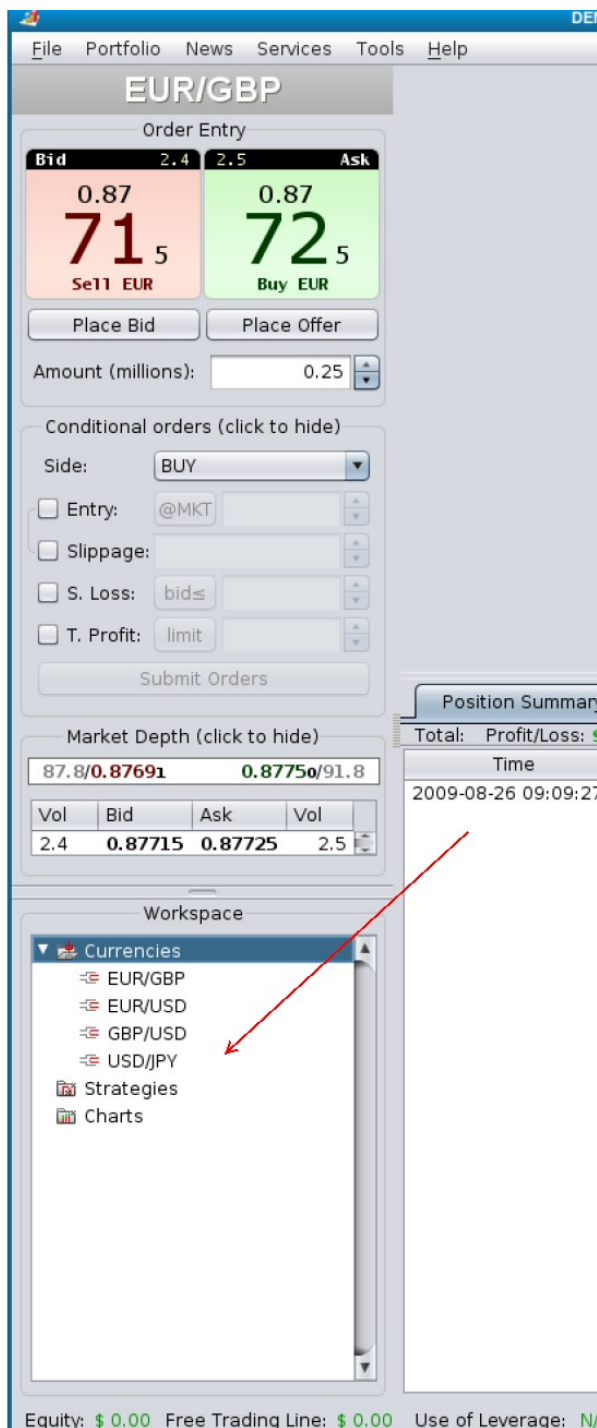


## How to open your strategy in Dukascopy JForex platform

First of all you need login into JForex platform using your username and password. Strategies management tree located in 'Workspace' panel which usually placed in left lower corner. You should see leafs '**Currencies**', '**Strategies**' and '**Charts**'. '**Strategies**' - this is what we looking for.



The screenshot displays the Dukascopy JForex platform interface for the EUR/GBP pair. The main window is titled 'EUR/GBP' and includes a menu bar with 'File', 'Portfolio', 'News', 'Services', 'Tools', and 'Help'. Below the menu bar is the 'Order Entry' section, which shows the current bid and ask prices (2.4 and 2.5) and the corresponding bid and ask rates (0.87715 and 0.87725). The 'Order Entry' section also includes buttons for 'Place Bid' and 'Place Offer', and a field for 'Amount (millions)' set to 0.25. Below the 'Order Entry' section is the 'Conditional orders (click to hide)' section, which includes a 'Side' dropdown set to 'BUY', and checkboxes for 'Entry', 'Slippage', 'S. Loss', and 'T. Profit'. Below the 'Conditional orders' section is the 'Market Depth (click to hide)' section, which shows a table of market depth data. The 'Market Depth' table has the following data:

Vol	Bid	Ask	Vol
2.4	0.87715	0.87725	2.5

Below the 'Market Depth' section is the 'Workspace' panel, which contains a tree view of the platform's structure. The tree view has the following structure:

- ▼ Currencies
  - EUR/GBP
  - EUR/USD
  - GBP/USD
  - USD/JPY
- Strategies
- Charts

A red arrow points to the 'Strategies' folder in the tree view. At the bottom of the interface, there is a status bar showing 'Equity: \$ 0.00', 'Free Trading Line: \$ 0.00', and 'Use of Leverage: N'.

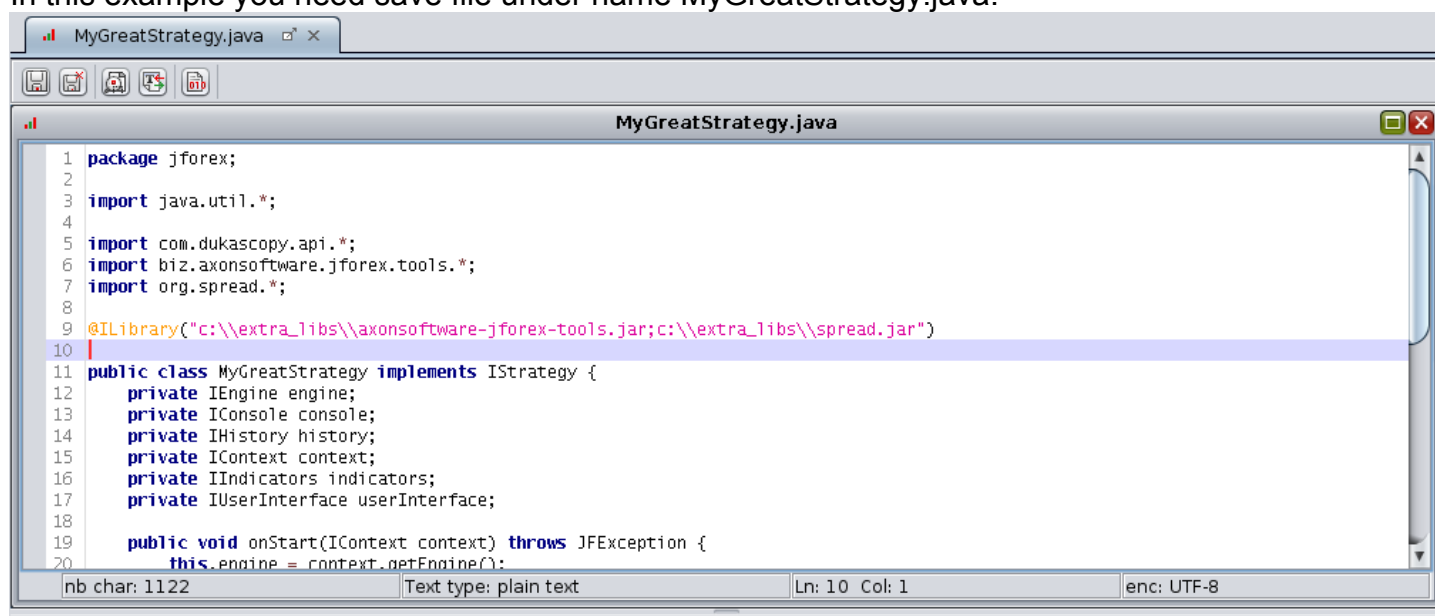
Lets say, you have code of your strategy opened somewhere in text editor from where you can simple copy and paste him.

Do right click on '**Strategies**' leaf and select '**New Strategy**' item. JForex platform will open new editor window with some initial strategy template. This template is useful when you need start development but as long we already have code we can dont heed him. Clear edit area by deleting template. Copy all code of your strategy into buffer(hint: to select all use hot key Ctrl+A) and paste him into JForex Platform editing area.

Once you did this save the strategy pressing 'Save' button or using hot key 'Ctrl+S'. Name of your file should be the same as name of main class of your strategy. You can get this name looking in the beginning for the code where you should find something like this:

```
public class MyGreatStrategy implements IStrategy {
```

....  
In this example you need save file under name MyGreatStrategy.java.



```

1 package jforex;
2
3 import java.util.*;
4
5 import com.dukascopy.api.*;
6 import biz.axonsoftware.jforex.tools.*;
7 import org.spread.*;
8
9 @ILibrary("c:\\extra_libs\\axonsoftware-jforex-tools.jar;c:\\extra_libs\\spread.jar")
10
11 public class MyGreatStrategy implements IStrategy {
12     private IEngine engine;
13     private IConsole console;
14     private IHistory history;
15     private IContext context;
16     private IIndicators indicators;
17     private IUserInterface userInterface;
18
19     public void onStart(IContext context) throws JFException {
20         this.engine = context.getEngine();

```

Another way add your strategy in platform is open saved file from hard drive. To to this select '**Open Strategy**' from menu and find file with code.

Now lets check is our code bug free and at least can be compiled. To do this right click on 'MyGreatStrategy' and select '**Compile**' Doing that you will transform your strategy in binary form acceptable for execution of JForex Platform and in same time do code verification for syntax errors. Depends on size of strategy compilation may take some time. Results(good or bad) you able to monitor in '**Messages**' tab which will change color of title once process is finished.

Imagine that we are leaving in ideal world and everything was ok this time. From that point we have 2 directions - do historical testing('Test') to verify logic or go trading('Run'). In most of the cases we suggest test what you have before trading. Yes, historical tester not cover full picture and sometimes his results far from the real market situation but he can show is behaviour of your strategy is meet your expectations.



Sometimes you need external libraries which was supplied with strategy code. In our example additionally to standard Java classes we use extra libraries: *axonsoftware-jforex-utils.jar* and *spread.jar*

To include this files in compilation you need provide full path to them in special tag `@Library` separated by ';' e.g. `@Library("c:\\extra_libs\\axonsoftware-jforex-utils.jar;c:\\extra_libs\\spread.jar")`  
Hint: use simple paths like `c:\\extra_libs`(or `/home/user/extra_libs` on Unix/MaxOS) this will reduce quantity of errors

Another important thing is adding symbols which is using in your strategy. Dukascopy JForex platform allow you manage subscriptions for symbols what mean that market updates will be send to you only if you subscribed.

If you strategy working with EUR/USD, USD/JPY and EUR/GBP - dont forget to include them in subscriptions.