

PROJEKT AKTIENHANDEL UND PORTFOLIOMANAGEMENT

UNTERRICHTSEINHEIT: 3 Tägiges Projekt in SWE und C#

Dauer: 3 Tage

Anteilig für Endnote : Projektnote

Arbeitsweise: Gruppenarbeit 1-2 Personen

Fragen an Frau Schicha oder Hr Röder per Teams oder im Unterricht.

ABGABE

Abgabe als **ZIP Datei**

- mit Diagrammen als JPEG,
- Code als VisualStudio-Projekt
- Word- oder Openoffice -Dokument für schriftliche Antworten
- Worddokument für die Queries und Prompts,
- Am Ende gibt es einen Minitest in Quizzform, an dem sie zeigen können, ob sie das Prinzip der Aufgabe verstanden haben.

Ziel: Mit dieser Aufgabe sollen die Themen Klassendiagramm und Aktivitätendiagramm vertieft werden und die objektorientierte Programmierung mit Vererbung und Listen umgesetzt werden.

Es gibt drei Teile

Abgabe bis Montag 13:00 Klassendiagramme und Aktionsdiagramme

Abgabe bis Mittwoch um 15:15 gezippte Datei mit Code und Dokumentation

Quizz dazu am Mittwoch 12:30 Uhr.

Aufgabe A) Mitteilung des Gruppennamens und der Teilnehmer der Gruppe an den Dozenten (Teams)

Gruppengröße 2-3 Personen

HINTERGRUND

Sie haben einen guten Job in einer Bank bekommen. Die Bank möchte für die Simulation von Wertpapieren und Portfolios einen Prototyp ohne grafische Oberfläche haben. Da Sie vielleicht noch keine Ahnung davon haben, was das alles bedeutet: Hier eine kleine Zusammenfassung:

- Ein **Portfolio** ist eine **Sammlung** von Wertpapieren für **einen Kunden**. Funktioniert ein wenig so wie ein Bücherschrank: Man kann Wertpapiere kaufen und diese werden im Portfolio hinterlegt. Oder man kann welche verkaufen und damit wieder herausnehmen. In der Regel hat man von einem Wertpapier immer mehrere.
- Liste handelbare Wertpapiere ist die Menge an Wertpapieren, die von der Bank aus zum Handel (Kauf und Verkauf in ein Portfolio) für den Kunden zur Verfügung stehen.

- **Wertpapiere** ist ein abstrakter Überbegriff für eine Menge an verschiedenen Dingen, mit denen eine Bank oder ein Investor am Wertpapiermarkt handelt. Es gibt viele verschiedene Arten von Wertpapieren, wir nutzen hier aber nur Aktien, ETFs, Anleihen, Optionen. Allen gemeinsam ist es, dass sie am Wertpapierhandel einen **Namen** und **ISIN Nummer** haben. Aber wie schon gesagt: Wertpapier können sie nicht handeln, nur die verschiedenen Formen davon.
- Dazu gehören **Aktien**. Diese sind so etwas wie Anteile an Unternehmen. Wenn es dem Unternehmen besser geht, dann steigt auch der Wert der Aktien und umgekehrt.
Eine Aktie soll die Wertpapierinformationen Namen und ISIN auch ein **Kürzel** (genannt ETR) und **Unternehmen** haben, zu dem es gehört. Und manche Aktien haben auch eine **Dividende**. Das ist eine jährliche Ausschüttung an Gewinn an den Aktienbesitzer, die als Prozentzahl festgelegt wird.
Beispiel Aktie Apple Inc. | AAPL | US0378331005. **Dividende** 0.1%
- Dann gibt es **Anleihen**, die haben auch alle Eigenschaften eines Wertpapiers, aber zudem noch eine **Laufzeit** bis zu einem bestimmten Datum Laufzeitende, indem die Anleihe fällig wird. Dann haben die **Anleihen** noch einen **Kuponwert**, dass ist der Wert, der die Höhe der Verzinsung pro Jahr in Prozent angibt
 - PCC Anleihe
Kupon:5,00%
ISIN DE000A351K90
 - Fälligkeit 1. Juli 2028
- Dann gibt es noch einen **ETF**. Dieser ist eine handelbare Einheit einer Gruppe von Wertpapieren (genannt Index). Man kann zum Beispiel einen DAX ETF kaufen
Alle Wertpapier Eigenschaften und eine „**Basis**“ .z.B. des deutschen Leitindex DAX aufbaut.
Z.B ISIN: DAX100ETF – Basisindex DAX
- **Optionsscheine**: Sind riskante Wertpapiere, bei dem man leicht mal den kompletten Wert verlieren kann. Sie haben eine Laufzeit, und am Ende der Laufzeit bieten diese die Option eine bestimmte Aktie (**Basis**) zu einem bestimmten Preis zu kaufen. Ein Optionsschein hat folgende Eigenschaften : Alle Wertpapier Eigenschaften + **Laufzeitende** + die Bezeichner entweder CALL (Aktienwert steigt) oder PUT (Aktienwert fällt) . Dies bezeichnet man als **Optionstyp**
z.B ISIN US0378331004
Basis: AAPL
Laufzeitende: 15.10.2023
Typ PUT

Typisch für Programmierung heute im Job ist es, dass man etwas programmieren muss, das man nicht unbedingt sofort versteht. Daher habe ich ihnen die oben genannte Aufteilung geschrieben. Nutzen Sie die fett markierten Eigenschaften für ihre Klassen und Attribute.

SCHRITTWEISER AUFBAU

Gehen Sie in folgender Weise vor:

1. Klassendiagramme Abgabe bis Montag 13:00 Uhr
2. Aktionsdiagramm Kauf / Verkauf Wertpapier in Portfolio (Benutzerführung) (Mo 13:00 Uhr)
3. Programmierung des Modells incl.
 - Portfolios Kauf / Verkauf für einen Kunden
 - Kursverlauf für Wertpapiere
4. Programmierung des Menüs
5. Einlesen des Wertpapierstamms aus einer Datei
6. Automatische Zufallserzeugung der Kurverläufe
7. Beantworten der Textaufgaben
8. Optional Zusatzaufgabe: Programmierung für mehrere Kunden

KLASSENDIAGRAMME

Zeichnen Sie alle Klassendiagramme mit den richtigen Verbindungen. Achten Sie darauf die richtigen Assoziationen darzustellen mit den richtigen Kardinalitäten und auch Komposition und Aggregation richtig zu zeichnen. Die Startklasse mit der Mainmethode wird für das Diagramm nicht benötigt. Gehen Sie zuerst den Text durch und schreiben Sie auf welche Klassen Sie benötigen werden. Überlegen Sie welche in horizontaler Ebene nebeneinander stehen und welche besser zur Veranschaulichung der Vererbung untereinander stehen.

AKTIONSDIAGRAMM

Für den Ablauf des Kaufens und Verkaufens in ein Portfolio soll ein Aktivitätendiagramm gezeichnet werden. Der Ablauf ist folgendermaßen:

... Es soll einen Wertpapierstamm von 10 Wertpapieren geben aus dem der Kunde auswählen kann (was er kaufen oder verkaufen möchte)

... Es sollen Wertpapiere gekauft werden und verkauft werden können. Diese landen im Kundenportfolio wie oben

Beispiel:

Console: Möchten Sie kaufen (k) oder verkaufen (v) oder zum Hauptmenü zurückkehren (x)

Ihre Eingabe: k

Wir haben folgende Wertpapiere zum Kauf bereit:

1 Aktie AAPL	aktueller Kaufkurs : 300\$	Dividende 0,1%
2 Anleihe PCC	aktueller Kaufkurs : 100\$	Laufzeit 20.01.2028

...

Welches Wertpapier möchten Sie kaufen?

2

Wieviele Wertpapiere möchten Sie kaufen?

10

OK Sie haben 10 Anleihen PCC zum Kurs von 100\$ gekauft
Hier sehen Sie ihr Portfolio:

[Wie oben Anzeige des Portfolios]

Ähnlich läuft es mit dem Punkt Verkaufen

Und dann gibt es noch den Menüpunkt (x)-> Zurück zum Hauptmenü

Zeichnen sie für diesen Ablauf ein Aktivitätendiagramm mit

Start/Ende/Aktionen/Verzweigungen/Verbindungsknoten

Abgabe spätestens Projekt-Montag um 13 Uhr. Machen Sie soweit sie kommen und geben Sie die Ergebnisse, die sie haben ab. Es geht nicht um Quantität, sondern um die Qualität der gezeichneten Diagramme. Also lieber weniger, dafür syntaktisch richtig.

DOKUMENTATION IN DER PROGRAMMIERUNG

Jede Klasse sollte mit Dokumentation erstellt werden. Also zumindest eine Dokumentation für die Klasse und jeweils einer von Ihnen gewählten Methode der Klasse. Positiv fällt es auch auf, wenn Sie kritischen Code mit Kommentaren ergänzen. So wird das in der Regel später im Job verlangt.

-> Jede Klasse `///<summary> ...`

-> Eine Methode pro Klasse `///<summary> ...`

-> komplexer Code (z.B. Menüführungsstruktur) `// Kommentar`

Aber auch nicht mehr. Konzentrieren Sie sich auf die Programmierung.

PROGRAMMIERUNG DES MODELLS

In den Klassen wird die Datenkapslung mit Hilfe von Properties umgesetzt. Es dürfen nur korrekte Objekte erstellt werden. Benutzen Sie dafür das Exception-Konzept. Verwenden Sie die DateTime-Klasse um ein Datum abzuspeichern. Benutzen Sie zur Speicherung von Daten generische Listen.

Erzeugen Sie sich eine Datei mit allen Informationen der zur Verfügung stehenden Wertpapiere und laden Sie die Daten beim Start der Anwendung in das Programm. Sorgen Sie dafür, dass die Datei beim Beenden des Programms aktualisiert wird. Dasselbe gilt für die gekauften Wertpapiere des Kunden/ der Kunden.

PROGRAMMIERUNG DES MENÜS

Das **Hauptmenü** soll zu Beginn an sichtbar sein und immer nach Abschluss der einzelnen Aktionen wieder dargestellt werden.

A - Anzeige des eigenen Portfolios

H - Handeln Kaufen und Verkaufen

K - Kursverlauf Anzeigen des Wertpapierkursverlaufs

X - Beenden

Beachten Sie, dass es hilfreich wäre eine Darstellungsklasse Menü zu erstellen.

PROGRAMMIERUNG KAUF UND VERKAUF

Ihr Programm soll für den Handel folgende Funktionalitäten zur Verfügung stellen:

In einem **Portfoliomanager** (Hauptklasse) sollen verschiedene Typen von Wertpapieren für ein **Portfolio** eines **Kunden** gekauft und verkauft und angezeigt werden können.

Beispiel einer Ausgabe eines Portfolios:

Portfolio Müller

Anzahl	Typ	Name	ETR	Kaufkurs	GesamtK	Kurs	Akt	Gesamt
3	Aktien	APPLE	AAPL	300	900	330		990
4	Anleihe	AnleihePCE	PCC	50\$	200\$...			
...								
Gesamt					1100			1240

Der Wertpapierkurs:

Jedes Wertpapier (egal ob Aktie Anleihe ETF oder Optionsschein) hat einen **Kurs** und einen **Kursverlauf**. Jeder Kurs hat ein **Datum** und einen **Wert**. Der aktuelle Kurs ist der letzte Kurs im Kursverlauf. Der Kaufpreis soll immer der erste Kurs im Kursverlauf sein. Daraus sieht man, ob der Kunde Gewinn oder Verlust mit der Aktie gemacht hat. Am Ende steht eine Summierung des GesamtKaufkurses und des GesamtAktKurses

PROGRAMMIERUNG KURSVERLAUF

Im Hauptmenü soll der Kunde die Möglichkeit haben den Kursverlauf zu sehen. Dazu wählt er ein Wertpapier aus der Liste der bei der Bank verfügbaren Wertpapiere. Die Wertpapier werden als Liste angezeigt:

Wir haben folgende Wertpapiere vorliegen

- 1 Aktie AAPL aktueller Kaufkurs : 300\$ Dividende 0,1%
- 2 Anleihe PCC aktueller Kaufkurs : 100\$ Laufzeit 20.01.2028

...

Für welches Wertpapier möchten Sie den Kurs sehen?

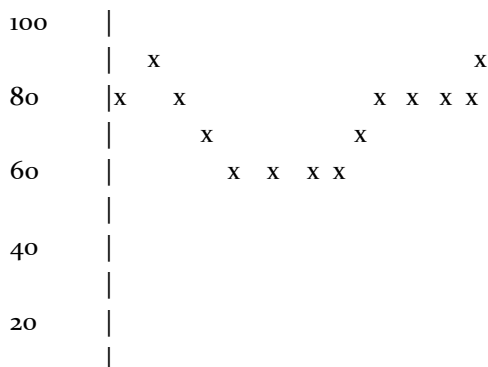
Der zufällige Kursverlauf soll bereits beim Anlegen der Stamm-Wertpapiere generiert werden. Zur Vereinfachung starten Sie mit einem zufälligen Wert zwischen 50 und 80 und generieren den nächsten Punkt daraus, indem Sie einen zufälligen Wert im Rahmen [-5 bis + 5] dazu addieren.

Machen Sie das für 14 Tage (Das Datum ist in der vereinfachten Form eine Zahl von 1-14) Für die Profis darf es als Datumsformat angelegt werden.

Darstellung eines zufälligen Kursverlaufs über 14 Tage, der folgendermaßen dargestellt werden soll:

Beispiel

Kurs Aktie AAPL



Die Bank verlangt von Ihnen nur eine Simulation der Börsenkurse nach folgendem Prinzip:
Der Kursverlauf darf nicht unter 0 fallen. Der Kursverlauf ist stetig, also ändert sich von einem Tag auf den nächsten nur maximal 10 %.

Alternativ für Profis - können Sie auch eine grafische Ausgabe mit einem Canvas im WPF mit C# erzeugen.

PROGRAMMIERUNG EINLESEN DER BANKDATEN

Für diesen Part sollen die Daten des Wertpapierstamms nicht aus Code erschaffen werden sondern mittels einer Methode `ausDateiLesen()` in einer Klasse `Wertpapierstamm` gelesen werden.
Der Kursverlauf soll NICHT eingelesen werden.

ZUSATZAUFGABE – FÜR MEHRERE KUNDEN

Dieser Aufgabenteil ist optional und bringt nochmals 10% mehr Punkte.
Programmieren Sie noch eine Klasse `Kundenliste` und `Kunde`. Für jeden Kunden soll ein Portfolio zur Verfügung stehen, mit dem alle Funktionen wie oben beschrieben durchgeführt werden können. Nue hinzu kommt:

- 1.) Erzeugen Sie zum Programmstart **3 Kunden**
- 2.) Erweitern Sie das Hauptmenü um den **Punkt Kunde wechseln**
- 3.) Ab dann sollen nur die Daten dieses Kunden dargestellt werden.

Alleine die Erweiterung des Klassendiagramms ohne Programmierung gibt Sonderpunkte.
Mit Programmierung wäre natürlich bestens!

ZUSATZAUFGABE FÜR SUPERPROFIS – REALE KURSE EINLADEN

Laden Sie Kursdaten für eine Aktie aus einer Onlinequelle für Aktiendaten. Eine kostenfreie Quelle gibt es bei :
<https://www.alphavantage.co/documentation/>

Der Aufruf der Daten (vorher kostenfreien Key beantragen und demo ersetzen)
https://www.alphavantage.co/query?function=TIME_SERIES_DAILY&symbol=IBM&apikey=demo

TEXTFRAGEN ZUM SYSTEM

1. Ausnahmen: Beachten Sie die möglichen Fehler, die während des Prozesses stattfinden können. An welchen Stellen sind Exceptions hilfreich?
2. Auch wenn sie es so nicht programmiert haben: Welche Klassen (nur Klassennamen) müssten vorhanden sein, damit man Darstellung und Datenklassen trennen kann. Wie wäre die Verbindung einer Datenklasse mit einer Darstellungsklasse im UML Diagramm gekennzeichnet. (Nur Worte als Antwort - kein Diagramm)
Gibt es auch Darstellungsklassen ohne Datenklassen?

3. Wie würden Sie den Unterschied zwischen Datenklassen und Funktionalitäts- und Hilfsklassen am Beispiel dieses Projekts erklären? Was sind die eigentlichen Datenklassen und was sind zusätzliche Klassen, die für den Ablauf und die Funktionalität hilfreich sind?

TIPPS

- Die Zentrale Klasse wird Wertpapier sein. Diese soll abstract (alternativ virtual) definiert werden.
- Klären Sie zuerst, was Datenklassen, Listenklassen und was Darstellungsklassen (zb Menüs) sind.
- Wenn Sie die Daten für den Wertpapierstamm später einlesen, sollten Sie in einer Hauptsteuerklasse einfach nur eine Methode ändern müssen (zb von generiereWertpapierStamm() nach ausDateiLesen()).
- Die main-procedure ist nicht die Hauptsteuerklasse. Bitte erschaffen Sie dafür eine eigene Hauptklassen, die aber von main instantiiert wird.

BEWERTUNG DES PROJEKTS.

10%	Aktivitätsdiagramm	
15%	Klassendiagramm	
10%	Schriftliche Aufgaben zum Verständnis	
15%	Wohlgeformte Programmierung der Klassen Richtige Darstellung, Dokumentation, Cleancode	
30%	Programmierung der Darstellung von Portfolio und Kursverlauf	
10%	Quizzergebniss	
10%	Dokumentation der Queries an Google oder der Prompts in ChatGPT (o.ä.)	
Einzelne Präsentationen am Ende		
Wir werden im Bedarfsfall einzelne Präsentation am Ende erbitten. Dadurch kann die Endnote sich um eine Note verbessern, gleichbleiben oder sich verschlechtern.		

Arbeitsstil: Selbst erarbeiten mit Google, Webseiten und ChatGPT. Aber auch durch Austausch untereinander, Fragen an den Mentor (also ich Euer SWE Dozent) oder Tutorials im Internet oder auch bei LinkedIn, bei der es eine große für Euch kostenfreie Bibliothek an guten Tutorials gibt.

Wenn Sie ChatGPT nutzen müssen alle Prompts protokolliert werden!!!

Sie müssen auch verstehen, was ChatGPT zurückliefert. Bei der Abgabe dieses Teils werden Fragen dazu gestellt.

WICHTIG: Es geht darum verschiedene Level zu erreichen: Das bedeutet, Sie dürfen erst den nächsten Level beginnen, wenn Sie mit dem Dozenten den aktuellen Level besprochen haben und abgezeichnet bekommen haben. Übergehen des Dozenten ohne Absprache ergibt Notenabzug.

GRUPPE: _____
Teilnehmer: _____



Level 1: Klassendiagramm



Level 2: Aktivitätendiagramm



Level 3: Grund-Klassen
Namen und Typ der Attribute
setter und getter Properties
Namen und Rückgabewerte der Methoden



Level 4: Funktionierendes Menü



Level 5: Funktionierendes Kaufen und Verkaufen



Level 6: Grafischer Kursverlauf
(Konsole oder Canvas)



Level 7: Grafischer Kursverlauf aus Onlinedaten
(Konsole oder Canvas)

Plan liegt dem Dozenten vor.