



Gymnázium, Praha 6, Arabská 14

Praha 6, Arabská 14, 160 00

ROČNÍKOVÝ PROJEKT



Předmět: Programování

Téma: Paper trading application

Autor: Josef Mitošinka

Třída: 2.E

Školní rok: 2022/2023

Vyučující: Mgr. Jan Lána

Třídní učitel: Mgr. Blanka Hniličková

Čestné prohlášení:

Prohlašuji, že jsem jediným autorem tohoto projektu, všechny citace jsou řádně označené a všechna použitá literatura a další zdroje jsou v práci uvedené.

Tímto dle zákona 121/2000 Sb. (tzv. Autorský zákon) ve znění pozdějších předpisů uděluji bezúplatně škole Gymnázium, Praha 6, Arabská 14 oprávnění k výkonu práva na rozmnožování díla (§ 13) a práva na sdělování díla veřejnosti (§ 18) na dobu časově neomezenou a bez omezení územního rozsahu.

V Praze dne 2.5.2022

.....

Josef Mitošinka

Anotace(česky):

Tématem mé ročníkové práce bylo udělat paper trading aplikaci. Program je napsána v programovacím jazyku Java. Jedná se o aplikaci která umožňuje uživatelům kupovat a prodávat kryptoměny, jež odpovídají reálným burzovním cenám. Veškeré nákupy jsou pouze "papírové", tudíž se při obchodování nepoužívají opravdové finanční prostředky.

Abstract(english):

The topic of my Annual's work was to make a paper trading application. The program is written in the Java programming language. It is an application that allows users to buy and sell cryptocurrencies that correspond to real market prices. All purchases are only "paper", so no real funds are used while trading.

Zadání:

Zadáním mé letošní ročníkové práce bylo udělat aplikaci v programovacím jazyku java, ve které budou uživatelé moci obchodovat s libovolnými kryptoměnami. Prodávat a kupovat. Uživatel dostane 1000\$, se kterými bude moci obchodovat. Veškeré ceny kryptoměn budou podle reálných burzovních cen.

Obsah

1 Úvod	1
2 Program	2
2.1 Uživatelské grafické rozhraní	2
2.2 Funkcionalita programu	4
2.2.1 onPortfolioButtonClick()	4
2.2.2 accountBalance()	4
2.2.3 saveCurrency()	5
2.2.4 updateBalanceOnBuy() + updateBalanceOnSell()	5
2.2.5 checkCurrency()	6
2.2.6 onBuyButtonClick() + onSellButtonClick()	6
2.2.7 onSearchBarAction()	8
2.2.8 currentPrice()	9
3 Závěr	10

1 Úvod

Tématem letošní ročníkové práce bylo napsat program v programovacím jazyku Java.

Jedním z hlavních problémů mnohých obchodníků s kryptoměnami je výdělečnost. Najít dobrou strategii, která vám je schopná dlouhodobě vydělávat finance je velmi obtížné a kvůli tomu je pouze jedno procento všech obchodníků na burzách výdělečné. Proto jsem vytvořil jsem aplikaci s názvem Riga trading. Jedná se o paper trading aplikaci, která umožňuje uživatelům kupovat a prodávat kryptoměny v reálném čase bez rizika na finančních ztrátách. Tuto dokumentaci, která popisuje moji aplikaci pro simulování obchodování s kryptoměnami v jazyce Java s použitím JavaFX, jsem vytvořil ve vývojovém prostředí IntelliJ IDEA.

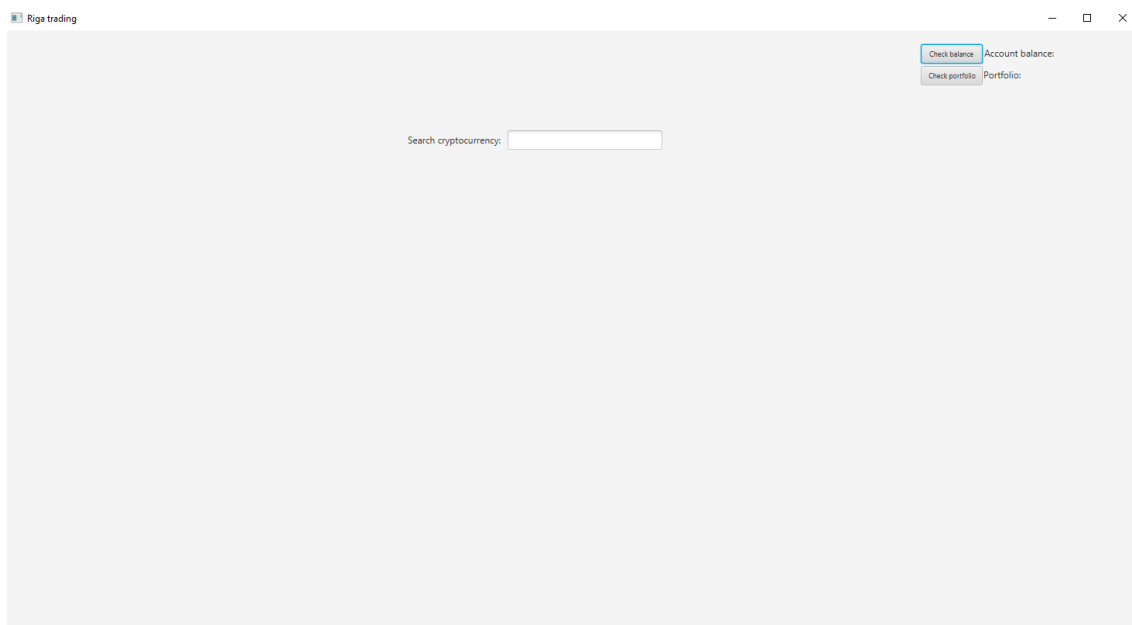
Aplikace umožňuje uživatelům sledovat aktuální ceny kryptoměn, nakupovat a prodávat kryptoměny a sledovat stav svého portfolia. Kromě toho aplikace poskytuje také informace o aktuálním zůstatku na účtu. Tato dokumentace popisuje jednotlivé funkce aplikace a způsob, jakým jsou implementovány.

Program se skládá ze čtyř tříd. První třída, HelloApplication.java. Jedná se o třídu ve které je metoda main() jež spouští celý program. Druhá třída, Scraper.java. Tato třída se stará o extrakci dat z internetu, konkrétně o ceny kryptoměn. Třetí třída, HelloController.java obsahuje veškerý kód díky kterému aplikace funguje. Poslední třída, hello-view.fxml se stará o grafické uživatelské rozhraní.

2 Program

2.1 Uživatelské grafické rozhraní

Při spuštění programu se uživateli zobrazí tato obrazovka. Nachází se zde několik možností co klient může udělat.

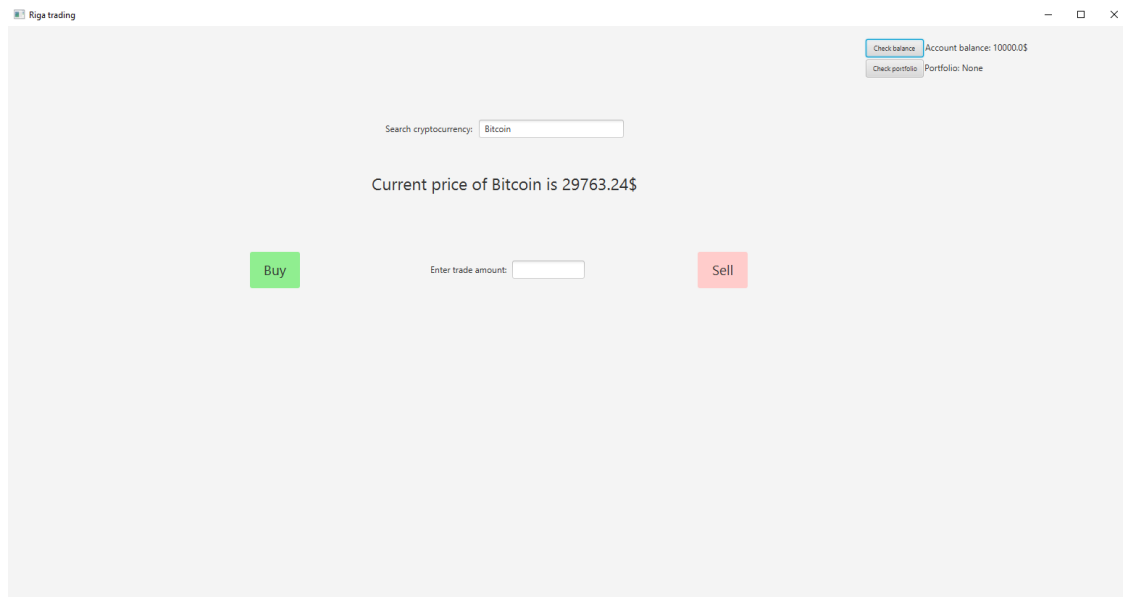


Obrázek 1: Hlavní strana

Jedna z možností je zkontrolovat zůstatek svého účtu. V pravém horním rohu se nachází tlačítko “Check balance” na které když uživatel klikne tak mu odhalí jeho aktuální zůstatek v dolarech. Pod tlačítkem “Check balance” se nachází druhé, které se nazývá “Check portfolio”. Má velmi podobnou funkci jako “Check balance”, ovšem místo zůstatku účtu kontroluje název kryptoměny kterou má uživatel v portfoliu a její počet, který zároveň převede do dolarů. Zhruba uprostřed obrazovky se nachází nápis “Search cryptocurrency” a hned vedle je pole, do něž je možné psát. Zde musí uživatel zadat název kryptoměny, kterou chce koupit nebo prodat.

Pokud již uživatel vlastní nějakou kryptoměnu a zadá jiný název než té co vlastní, bude vybídnut k přepsání vlastněné na novou, tudíž přijde o stávající kryptoměnu a místo ní bude moci koupit novou.

Po zadání kryptoměny se klientovi ukáže nové pole, které slouží k zadání částky na prodej či nákup již dříve zadané kryptoměny. K tomu se



Obrázek 2: Koupení či prodej kryptoměny
zobrazí dvě tlačítka a informace o její ceně. Jedno tlačítko slouží na nákup, “Buy”, a druhé na prodej, “Sell”. Po kliknutí na vybrané tlačítko si uživatel buď prodá svou kryptoměnu, nebo přikoupí více .

2.2 Funkcionalita programu

V této části dokumentace budu rozebírat funkce v programu a některé problémy při programování.

2.2.1 onPortfolioButtonClick()

```
public void onPortfolioButtonClick() throws IOException {
    checkCurrency();
    if (cryptoCurrency == null || cryptoCurrency.equals("")) {
        portfolioLabel.setText("Portfolio: None");
    } else {
        portfolioLabel.setText("Portfolio: " + cryptoCurrency + "-" +
            Math.round(amount * 100.0) / 100.0 + "-" + Math.round(amount *
            scraping.currentPrice(cryptoCurrency) * 100.0) / 100.0 + "$");
    }
}
```

Tato funkce se stará o zjištění stavu portfolia. Při kliknutí na tlačítko s nápisem “Check portfolio” napíše uživateli aktuální stav jeho portfolia který přečte z textového souboru data.txt. V případě prázdného portfolia se objeví nápis “None” jako symbol že uživatel nevlastní žádné kryptoměny. V opačném případě, když portfolio není prázdné, se klientovi zobrazí nápis s vlastněnou kryptoměnou, zaokrouhleným počtem tokenů vlastněné kryptoměny a aktuálním zaokrouhlenou cenou vlastněných tokenů v dolarech.

2.2.2 accountBalance()

```
public double accountBalance() {
    String data = "";

    File balanceFile = new File("Balance.txt");

    //Data reading
    try {
        FileReader reader = new FileReader(balanceFile);
        BufferedReader breader = new BufferedReader(reader);
        StringBuffer sb = new StringBuffer();
        String line;
        while ((line = breader.readLine()) != null) {
            sb.append(line);
        }
        reader.close();
        data = sb.toString();
    } catch (IOException ex) {
        ex.printStackTrace();
    }
}
```

```

        Money = Double.parseDouble(data);

        out.println("Current balance is: " + Money + "$");

        balance.setText("Account balance: " + Money + "$");

        return Money;
    }

```

Funkce `accountBalance()` je velmi podobná předchozí funkci `onPortfolioButtonClick()`, jediný rozdíl je v souboru který čte. Tato funkce používá soubor `Balance.txt` jako zdroj svých dat. Jednoduše přečte částku zapsanou v textovém souboru `Balance.txt` a zobrazí ji v dolarech uživateli.

2.2.3 saveCurrency()

```

public void saveCurrency() {
    Path dataPath = Paths.get("Data.txt");
    while (true) {
        try {
            Files.writeString(dataPath, searchBar.getText() + "-" +
amount, StandardCharsets.UTF_8);
            break;

        } catch (IOException ex) {
            out.println("Error, failed to save data.");
        }
    }
}

```

Tato funkce má velmi důležitou roli v mém projektu. Jedná se o funkci která ukládá změny v portfoliu. Při zakoupení nebo prodeji kryptoměny se vždy zavolá tato funkce, která aktualizuje textový soubor `Data.txt`.

2.2.4 updateBalanceOnBuy() + updateBalanceOnSell()

```

public void updateBalanceOnBuy() {
    Path dataPath = Paths.get("Balance.txt");
    accountBalance();
    Money = Money - Double.parseDouble(amountBar.getText());
    while (true) {
        try {
            Files.writeString(dataPath, String.valueOf(Money),
StandardCharsets.UTF_8);
            break;

        } catch (IOException ex) {
            out.println("Error, failed to save data.");
        }
    }
}

```

```

    }
}

```

Tyto funkce dvě funkce slouží k aktualizaci stavu účtu. Při prodání či koupení kryptoměny je potřeba aktualizovat finanční stav účtu, ke kterému slouží metody `updateBalanceOnBuy()` a `updateBalanceOnSell()`. Při prodání kryptoměny se spustí funkce `updateBalanceOnSell()`, která uživateli přidá na účet peníze a naopak `updateBalanceOnBuy()` je uživateli odečte.

2.2.5 checkCurrency()

```

public void checkCurrency() {
    String data = "";
    File balanceFile = new File("Data.txt");

    //Data reading
    try {
        FileReader reader = new FileReader(balanceFile);
        BufferedReader bfreader = new BufferedReader(reader);
        StringBuffer sb = new StringBuffer();
        String line;
        while ((line = bfreader.readLine()) != null) {
            sb.append(line);
        }
        reader.close();
        data = sb.toString();

    } catch (IOException ex) {
        ex.printStackTrace();
    }
    if (!data.equals("")) {
        String[] dataString = data.split("-");
        cryptoCurrency = dataString[0];
        amount = Double.parseDouble(dataString[1]);
    }
}

```

`CheckCurrency()` funkce umí číst data ze souboru `Data.txt`. V případě, že v nějaké metodě je potřeba pracovat s koupenou kryptoměnou nebo jejím počtem, zavolá se tato funkce, která ze souboru `Data.txt` přečte vlastněnou kryptoměnu a počet tokenů co uživatel vlastní a uloží je do již vytvořených proměnných, aby s nimi bylo možné pracovat.

2.2.6 onBuyButtonClick() + onSellButtonClick()

```

@FXML
public void onBuyButtonClick() throws IOException {
    buyButton.setVisible(false);
}

```

```

sellButton.setVisible(false);
amountBar.setVisible(false);
amountLabel.setVisible(false);
currentPrice = scraping.currentPrice(searchBar.getText());
if (amountBar.getText() == null ||
!amountBar.getText().matches("[0-9]+")) {
    errorMessage.setVisible(true);
    errorMessage.setText("Invalid amount");
    amountBar.setVisible(false);
}
accountBalance();
checkCurrency();
out.println(Money);
if (Double.parseDouble(amountBar.getText()) > Money) {
    errorMessage.setVisible(true);
    amountBar.setVisible(false);
    errorMessage.setText("Insufficient balance.");
} else {
    amount = amount + Double.parseDouble(amountBar.getText()) /
scraping.currentPrice(searchBar.getText());
    errorMessage.setText("Bought " +
Double.parseDouble(amountBar.getText()) / currentPrice + " of " +
searchBar.getText() + " at market price: " + currentPrice + "$");
    portfolioLabel.setText("Portfolio: " + cryptoCurrency + "-" +
Math.round(amount * 100.0) / 100.0 + "-" + Math.round(amount *
currentPrice * 100.0) / 100.0 + "$");
    errorMessage.setVisible(true);
    saveCurrency();
    updateBalanceOnBuy();
    accountBalance();
}
}

```

Funkce `onBuyButtonClick()`, jak již název napovídá, slouží pro koupení kryptoměny. Po stisknutí tlačítka “Buy” se spustí série operací, které má tato metoda na starost. První krok ze série operací je zneviditelnění několika komponentu. Následně má za úkol zkontrolovat, zda-li zadaná částka nepřesahuje aktuální stav účtu a zda-li obsahuje pouze čísla. Pokud je částka menší než stav účtu a obsahuje pouze čísla, transakce je provedena a kryptoměna je přidána do portfolia pomocí již dříve zmíněné funkce `saveCurrency()` a stav účtu je aktualizován přes `updateBalanceOnSell()`. Na konci je zavolána metoda `accountBalance()`, aby se uživateli automaticky aktualizoval nápis se stavem účtu. Funkce `onSellButtonClick()` má velmi podobnou funkcionalitu, jediný rozdíl je, že počet tokenů odečítá a stav účtu přičítá.

2.2.7 onSearchBarAction()

```
@FXML
public void onSearchBarAction() throws IOException {
    currentPrice = scraping.currentPrice(searchBar.getText());

    out.println(currentPrice);

    checkCurrency();

    out.println(cryptoCurrency);

    if (cryptoCurrency == "" || cryptoCurrency == null ||
cryptoCurrency.equals(searchBar.getText())) {
        amountBar.setVisible(true);
        amountLabel.setVisible(true);
        yesButton.setVisible(false);
        noButton.setVisible(false);
        errorMessage.setVisible(false);
        priceLabel.setVisible(true);
        priceLabel.setText("Current price of " + searchBar.getText() + "
is " + currentPrice + "$");
        priceLabel.setAlignment(Pos.CENTER);
        buyButton.setText("Buy");
        sellButton.setText("Sell");
        buyButton.setVisible(true);
        sellButton.setVisible(true);

        amountBar.setVisible(true);
        amountLabel.setVisible(true);
    } else {
        amountBar.setVisible(false);
        amountLabel.setVisible(false);
        buyButton.setVisible(false);
        sellButton.setVisible(false);
        priceLabel.setVisible(false);
        errorMessage.setText("You already own 1 cryptocurrency, do you
want to overwrite it?");
        errorMessage.setVisible(true);
        yesButton.setVisible(true);
        noButton.setVisible(true);
    }
}
```

Nyní se dostáváme k další velmi důležité funkci `onSearchBarAction()`. Tato funkce kontroluje, zda-li uživatel zadal kryptoměnu co již vlastní či nikoli. V případě že má klient prázdné portfolio, může zadat libovolnou kryptoměnu a po zadání mu je vypsána její aktuální cena a je mu nabídnuto ji koupit. V případě že již uživatel má v portfoliu kryptoměnu, ale zadaná měna se shoduje s tou v portfoliu, je opět vybědnut ke koupi či prodeji kryptoměny. Ovšem pokud zadá

kryptoměnu co nevlastní a uživatelské portfolio není prázdné, bude vybídnut k přepsání kryptoměny, což znamená, že o aktuálně vlastněnou kryptoměnu přijde. Poté je mu umožněno koupit opět novou kryptoměnu.

2.2.8 currentPrice()

```
public double currentPrice (String cryptocurrencyName) throws
IOException {
    String CryptoCurrencyNameUrl = cryptocurrencyName+"/";

    Document document =
Jsoup.connect(webURL+CryptoCurrencyNameUrl).get();

    elements = document.getElementsByClass("priceValue");

    if (elements.size() == 0) {
        throw new NoSuchElementException("Could not find price element");
    }

    String priceStr = elements.first().text().replaceAll("[^0-9.]+", "");
    double price = Double.parseDouble(priceStr);
    return price;
}
```

Tato funkce se narozdíl od ostatních zmíněných funkcí nenachází v HelloController.java, nýbrž je ve vlastní třídě Scraper.java. Jedná se o nejdůležitější funkci tohoto celého programu. Funkce currentPrice() zjišťuje ceny kryptoměn pomocí extrakce dat z webové stránky coinmarketcap¹. Při extrakci dat používám java knihovnu jsoup², která mi umožňuje najít hledané ceny. Funkce najde HTML třídu “priceValue” na URL adrese coinmarketcap/hledanáKryptoměna, která obsahuje cenu zadané kryptoměny. Na vytvoření této funkce jsou použity návody na stránce youtube³.

¹n.d. CoinMarketCap: Cryptocurrency Prices, Charts And Market Capitalizations. Accessed April 26, 2023. <https://coinmarketcap.com/>.

²Hedley, Jonathan. n.d. jsoup: Java HTML parser, built for HTML editing, cleaning, scraping, and XSS safety. Accessed April 26, 2023. <https://jsoup.org/>.

³“How to Web Scrape with Java using Jsoup Library | Extract Website Data + Source Code.” 2020. YouTube. <https://www.youtube.com/watch?v=BEvRZUEQ3Dc>.

3 Závěr

Tato dokumentace popisuje aplikaci Riga trading, která umožňuje uživatelům obchodovat s kryptoměnami v reálném čase. S programem jsem poměrně spokojený až na pár věcí, jako je způsob ukládání kryptoměn a stavu účtu. Ukládání dat do textového souboru, který je možný běžně přepsat, není nejlepší způsob. Druhá věc je počet kryptoměn, které uživatel může vlastnit. Mít pouze jednu kryptoměnu v celém portfoliu je poměrně nepraktické. Mimo tyto dva nedostatky jsem s prací spokojen.

Seznam obrázků

[1] Obrázek 1: Hlavní strana

[2] Obrázek 2: Koupení či prodej kryptoměny

Reference

[1] n.d. CoinMarketCap: Cryptocurrency Prices, Charts And Market

Capitalizations. Accessed April 26, 2023. <https://coinmarketcap.com/>.

[2] Hedley, Jonathan. n.d. jsoup: Java HTML parser, built for HTML editing, cleaning, scraping, and XSS safety. Accessed April 26, 2023. <https://jsoup.org/>.

[3] "How to Web Scrape with Java using Jsoup Library | Extract Website Data + Source Code." 2020. YouTube.

<https://www.youtube.com/watch?v=BEvRZUEQ3Dc>.