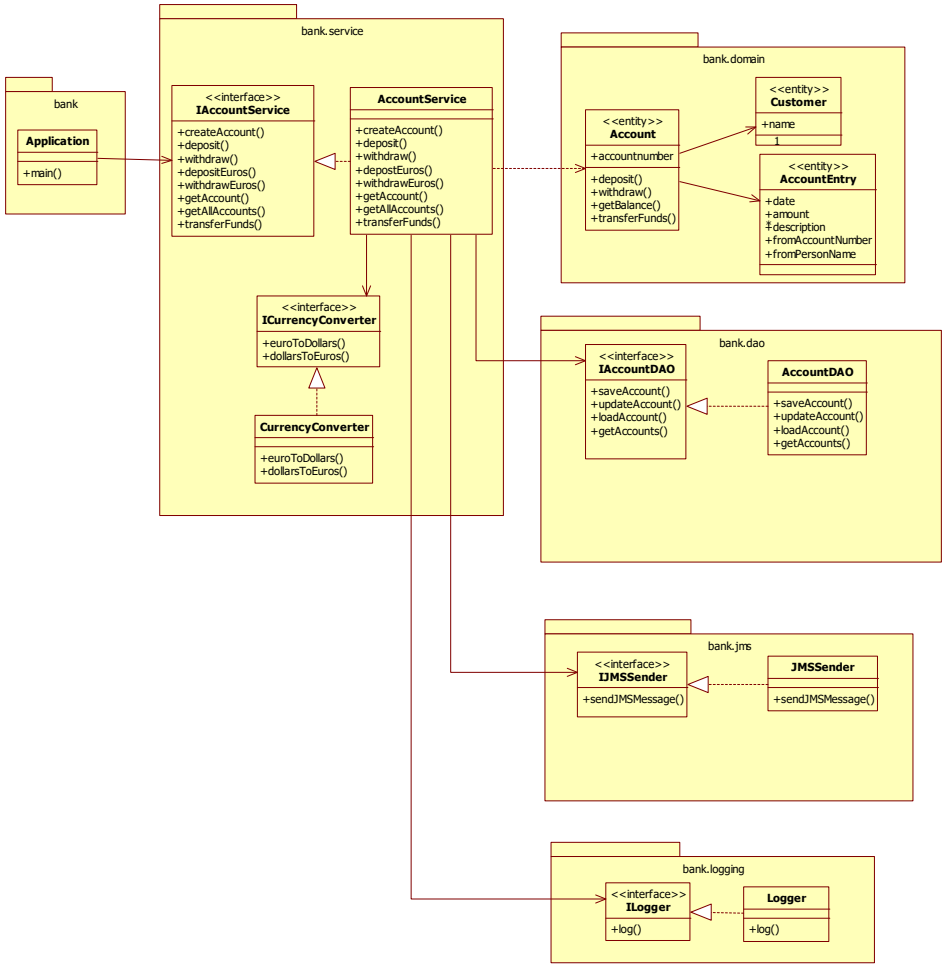
The Setup:

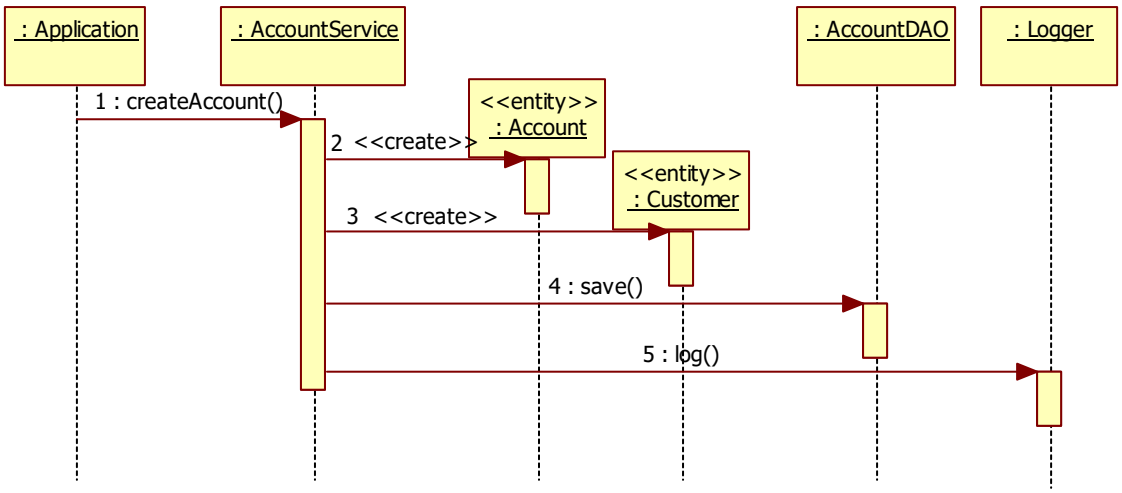
This exercise introduces the bank application. The bank application is a small application that embodies most of the architectural needs of a more real world enterprise application. Although the application that we start with in this exercise does not use any of Spring's features (yet), many areas in this application could benefit from them.

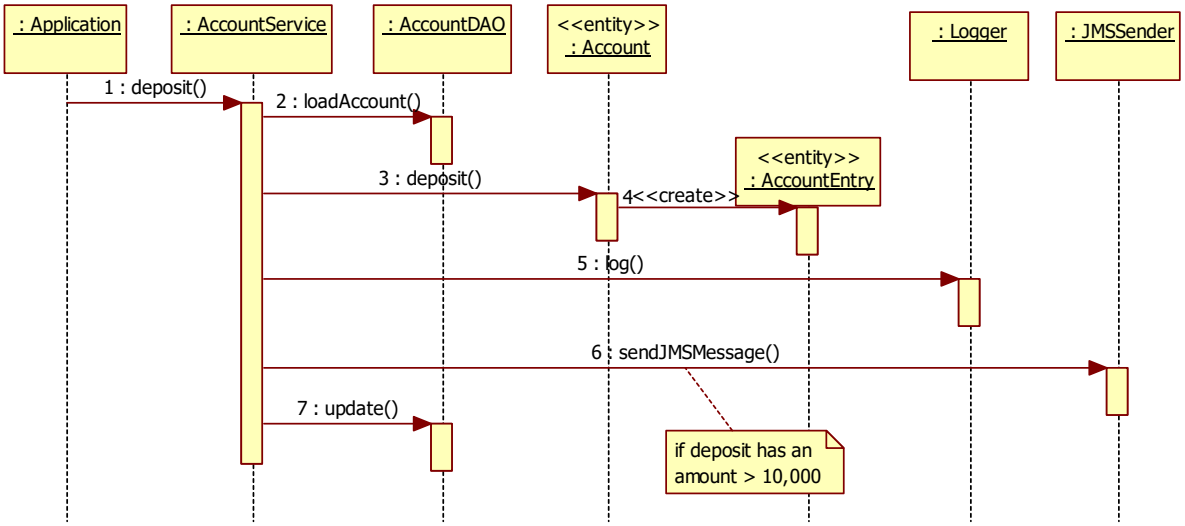
In this exercise, we will start by adding dependency injection to the application. In subsequent exercises we will continue to build on this, adding new features as they are covered.

Start by opening the project **exercise-12-1** you download sources from Azure (i.e., **C:\CS544\exercises\exercise-12-1**) and add the **Spring dependencies** to it.

The Application:







The bank application uses an AccountService object to perform the various bank-related services such as creating accounts, depositing money (Euros or Dollars), withdrawing money (Euros or Dollars), and transferring funds between accounts.

The AccountService object manipulates the domain objects Account, Customer and AccountEntry through the methods mentioned above. An Account object and a Customer object are created when CreateAccount() is called, and an AccountEntry is created for every deposit or withdrawal. All changes are then saved to the database through the AccountDAO.

Since the Accounts only work internally with dollar amounts, all euro deposits and withdrawals are first converted to dollars using a CurrencyConverter object.

All AccountService methods are logged through the Logger object, and whenever an amount greater than 10,000 dollars is deposited or transferred into an account, an additional JMS message is sent to the taxation services department.

Running **App.java** in the **exercise-12-1.bank** package should output:

```
Jun 12, 2009 11:45:24 PM bank.logging.Logger log
INFO: createAccount with parameters accountNumber= 1263862 , customerName= Frank Brown
Jun 12, 2009 11:45:24 PM bank.logging.Logger log
INFO: createAccount with parameters accountNumber= 4253892 , customerName= John Doe
Jun 12, 2009 11:45:24 PM bank.logging.Logger log
INFO: deposit with parameters accountNumber= 1263862 , amount= 240.0
Jun 12, 2009 11:45:24 PM bank.logging.Logger log
INFO: deposit with parameters accountNumber= 1263862 , amount= 529.0
CurrencyConverter: converting 230.0 dollars to euros
Jun 12, 2009 11:45:24 PM bank.logging.Logger log
INFO: withdrawEuros with parameters accountNumber= 1263862 , amount= 230.0
Jun 12, 2009 11:45:24 PM bank.logging.Logger log
INFO: deposit with parameters accountNumber= 4253892 , amount= 12450.0
JMSSender: sending JMS message =Deposit of $ 12450.0 to account with accountNumber=
4253892
CurrencyConverter: converting 200.0 dollars to euros
Jun 12, 2009 11:45:24 PM bank.logging.Logger log
INFO: depositEuros with parameters accountNumber= 4253892 , amount= 200.0
Jun 12, 2009 11:45:24 PM bank.logging.Logger log
INFO: transferFunds with parameters fromAccountNumber= 4253892 , toAccountNumber=
1263862 , amount= 100.0 , description= payment of invoice 10232
Statement For Account: 4253892
Account Holder: John Doe
-Date--------------------------Description-------------------Amount-------------
  Fri Jun 12 23:45:24 GMT 2009                       deposit            12450.00
  Fri Jun 12 23:45:24 GMT 2009                       deposit              314.00
  Fri Jun 12 23:45:24 GMT 2009      payment of invoice 10232              -100.00
--------------------------------------------------------------------------------
                                     Current Balance:            12664.00

Statement For Account: 1263862
Account Holder: Frank Brown
-Date--------------------------Description-------------------Amount-------------
  Fri Jun 12 23:45:24 GMT 2009                       deposit              240.00
  Fri Jun 12 23:45:24 GMT 2009                       deposit              529.00
  Fri Jun 12 23:45:24 GMT 2009                      withdraw             -361.10
  Fri Jun 12 23:45:24 GMT 2009      payment of invoice 10232              100.00
--------------------------------------------------------------------------------
                                     Current Balance:              507.90
```

The Exercise:

Change the bank application in such a way that the Logger, CurrencyConverter, AccountDAO and JMSSender are injected into the AccountService, rather than being instantiated with ***new***. In other word, AccountService should no longer contain these lines:

```
accountDAO = new AccountDAO();
currencyConverter = new CurrencyConverter();
jmsSender =  new JMSSender();
logger = new Logger();
```

Also update **App.java** so that it retrieves the AccountService from the Spring context.