Cryptocurrencies prices

As everybody's trading cryptocurrencies nowadays you just got hired to implement a cryptocurrencies register. For now you are only working with three types of currencies (BITCOIN, ETHEREUM and RIPPLE). On the main frame the user should be able to set the current price in $ for any of the already mentioned coins.

When adding another price for the same coin the current price on the main frame should get updated but pay attention that you should store somehow a history of the prices for that coin. When the *Show* button gets pressed a new frame should get created. In this new frame display the last prices that were set for that specific coin. Limit the number of prices shown in history to at most 5 even if your history for that coin might contain more than 5 entries. See the attached snapshots for further details.

Take into account to add validation messages in case something goes wrong (i.e. `NumberFormatException`).

No database required! Use suitable collections to store all the data you need.

Points to be followed

(1p) 1. Functionality of the final solution.
(1p) 2. GUI looks and works similar to the one provided in snapshots.
(1p) 3. Current price gets updated accordingly.
(1p) 4. History works in the next cases: 0 prices in history, 1 -> 5 prices in history and > 5 prices in history.
(1p) 5. Validation messages as expected.
(1p) 6. Application stops on X button pressed.

Notes: You've got 1 hour and 40 minutes to perform the required tasks. You are requested to create an executable jar from your project (Right click on your project > Export.. > Executable JAR File) and upload it to moodle once you have completed the tasks. You don't have to recreate the same structure or UI as provided in the snapshots; similar versions are accepted! Good luck!

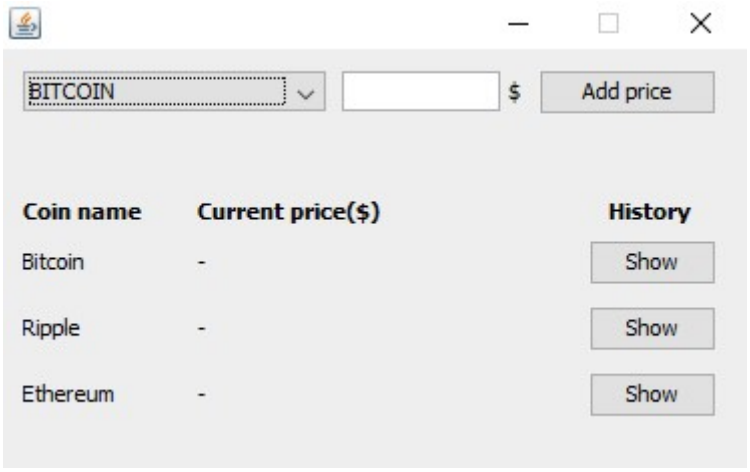

Figure 1: Main frame on starting the application

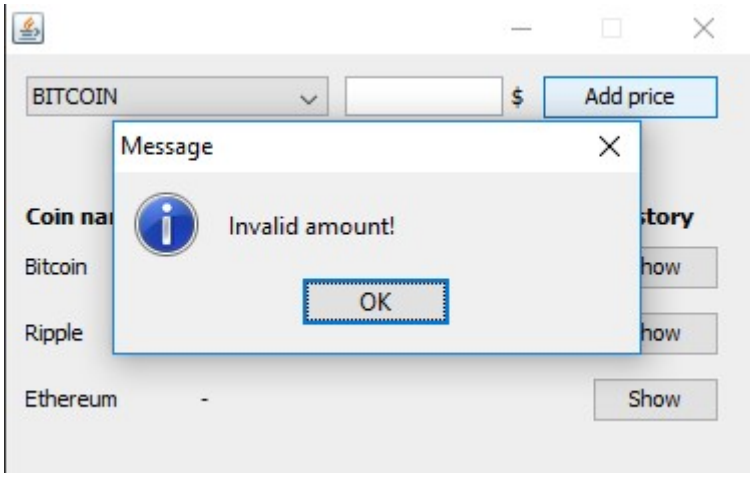

Figure 2: Validation message on invalid amount entered

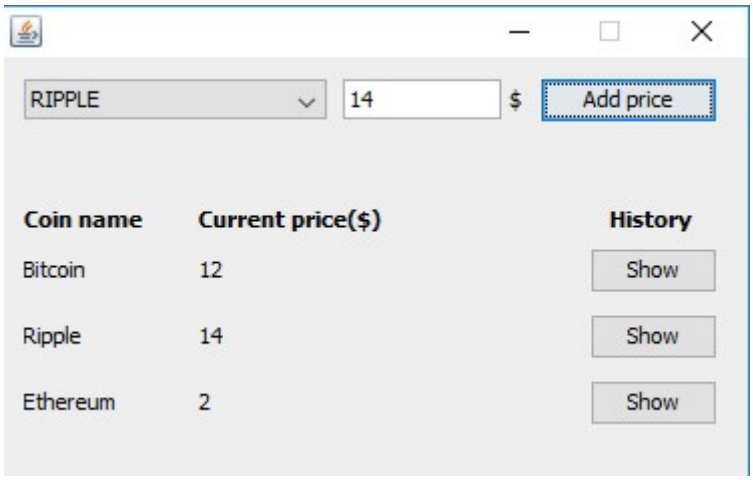

Figure 3: Successfully added prices displayed for each of the coins

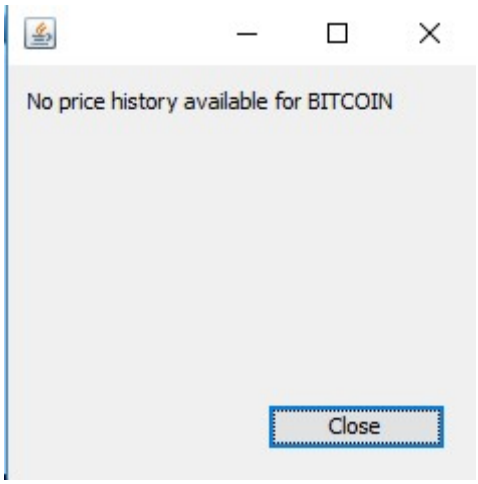

Figure 4: Coin price history if no prices set yet

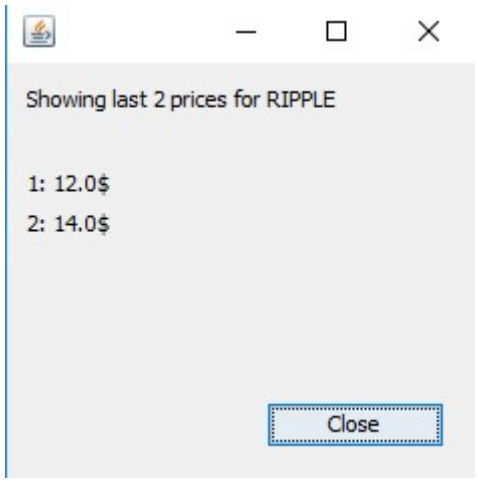

Figure 5: Coin price history containing the last maximum 5 prices

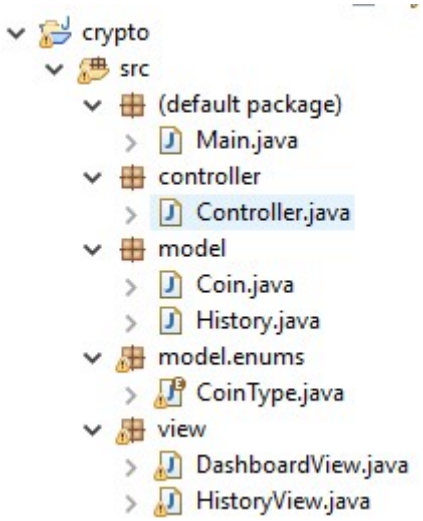

Figure 6: Example of structuring the code