תקציר

כי אם לא תקנה – איך תמכור ???

מס' גרסה: **2**

שפת ג'אווה ומבוא לפיתוח ווב

Rizpa Stock Exchange (RSE) אביב 2020 -

מרצה: **אביעד כהן** [aviadco@mta.ac.il](mailto:aviadco@mta.ac.il)

בודק: **איתי כהן**  itaych@mta.ac.il

התרגיל מנוסח בלשון זכר, אך מכוון לכלל המגדרים האפשריים בצורה שווה

תוכן העניינים

[דרישות הקורס 4](#_Toc61019629)

[כללי 4](#_Toc61019630)

[איך להגיש תרגילים באיחור, ולהישאר בחיים 5](#_Toc61019631)

[הנחיות כלליות לכתיבת התרגיל 6](#_Toc61019632)

[תרגיל reflection – תרגיל עצמאי ביחידים (5%) - הגשה: 16.3.2021 8](#_Toc61019633)

[כללי 8](#_Toc61019634)

[מבנה התרגיל 8](#_Toc61019635)

[ניקוד 8](#_Toc61019636)

[איך בודקים ? 9](#_Toc61019637)

[מה מגישים ? 9](#_Toc61019638)

[מקרה בדיקה לדוגמא 9](#_Toc61019639)

[RIzpa Stock Exchange – מע' למסחר במניות 11](#_Toc61019640)

[מטרת התרגיל(ים) בקורס 11](#_Toc61019641)

[כללי 11](#_Toc61019642)

[מניות 11](#_Toc61019643)

[סוגי פקודות 12](#_Toc61019644)

[מנוע המסחר 12](#_Toc61019645)

[תהליך המסחר ועדיפויות הפקודות 12](#_Toc61019646)

[משתמשים 13](#_Toc61019647)

[הנחיות ספציפיות למימוש מערכת RIzpa Stock Exchange 13](#_Toc61019648)

[תרגיל 1 – מימוש **Rizpa Stock Exchange** כאפליקציית Console (25%) - הגשה: 12.4.2021 14](#_Toc61019649)

[פרטים יבשים 14](#_Toc61019650)

[דרישות 14](#_Toc61019651)

[חלוקה למודולים 18](#_Toc61019652)

[איך מתחילים ? (המלצה...) 18](#_Toc61019653)

[בונוסים 18](#_Toc61019654)

[סבבה, סיימתי. מה ואיך להגיש ? 19](#_Toc61019655)

[שאלות ותשובות 19](#_Toc61019656)

[תרגיל 2 – מימוש**Rizpa Stock Exchange** כאפליקציית JavaFX (30%) – הגשה: 24.5.2021 20](#_Toc61019657)

[פרטים יבשים 20](#_Toc61019658)

[דרישות 20](#_Toc61019659)

[חלוקה למודולים 22](#_Toc61019660)

[איך מתחילים ? (המלצה...) 22](#_Toc61019661)

[בונוסים 22](#_Toc61019662)

[סבבה, סיימתי. מה ואיך להגיש ? 22](#_Toc61019663)

[שאלות ותשובות 23](#_Toc61019664)

[תרגיל 3 – מימוש **Rizpa Stock Exchange** כאפליקציית WEB (40%) – הגשה: 17.7.2021 24](#_Toc61019665)

[פרטים יבשים 24](#_Toc61019666)

[דרישות 24](#_Toc61019667)

[חלוקה למודולים 28](#_Toc61019668)

[איך מתחילים ? 28](#_Toc61019669)

[בונוסים 28](#_Toc61019670)

[סבבה, סיימתי. מה ואיך להגיש ? 29](#_Toc61019671)

[שאלות ותשובות 29](#_Toc61019672)

[נספח א' – תיאור תרחיש קנייה ומכירה לפקודת LMT 30](#_Toc61019673)

[נספח ב' – תיאור תרחיש קנייה ומכירה לפקודת MKT 36](#_Toc61019674)

[נספח ג' – תיאור תרחיש קנייה ומכירה לפקודת FOK 38](#_Toc61019675)

[נספח ד' – תיאור תרחיש קנייה ומכירה לפקודת IOC 40](#_Toc61019676)

[נספח ה' – תיאור מבנה המערכת באמצעות קובץ XML 42](#_Toc61019677)

[סכמת קובץ XML – גרסה I 42](#_Toc61019678)

[סכמת קובץ XML – גרסה II 43](#_Toc61019679)

[סכמת קובץ XML – גרסה III 44](#_Toc61019680)

[נספח ו' – קבצי סכמה לוידוא קובץ ה XML 45](#_Toc61019681)

דרישות הקורס

## כללי

1. בקורס אין בחינה אך חובה להגיש תרגילים (סה"כ 4).
2. את מרבית התרגילים (3) ניתן להגיש בזוגות, אך לא בשלישיות / רביעיות / חמישיות או יותר

(כן, גם אם מדובר בשלישיה / רביעיה / חמישיה הצועדת יחדיו לאורך שנים מאז גיל הגן והגישה עד עכשיו את כל הפרוייקטים ביחד).

את תרגיל ה - [reflection](#_כללי) חובה להגיש ביחידים.

1. בעבודה משותפת על תרגיל יש להקפיד על מעורבות אקטיבית של כלל המגישים בכל חלקי התרגיל.
2. במידה והוגדר בונוס לתרגיל מסוים, ציון הבונוס יתווסף לציון התרגיל בלבד (ולא לציון הסופי של הקורס כולו).
3. התרגילים יוגשו דרך מערכת Mama. מוגדר רכיב 'מטלה' נפרד לכל תרגיל.
4. לפני שליחת התרגיל יש לבדוק שהוא עובד ומכיל את הקבצים המעודכנים ביותר, על מערכת "נקייה".

בצעו את סט הפעולות שאתם מצפים מן הבודק לבצע וודאו כי הכל מתנהל כראוי וכסדרו.

1. ניתן להחליף את השותפ/ה בכל תרגיל, ללא צורך באישור או הודעה למרצה.

הניקוד על כל תרגיל נזקף לזכות הסטודנטים שבצעו אותו בלבד.

1. טרם הגשת התרגיל המתגלגל הראשון, תידרשו לשלוח מייל שבו מידע על צוות מגיש/י התרגיל.   
   במידה ויחול שיוי בציוות בתרגילים הבאים – יש לשלוח עדכון מתאים במייל לקראת ההגשה של התרגיל הבא.
2. יש להעלות את התרגיל רק עבור אחד מבני הזוג ולהוסיף את שם בת/בן הזוג ומספר תעודת הזהות שלה/ו גם באתר וגם בקובץ ה readme המצורף (פרטים בהמשך).

## איך להגיש תרגילים באיחור, ולהישאר בחיים

1. ניתן להגיש תרגיל עד שבוע איחור, כאשר עבור כל 24 שעות איחור – תורד נקודה אחת מציון התרגיל ; תרגיל שיוגש באיחור של יותר משבוע (ללא סיבה מוצדקת) – **פשוט לא יבדק**.
2. במידה והגשתם תרגיל אולם הבודק נתקל בבדיקתו במצב שפשוט לא מאפשר את המשך הבדיקה (למשל כישלון בטעינת קובץ הבדיקה) – הרי שאתם מוגדרים כתקלת level 0. במקרה של תקלה שכזו איידע אתכם ואאפשר לכם לבדוק, לתקן ולהגיש מחדש את התרגיל כדי שאפשר יהיה לבודקו אחרי הכל.   
   שימו לב **כי בכל במקרה של הגשה חוזרת** בגלל level 0 – הציון לתרגיל יתחיל מ 90, ללא שום קשר לאופי הבעיה ו/או התיקון (גם אם התיקון היה "קטן". גם אם התיקון היה בגלל בלבול בהגשה של גרסה קדומה יותר של הקבצים. גם אם הכלב אכל לכם את שיעורי הבית)
3. בתרגילים השונים ניתן לממש בונוסים (פרטים בהמשך).

המטרה של הבונוס היא לעזור לכם להעלות את הציון ולא להורידו !

רוצה לאמר: אל תגישו באיחור רק בשביל להספיק לפתח בונוס.

בונוס מפתחים **אם ורק אם** סיימתם את כל דרישות הבסיס להגשה, יש לכם עוד מספר ימים, וברצונכם לנסות ולהגדיל את הציון ע"י בונוס.

היות וכך, ולמען הסר כל ספק: **לא ייבדקו** הבונוסים עבור תרגילים שהוגשו באיחור (שאינו מוצדק).

בהתאם לכך, אני שומר לעצמי חירות רבה יותר בשינוי כזה או אחר של מי מסעיפי הבונוס, גם במהלך התרגיל עצמו.

1. עומס בלימודים, בעבודה, בחיים, בגלל הילדים או ההורים, שכנים וחברים (או בכל תחום אחר) אינו נחשב כסיבה לגיטימית להארכה.
2. במקרה של מחלה או מילואים יש להוסיף סריקה של אישור המחלה או המילואים להגשת התרגיל; בהגשת תרגיל באיחור בשל שירות מילואים יש לצרף להגשת התרגיל קובץ סרוק של טופס השחרור מהמילואים (ולא את צו הקריאה).
3. במקרה של בקשה להארכה (מכל סיבה שהיא, לרבות מילואים ומחלה) יש לפנות למרצה מראש על מנת לקבל אישור (במייל או פרונטלית בכיתה)
4. לאנשי הקבע – הישארות של שבת בבסיס אינה נחשבת כסיבה לגיטימית להארכה (מכיוון שזה חלק מהסדר העבודה בצה"ל); יציאה לאבט"ש כן נחשבת כמילואים ויש להגיש אישור ממפקד הבסיס.
5. סטודנט המגיש תרגיל באיחור של עד שבוע עם סיבה מוצדקת ישלח את הנימוקים לאיחור ביחד עם התרגיל.
6. ניתן לערער על ציון של תרגיל לכל היותר שבוע מיום פרסום המשוב והציון במע' המאמא.

כדי לערער יש לשלוח מייל למרצה בצירוף כל הסיבות והטענות שלכם.

## הנחיות כלליות לכתיבת התרגיל

* במהלך הקורס יוצגו דוגמאות והסברים מבוססים על כתיבה בסביבת הפיתוח (IDE) – Intelij IDEA.

אתם מוזמנים (ומעודדים בזאת) לפתח גם כן את התרגיל בסביבת העבודה intelij.

ניתן לקבל רישיון חינם לשימוש בגרסת ultimate, רק בשל היותכם סטודנטים במכללה (כבר שווה !)

יחד עם זאת, כל אחד רשאי לבחור לעבוד בסביבת העבודה הנוחה והמוכרת לו. כך או אחרת הגשת התרגיל אינה כוללת את סביבת הפיתוח אלא אך ורק הרצה ידנית מ cmd (כמו פעם...).

שימו לב: מבחינתכם, לבודק פשוט אין intelij (או כל ide אחר) ולכן זו אפילו לא אופציה.

חיסכו ממני (ומכם) את כתיבת המייל המבקש זאת.

* יש להגיש את התרגילים בתור קובץ zip/rar (לא 7Z !)

הקובץ יכיל:

1. כל הקבצים הרלבנטים להפעלת התרגיל (jar/war – פרטים בגוף התרגיל).
2. קובץ אצווה ( == batch) שיכיל את הפקודה שמריצה את התרגיל.
3. קובץ readme שיכיל את פרטי המגיש/ים, כמו גם הנחיות כלליות להרצה התרגיל וכל הנחות שלקחתם במהלך התרגיל ואתם סבורים שחשוב כי הבודק יכיר. דמיינו כי בכל שאלה/תקלה שיתקל בהן הבודק, יעמוד לרשותו רק קובץ ה readme שלכם. דאגו להבהיר ולהסביר את כל הדברים שיכולים להשתבש ו/או שבעטיים ייתכנו בעיות/שאלות/תהיות וכיוצב'.

כמו כן, כל הנחה שאתם מניחים בעצמכם לגבי אופן מימוש התרגיל (בין אם בלוגיקת התרגיל ובין אם בהנחה טכנולוגית) צריכה להיות רשומה בקובץ.

על קובץ הreadme להיות בפורמט word או pdf (**לא notepad !**). חי נפשי – אם מישהו מגיש readme כקובץ טקסט פשוט תרד לו נקודה...

* דווקא בגלל שאין זהו קורס שבו יכנסו לנבכי הקוד ויבדקו כל שורה ושורה, יש להקפיד ביתר שאת על קוד נקי, מסודר, קריא ויעיל. בפרט:
* הימנעו משכפול קוד
* פונקציות ארוכות מדי (יותר מגודל עמוד)
* בחירת שמות גרועים למחלקות, לפונקציות ולמשתנים
* הזחה (אינדנטציה) נכונה
* imports מיותרים
* יש להקפיד להשתמש ב-modifiers בצורה נבונה:
* מחלקה שלא אמורים לבנות אובייקטים שלה אמורה להיות מוגדרת כ-abstract
* קבועים יש לסמן כ- final
* משתנים של המחלקה רצוי להגדיר כ-private אלא אם יש סיבה לגיטימית לבחירה אחרת.
* יש להקפיד על מוסכמות בסגנון הכתיבה – שמות מחלקות יתחילו באות גדולה, שמות חבילות, משתנים ופונקציות באות קטנה, שמות קבועים יהיו מורכבים רק מאותיות גדולות וכו'. ראו מסמך java coding conventions שהועלה למאמא.
* התמודדות עם קלט שאינו תקין (במקומות הרלבנטים) היא חלק בלתי נפרד מחווית המפתח (לטוב ולרע...).

יש לוודא קלט תקין מהמשתמש בכל שלב ולהחזיר הודאות שגיאה קריאות, אינפורמטיביות במידה והקלט אינו תקין. (למשל: לא להגיד שהקובץ לא תקין – אלא מה לא תקין בקובץ בצורה מפורטת...)

* כל הקלט והפלט בתרגילים השונים יהיה באנגלית בלבד.

אין להציג או לתמוך בקבלת קלט ו/או הצגת פלט בעברית או בכל שפה אחרת.

כל הקלטים באנגלית יהיו case insensitive, כלומר אין חשיבות ל capital case. דוגמא: MoMo=mOmO

* יש לדאוג שבאף מקרה לא ייזרק Exception שלא טופל ושיגרום לתעופה של התוכנית; יש לטפל ב-Exceptions בנקודה שבה ניתן לעשות כן.
* הוראות שגויות שייגרמו לאפליקציה שלא לרוץ יורידו נקודות, ולכן רצוי מאוד שתנסו להתקין את האפליקציה בעצמכם לפי ההוראות שתכתבו.
* **זהו תרגיל מתגלגל. המטרה היא לבנות בסיס ראשוני בתרגיל הראשון, ולהמשיך ולהשתמש בו, ככל האפשר (ואפשר !) במהלך התרגילים הבאים. השקיעו חשיבה ותכנון בעיצוב הפתרון תוך מחשבה על איך מה שתעשו היום ישרת אתכם מחר. (זה כלל נכון לחיים, לא רק לתרגיל זה).**
* חלק מהעבודה בתרגילים היא קבלת החלטות בנושאים שאינם מפורטים במדויק. המטרה היא לתרגל את הנושאים המרכזיים הנלמדים בקורס, ולא לתפור מוצר לפי דרישות של לקוח. על כן, בכל מקום שלא מופיעה דרישה מדויקת – מוטל עליכם לבחור בדרך ההגיונית ביותר שנראית לכם ולציין את בחירתכם בקובץ ה Readme אשר מוגש עם התרגיל. אם יש ספק לגבי אופן פעולתכם אתם מעודדים לשאול האם הפתרון שאתם חושבים לתת לסוגיה מסוימת הוא קביל ולגיטימי (שאלות בפורום, מייל למרצה וכו)
* **ווידוא הגשת התרגיל טרם הגשתו:**
* **יש לוודא כי ההגשה שלכם רצה היטיב על מע' נקייה, באופן שבו גם הבודק יריץ אותה,** על מערכת נקייה וללא תוצרי לוואי אחרים של הפעלות קודמות שלכם.
  + **הבודק יבצע את הבדיקה על מע' windows 10. כל מי שמפתח על גבי mac/linux – זכותכם – אבל גם חובתכם לוודא כי אתם רצים היטיב על windows 10. למען הסר ספק, לא תתבצע בדיקה על מע' הפעלה אחרת. כמו כן לא תהיה התחשבות בתקלות שמקורם רק בשל עבודה על מע' הפעלה שונות (ולא שאמורות להיות תקלות כאלה..)**
  + **יש לוודא כי כל קבצי הבדיקה השונים שהועלו ל mama נטענים בהצלחה ע"י המע' שלכם טרם ההגשה. בדיקת הבודק תתחיל מבדיקה בסיסית המבוססת בצורה גסה על קבצים אלה. חבל ליפול Level 0 על שטות שיכולתם לעלות עליה בשנייה עוד בשלב הפיתוח.**
  + **כאמור, הגשה חוזרת בשל תקלות level 0 תתחיל מראש מציון של 90. בלי שום יוצא מן הכלל. הקדימו תרופה למכה.**
* **בחלק מהתרגילים ניתנת אפשרות למימוש דרישות בונוס.**
  + **ישנם 2 סוגי בונוסים:**
    1. **בונוס בתוך טווח התרגיל - יכול להביא אתכם לכל היותר לציון 100, ולחפות במקרה והורדו לכם נקודות בשל תקלות.**
    2. **בונוס מחוץ לטווח התרגיל – יכול להעלות את ציונכם אף מעבר ל 100 (וכן, יש כפל מבצעים לטובת הלקוח).**
  + **בכל מקרה יש לבצע את הבונוס אם ורק אם סיימתם את כל דרישות הבסיס ההכרחיות לתרגיל.**
  + **חלק מהבונוסים בתרגילים השונים הם כאלה שנועדו "להקדים תרופה למכה" – מימוש דרישה בתרגיל n אשר בכל מקרה תגיע כדרישה חובה בתרגיל n+1.**

**הדבר נועד לעודד אתכם להוריד את העומס הצפוי בתרגיל n+1, מתוך הנחת יסוד שתרגיל n הוא קל יותר ומרווח יותר.**

**תכננו את עבודתכם בהתאם ושאפו "להקדים תרופה למכה", במידת האפשר. (וגם זה כלל חשוב לחיים, בלי קשר לתרגיל ולקורס).**

* + **פירוט הבונוסים, משקלם ונקודותיהם מפורט בגוף התרגיל הספציפי.**
  + **אם כבר מממשים בונוס, יש לממש את כולו, עפ"י דרישתו כדי לזכות במלוא הניקוד שהוא מקנה. בכל מקרה ההחלטה על ניקוד הבונוס היא בידי הבודק/מרצה בלבד (אל דאגה, המגמה היא להיות נדיבים ככל האפשר...)**
  + **כאמור, ולמען הסר כל ספק – ניקוד הבונוס מתווסף לניקוד התרגיל הספציפי שבו הוא מומש ולא לניקוד הסופי של הקורס. לא ניתן לקבל ציון סופי בקורס שהוא מעל ל 100 (גם אם בזכות הבונוסים הגעתם לציון כזה(.**

תרגיל reflection – תרגיל עצמאי ביחידים (5%) - הגשה: 16.3.2021

## כללי

**מועד הגשה: 16.3.2021 צורת הגשה: ביחידים בלבד**

בתרגיל זה תזכו להכיר ולתרגל את מנגנון ה Reflection בג'אווה.

המצגת מכילה ידע בסיסי המהווה נק' פתיחה בלבד לנושא זה, ולכן כחלק מהתרגיל תדרשו גם ללמוד לבד נושאים נוספים הקשורים לעולם ה reflection שייתכן ואינם מכוסים (או אינם מכוסים כהלכה) במצגת.

שימו לב כי במצגת ישנם 3 נושאים. התרגיל הוא על הנושא הראשון בלבד של Reflection.

מהות התרגיל היא לכתוב investigator שיודע לקבל מופע (instance) של איזה שהוא אובייקט, ואז יודע "לחקור" אותו ולענות על מספר שאלות בהקשרו.

במהלך התרגיל תצטרכו להשתמש אך ורק ביכולת ה Reflection של ג'אווה כפי שמוסברות במצגת.

**אין להשתמש (ואין שום צורך) בשום ספריית צד שלישי כדי לבצע את המטלות השונות !**

צפי העבודה על התרגיל, בהינתן שקראתם והבנתם את המצגת, הוא 3-4 שעות.

צפי אורך הקוד שעליכם לכתוב הוא לא יותר מ 250 שורות (אני עשיתי זאת ב 160 שורות מרווחות היטיב..)

הבדיקה לתרגיל תבוצע בצורה אוטומטית, ע"י קוד שיטען את הקובץ שלכם ויריץ את סט הבדיקות שנגזרות ממנו, כלומר יקרא לכל השיטות המוגדרות ב interface שמימשתם, תוך השוואת הערך המוחזר מהקריאה אל ערך מצופה.

## מבנה התרגיל

התרגיל מכיל ממשק בשם Investigator, אותו עליכם לממש. מהות השיטות בממשק היא לחקור instance של class אחר.

כל שיטה בממשק מתועדת היטיב מבחינת מה היא צריכה לעשות, מה הפרמטרים שהיא מקבלת ומה היא צריכה להחזיר.

חלק מהתרגיל כולל גם התמודדות עם התיעוד והבנה בעזרתו בלבד מה עליכם לעשות בכל שיטה ושיטה.

**הערות חשובות**:

1. שימו לב כי ה class שאתם מממשים חייב להכיל default public constructor !
2. שימו לב כי עליכם למקם את הממשק שקיבלתם (Investigator.java) בדיוק תחת ה package שנקרא **reflection.api** (ובהתאם לכך גם היררכיית הספריות כמובן).
3. במידה ויש מטודות הדורשות תפיסת exception, יש לתפוס אותו אולם אין להדפיסו ! (זה יוצר אי-סדר בהדפסות הפלט). במידה ומתרחשת תקלה או שתזרקו הלאה את ה exception ותוכנית הבדיקה תתמודד איתו בדרכה (הורדת ניקוד על הסעיף המדובר) או שתחזירו ערך כלשהוא כתוצאת המטודה (גם אם הוא לא נכון).
4. התוכנית מתחילה מקריאה לפונקיה load (אחת הפונקציות המתוארות בממשק). בפונקציה זו תקבלו את ה instance אותו עליכם לחקור. אפשר להניח כי זו הפונקציה הראשונה שתקרא, וכי היא תיקרא בדיוק פעם אחת.
5. כל מטודה בתרגיל עובדת בפני עצמה ופועלת על ה instance שקיבלתם במטודה load. אין להניח או להסתמך על סדר בקריאת המטודות (למען האמת בכל בדיקה הסדר הוא רנדומלי)
6. אין שום צורך (מבחינת התרגיל) לכתוב פונקציית main בקובץ התוכנית שלכם. אני לא אפעיל את התוכנית שלכם, אלא אפעיל את תוכנית הבדיקה שלי (שם יש main) והיא, בתורה, תטען את התוכנית שלכם.

## ניקוד

התרגיל שווה עד 5 נקודות **מהציון הסופי** (!!)

(לא רע ל 3 - 4 שעות עבודה ו 200 שורות קוד...)

## 

## איך בודקים ?

קיבלתם את תוכנית הבדיקה עצמה, אותה הבודק הולך להפעיל לטובת בדיקת התרגיל.

התוכנית מקבלת כפרמטר חיצוני את שם הקובץ המקומפל שלכם (.class) ובודקת אותו על מספר מקרי בדיקה.

בעותק התוכנית שקיבלתם כרגע, יש דוגמא למקרה בדיקה פשוט עליו תוכלו לנסות, להתנסות ולוודא אם אתם בכיוון הנכון או אם לאו.

כדי להפעיל את תוכנית הבדיקה עליכם לנווט לספרייה בה נמצאת תוכנית הבדיקה, ולהקליד ב CMD:

…\> RunTester <your .class file name>

אם הכל עובד כשורה, תקבלו את הפלט הבא:

Test name: Rectangle Class

Testing Rectangle Class basics...

Testing [ getTotalNumberOfMethods ]: expecting answer [6] and got [6]

…

Test Score: 100

## מה מגישים ?

עליכם להגיש קובץ zip, הכולל **בדיוק** 2 קבצים (ו 2 קבצים בלבד !):

1. קובץ התוכנית שלכם בלבד, בגרסתו המקומפלת (.class).
2. קובץ קוד המקור (למקרה של בעיות חמורות בלבד)

שם קובץ הזיפ צריך להכיל את שמכם ואת הת.ז. (אין להגיש קובץ readme בתרגיל זה...)

**מה לא מגישים ?**

1. את קובץ הממשק שקיבלתם
2. תיקיית פרויקט...
3. כל זבל אחר שבמקרה יושב לכם ליד הקוד...

## 

## מקרה בדיקה לדוגמא

במקרה זה מתואר האובייקט Rectangle היורש מאובייקט שנקרא Polygon.

מימוש זה כבר מוטמע בתוכנית הבדיקה שקיבלתם ומופע שלו יינתן כקלט לתוכנית שלכם.

המופע יאותחל בצורה הבאה:

**rectangle** = **new** Rectangle(4,6);

(המימוש הוא חלקי ולא תמיד הגיוני – אז אל תתפסו לקטנות...)

**public class** Polygon {  
  
 **private** Set<Point> **points**;  
  
 **public** Polygon() {  
 **points** = **new** HashSet<>();  
 }  
  
 **public int** getTotalPoints() {  
 **return points**.size();  
 }  
  
 **protected void** addPoint(**int** x, **int** y) {  
 **points**.add(**new** Point(x, y));  
 }  
}

**public class** Rectangle **extends** Polygon **implements** Comparable, Serializable {  
  
 **private int x**;  
 **private int y**;  
 **private final int SCALE** = 2;  
  
 **public static void** PRINT\_SOMETHING() {  
 System.***out***.println(**"this is a static method"**);  
 }  
  
 **public** Rectangle() {  
 **x** = -1;  
 **y** = -1;  
 }  
  
 **public** Rectangle(**int** x, **int** y) {  
 **this**.**x** = x;  
 **this**.**y** = y;  
 updateParent();  
 }  
  
 **private void** updateParent() {  
 addPoint(0, 0);  
 addPoint(**x**, 0);  
 addPoint(0, **y**);  
 addPoint(**x**, **y**);  
 }  
  
 **public int** calcArea() {  
 **return x** \* **y**;  
 }  
  
 **public int** calcPerimeter() {  
 **return** twice(**x**) + twice(**y**);  
 }  
  
 **private int** twice(**int** num) {  
 **return** 2 \* num;  
 }  
  
 @Override  
 **public int** compareTo(Object o) {  
  
 **return this**.calcArea() - ((Rectangle)o).calcArea();  
 }  
}

RIzpa Stock Exchange – מע' למסחר במניות

## מטרת התרגיל(ים) בקורס

חלק א' – מימוש מנוע מע' המסחר המציע פוקציונלית בסיסית, תוך תפעול המע' כאפליקציית console.

חלק ב' – הרחבת המע' במספר פיצ'רים, תוך פיתוח תצוגה גרפית כאפליקציית Java FX (Desktop).

חלק ג' – מימוש אפליקציית web המציעה מע' מקוונת המעודדת מסחר הוגן בין כמה משתתפים.

## כללי

הדרישות "הכלליות" המובאות בהמשך, תקפות לתיאור המע' בכללותה, ומהוות חלק בלתי נפרד מהגדרת כל תרגיל ותרגיל.   
בכל תרגיל נממש רק (עוד) חלק מהדרישות.

הבורסה לניירות ערך היא מוסד המאפשר קנייה ומכירה של מניות (ניירות ערך) המייצגות בעלות חלקית בחברה. הבורסה מקיימת בתוכה שוק בין מוכרים לקונים שבו המוכרים מציעים למכירה את הסחורה שבחזקתם (מניות) והקונים מעוניינים לרכוש אותה מידיהם. ברגע שהמחיר נסגר בין המוכר לקונה – העסקה מתבצעת, הסחורה (המניות) עוברות מיד ליד והתמורה (הכסף) מועברת חזרה.

בעבר, היו עומדים סוחרים בחדר הבורסה (אזור הנקרא 'רצפת המסחר') והיו כורזים מי מוכר איזו מנייה, באיזה מחיר ובאיזו כמות. מנגד היו סוחרים אחרים מביעים את רצונם לקנות את המניה (בתכלס צועקים וצורחים, דוחפים ונדחפים) כדי להיות הראשונים שזוכים בהצעת הרכישה המעניינת.   
אין צורך להכביר במילים על הבעיות הרבות הטבועות בשיטה ישנה זו.

והנה, באופן טבעי, עם התפתחות העולם האלקטרוני והמיחשובי, מזה שנים רבות (שנות ה 90) עברו בורסות ניירות הערך בשווקים השונים לעבוד עם מע' מסחר אלקטרונית, המחליפה את השיטה הישנה (האנלוגית ?) שבה היה מתנהל המסחר בעבר.

מע' מסחר אלקטרונית מאפשרת, בראש ובראשונה, שקיפות של כל המידע לכולם בלי שום יוצא מן הכלל (למנשה מחולון ולמזל ראש הדרקטוריון של בית השקעות כזה או אחר). שנית, המסחר יכול להתבצע בכמה מניות בצורה סימולטנית. כמו כן המסחר רציף ומאפשר הזרמת פק' קנייה ומכירה לכל אורך שעות המסחר באמצעים דיגיטלים מהירים ובטוחים לעין שאור מאשר היה נהוג בעבר.

**והכי חשוב: יותר לא צועקים ודוחפים !**

בתרגיל זה נפתח מע' מסחר (דמיונית) שלמה הכוללת הן את מנוע מע' המסחר עצמו והן את הממשק המאפשר למשתמשים השונים להירשם, לנהל חשבונות למכור ולקנות ניירות ערך וכו'.

המע' שלנו תקרא בפשטות "רצפה" (Rizpa) – הומאז' למושג 'רצפת המסחר' שהיה נהוג אי אז...

(שימו לב: החלק הארי של הגדרת המע' מבוסס על האופן האמיתי שבו עובדת מע' 'רצף' – מע' המסחר האלקטרונית של הבורסה בת"א. לקחתי לעצמי את החירות לפשט ו/או להיות יצירתי היכן שאפשר ולהתמקד בעיקר כך שאנחנו מממשים חלקים מינימלים מהמע' הכוללת עצמה. למתעניינים/ות הנה קישור לאתר הבורסה להסבר מלא על מע' [רצף](https://info.tase.co.il/Heb/trading/trading_phases/Pages/trading_phases.aspx).)

## מניות

במע' 'רצפה' סוחרים אך ורק במניות (לא נתמוך בשום מכשיר השקעה אחר כגון אופציות, אג"ח וכו')

כל חברה מסחרית יכולה להנפיק עצמה בבורסה (לפי שווי מסוים) ובכך להפיץ את מניותיה לציבור הרחב. מניה היא 'נייר ערך' המייצג בעלות יחסית בחברה לה הוא שייך. אם לחברה מסוימת יש סה"כ 100 מניות ואני מחזיק ב 10 מהן – אני למעשה מחזיק בעשירית משווי החברה. שווי החברה נגזר (בין היתר) מסך המניות שלה.

לכל מנייה יש מחיר. ביום ההנפקה נקבע לה מחיר פתיחה, והחל מכאן ניתן לסחור בה ולהעבירה מיד ליד בעסקאות של קנייה ומכירה. כך יוצא כי בכל עסקה מחירה של המנייה יכול להשתנות ובהתאם לכך גם שווי החברה (המורכבת מסך מניותיה).

מנייה מאופיינת בצורה הבאה:

* שם החברה לה היא שייכת - מחרוזת תווים ייחודית (אין 2 חברות בעלות אותו השם)
* שם המנייה (SYMBOL) – שם מקוצר למנייה. מופיע באותיות לועזיות גדולות בלבד
* שער המנייה – המחיר של העסקה האחרונה שבוצעה במניה

למנייה מוגדר מחזור עסקאות. מחזור עסקאות הוא שווי סך העסקאות (כמות X מחיר) שבוצעו במנייה בפרק זמן מסוים (נגיד יום, חודש וכו')

## סוגי פקודות

RSE תנהל פק' קנייה ומכירה של ניירות ערך.

עבור כל פק' יש לציין:

* שם נייר הערך (ה SYMBOL)
* כיוון הפקודה (קניה או מכירה)
* סוג הפקודה (LMT/FOK/IOC/MKT – ראו בהמשך)
* כמות מניות למכירה
* שער מבוקש (בפק' מבוססות LMT)
* תאריך כניסה למע' (חותמת זמן – timestamp)

המע' תתמוך ב 4 סוגי פקודות עיקריות, הן לקניה והן למכירה (במציאות יש סוגי פקודות נוספים):

1. LMT – מגדירה גבול (LIMIT) למחיר המנייה שבו (או יותר נכון עד אליו/ממנו) מעוניינים לבצע את הפעולה
   1. בפק' קנייה: אני מעוניין לקנות לכל היותר במחיר הזה
   2. בפק' מכירה: אני מעוניין למכור לכל הפחות במחיר הזה
2. MKT – פק' שוק (MARKET) – בצע את הפק' במחיר הנוכחי האפשרי (קנה עכשיו או מכור עכשיו במחיר המתאפשר)
3. FOK – בדומה לפק' LMT, אולם הפקודה חייבת להתבצע במלואה ומיידית או אחרת נזנחת כולה
4. IOC – בדומה לפק' LMT, אולם הפקודה חייבת להתבצע מיידית, ושאריותיה, אם קיימות נזנחות ונשכחות.

## מנוע המסחר

ככל שהמנייה מבוקשת יותר, מי שמחזיק את המנייה יסכים/יציע למכור אותה במחיר גבוה יותר. אם יימצאו קונים במחיר המדובר – הרי שתתבצע עסקה ואז מחיר המנייה (נקרא גם שער המנייה) ייקבע על פי עסקה זו.

לעומת זאת, אם אנשים לא מאמינים בחברה (או שמתרחש סקנדל בחברה) כל מי שמחזיק את המנייה ירצה "להיפטר" ממנה ולכן יהיה מוכן למכור אותה, אפילו במחיר נמוך. אם יימצאו קונים שמעוניינים לקנות את המנייה של החברה (חרף הסקנדל) – הם יוכלו לקנותה ואז ייקבע מחירה לפי עסקה זו.

מחיר מנייה נקבע לפי העסקה האחרונה שבוצעה בה, ועל כן יכול להשתנות באופן תדיר.

מנוע המסחר הוא למעשה לב המע'. תפקידו לקבל את פק' המסחר, ולבצע את המפגש בין פק' קנייה מתאימה לפק' מכירה מתאימה ולהיפך. בקרות מפגש שכזה בין 2 פקודות נגדיות, מתבצעת עסקה, הכספים והמניות מועברים בין הקונה למוכר ומחיר המנייה מתעדכן על פי מחיר העסקה. העסקה נרשמת אז ברשימת העסקאות שבוצעו והתהליך חוזר חלילה. תהליך זה מתבצע עבור כל פק' ועבור כל נייר ערך (מנייה) שהמע' מנהלת.

## תהליך המסחר ועדיפויות הפקודות

במסגרת נייר ערך מסויים, לפקודות השונות במע' בגינו ניתנת עדיפות בהתאם לשער הפקודה ואם מדובר על שערים דומים – בהתאם לתאריך קבלת הפקודה:

עבור פקודות קנייה, הפקודה בעלת השער הגבוה יותר היא העדיפה ביותר.   
במידה ויש פקודות באותו השער – אזי הפקודה המוקדמת יותר מקבלת עדיפות גבוהה יותר.

עבור פקודות מכירה, הפקודה בעלת השער הנמוך יותר היא העדיפה ביותר.   
במידה ויש פקודות באותו השער – אזי הפקודה המוקדמת ביותר מקבלת עדיפות גבוהה יותר.

בהגיע פקודה לביצוע במע', היא תופגש מול רשימת הפקודות הנגדיות לה (קנייה תופגש מול מכירה ומכירה תופגש מול קנייה) תוך התייחסות לעדיפות סדר הפקודות כפי שתואר לעיל ותוך נסיון לקיים עסקה על פי הכללים הבאים:

פקודת קנייה בשער X (רוצה לאמר: 'אני רוצה לקנות לכל היותר ב X') תופגש עם פקודת מכירה בשער Y (רוצה לאמר: 'אני רוצה למכור לכל הפחות בשער Y') כל עוד X >= Y.   
פקודת מכירה בשער Y (רוצה לאמר: 'אני רוצה למכור לכל הפחות ב Y') תופגש עם פקודת קנייה בשער X (רוצה לאמר: 'אני רוצה לקנות לכל היותר בשער X') כל עוד X >= Y

במידה וניתן לבצע עסקה, העסקה תתבצע במחיר המוגדר על פי הפקודה הנגדית שאיתה הופגשה הפקודה המדוברת.  
למשל:   
אם אני מגיש פקודת קנייה שמופגשת מול פקודת מכירה וניתן לקיים עסקה – גובה העסקה יהיה כפי המוגדר בפקודת המכירה  
אם אני מגיש פקודת מכירה שמופגשת מול פקודת קנייה וניתן לקיים עסקה – גובה העסקה יהיה כפי המוגדר בפקודת הקנייה.

יש לבצע את המעבר על הפקודות הנגדיות עד למיצוי הכמות לקנייה/מכירה בפקודה או עד לסיום המעבר על רשימת הפקודות הנגדיות.

ייתכן והפקודה לא תוכל להתבצע במלואה. את שארית הפקודה (או כולה, במידה ולא יכלה להתבצע כלל) יש להוסיף לרשימת הפקודות הרלבנטיות (קנייה או מכירה) על פי סדר העדיפויות כפי שהוגדר והוסבר מעלה. כך למעשה מנוע המסחר מקיים בכל רגע נתון 3 רשימות עיקריות (עבור כל מנייה):

1. רשימת הוראות הקנייה הממתינות לביצוע, מסודרות על פי סדר העדיפויות כפי שהוגדר לעיל.
2. רשימות הוראות המכירה הממתינות לביצוע, מסודרות על פי סדר העדיפויות כפי שהוגדר לעיל.
3. רשימת העסקאות שבוצעו בהצלחה.

עיקבו **באדיקות** על [נספח א'](#appendixA) המציג השתלשלות מאורעות אפשרית דרכה תוכלו ללמוד את התהליך על בוריו.

## משתמשים

המע' תתמוך בניהול משתמשים. לכל משתמש יהיה חשבון ומידע על המניות שבבעולותו.

משתמש יוכל להגיש פק' קניה ומכירה למע' המסחר ויקבל דיווחים כאשר/באם העסקאות שהגיש בוצעו.

כל משתמש יוכל לראות את היסטוריית העסקאות שביצע כמו גם את פרטי החשבון ההיסטוריים שלו.

בנוסף יוכל משתמש לקבל מידע הסטורי על כל מניה ומנייה המנוהלת במע'.

## הנחיות ספציפיות למימוש מערכת RIzpa Stock Exchange

1. **המטרה היא לבנות מנוע מערכת גנרי, כזה שידע לקבל את הפרטים לגבי המניות השונות מתוך קובץ נתונים בפורמט XML (עבודה עם XML'ים תילמד במהלך הקורס כמובן).**

**מנוע המערכת הגנרי ילך וישתכלל מתרגיל לתרגיל, בהתאם לפיצ'רים השונים. כך תוכלו לחוות מהלך שלם של מוצר החל מרעיון קטן במימוש בסיסי וכלה במנוע מע' המניע אפליקציית ווב שלמה.**

1. **כחלק מהמע' תצטרכו לחשוב ולבחור לבד את מבני הנתונים השונים שישרתו את הצרכים של דרישות המע'. זהו לא קורס במבני נתונים או באלגוריתמים, ומבני הנתונים/אלגוריתמים שתבחרו לממש לא חייבים להיות היעילים ביותר או האופטימליים. מספיק שהם יעבוד בצורה נכונה (ללא טעויות) ובזמן סביר.**

תרגיל 1 – מימוש **Rizpa Stock Exchange** כאפליקציית Console (25%) - הגשה: 12.4.2021

## פרטים יבשים

צפי תחילת עבודה: **14.3.21** תאריך הגשה: **12.4.21**

צפי זמן לביצוע: **3 שבועות** ציון אפשרי מקסימלי: 110

משקל התרגיל: **25%** קושי: **סביר**

**מטרת התרגיל העיקרית**

1. הקמת מנוע המע' הבסיסי
2. יצירת ממשק console לתפעול המערכת

## דרישות

1. בתרגיל זה תממשו את מנוע המערכת, אשר יודע לקבל נתונים על מניות במע'. המע' תאפשר למשתמש לצפות במידע על המניות השונות (כולל מידע של הסטורית העסקאות) כמו גם לאפשר למשתמש להזין פק' קנייה ומכירה עבור מנייה מסויימת ובכך להניע את תהליך המסחר.

את מנוע המע' תלווה שכבת UI מסוג של console application אשר תאפשר למשתמש "לדבר" עם המע' באמצעות הצגת סדרה של תפריטים ופעולות אפשריות. המשתמש יבחר פעולה לביצוע, יכניס קלט נדרש (במקרים הנדרשים) ויקבל חזרה את הפלט מהמע' וחוזר חלילה.

1. בתרגיל זה תתמכו בגרסה "קלת משקל" של המע':

* כל הפרטים לגבי המניות השונות יוגדרו במלואם בקובץ ה XML.
* אין "לקוחות" או ניהול חשבונות בתרגיל זה. הלקוח האחד והיחיד הוא המשתמש. אין לו שם או מספר סידורי.
* המע' תתמוך בסוג פקודה אחד בלבד: LMT

1. פרטיה הטכנים של המערכת יינתנו מקובץ XML (כמפורט [בנספח](#xmlAppendix)).

עליכם לוודא בדיקת קלט לקובץ ה XML ולוודא כי הקובץ מכיל מידע תקין ואמין.

(מובטח כי הקובץ יהיה תקין schema-wise אבל לא בהכרח תקין application-wise...)

במידה והקובץ מתגלה כתקול, יש להודיע על כך בצורה מסודרת ונאותה למשתמש. יש לפרט מהי סיבת התקלה (ולא להסתפק במשפט סתום-בלום כמו: "הקובץ תקול...").

בפרט יש לוודא את הפרטים הבאים:

* 1. הקובץ אכן קיים, והוא מסוג XML (די לבדוק לשם כך כי הוא נגמר בסיומת xml)
  2. אין מניות בעלות אותו SYMBOL
  3. אין מניות שונות לאותה החברה (שם חברה לא מופיע פעמיים)

1. יש לוודא תקינות קלט כחלק מכל אינטרקציה עם המשתמש בכל מקום שבו זה רלבנטי:

אם אתם מצפים לקבל מספר – לא לקרוס כי הכניסו לכם בטעות (או בכוונה) טקסט וכו'.

בכל מקרה של תקלה יש להיות מאוד ברורים במסר שמעבירים חזרה למשתמש: מה קרה ? מה הייתה מהות התקלה ? היכן שזה רלבנטי, איך לתקנה וכו'.

חישבו איך להיות ידידותיים למשתמש ולעולם אל תניחו כי מי שמשתמש באפליקציה שלכם הוא מתכנת בעצמו או מישהו שמגיע מהתחום ו"מכיר" איך דברים עובדים לבד. (זה הזמן לחשוב על...)

1. **אין צורך** להשתמש בצבעים שונים במהלך תרגיל זה בעת ההדפסה ל console.

יתרה מזאת, ישנו צפי רב (ניסיון מהסמסטרים הקודמים) כי ניסיון לעשות כן תוך שימוש בספריות צד שלישי קורס אצל הבודק,

מעוות את כל תצוגת המסך וגורם לחוסר יכולת לבדוק את ההגשה.

גם אם בדקתם את זה אצלכם וזה עבד.

גם אם בדקתם במחשב של השכנה וזה עבד.

כמו כן **אין** לנקות את המסך בין פקודה לפקודה.

1. בתרגיל זה בכל מקום שבו מוצג תאריך, יש להציג רק את זמן השעון בפורמט הבא:

HH:mm:ss:SSS

קיראו והבינו לבד כיצד ניתן להוציא חתימת זמן נוכחית, כמו גם להציגה בתצורה הנדרשת.

(הנה דוגמא [אחת](https://stackoverflow.com/questions/23068676/how-to-get-current-timestamp-in-string-format-in-java-yyyy-mm-dd-hh-mm-ss) מני רבות. אני ממליץ על תשובתו של zgc)

1. עליכם לכתוב ממשק משתמש בתצורת console.

ממשק המשתמש יכיל סט סופי של פקודות שדרכן ניתן יהיה להפעיל את המערכת.

אחרי הצגת תפריט הפקודות יש לחכות לקלט מהמשתמש באשר לפעולה אותה הוא רוצה לבצע. לאחר ביצוע הפעולה (שאולי תגרור בקשת קלט נוסף מהמשתמש) יש להציג את הפלט החוזר ממנה (לכל פקודה יש פלט החוזר ממנה) ואז להציג שוב את התפריט וחוזר חלילה.

**שימו לב**: ישנן פקודות שאין הגיון לבצע אותן אם לא קדמו להם פקודות אחרות. במידה וזה קורה יש להציג הודעת שגיאה רלבנטית למשתמש ולאפשר את המשך מהלך פעילות המע'.

רשימת הפקודות שיש לתמוך בהן:

1. קריאת קובץ פרטי המע'

פקודה זו טוענת את פרטי המערכת מתוך קובץ נתונים בפורמט XML.

קבצי דוגמא מתאימים הועלו מבעוד מועד לאתר הקורס ואתם מוזמנים להורידם ולבחון אותם בהתאם.   
(אתם מעודדים לייצר לעצמכם קבצי בדיקה נוספים כדי לבדוק את המע' בצורה יסודית וטובה יותר כאוות נפשכם).

יש לבקש מהמשתמש נתיב מלא לקובץ ה XML אותו הוא רוצה לטעון למע'.   
הנתיב יכול להכיל רווחים בתוכו (למשל "program files") ויש לוודא כי הדבר לא מכשיל אתכם (ולא שהוא אמור).   
הנתיב יכיל רק אותיות באנגלית (לא ג'יבריש של אותיות בעברית וכו')

הקובץ יהיה קובץ XML שפרטיו וחוקיו המפורטים מובאים [בנספח](#xmlAppendix) לתרגיל זה.   
אתם מצופים לעבור על פרטים אלה ולהתייחס אליהם כחלק מהגדרת התרגיל.

יש לוודא תקינות הנתונים המופיעים בקובץ כפי שהוגדרו בסעיף 3.

במידה והקובץ לא תקין יש לדווח זאת למשתמש בצורה ברורה כך שניתן יהיה להבין מה לא תקין בקובץ.

אין לקרוס על exception במידה וקובץ מתגלה כאינו תקין ; יש לאפשר למע' להמשיך לפעול במצב זה.

(כחלק מבדיקת התרגיל יטענו למערכת קבצים שאינם חוקיים כדי לבדוק מהי התגובה).

במידה והקובץ נמצא תקין – יש לטעון את פרטיו למע' ולדווח על כך למשתמש.

**דגשים:**

1. יש לאפשר למשתמש לטעון כמה קבצים אחד אחרי השני (כלומר להפעיל את הפקודה כמה פעמים רצוף).

כל קובץ תקין "דורס" לחלוטין את כל פרטי הקובץ (התקין) שהיה טעון לפניו במע' (ככל שהיה כזה).

כל נסיון טעינה של קובץ תקול לא דורסת את פרטי הקובץ (התקין) האחרון שהיה במע' (ככל שהיה כזה)

1. פקודה זו מוצגת ומאופשרת תמיד. אפשר לבחור בה בכל רגע נתון במע'.
2. הצגת המניות במע'

פקודה זו תציג למשתמש את כל המניות המוגדרות במע'.

עבור כל מנייה יש להציג:

* שם המנייה (SYNBOL)
* שם החברה לה המניה שייכת
* מחיר המנייה הנוכחי
* סה"כ עסקאות שבוצעו במנייה עד עכשיו
* מחזור העסקאות שבוצעו במנייה

**דגשים**:

* אין לאפשר פקודה זו אם לא טעון קובץ תקין במע' (או להציג פלט מתאים).

1. הצגת מידע על מנייה בודדת

פקודה זו תאפשר ללקוח לבחור מנייה אחת ולצפות במידע מפורט יותר לגביה בלבד.

את המנייה יש לבחור על פי שמה. יש לאפשר למשתמש להכניס את השם בכל case שהוא ירצה ולהציג לו את פרטי המניה הנדרשת (למשל אפשר להכניס GoGLe והמע' צריכה להבין שמדובר במנייה שה SYMBOL שלה הוא GOGLE)

המידעים שיש להציג עבור כל מנייה:

* כל המידעים המפורטים עבור כל מנייה על פי פקודה מס' 2
* פירוט של כל העסקאות שבוצעו במנייה עד כה, מוצגות על פי תאריך מהקדום ביותר (למטה) עד העדכני ביותר (למעלה, הראשון)

עבור כל עסקה:

* + תאריך (למעשה השעה, בפורמט שהוגדר מעלה)
  + כמות מניות שנמכרה
  + מחיר המכירה
  + סה"כ שווי העסקה

דגשים:

1. אין לאפשר פקודה זו אם לא טעון קובץ תקין במע' (או להציג פלט מתאים).
2. במידה ושם המנייה לא קיים יש להציג למשתמש הודעה מתאימה
3. ביצוע הוראת מסחר

פקודה זו תאפשר למשתמש להגיש למע' הוראת מסחר לביצוע.

ההוראה יכולה להיות לקנייה או למכירה (בתרגיל זה אין מושג של משתמש ואין חשיבות להאם הוא מחזיק את המניה שהוא רוצה למכור או לא)

בתרגיל זה נתמוך בפקודת LIMIT בלבד. בפקודה זו יש לבחור מגבלה על גובה העסקה ולציית למגבלה.

כאשר המשתמש מכניס פקודה לביצוע עליו להכניס את הפרטים הבאים:

* האם הפקודה היא לקנייה או מכירה
* שם המניה לפקודה. יש להכניס את ה SYMBOL שיכול להינתן בכל CASE שהמשתמש יחפץ
* כמות לקנייה או מכירה
* שער גבול לביצוע ההוראה.

אם זו הוראת קנייה הרי שהפקודה היא לקנות **לכל היותר** במחיר המדובר

אם זו הוראת מכירה הרי שהפקודה היא למכור **לכל הפחות** במחיר המדובר.

הפקודה מתקבלת במנוע המע' ויש לבדוק את תקינותה. בפרט:

* שם המנייה קיים ומוכר במע'
* כמות לקנייה או מכירה גדולה מ 0

באם הפקודה אינה תקינה – יש להציג הודעה מתאימה למשתמש ולחזור לתפריט הראשי.

באם הפקודה תקינה, תוצמד לה חותמת זמן נוכחית (timestamp) והיא תתחיל את תהליך המפגש מול פקודות נגדיות. פקודת קנייה תופגש מול פקודות המכירה הקיימות ופקודת מכירה תופגש מול פקודות הקנייה הקיימות.   
באם נמצאה התאמה (חלקית או מלאה) מתבצעת עסקה והיא נרשמת ברשימת העסקאות שבוצעו במנייה דנן.

אם הפקודה לא בוצעה במלואה, או בכלל, היא תתווסף לרשימת הפקודות המתאימה במנוע המע', בהמתינה למפגש עתידי עם פקודה נגדית.

(עיקבו על [נספח א'](#appendixA) כדי לקבל מושג איך מתנהל תרחיש מכירה/קנייה מתחילתו ועד סופו על שלל אפשרויותיו)

הפלט למשתמש צריך לציין אם הפקודה בוצעה ונוצרה עסקה (או עסקאות) בעקבותיה (מלאה או חלקית). במקרה זה יש לפרט את פרטי העסקה (או העסקאות) שנוצרה.

אם הפקודה לא בוצעה (כולה או חלקה) ועל כן נכנסה לספר הפקודות – יש להודיע גם זאת למשתמש בצורה מסודרת וברורה.

דגשים:

* אין לאפשר פקודה זו אם לא טעון קובץ תקין במע' (או להציג פלט מתאים).

1. הצגת רשימות הפקודות לביצוע

פקודה זו תציג למשתמש את נבכי מנוע המסחר.

עבור כל נייר ערך בנפרד, יש להראות למשתמש 3 רשימות:

1. רשימת הפקודות הממתינות לקנייה
2. רשימת הפקודות הממתינות למכירה
3. רשימת העסקאות שבוצעו

(שימו לב: כל אחת מהרשימות יכולה להיות גם רשימה ריקה ; עדיין יש להציגה ולציין שאין בה אברים)

עבור כל פקודה/עסקה, בכל רשימה יש להציג את המידע הבא:

1. תאריך ביצוע הפקודה (השעה בפורמט שצויין לעיל)
2. כמות
3. שער
4. מחזור (כמות X שער)

אברי הרשימות יוצגו כמובן על פי סדר העדיפויות של הפקודות, כפי שהוסבר בחלק הכללי של התרגיל (מעדיפות ראשונה למעלה עד לעדיפות האחרונה למטה)

בנוסף יש להציג סיכום של המחזור הכולל של כל אחת מהרשימות, כלומר סך המחזור עבור הפקודות הממתינות לקנייה (מחזורים בהמתנה), מכירה וסך המחזור של העסקאות שבוצעו.

1. יציאה מהמע'

פקודה זו יוצאת מהמערכת.

## חלוקה למודולים

בתרגיל זה **חובה** לייצר (לפחות) 2 מודולים (מהם תפיקו בהמשך 2 jar'ים):

* + 1. ממשק ה ui, המציג את התפריטים השונים, אחראי על קליטת קלט מהמשתמש והחזרת הפלט למשתמש.

שימו לב זהו המודול "האקטיבי", המניע את כל המע'. הוא זה האחראי על פנייה ותפעול מנוע המערכת.

כפועל יוצא, כל ההדפסות של מידעים למשתמש (System.out.println) מתבצעות **אך ורק** מתוך מודול זה ; במודול זה יושבת מטודת ה main ; מודול זה אחראי על לולאת תפעול המע' העיקרית, הצגת התפריטים, איסוף הקלט מהמשתמש, הצגת הפלטים למשתמש וכו'.

* + 1. מנוע המערכת, האחראי על קבלת הפקודות (ממודול ה ui), ביצועם והחזרת פלטים מתאימים.

שימו לב שמודול זה "פסיבי", והוא **רק** מגיב לבקשות ולפקודות המתקבלות ממקורות בלתי ידועים לו (בתרגיל זה מודול #1). בתרגילים הבאים מקורות נוספים יפנו אליו לקבלת מידע וחשוב מאוד להקפיד על כך **שמודול זה אינו מכיר/מודע למי פונה אליו.**

## איך מתחילים ? (המלצה...)

ככלל, מומלץ להתחיל משלד ראשוני שמאפשר קליטת קלט מהמשתמש, העברתו למנוע המע' והחזרת התשובה למשתמש. ברגע שתעשו זאת מההתחלה במצב שבו אתם כבר עובדים עם 2 מודולים – יקל עליכם להרחיב את המע' עוד ועוד ולתמוך בפקודות הנדרשות.

התחילו בהקמת מבני הנתונים של המע' המתארים מנייה. תוכלו להתחיל למלא את מבני הנתונים האלה עצמאית אפילו לפני שתממשו את קריאת המידע מה XML. התחילו בתמיכה בפקודות 2 ו 3 המציגות את המידעים הסטטים שכבר מגיעים מקובץ התרגיל. המשיכו למימוש פקודה 1 שמאפשרת ממש טעינה של מידעים מקובץ (והשתמשו בפקודות 2 ו 3 כדי לוודא את נכונות המידעים והקשרים בין מבני הנתונים שלכם).

רק לאחר מכן תימכו באפשרות להגיש ולבצע את הוראות המסחר השונות. מומלץ. לפתח את פקודה 4 ו5 במקביל כך שתוכלו להיעזר בפקודה 5 כדי לוודא את נכונות ביצוע פקודה 4.

## בונוסים

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | סוג | מהות | למה שווה לי ? | כמה שווה לי ? |
| 1 | הקדמת תרופה למכה | מימוש תמיכה בפקודת MKT.  כחלק מפקודה 4 בתפריט, שבה מגישים פקודת קנייה/מכירה למע', אפשרו למשתמש לבחור את סוג הפקודה: LMT או MKT.  במידה וזו פקודת MKT עיקבו אחר הוראות הביצוע שלה כפי שהן מופיעות בתרגיל 2 סעיף 3 כמו גם [בנספח ב'](#appendixB) | כי גם ככה הולכים לממש זאת בהמשך – אז למה לא ?? | 10 נקודות  (עד ל 100) |
| 2 | הגדלת ראש | אפשרו למשתמש לשמור את מצב המע' לקובץ.  הוסיפו פקודה לתפריט שתאפשר למשתמש להכניס נתיב שאליו הוא ירצה שמצב המע' יישמר.  עליכם לשמור את המצב של כל המניות הנוכחיות, של כל ספרי הפקודות והעסקאות שבוצעו עד עכשיו.  הוסיפו פקודה נוספת לתפריט שתאפשר למשתמש לטעון את כל המע' ישירות מהקובץ ששמרתם אותה אליה. (שימו לב, זוהי פקודה נפרדת מפקודת החובה שכבר קיימת לטעינת המע' מקובץ ה XML).  לבחירתכם פורמט ואופן שמירת המידע של המע' לקובץ. | טעימה קלה מדרכי שמירת נתונים | 10 נקודות  **(מעל ל 100)** |

## סבבה, סיימתי. מה ואיך להגיש ?

יש להגיש קובץ zip המכיל:

1. 2 jar'ים (לפחות) שהם כל הקוד שלכם, בצירוף קובץ אצווה (batch) שהפעלתו תריץ את התוכנית

(כלומר תבצע java -jar <class name> וכו').

1. קבצי קוד המקור של הפרויקט שלכם.
2. קובץ ההגשה יכיל גם קובץ **readme** שיכיל הסבר על המערכת, בחירותיכם השונות במקומות שבהם היו לכם בחירה, כמו גם כל דבר נוסף העולה על דעתכם שחשוב שהבודק תדע.
3. יש לכלול בקובץ ה **readme** גם תיעוד והסבר כללי (וממצה) של המחלקות העיקריות ותפקידם.
4. יש לכלול בקובץ ה **readme** גם פירוט של המגישים שם, ת.ז. ואי מייל זמין ורלבנטי (!!) – במידה ויהיה צורך ליצור קשר.
5. במידה ומימשתם את מי מהבונוסים לעיל, ציינו זאת בקובץ ה readme כדי שהבודק ידע לבדוק בהתאם.

בונוס שימומש אבל לא יתועד – לא ייבדק !

## שאלות ותשובות

|  |  |  |  |
| --- | --- | --- | --- |
| # | תאריך | שאלה | תשובה |
| 1 |  |  |  |

תרגיל 2 – מימוש**Rizpa Stock Exchange** כאפליקציית JavaFX (30%) – הגשה: 24.5.2021

## פרטים יבשים

צפי תחילת עבודה: **18.4.21** תאריך הגשה: **24.5.21**

צפי זמן לביצוע: **4 שבועות** ציון אפשרי מקסימלי: **110**

משקל התרגיל: **30%** קושי: **מאתגר**

**מטרות התרגיל העיקרית**

1. מימוש האפליקציה כ desktop application באמצעות Java FX
2. הרחבת מנוע המערכת לתמיכה בפיצ'רים נוספים (משתמשים, פקודות מסחר נוספות)

## דרישות

* 1. בתרגיל זה תממשו את המערכת כ desktop application באמצעות ארכיטקטורת JavaFX כפי שנלמד בכיתה.

בתרגיל זה אין יותר סדרה של "תפריטים" כפי שהיו בתרגיל 1. כל ממשק המשתמש הוא ממשק גרפי בלבד, המבוסס על JavaFX. כל האינטראקציה עם המשתמש מתרחשת דרך פעולותיו על גבי הפקדים השונים שתציגו לו (כפתורים, תיבות טקסט וכו').

1. העבודה במע' תחל בטעינת קובץ XML (כמו שהיה בתרגיל 1).

לבד מהמידע לגבי הפריטים/חנויות אפשריות, הקובץ יכיל גם מידע לגבי זהות של לקוחות רשומים במע', כמו גם אחזקות המניות שלהם.

קבצי ה XML יהיו מוגדרים על פי סכמה חדשה של תרגיל 2 (ראו [נספח XML](#_סכמת_קובץ_XML_2) כדי להבין מה התוספות/הבדלים יחסית לסכמה של תרגיל 1)

את קובץ ה XML יש לטעון באמצעות FileChooser Dialog בלבד, כפי שהוצג בדוגמא המסכמת. הטעינה מאחורי הקלעים תתבצע באמצעות task כפי שנלמד בכיתה תוך הצגת חיווי דינמי למשתמש לגבי אופן התקדמות הטעינה (במידת הצורך, הטמיעו השהיה מלאכותית כדי לסמלץ טעינה מורכבת וארוכה)

יש לבצע בדיקת תקינות על תוכנו של קובץ ה XML.   
כל הבדיקות כפי שבוצעו בתרגיל 1 ובנוסף:

1. בתוך קבוצת הלקוחות, אין 2 לקוחות בעלי שם זהה.
2. כל מנייה שהלקוח מחזיק – אכן מוגדרת וקיימת במסגרת הקובץ (על פי הסימבול – case sensitive).
3. בתרגיל זה נגדיר סוג חדש של פקודת מסחר: פקודת שוק (MKT)

בפקודה זו (המתאפשרת הן לקנייה והן למכירה) אין הגבלת שער. המוטיבציה היא לבצע את העסקה (כולה או חלקה) בשער האפשרי הטוב ביותר במצב הנתון במע', על פי רשימות הפקודות הנגדיות הרלבנטיות, המסודרות על פי סדר העדיפויות כמובן.

במידה והפקודה לא יכולה להתבצע במלואה, היא נכנסת לרשימת הפקודות הרלבנטית (קנייה או מכירה) עם ערך שער של העסקה האחרונה שבוצעה בנייר, ומעכשיו נחווית כפקודת LMT לכל עניין ודבר מבחינת תפעולה.   
שימו לב שעדיין צריך לזכור (ולהציג) כי זו פקודה מסוג MKT.

עיקבו אחר [נספח ב'](#appendixB) (כהמשכו של נספח א'), בו תראו דוגמא ממצה כיצד פקודה זו באה לידי ביטוי.

1. בתרגיל זה נתמוך בהגדרה של משתמשים שונים. (אך עדיין לא חשבונות וטרנזקציות).

המידע לגבי המשתמשים השונים יגיע כחלק מקובץ ה XML.

עבור כל משתמש יינתן שם ייחודי לו, כמו גם רשימה של מניות שהוא מחזיק בהן יחד עם כמות מכל מנייה.

(אני ממליץ להשתמש ב tabPane כדי להציג כל לקוח ב tab משלו. כך תייצרו הפרדה מובנית בין פרטי לקוח אחד למשנהו)

1. תהליך המסחר:

המשתמש במע' יבחר את הלקוח אשר בשמו הוא רוצה לבצע את הפעולות השונות.

בבחירת משתמש יוצגו הפרטים השונים שלו:

* שם המשתמש
* רשימת המניות אותן הוא מחזיק (SYMBOL, כמות, שער נוכחי של המנייה)
* סך שווי אחזקותיו (סכימת כל כמות המניות X שער נוכחי)

באמצעות כפתור ייעודי, המשתמש יוכל להתחיל את תהליך הגשת פקודת מסחר, המורכבת מהשלבים הבאים (סדר השלבים הנ"ל הוא חובה, נא להיצמד אליו ולא לשנות את הסדר):

1. יש לבחור אם הפקודה היא לקניה או למכירה.
2. יש לבחור את המנייה הרלבנטית לפקודה:

בפקודת קנייה יש לאפשר לבחור את כל אחד מהניירות הקיימים במע', הן אלה שבחזקתו (רוצה לקנות עוד) והן אלה שאינן כרגע בחזקתו.

בפקודת מכירה יש לאפשר לבחור אך ורק את הניירות הנמצאים בחזקתו.

1. יש לבחור את סוג הפקודה: LMT או MKT.

עבור פקודת LMT יש לבקש גם את הגבלת השער.   
עבור פקודת MKT אין הגבלת שער (היא מתרחשת במחיר השוק הנוכחי)

1. יש לבחור את הכמות של המניות שהוא רוצה לקנות/למכור.

עבור קנייה אפשר לקבל כל כמות (גדולה מ 0)  
עבור מכירה אפשר לקבל את מקסימום הכמות שהמשתמש מחזיק בה כרגע (וכמובן גדולה מ 0).

1. המשתמש יחתום את הפעולה ויאשר כי הוא רוצה להגיש את הפקודה למע' המסחר.

ברגע שהפקודה מוגשת למע' המסחר היא עוברת את תהליך ההפגשה.   
במידה והפקודה שהוגשה מומשה בהצלחה (כולה או חלקה) יש להציג הודעה אינפורמטיבית למשתמש כמו גם להציג את המידע במקומות הרלבנטים.

במידה והפקודה לא מומשה כלל – יש להציג הודעה אינפורמטיבית למשתמש גם כן.

1. צפייה בספרי הפקודות והעסקאות:

אפשרו למשתמש להציץ בנבכי מע' המסחר בכל רגע נתון (אפשר ע"י כפתור ייעודי או אזור מיוחד במסך).

המשתמש יוכל לבחור מנייה אחת (מתוך סך המניות המנוהלות בתיק) ועבור כל אחת לראות את המידעים הבאים:

* רשימות הפקודות הממתינות לביצוע (קנייה ומכירה), מסודרות וממוינות כמובן על פי עדיפויותן. יש להציג רשימות נפרדות עבור קנייה ועבור מכירה.

עבור כל פקודה יש להציג:

* + תאריך (חתימת זמן כפי שהוגדרה לעיל כבר בתרגיל 1)
  + סוג (MKT / LMT)
  + כמות
  + מחיר
  + שם יוזם הפקודה
* רשימת העסקאות שבוצעו. עבור כל עסקה, לבד מהמידע הנ"ל, יש לציין את שם הקונה והמוכר בעסקה

(בהתאם להמלצתי מעלה, יכולת זו יכולה להיות ממומשת ב tab נוסף שנקרא admin...)

1. כחלק מבדיקת המערכת ישונה גודל המסך (resize) ותיבדק המע' שלכם במסך בגודל שונה. עליכם לדאוג לסידור נכון של רכיבי ה ui ולוודא את תקינותם גם בגודל קטן. מומלץ מאוד להשתמש ב scroll pane (יש ללמוד עליו לבד) כדי להציג תוכן גדול בתוך שטח מסך קטן.

**למען הסר כל ספק ומניעת כל התחכמות שהיא, אין "לטפל" ב resize ע"י כך שפשוט תמנעו מהמסך להיות resizable.**

במידה ומימשתם בונוסים בתרגיל הקודם אין הכרח לגרום להם לפעול גם בתרגיל זה, אולם אם זה מתאפשר זה יחמם את ליבי. (למען הסר ספק, חימום ליבי אינו מתורגם להעלאת נקודות).

## חלוקה למודולים

בתרגיל זה יש לממש את מודול ה Java FX כ jar עצמאי שעובד מול מודול מנוע המערכת (שגם הוא משתכלל כפוקנציה של הדרישות השונות).

מומלץ לפתח את המודול של JavaFX באותו פרוייקט ב intelij, כמודול נוסף ליד מודול מנוע המערכת.

(שימו לב: אין צורך לגרום למודול הקונסול של תרגיל 1 להמשיך לעבוד גם עם המנוע של תרגיל 2.)

## איך מתחילים ? (המלצה...)

התחילו בהגיית המראה הכללי של המע'. ואז המשיכו לייצר את המע' ב java fx על פי הגדרתה הראשונית.

התחילו בהוספת כפתור של טעינת הקובץ. יצרו את המצב שבו אתם מצליחים לטעון את הקובץ בהצלחה. (כרגע בלי ה TASK אלא רגיל כמו שהיה בתרגיל 1). עכשיו כשהמידעים כבר בפנים, התחילו לעבור על תצוגת המידעים השונים.

התחילו בהגיית איך כל רכיב נראה (רכיב המציג רשימת פקודת; רכיב המציג מידע על מניות של לקוח וכו'), ואז צרו את ה FXML, css, controller של הרכיבים הללו בהתאם לדרישות השונות. עכשיו שלבו מופעים של הרכיבים האלה באפליקציה הראשית שלכם. ממליץ להתחיל בהצגת הלקוחות על כלל פרטיהם, ואחרי כן להתקדם להצגת תוכנו של מנוע המסחר.

המשיכו בתהליך הגשת פקודת מסחר חדשה, כפי שהיה נהוג בתרגיל 1 (רק LMT).   
כשהכל עובד ויש ביכולתכם לצפות במידע של מנוע המסחר – התקדמו למימוש פקודת המכירה מסוג MKT

השאירו את ההתעסקות עם טעינת הקובץ באמצעות task לסוף התרגיל.

## בונוסים

היכן שזה רלבנטי (בונוס 1 ו 2) על הבונוסים להגיע "מכובים" בתור התחלה כך שהבודק "יפתח" אותם רק לטובת הבדיקה שלהם

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | סוג | מהות | למה שווה לי ? | כמה שווה לי ? |
| 1 | פשוט מגניב ! | אנימציות  ממשו לפחות 2 סוגי אנימציות לבחירתכם, כפי שנלמד בכיתה. תעדו בקובץ ה README איזה אנימציות מימשתן, והיכן צפוי המשתמש להיתקל בהן.  האנימציה צריכה להתבצע לכל היותר במשך 2 שניות  יש לאפשר גם לנטרל את האנימציות הנ"ל, כך שהיא לא תאיט את התקדמות המערכת | כי זה מה-זה קל... | 5 נקודות  (ועד ל 100) |
| 2 | הגדלת ראש נאה | החלפת skin למע'  בבונוס זה תממשו (לפחות) 2 ערכות צבעים נוספות על הערכה הדיפולטית, ותאפשרו למשתמש להחליף את ערכות הצבעים. שימו לב ערכת הצבעים כוללת (לכל הפחות):   1. החלפת רקע של כל המסך 2. החלפת המראה של הכפתורים על המסך 3. החלפת הפונט והגודל של כל ה label על המסך | כי זה בכל זאת תרגיל בממשק גרפי – ומה יותר גרפי מזה ??? | **5 נקודות**  **(ועד ל 100)** |
| 3 | הגדלת ראש חביבה | ממשו בסעיף 6 את היכולת לראות גרף של מנייה נבחרת  הגרף יציג את מחיר המנייה כפי שהשתנה לאורך העסקאות השונות. מדובר על גרף קווי רגיל כאשר ציר ה X הוא ציר הזמן וציר ה Y הוא מחיר המנייה |  | **10 נקודות**  **(מעל ל 100)** |

## סבבה, סיימתי. מה ואיך להגיש ?

יש להגיש קובץ zip המכיל:

1. 2 jar'ים (לפחות) המהווים את כל הקוד שלכם, בצירוף קובץ אצווה (batch) שהפעלתו תריץ את התוכנית
2. קבצי קוד המקור של הפרויקט שלכם.
3. קובץ ההגשה יכיל גם קובץ readme שיכיל הסבר על המערכת, בחירותיכם השונות במקומות שבהם היו לכם בחירה, כמו גם כל דבר נוסף העולה על דעתכם שחשוב שהבודקת תדע.
4. יש לכלול בקובץ ה **readme** גם תיעוד והסבר כללי (וממצה) של המחלקות העיקריות החדשות ותפקידם. (חישבו מה יסייע לבודק להיכנס ביתר קלות לקוד שלכם ולהבין מי נגד מי...)
5. במידה ומימשתם את מי מהבונוסים לעיל, ציינו זאת בקובץ ה readme כדי שהבודק ידע לבדוק בהתאם.

בונוס שימומש אבל לא יתועד – לא ייבדק !

## שאלות ותשובות

|  |  |  |  |
| --- | --- | --- | --- |
| # | תאריך | שאלה | תשובה |
|  |  |  |  |

תרגיל 3 – מימוש **Rizpa Stock Exchange** כאפליקציית WEB (40%) – הגשה: 17.7.2021

## פרטים יבשים

צפי תחילת עבודה: **6.6.21** תאריך הגשה: **17.7.21**

צפי זמן לביצוע: **5 שבועות** ציון אפשרי מקסימלי: **130**

משקל התרגיל: 40**%** קושי: **ווא'עליה אלל'בחטי**...

**מטרות התרגיל העיקרית**

1. מימוש המערכת כאפליקציית web.
2. מימוש חשבונות משתמשים וניהול כספים
3. מימוש הנפקת חברות חדשות
4. מימוש פקודות מסחר נוספות

## דרישות

1. בתרגיל זה נפתח מע' web שתציע למשתמשים ממשק נוח לסחור במניות.

המע' תאפשר למשתמשים להירשם אליה בצורה דינמית (אין יותר משתמשים הרשומים בקובץ ה XML.)

במע' יוגדרו 2 סוגים של משתמשים: "סוחר" ומנהל מע' (admin).

משתמש מסוג "סוחר" יכול להעלות קובץ XML בפורמט של תרגיל 3 המגדיר מניות במע' ואת אחזקותיו במניות אלה.

המע' תנהל אם כן את סך כל המניות (והמשתמשים) הנגזרים מסך כל קבצי המידע שהועלו אליה על ידי המשתמשים השונים.

המשתמשים יוכלו לבחור מניה מסויימת ולבצע פקודות בהקשרה (קנייה/מכירה) ולקבל פידבק, מידעים והתראות בזמן אמת לגבי מצב התיק שלהם וביצוע הפקודות.

בנוסף נגדיר פקודות מסחר חדשות (FOC\IOC), כמו גם נאפשר הנפקה של חברה – יצירה של מנייה חדשה לחלוטין במע' יש מאין !

למשתמש מסוג admin תהיה גישה לצפייה בספר הפקודות הפנימי של המע', אולם הוא אינו בעל חשבון ואינו יכול להיות סוחר פעיל במע', לקנות ולמכור מניות.

1. בתרגיל זה כל משתמש מתחבר למע' מ browser שלו, והאינטראקציה שלו עם המשתתפים האחרים מתבצעת למעשה אך ורק ע"י השרת המתווך בינהם.

פורמלית, יש לתמוך בחווית משחק תקינה מעל chrome בלבד.

גם אם פיתחתם מעל FF או IE (טפו..) – וודאו כי אתם עובדים בצורה מוצלחת מעל chrome.

האפליקציה תיבדק רק מעל chrome.

בקיצור chrome,

קאפיש ?

1. ככלל, כל פרטי המידע במע' צריכים להתעדכן בצורה אוטומטית עבור כל משתמש (=== כל browser) בצורה של Pull כפי שלמדנו בשיעור והודגם בדוגמא המסכמת.
2. אין צורך לתמוך ב back כאמצעי למעבר בין המסכים השונים. כמו כן הניחו כי לא נתעלל באפליקציה ולא נשנה את שורת הכתובת, לא נמחק cookies וכו'. הבדיקה תתבצע אך ורק ע"י שימוש בפונקציונליות שתאפשרו במסכי המשחק השונים (משל היינו הוריכם...)
3. אין צורך לבצע שמירה של נתונים מעבר למופע הנוכחי של השרת. במילים אחרות כשהשרת יורד - כל הנתונים נעלמים.

לא שומרים את המשתמשים שנרשמו, את המניות, הסטוריית עסקאות וכו'

1. יש לדאוג ולוודא כי מסכי המערכת לסוגם מתנהגים יפה ב resize ולא מחסירים שום פרט.

הם יכולים להיות מכוערים, אבל שישארו פונקציונלים (כשם שהיה נהוג בתרגיל 2).

1. בתרגיל זה ישנם 3 מסכים עיקריים:
   1. **מסך רישום למע' (sign up)**

במסך זה יש לאפשר למשתמש להיכנס למע'.

לכל משתמש יש שם ייחודי לו ורק לו.

לכל משתמש יש תפקיד במע': סוחר/מנהל מע' (admin). יש לבחור למשתמש בדיוק תפקיד אחד.

אחרי רישום מוצלח תעברו למסך מס' 2...

* 1. **מסך ריכוז משתמשים ומניות**

במסך זה יוכלו המשתמשים השונים לראות מי הם המשתמשים האחרים במע' המחוברים כרגע, כמו גם את סך כל המניות המנוהלות במע'. המידע הבא מוצג עבור שני סוגי המשתמשים:

יש להציג מידע על שמות כל המשתמשים הפעילים כרגע במע'.

עבור כל משתמש:

* + 1. שם המשתמש
    2. תפקיד ("סוחר" או "admin")

יש להציג מידע על כלל המניות שהמע' מכירה ומנהלת. עבור כל מנייה:

* 1. שם החברה
  2. SYMBOL
  3. שער נוכחי
  4. מחזור מסחר מצטבר

רשימות אלה (של המשתמשים והמניות) מתעדכנות ב live ומציגות את מצב המע' (המשתמשים/מניות) כפי שהוא בכל רגע נתון (יש לעבוד בשיטות שלמדנו של Pull. המלצתי: all)

בהתאם לתפקיד של המשתמש (כפי שנקבע בעמוד הלוגין) ייתכנו מספר אפשרויות שונות לפעולה בעמוד זה

**עבור משתמש מסוג סוחר:**

משתמש מסוג של "סוחר" (ורק הוא) יכול לבחור קובץ ממע' הקבצים ולהעלות אותו לשרת.

חובה לעבוד עם FileChooser Dialog, כפי שהוצג בכיתה.

קבצי ה XML יהיו מוגדרים על פי סכמה חדשה של תרגיל 3   
(ראו [נספח XML](#_סכמת_קובץ_XML_3) כדי להבין מה התוספות/הבדלים יחסית לסכמה של תרגיל 2)

קובץ ה XML לא יכיל הגדרת לקוחות (כפי שהיה בתרגיל 2) אלא רק הגדרה של מניות ואחזקות בהן - של המשתמש שהעלה את המידע.

משתמש יכול להעלות יותר מקובץ אחד. במידה ואותה המנייה (על פי ה SYMBOL) קיימת כבר במע' וגם בקובץ שמועלה כרגע לשרת – נשארים עם הגדרת המנייה כפי שהיא במע' עצמה, כלומר עם השער הידוע שלה, ולא לוקחים את השער המפורט בקובץ

בהגיעו לשרת, הקובץ יעבור בדיקת תקינות (כל הבדיקות הרלבנטיות, כפי שהוגדרו בתרגיל 1 ו 2).

בנוסף יש לוודא כי כל אחזקות המניות של המשתמש – אכן מוגדרות בקובץ עצמו (כלומר משתמש לא יכול להעיד על כך שהוא הבעלים של מנייה X מבלי שזו מתוארת בקובץ ה XML שהוא עצמו מעלה.

במידה ונמצא תקין הודעה מתאימה תוחזר למשתמש ומעכשיו כל המשתמשים מכל הסוגים במע' יכולים לראות את נתוני המניות שהתווספו מהקובץ הזה ומופיעים בטבלת המניות שהמע' מנהלת.

במידה ונמצא תקול, הודעה מתאימה תוחזר למשתמש שהעלה אותו בלבד והקובץ (והמידע שבו) כמובן לא נכנס למע'.

ניהול חשבון

משתמש מסוג "סוחר" יכול לנהל חשבון כספי ולבצע העברת כספים ("בכאילו") בינו ובין לקוחות אחרים בהתאם למניות שקנה/מכר.

המשתמש יראה בעמוד זה את יתרת הזכות שיש לו בחשבון.   
אין צורך לתמוך או לבדוק מצב של כניסה ליתרה שלילית.

ברגע שמתבצעת עסקה, הכספים מועברים מהקונה למוכר מיידית וניתן לראות זאת בפרטי התנועות בחשבון של שני הצדדים.

יש להציג כפתור ייעודי שלחיצה עליו תאפשר למשתמש הטענה של חשבונו הפרטי בסכום כסף. יש לבקש סכום זה מהמשתמש כמובן ולקחת את חותמת הזמן הנוכחית (time stamp) כמועד שבו מתבצעת הפעולה

יש לאפשר למשתמש לראות את כלל התנועות בחשבון מיום הקמת המע'.

יש להציג את הפרטים הבאים עבור כל סוג פעולה (תנועה) בחשבון:

* + 1. סוג: הטענה, קניית מנייה, מכירת מנייה (במקרים של מנייה – לציין את ה SYMBOL שלה)
    2. תאריך ביצוע הפועלה.
    3. סכום הפעולה (חיובי למכירה והטענה ; שלילי לקנייה)
    4. יתרה לפני ביצוע הפעולה
    5. יתרה אחרי ביצוע הפעולה

הנפקת חברה (מנייה חדשה)

משתמש מסוג "סוחר" יכול לבצע הנפקה של חברה חדשה (תהליך יצירת מנייה חדשה במע')

בלחיצה על כפתור ייעודי יש לבקש מהמשתמש את הפרטים הבאים:

* 1. שם החברה
  2. SYMBOL המנייה
  3. כמות הנפקה
  4. שווי חברה מוערך

יש לבדוק כי אין כאן נסיון להנפיק עוד מניות לחברה קיימת (כלומר שם החברה וה SYMBOL ייחודיים). במידה וזה המצב – יש לשלוח הודעה מתאימה למשתמש ולבטל את הפעולה.

אחרי אישור הפעולה, תיווצר במע' מנייה חדשה. המשתמש שיצר אותה הוא הבעלים של כל כמות המניות שהונפקה.   
שער המנייה הוא תוצאת החלוקה של שווי החברה המוערך בכמות מניות ההנפקה.   
אין צורך להתייחס לנקודה עשרונית במקרה וכזו צצה (קחו רק את החלק השלם).

מרגע זה המנייה קיימת במע' וכל המשתמשים יכולים לראות אותה ולהתחיל לסחור בה כרגיל, כאילו הייתה מניה רגילה שנטענה מקובץ.

**עבור משתמש מסוג admin:**

משתמש מסוג admin לא יכול להעלות קבצי אחזקות מניות.   
כמו כן אין לו "חשבון" במע' (הוא לא יכול להגיש פקודות מסחר).

2 סוגי המשתמשים (סוחר ו admin) יכולים ללחוץ על שורת מנייה אחת ולעבור למסך מס' 3 שבו ניתן לראות מידע מפורט על המנייה הספציפית...

* 1. **מסך פעולות על מנייה ספיציפית**

במסך זה 2 סוגי המשתמשים יוכלו לראות פרטים מקיפים על מנייה נבחרת ולבצע בגינה פעולות.

עבור המנייה הנבחרת יש להציג את הפרטים הבאים:

1. שם המנייה
2. שם החברה
3. שער נוכחי
4. מחזור מסחר נוכחי
5. רשימת עסקאות שבוצעו במניה. עבור כל עסקה יש להציג
   1. תאריך
   2. כמות
   3. שער

הרשימה תהיה ממוינת לפי תאריך ביצוע העסקה. הקדום ביותר למטה ; העדכני ביותר למעלה.

יש להציג כפתור ייעודי המאפשר למשתמש לחזור חזרה לעמוד מס' 2 (אין להסתמך על back של ה browser)

בהתאם לתפקיד של המשתמש (כפי שנקבע בעמוד הלוגין) ייתכנו מספר אפשרויות שונות לפעולה בעמוד זה.

**עבור משתמש מסוג "סוחר":**

משתמש מסוג סוחר יוכל גם לראות את כמות אחזקותיו במנייה הנ"ל. (0 אם הוא לא מחזיק אותה כלל)

בנוסף המשתמש יוכל להגיש מעמוד זה פק' מכירה או קנייה למנייה:

יש להציג למשתמש כפתורים ייעודיים ולאפשר לו למלא את הפרטים הנדרשים לקנייה/מכירה של המנייה (כפי שהיה בתרגיל 2). עם הגשת הפקודה למע' היא תתבצע בהתאם להתנהלות בתרגיל 2 (מתבצעת כולה, חלקה או בכלל לא).

בתרגיל זה תתמכו ב 2 פק' מסחר חדשות:

1. פקודת FOK – FILL OR KILL – פקודת מגבלת שער זו (כמו LMT) מתבצעת **במלואה**, מיידית ברגע הגשתה למע' – ואם לא – היא נזנחת לחלוטין ולא נרשמת כממתינה בספרי הפקודות (ראו נספח ג' לפירוט על אופן פעולת הפקודה).

במידה ולא בוצעה בכלל – יש להקפיץ התראה מתאימה למשתמש שהגיש אותה.

1. פקודת IOC – IMMIDIATE OR CANCEL – פקודת מגבלת שער זו (כמו LMT) מתבצעת במלואה או בחלקה מיידית ברגע הגשתה למע'. כל שאריות שלה שלא התבצעו (או כולה) – נזנחות ולא נכנסות כממתינות בספרי הפקודות (ראו נספח ד' לפירוט על אופן פעולת הפקודה).

במידה ולא בוצעה בכלל – יש להקפיץ התראה מתאימה למשתמש שהגיש אותה.

התראות

כל משתמש שהגיש פקודת מכירה או קנייה יקבל התראה כאשר הפקודה מתבצעת (כולה או חלקה).   
על ההתראה לכלול מידע טקסטואלי שמסביר בצורה מלאה את מהות הפעולה שבוצעה (איזו מניה נמכרה/נקנתה, באיזה שער, באיזו כמות, האם נשארו שאריות וכו')

את ההתראות יש לקבל בעמוד מס' 2 כמו גם בעמוד מס' 3 (גם אם מסתכלים על מנייה אחרת מסוג ההתראה המתקבלת).

**עבור משתמש מסוג "admin":**

משתמש מסוג admin לא יכול להגיש בקשות קנייה/מכירה או לקבל התראות.

בנוסף למידע הכללי לעיל הוא יכול גם לראות את המידע מספר הפקודות הפתוחות (קנייה ומכירה).

המידע שה admin רואה מתעדכן גם הוא בצורה אוטומטית כדי לשקף את אמינות הנתונים המתרחשת בנבכי שרת המע'.

כך שאפשר שמשתמש אחד יהיה admin ויציץ בעמוד של מנייה מסויימת, בזמן שמשתמשים אחרים מגישים פקודות קנייה/מכירה וברגע המפגש בין הפקודות ניתן לראות את ההתנהלות בנבכי המנוע דרך מסך ה admin כשהיא מעודכנת ב runtime באמצעים שנלמדו בכיתה (delta fetching או all the information all the time)

## חלוקה למודולים

יש לייצר מודול נוסף שממנו ייבנה war המכיל את כל ה jar'ים הנדרשים (Jar של מנוע המע' ; gson.jar ; אולי אחרים ?).

מודול זה מכיל גם את קוד ה UI (HTML, CSS, JS) וגם את קוד הסרבלטים המקשרים למודול המנוע.

## איך מתחילים ?

אני ממליץ להתחיל במעבר יסודי על הדוגמא המסכמת של הקורס.   
היא תקנה לכם שליטה בסיסית ברזי אפליקצית ווב (בסיסית) ובתפקידים השונים של כל רכיב ושחקן בסיפור.

לאחר מכן התחילו בעבודה על פי סדר העמודים:

ראשית התחילו בעמוד הלוגין. הוא יכול להיות זהה בעיקרו לעמוד הלוגין כפי שהוצג בדוגמא המסכמת (הרגישו חופשי "להעתיק" משם את המידע הנדרש לכם. מתנה ממני אליכם. כזה אני. גדוש בטוב.)

תו"כ כתיבת עמוד זה תתרגלו את שלב ה UI ל שרת וחזרה ל UI. זה גם ידרוש מכם להקים בצד השרת אובייקט כזה או אחר שמנהל את המשתמשים וכן הלאה.

לאחר מכן התחילו לעבוד על עמוד מס' 2. ביחרו פיצ'ר נוסף וממשו אותו בצורה מלאה מה-UI, עבור לסרבלט המתאים בשרת וחזרה ל UI, בו אתם מייצרים את ה HTML בהתאם לתשובה שחזרה מהשרת.

ממליץ מאוד (כמו ב JAVA FX) לייצר עמוד HTML סטטי שמכיל "סקיצה" ראשונית של המידעים שאתם רוצים להציג כך שתוכלו להבין איך הוא אמור להיראות, מי הרכיבים שבו, איך לעצב אותו מבחינת CSS וכן הלאה. זה גם ייתן לכם הבנה ראשונית של מהם כל קריאות ה HTTP שאתם עתידים לייצר מכל רכיב ורכיב, באיזו תדירות הם יוצאים ? איזה סרבלטים משרתים אותם (האם סרבלט אחד לכולם או כמה "סרבלטונים" קטנים ?) וכן הלאה. אחרי שיש לכם סקיצה – תוכלו ב JS להתחיל לייצר את התוכן ה HTML של כל אזור ואזור ולעבוד על רכיב ורכיב "לעומק" (UI -> SERVER -> UI).

כך מסיימים את עמוד 2 ואז עוברים באותו אופן לעמוד 3. אחרי שכל התשתית והשלד הראשי של האפליקציה בנוי – מוסיפים את כל הכפתורים השונים ומפתחים כל פיצ'ר עד סופו.

## בונוסים

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| # | סוג | מהות | למה שווה לי ? | כמה שווה לי ? |
| 1 | הגדלת ראש מגניבה ! | Chat:  אפשרו למשתמשים לקיים chat בינהם תו"כ התחברותם למע'.  ה chat צריך להתנהל כולו אל מול השרת כמובן בצורה שבה כולם רואים את מה שכולם כותבים. (מאוווווווווווד דומה לדוגמא הסופית)  הצ'ט מתבצע בין המשתמשים הרשומים והמחוברים למערכת בלבד. | היש מדהים מזה ??? | 10 נקודות  (ועד ל 100) |
| 2 | הגדלת ראש בלתי ניתנת להכלה במגניבותה ! | בעמוד מס' 3 הציגו גרף המתאר את שינוי המנייה לאורך זמן (כשם שהיה בבונוס של תרגיל 2). חיקרו ועיבדו אם איזו ספריית JS שתמצאו לנכון (או בצורה native). | כי בכל זאת מע' מסחר במניות... מה לא יהיו גרפים ?? | **10 נקודות**  **(מעל ל 100)** |
| 3 | הגדלת ראש סופר דופר ! | ממשו את שכבת ה UI של תרגיל 3 כחלק מאפליקציית Java FX.  אפשר ורצוי כמובן להסתמך על האפליקציה הקיימת שכבר מימשתם בתרגיל 2.  בבונוס זה יש לאפשר למשתמש להירשם למע', לאחר מכן לייצר עבורו עמוד (java fx) המקביל לעמוד 2 שבו יוכל לראות את פרטי המניות השונות, כמו גם את פרטי החשבון שלו, כפי שמתואר בעמוד מס' 2 (אפשר לוותר על הצגת רשימת המשתמשים במע'). משם יוכל לבחור את אחת המניות ולבצע בה מסחר כשם שכבר היה בתרגיל 2 !  **שימו לב**:  יש כמובן לתמוך בקבלת התראות מתאימות כפי שמוגדרות בתרגיל 3.  עליכם לבחור ולהשתמש כאן ב HTTP Client שיפעל מול אותו השרת, כפי שקורה בתרגיל 3 (למעשה בונוס זה ייבדק כך שחלק מהמשתמשים הם ב browser וחלקם ב java fx.) | [Oh My God !!!](https://www.youtube.com/watch?v=iMs9feeSknk) | **20 נקודות (מעל ל 100)** |

## סבבה, סיימתי. מה ואיך להגיש ?

יש להגיש קובץ zip המכיל:

1. WAR אחד בדיוק שיושם בספריית tomcat ויעבור deployment אוטומטי. על WAR זה לכלול את כל התלויות שלכם,

אין להניח כי יסופקו לכם תלויות מבחוץ (למשל Gson וכו').

1. קבצי קוד המקור של הפרויקט שלכם (גם צד השרת וגם צד הלקוח).
2. קובץ ההגשה יכיל גם קובץ readme שיכיל הסבר על המערכת, בחירותיכם השונות במקומות שבהם היו לכם בחירה, כמו גם כל דבר נוסף העולה על דעתכם שחשוב שהבודקת תדע.

**בפרט חייב להכיל קובץ ה readme את ה url שאיתו מנווטים אל מסך ה login של המשחק שלכם.**

1. יש לכלול בקובץ ה **readme** גם תיעוד והסבר כללי (וממצה) של המחלקות העיקריות החדשות ותפקידם. (חישבו מה יסייע לבודקת להיכנס ביתר קלות לקוד שלכם ולהבין מי נגד מי...)
2. במידה ומימשתם את מי מהבונוסים לעיל, ציינו זאת בקובץ ה readme כדי שהבודקת תדע לבדוק בהתאם.

בונוס שימומש אבל לא יתועד – לא ייבדק !

הגשה באיחור, שאינה באישור, תבטל כל מימוש בונוס. אין להגיש באיחור בשביל להספיק לעשות

בונוסים. תכננו את הזמן בהתאם.

## שאלות ותשובות

|  |  |  |  |
| --- | --- | --- | --- |
| # | תאריך | שאלה | תשובה |
| 1 |  |  |  |

נספח א' – תיאור תרחיש קנייה ומכירה לפקודת LMT

נספח זה יציג התנהלות טיפוסית של מנוע המסחר במקרים השונים שבהם מגיעות פקודות המסחר השונות.

כל המוצג בהמשך מתייחס לפקודות הניתנות ונשמרות בהקשר של מנייה אחת )תהליך דומה קורה עבור כל מנייה ומנייה בתוך מנוע המסחר(.

מטעמי פשטות, התאריך מובא כאן כמספר רץ. במציאות הוא חותמת זמן של יצירת הפקודה/סגירת העסקה.

הרשימות מסודרות לפי עדיפויות על פי הוראות התרגיל בכל רגע נתון. העדיפות הגבוהה ביותר היא העליונה ביותר והולך ופוחת.

ניתן לראות 3 רשימות: פקודות הממתינות למכירה ; פקודות הממתינות לקנייה ; עסקאות שבוצעו. כמו כן בכל רגע נתון ניתן לראות מהו השער של המנייה כפי שנגזר מהעסקאות שבוצעו

עבור מנייה X המתחילה בשער 0, בראשית המע' כל רשימות הפקודות והעסקאות ריקות לחלוטין:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 0 |  |  |  |  |  |  |  |  |  |  |  |

1. הגשת פקודת ראשונה:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 100 | 20 | 1 | LMT | מכירה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 0 |  |  |  |  |  |  |  | 100 | 20 | 1 | 1 |

הסבר:

את פקודת המכירה צריך להפגיש עם כל פקודות הקנייה. אין כרגע פקודות קנייה ולכן היא נכנסת להמתנה ברשימת פקודות המכירה

1. הגשת פקודה שנייה:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 110 | 10 | 2 | LMT | מכירה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 0 |  |  |  |  |  |  |  | 100 | 20 | 1 | 1 |
|  |  |  |  |  |  |  | 110 | 10 | 2 | 2 |

הסבר:

פקודת המכירה צריכה לפגוש פקודות קנייה ממתינות. אין כאלה, ועל כן היא נכנסת לרשימת הפקודות הממתינות.

עדיפות פקודת מכירה היא לפי השער הנמוך ביותר. ועל כן הפקודה הנוכחית בשער 110 תיכנס אחרי הפקודה הקיימת בשער 100

1. הגשת פקודה שלישית:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 90 | 10 | 3 | LMT | מכירה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 0 |  |  |  |  |  |  |  | 90 | 10 | 3 | 1 |
|  |  |  |  |  |  |  | 100 | 20 | 1 | 2 |
|  |  |  |  |  |  |  | 110 | 10 | 2 | 3 |

הסבר:

פקודת המכירה צריכה לפגוש פקודות קנייה ממתינות. אין כאלה, היא נכנסת לרשימת הפקודות הממתינות. עדיפות פקודות המכירה היא לפי השער הנמוך ביותר ולכן פקודה זו מדלגת לראש התור (גם אם כרונולוגית קרתה אחרונה)

1. הגשת פקודה רביעית:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 90 | 10 | 4 | LMT | קנייה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 90 | 900 | 90 | 10 | 4 |  |  |  | 100 | 20 | 1 | 1 |
|  |  |  |  |  |  |  | 110 | 10 | 2 | 2 |

הסבר:

פקודת הקנייה מופגשת מול כל פקודות המכירה הממתינות, על פי סדר עדיפויותן.

נמצאה פקודת מכירה במחיר שהוא נמוך או שווה לפקודת הקנייה ולכן התבצעה עסקה. כל כמות הקנייה מוצתה אל מול פקודת המכירה הרלבנטית ולכן כל פקודות המחירה נמחקת מהרשימה. כמו כן פקודת הקנייה מוצתה לחלוטין ולכן לא נותר מה להכניס לרשימות הקנייה.

כתוצאה מכך נוצרה עסקה ראשונה במנייה. העסקה קובעת את מחיר המנייה לגובה העסקה האחרונה (90)

1. הגשת פקודה חמישית:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 120 | 25 | 5 | LMT | קנייה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 110 | 900 | 90 | 10 | 4 |  |  |  | 110 | 5 | 2 | 1 |
| 2000 | 100 | 20 | 5 |  |  |  |  |  |  |  |
| 550 | 110 | 5 | 5 |  |  |  |  |  |  |  |

הסבר:

פקודת הקנייה מופגשת מול כל פקודות המכירה הממתינות על פי סדר עדיפויותן.

כל עוד המחיר בקנייה גבוה או שווה למחיר בפקודת המכירה, ונותרה עוד כמות בפקודת הקנייה – גומעים את פקודות המכירה הרלבנטיות.

פקודת המכירה הראשונה שהמתינה (מתאריך 1) גמעה 20 מניות ובעקבותיה בוצעה עסקה בגובה 100.

בעקבות כך נותרו עוד 5 מניות לביצוע.

פקודת המכירה השנייה שהמתינה (מתאריך 2) גמעה עוד 5 מניות ובעקבותיה בוצעה עסקה בגובה 110.

מפקודת המכירה השנייה נשארו לביצוע עוד 5 מניות בלבד. כל פקודת הקנייה מוצתה עד תום.

שער המנייה נקבע לפי גובה העסקה האחרונה שבוצעה (110)

1. הגשת פקודה שישית:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 90 | 30 | 6 | LMT | קנייה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 110 | 900 | 90 | 10 | 4 | 90 | 30 | 6 | 110 | 5 | 2 | 1 |
| 2000 | 100 | 20 | 5 |  |  |  |  |  |  |  |
| 550 | 110 | 5 | 5 |  |  |  |  |  |  |  |

הסבר:

פקודת הקנייה מופגשת עם פקודות המכירה הממתינות. אין שום פקודה שמחירה נמוך או שווה למחיר בפקודת הקנייה ולכן היא נכנסת לרשימת ההמתנה של הפקודות לקנייה

1. הגשת פקודה שביעית:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 80 | 20 | 7 | LMT | קנייה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 110 | 900 | 90 | 10 | 4 | 90 | 30 | 6 | 110 | 5 | 2 | 1 |
| 2000 | 100 | 20 | 5 | 80 | 20 | 7 |  |  |  | 2 |
| 550 | 110 | 5 | 5 |  |  |  |  |  |  |  |

הסבר:

פקודת הקנייה מופגשת עם פקודות המכירה הממתינות. אין שום פקודה שמחירה נמוך או שווה למחיר בפקודת הקנייה ולכן היא נכנסת לרשימת ההמתנה של הפקודות לקנייה. עדיפות פקודות הקנייה היא לפקודות במחיר הגבוה ביותר ולכן היא נכנסת לעדיפות נמוכה יותר יחסית לפקודה הקיימת.

1. הגשת פקודה שמינית:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 100 | 10 | 8 | LMT | קנייה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 110 | 900 | 90 | 10 | 4 | 100 | 10 | 8 | 110 | 5 | 2 | 1 |
| 2000 | 100 | 20 | 5 | 90 | 30 | 6 |  |  |  | 2 |
| 550 | 110 | 5 | 5 | 80 | 20 | 7 |  |  |  | 3 |

הסבר:

פקודת הקנייה מופגשת עם פקודות המכירה הממתינות. אין שום פקודה שמחירה נמוך או שווה למחיר בפקודת הקנייה ולכן היא נכנסת לרשימת ההמתנה של הפקודות לקנייה. עדיפות פקודות הקנייה היא לפקודות במחיר הגבוה ביותר ולכן היא נכנסת לעדיפות גבוהה יותר יחסית לפקודות הקיימות.

1. הגשת פקודה תשיעית:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 90 | 5 | 9 | LMT | קנייה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 110 | 900 | 90 | 10 | 4 | 100 | 10 | 8 | 110 | 5 | 2 | 1 |
| 2000 | 100 | 20 | 5 | 90 | 30 | 6 |  |  |  | 2 |
| 550 | 110 | 5 | 5 | 90 | 5 | 9 |  |  |  | 3 |
|  |  |  |  | 80 | 20 | 7 |  |  |  | 4 |

הסבר:

פקודת הקנייה מופגשת עם פקודות המכירה הממתינות. אין שום פקודה שמחירה נמוך או שווה למחיר בפקודת הקנייה ולכן היא נכנסת לרשימת ההמתנה של הפקודות לקנייה. היות ויש כבר פקודה במחיר שלה (מתאריך 6), אזי שכן העדיפות נקבעת על פי התאריך: הפקודה הקדומה יותר (6) נמצאת בעדיפות גבוהה יותר מהפקודה המאוחרת יותר (מתאריך 9)

1. הגשת פקודה עשירית:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 90 | 5 | 10 | LMT | מכירה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 100 | 900 | 90 | 10 | 4 | 100 | 5 | 8 | 110 | 5 | 2 | 1 |
| 2000 | 100 | 20 | 5 | 90 | 30 | 6 |  |  |  | 2 |
| 550 | 110 | 5 | 5 | 90 | 5 | 9 |  |  |  | 3 |
| 500 | 100 | 5 | 10 | 80 | 20 | 7 |  |  |  | 4 |

הסבר:

פקודת המכירה מופגשת עם פקודות הקנייה הממתינות. הפקודות תפגשנה כל עוד מחיר הקנייה גבוה או שווה למחיר בפקודת המכירה. פקודות הקנייה הראשונה גומעת את כל הכמות של פקודת המכירה ולכן מייצרת עסקה בגובה המדובר. מפקודת הקנייה נותרה עוד שארית לביצוע. היא נותרת בעדיפות המקורית שלה

1. הגשת פקודה אחת עשר:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 95 | 25 | 11 | LMT | מכירה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 100 | 900 | 90 | 10 | 4 | 90 | 30 | 6 | 95 | 20 | 11 | 1 |
| 2000 | 100 | 20 | 5 | 90 | 5 | 9 | 110 | 5 | 2 | 2 |
| 550 | 110 | 5 | 5 | 80 | 20 | 7 |  |  |  | 3 |
| 500 | 100 | 5 | 10 |  |  |  |  |  |  | 4 |
| 500 | 100 | 5 | 11 |  |  |  |  |  |  | 5 |

הסבר:

פקודת המכירה מופגשת עם פקודות הקנייה הממתינות. הפקודות תפגשנה כל עוד מחיר הקנייה גבוה או שווה למחיר בפקודת המכירה. פקודות הקנייה הראשונה (תאריך 8) גומעת רק 5 מתוך 25 המניות שיש למכור. עבור שאר 20 המניות אין קונה (הפקודות הבאות בתור במחיר נמוך יותר ממחיר המכירה) ולכן שארית הפקודה נכנס להמתנה. בפקודות מכירה העדיפות היא למחיר הנמוך ביותר ולכן הפקודה נכנסת לעדיפות ראשונה

1. הגשת פקודה שתים עשרה:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 90 | 20 | 12 | LMT | מכירה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 90 | 900 | 90 | 10 | 4 | 90 | 10 | 6 | 95 | 20 | 11 | 1 |
| 2000 | 100 | 20 | 5 | 90 | 5 | 9 | 110 | 5 | 2 | 2 |
| 550 | 110 | 5 | 5 | 80 | 20 | 7 |  |  |  | 3 |
| 500 | 100 | 5 | 10 |  |  |  |  |  |  |  |
| 500 | 100 | 5 | 11 |  |  |  |  |  |  |  |
| 1800 | 90 | 20 | 12 |  |  |  |  |  |  |  |

הסבר:

פקודת המכירה מופגשת עם פקודות הקנייה הממתינות. הפקודות תפגשנה כל עוד מחיר הקנייה גבוה או שווה למחיר בפקודת המכירה. פקודות הקנייה הראשונה מתאימה במחיר וגומעת את כל כמות המכירה הנדרשת ועדיין נשארת עם עוד 10 מניות לקנייה.

1. הגשת פקודה שלושה עשר:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 90 | 12 | 13 | LMT | מכירה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 90 | 900 | 90 | 10 | 4 | 90 | 3 | 9 | 95 | 20 | 11 | 1 |
| 2000 | 100 | 20 | 5 | 80 | 20 | 7 | 110 | 5 | 2 | 2 |
| 550 | 110 | 5 | 5 |  |  |  |  |  |  |  |
| 500 | 100 | 5 | 10 |  |  |  |  |  |  |  |
| 500 | 100 | 5 | 11 |  |  |  |  |  |  |  |
| 1800 | 90 | 20 | 12 |  |  |  |  |  |  |  |
| 900 | 90 | 10 | 13 |  |  |  |  |  |  |  |
| 180 | 90 | 2 | 13 |  |  |  |  |  |  |  |

הסבר:

פקודת המכירה מופגשת עם פקודות הקנייה הממתינות. הפקודות תפגשנה כל עוד מחיר הקנייה גבוה או שווה למחיר בפקודת המכירה. פקודות הקנייה הראשונה ממוצת כולה ומולידה עסקה של 10 מניות בגובה 90. מפקדות המכירה המקורית נותרו עוד 2 מניות למכירה, והן פוגשות את פקודת הקנייה הבאה בתור (תאריך 9) ומותירים בה רק עוד 3 מניות לקנייה. כתוצאה מכך מתבצעת עוד עסקה במחיר 90 עבור עוד 2 מניות. שימו לב מדובר ב 2 עסקאות נפרדות והן צריכות להירשם ולהופיע ככאלה.

1. הגשת פקודה ארבע עשר:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 85 | 13 | 14 | LMT | מכירה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 90 | 900 | 90 | 10 | 4 | 80 | 20 | 7 | 85 | 10 | 14 | 1 |
| 2000 | 100 | 20 | 5 |  |  |  | 95 | 20 | 11 | 2 |
| 550 | 110 | 5 | 5 |  |  |  | 110 | 5 | 2 | 3 |
| 500 | 100 | 5 | 10 |  |  |  |  |  |  |  |
| 500 | 100 | 5 | 11 |  |  |  |  |  |  |  |
| 1800 | 90 | 20 | 12 |  |  |  |  |  |  |  |
| 900 | 90 | 10 | 13 |  |  |  |  |  |  |  |
| 180 | 90 | 2 | 13 |  |  |  |  |  |  |  |
| 270 | 90 | 3 | 14 |  |  |  |  |  |  |  |

הסבר:

פקודת המכירה מופגשת עם פקודות הקנייה הממתינות. הפקודות תפגשנה כל עוד מחיר הקנייה גבוה או שווה למחיר בפקודת המכירה. פקודות הקנייה הראשונה ממוצת כולה ומולידה עסקה של 3 מניות בגובה 90. פקודת המכירה הבאה כבר לא מתאימה מבחינת מחיר (80 מול 85). לכן שארית פקודת המכירה (10 מניות במחיר 85) נכנסות לרשימת ההמתנה של פקודות מכירה. עדיפות בפקודות מכירה היא למחיר הנמוך ביותר ולכן היא ממוקמת בראש הרשימה.

נספח ב' – תיאור תרחיש קנייה ומכירה לפקודת MKT

נספח זה יציג התנהלות טיפוסית של מנוע המסחר במקרים השונים שבהם מגיעות פקודות המסחר השונות (קנייה/מכירה) אך מסוג של פקודות MKT בלבד.

כל המוצג בהמשך מתייחס לפקודות הניתנות ונשמרות בהקשר של מנייה אחת.   
תהליך דומה קורה עבור כל מנייה ומנייה בתוך מנוע המסחר.

מטעמי פשטות, התאריך מובא כאן כמספר רץ. במציאות הוא חותמת זמן של יצירת הפקודה/סגירת העסקה.

הרשימות מסודרות לפי עדיפויות על פי הוראות התרגיל בכל רגע נתון. העדיפות הגבוהה ביותר היא העליונה ביותר והולך ופוחת.

התרחיש יתחיל מסופו של התרחיש המוצג בנספח א':

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 90 | 270 | 90 | 3 | 14 | 80 | 20 | 7 | 85 | 10 | 14 | 1 |
|  |  |  |  |  |  |  | 95 | 20 | 11 | 2 |
|  |  |  |  |  |  |  | 110 | 5 | 2 | 3 |

1. הגשת פקודה ראשונה:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| N/A | 15 | 15 | MKT | מכירה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 80 | 270 | 90 | 3 | 14 | 80 | 5 | 7 | 85 | 10 | 14 | 1 |
| 1200 | 80 | 15 | 15 |  |  |  | 95 | 20 | 11 | 2 |
|  |  |  |  |  |  |  | 110 | 5 | 2 | 3 |

הסבר:

פקודת המכירה MKT מופגשת עם פקודות הקנייה הנגדיות, והיא גומעת את הפקודות ככל שהיא יכולה מבחינת הכמות (לא אכפת לה מהמחיר – היא פועלת במחיר השוק). פקודת הקנייה הראשונה מספקת את כל פקודת המכירה ואף עוד נותרה שארית לפקודת הקנייה. העסקה מתבצעת בשער המוצע של פקודות הקנייה (== מחיר השוק) ומולידה עסקה של 15 מניות במחיר 80.

1. הגשת פקודה שנייה:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| N/A | 15 | 16 | MKT | מכירה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 80 | 270 | 90 | 3 | 14 |  |  |  | 80 | 10 | 16 | 1 |
| 1200 | 80 | 15 | 15 |  |  |  | 85 | 10 | 14 | 2 |
| 400 | 80 | 5 | 16 |  |  |  | 95 | 20 | 11 | 3 |
|  |  |  |  |  |  |  | 110 | 5 | 2 | 4 |

הסבר:

פקודת המכירה מופגשת עם פקודות הקנייה הנגדיות, והיא גומעת את כל פקודת הקנייה שנותרת ומייצרת עסקה של 5 מניות בגובה 80. נותרה עוד שארית לפקודת המכירה ולכן היא נכנסת לרשימת הפקודות הממתינות לביצוע. כשפקודת מרקט נכנסת לרשימות ההמתנה מתייחסים אליה כאל פקודת LMT כאשר מגבלת השער היא שער העסקה האחרונה שבוצעה. על פי שער זה (80) – היא מתמקמת בראש סדר העדיפויות של פקודות המכירה

1. הגשת פקודת שלישית:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| N/A | 20 | 17 | MKT | קניה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 85 | 270 | 90 | 3 | 14 |  |  |  | 95 | 20 | 11 | 1 |
| 1200 | 80 | 15 | 15 |  |  |  | 110 | 5 | 2 | 2 |
| 400 | 80 | 5 | 16 |  |  |  |  |  |  |  |
| 800 | 80 | 10 | 17 |  |  |  |  |  |  |  |
| 850 | 85 | 10 | 17 |  |  |  |  |  |  |  |

הסבר:

פקודת הקנייה מופגשת עם פקודות המכירה הממתינות לה. זו פקודת MKT כלומר השאיפה היא לקנות במחיר השוק. פק' המכירה הראשונה ממוצת כולה ומולידה עסקה של 10 מניות במחיר 85. נשארו עוד 10 מניות לקנייה מהפקודה המקורית, והן פוגשות את פקודת המכירה הבאה בתור והיא ממצה את פקודת הקנייה לחלוטין ומולידה עסקה נוספת הפעם של 10 מניות במחיר 85.

1. הגשת פקודה רביעית:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| N/A | 30 | 18 | MKT | קניה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 110 | 270 | 90 | 3 | 14 | 110 | 5 | 18 |  |  |  | 1 |
| 1200 | 80 | 15 | 15 |  |  |  |  |  |  |  |
| 400 | 80 | 5 | 16 |  |  |  |  |  |  |  |
| 800 | 80 | 10 | 17 |  |  |  |  |  |  |  |
| 850 | 85 | 10 | 17 |  |  |  |  |  |  |  |
| 1900 | 95 | 20 | 18 |  |  |  |  |  |  |  |
| 550 | 110 | 5 | 18 |  |  |  |  |  |  |  |

הסבר:

פקודת הקנייה מופגשת עם פקודות המכירה הממתינות לה. זו פקודת MKT כלומר השאיפה היא לקנות במחיר השוק. פק' המכירה הראשונה ממוצת כולה ומולידה עסקה של 20 מניות במחיר 95. נשארו עוד 10 מניות לקנייה מהפקודה המקורית, והן פוגשות את פקודת המכירה הבאה בתור ומולידים עוד עסקה של 5 מניות במחיר של 110. אין יותר פקודות מכירה ממתינות ולכן שארית פקודת הקנייה נכנס לרשימת פקודות הקנייה הממתינות. מגבלת השער לפקודה שנקבע הוא שער העסקה האחרונה שבוצע (110)

נספח ג' – תיאור תרחיש קנייה ומכירה לפקודת FOK

נספח זה יציג התנהלות טיפוסית של מנוע המסחר במקרים השונים שבהם מגיעות פקודות המסחר השונות (קנייה/מכירה) אך מסוג של פקודות FOK בלבד.

כל המוצג בהמשך מתייחס לפקודות הניתנות ונשמרות בהקשר של מנייה אחת.   
תהליך דומה קורה עבור כל מנייה ומנייה בתוך מנוע המסחר.

מטעמי פשטות, התאריך מובא כאן כמספר רץ. במציאות הוא חותמת זמן של יצירת הפקודה/סגירת העסקה.

הרשימות מסודרות לפי עדיפויות על פי הוראות התרגיל בכל רגע נתון. העדיפות הגבוהה ביותר היא העליונה ביותר והולך ופוחת.

התרחיש יתחיל ממצב הפקודות הממתינות הבא:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 110 | 550 | 110 | 5 | 18 | 110 | 5 | 18 | 100 | 10 | 19 | 1 |
|  |  |  |  |  | 100 | 10 | 21 | 110 | 10 | 20 | 2 |

1. הגשת פקודה ראשונה:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 120 | 30 | 22 | FOK | קניה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 110 | 550 | 110 | 5 | 18 | 110 | 5 | 18 | 100 | 10 | 19 | 1 |
|  |  |  |  |  | 100 | 10 | 21 | 110 | 10 | 20 | 2 |

הסבר:

פקודת הקנייה מסוג FOK מופגשת עם פקודות המכירה הממתינות. באם ניתן לבצע את כל הפקודה בכללותה (על פי תנאי המחיר כמו בפקודת LMT) ולמצות את מלוא הכמות – היא מתבצעת. אם לא – היא נשכחת.

במקרה זה יש מפגש של מחיר – אולם הכמות הנדרשת לקנייה גדולה מסך הכמות המתאפשרת למכירה (פקודות מתאריך 19 ו 20) ועל כן כל הפקודה לא מבוצעת בכלל (!) ואף לא נכנסת לרשימת הפקודות להמתנה

1. הגשת פקודה שנייה:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 105 | 20 | 23 | FOK | קניה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 110 | 550 | 110 | 5 | 18 | 110 | 5 | 18 | 100 | 10 | 19 | 1 |
|  |  |  |  |  | 100 | 10 | 21 | 110 | 10 | 20 | 2 |

הסבר:

פקודת הקנייה מסוג FOK מופגשת עם פקודות המכירה הממתינות. אחרי מיצוי פקודת המכירה הראשונה (105 מול 100) יישאר לבצע עוד כמות של 10 מפקודות הקנייה המקורית. אולם לא ניתן להשלים את העסקה הנ"ל (בגלל בעיית מחיר 110 מול 105) ולכן לא ניתן לבצע את פקודת ה FOK מיידית ובמלואה. על כן אין שום עסקה (גם לא של הפקודה הראשונה) והפקודה לא נכנסת לרשימת פקודות הקנייה להמתנה.

1. הגשת פקודה שלישית:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 110 | 15 | 23 | FOK | קניה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 110 | 550 | 110 | 5 | 18 | 110 | 5 | 18 | 110 | 5 | 20 | 1 |
| 1000 | 100 | 10 | 23 | 100 | 10 | 21 |  |  |  | 2 |
| 550 | 110 | 5 | 23 |  |  |  |  |  |  |  |

הסבר:

פקודת הקנייה מסוג FOK מופגשת עם פקודות המכירה הממתינות. אחרי מיצוי פקודת המכירה הראשונה (105 מול 100) יישאר לבצע עוד כמות של 10 מפקודות הקנייה המקורית. הפקודה השנייה יכולה לספק את הצרכים הנ"ל (ואף נותרת ממנה שארית) ולכן אפשר לאמר כי (סוף סוף) כל פקודת ה FOK מוצתה לחלוטין, ולכן העסקאות שנגזרו ממנה מגיעות לכדי ביצוע.

1. הגשת פקודה רביעית:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 100 | 20 | 24 | FOK | מכירה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 110 | 550 | 110 | 5 | 18 | 110 | 5 | 18 | 110 | 5 | 20 | 1 |
| 1000 | 100 | 10 | 23 | 100 | 10 | 21 |  |  |  | 2 |
| 550 | 110 | 5 | 23 |  |  |  |  |  |  |  |

הסבר:

פקדות המכירה מסוג FOK מופגשת עם פקודות הקנייה הממתינות. אמנם היתן לקיים עסקאות מבחינת המחיר, אולם סך הכמות לקנייה אינה מספיקה כדי לספק את הפקודה במלואה (20 מול 15) ולכן אף עסקה לא מתבצעת.

1. הגשת פקודה חמישית:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 100 | 10 | 25 | FOK | מכירה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 100 | 550 | 110 | 5 | 18 | 100 | 5 | 21 | 110 | 5 | 20 | 1 |
| 1000 | 100 | 10 | 23 |  |  |  |  |  |  |  |
| 550 | 110 | 5 | 23 |  |  |  |  |  |  |  |
| 550 | 110 | 5 | 25 |  |  |  |  |  |  |  |
| 500 | 100 | 5 | 25 |  |  |  |  |  |  |  |

הסבר:

פקדות המכירה מסוג FOK מופגשת עם פקודות הקנייה הממתינות. פקודת הקנייה הראשונה מתבצעת ומותירה עוד כמות של 5 בפקודת המכירה המקורית (מולידה עסקה של 5 מניות במחיר 110). פקודת הקנייה הבאה בתור יכולה להתבצע גם כן ולמצות בכך את מלוא פקודת ה FOK, ואף נותר בה עוד כמות של 5 לביצוע.

נספח ד' – תיאור תרחיש קנייה ומכירה לפקודת IOC

נספח זה יציג התנהלות טיפוסית של מנוע המסחר במקרים השונים שבהם מגיעות פקודות המסחר השונות (קנייה/מכירה) אך מסוג של פקודות IOC בלבד.

כל המוצג בהמשך מתייחס לפקודות הניתנות ונשמרות בהקשר של מנייה אחת.   
תהליך דומה קורה עבור כל מנייה ומנייה בתוך מנוע המסחר.

מטעמי פשטות, התאריך מובא כאן כמספר רץ. במציאות הוא חותמת זמן של יצירת הפקודה/סגירת העסקה.

הרשימות מסודרות לפי עדיפויות על פי הוראות התרגיל בכל רגע נתון. העדיפות הגבוהה ביותר היא העליונה ביותר והולך ופוחת.

התרחיש יתחיל ממצב הפקודות הממתינות הבא:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 110 | 550 | 110 | 5 | 18 | 110 | 5 | 18 | 100 | 10 | 19 | 1 |
|  |  |  |  | 100 | 10 | 21 | 110 | 10 | 20 | 2 |

1. הגשת פקודה ראשונה:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 120 | 30 | 22 | FOK | קניה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 110 | 550 | 110 | 5 | 18 | 110 | 5 | 18 |  |  |  | 1 |
| 1000 | 100 | 10 | 22 | 100 | 10 | 21 |  |  |  | 2 |
| 1100 | 110 | 10 | 22 |  |  |  |  |  |  |  |

הסבר:

פקודת הקנייה מסוג IOC מופגשת עם פקודות המכירה הממתינות. באם ניתן לבצע את הפקודה, כולה או חלקה, (על פי תנאי המחיר כמו בפקודת LMT)– היא מתבצעת. אם נותרו ממנה שאריות – הן לא נכנסות לרשימות הפקודות הממתינות

במקרה זה יש מפגש של מחיר – ואחרי מיצוי כל פקודות המכירה הקיימות (תאריך 19 ו 20), עדיין נותר עוד 10 מניות בפקודות הקנייה המקורית.שארית פקודה זו לא נכנסת לרשימת פקודות הקנייה הממתינות.

1. הגשת פקודה שנייה:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 120 | 20 | 23 | FOK | מכירה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 110 | 550 | 110 | 5 | 18 | 110 | 5 | 18 |  |  |  | 1 |
| 1000 | 100 | 10 | 22 | 100 | 10 | 21 |  |  |  | 2 |
| 1100 | 110 | 10 | 22 |  |  |  |  |  |  |  |

הסבר:

פקדות המכירה מסוג IOC מופגשת עם פקודות הקנייה הממתינות. לא ניתן לבצע עסקה בגלל מגבלות המחיר. הפקודה בכללותה לא נכנסת לרשימת פקודות המכירה הממתינות.

1. הגשת פקודה שלישית:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **מחיר (שער)** | **כמות** | **תאריך** | **סוג** | **כיוון** |
| 100 | 10 | 24 | FOK | מכירה |

מצב רשימות הפקודות הפנימיות במנוע המסחר לאחר עיבוד הפקודה:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **עסקאות שבוצעו** | | | | **קנייה** | | | **מכירה** | | |  |
| **שער** | **מחזור** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **מחיר** | **כמות** | **תאריך** | **עדיפות** |
| 100 | 550 | 110 | 5 | 18 | 100 | 5 | 21 |  |  |  | 1 |
| 1000 | 100 | 10 | 22 |  |  |  |  |  |  |  |
| 1100 | 110 | 10 | 22 |  |  |  |  |  |  |  |
| 550 | 110 | 5 | 24 |  |  |  |  |  |  |  |
| 500 | 100 | 5 | 24 |  |  |  |  |  |  |  |

הסבר:

פקדות המכירה מסוג IOC מופגשת עם פקודות הקנייה הממתינות. פקודת הקנייה הראשונה מתבצעת ומותירה עוד כמות של 5 בפקודת המכירה המקורית (מולידה עסקה של 5 מניות במחיר 110). פקודת הקנייה הבאה בתור יכולה להתבצע גם כן ולמצות בכך את מלוא פקודת המכירה (ואף נותר בה עוד כמות של 5 לביצוע). היא מולידה עסקה של 5 מניות במחיר 100.

נספח ה' – תיאור מבנה המערכת באמצעות קובץ XML

תיאור מבנה המערכת מפורט כחלק מקובץ ה XML

במהלך הבדיקה (של שלל התרגילים), תיבדק המערכת באמצעות מספר קבצים שונים, חלקם חוקיים וחלקם תקולים (תקולים אפליקטיבית, תקינים סכמטית), במטרה לראות האם וכיצד המערכת מגיבה לשגיאות.

בחנו היטיב את קבצי הדוגמא שהועלו למע' ה Mama וודאו כי אתם מבינים את פרטיהם ומבניהם.

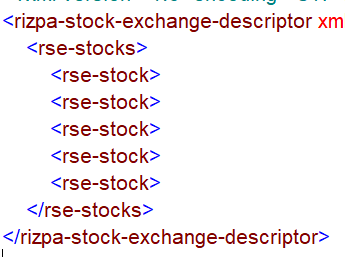
היכן שמצויין case insensitive הכוונה היא שאין חשיבות ל case של האותיות באנגלית. במקרה זה הערך milk זהה לערך MiLk

היכן שמצויין שהמחרוזת יכולה להכיל רווחים – המדובר הוא רק על רווחים בתוך המחרוזת ולא על רווחים בתחילתה/סופה

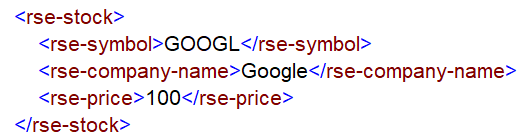
## סכמת קובץ XML – גרסה I

הקובץ מתאר מבנה של מניות סחירות במע':

|  |  |  |  |
| --- | --- | --- | --- |
| # | סוג | שם | מהות |
| 1 | Element | rizpa-stock-exchange-descriptor | מתאר את המבנה הכללי של מע' רצפה |
| 2 | Element | rse-stocks | מבנה המתאר את כל המניות המוגדרות במע'. יכול להכיל לפחות תיאור מנייה אחת ועד אינסוף. |



**rse-stock**



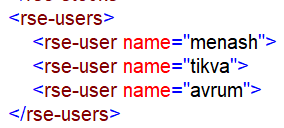
|  |  |  |  |
| --- | --- | --- | --- |
| # | סוג | שם | מהות |
| 1 | Element | rse-stock | מבנה המתאר מנייה אחת בלבד |
| 2 | Element | rse-symbol | מתאר את הסמל של המנייה. אותיות באנגלית בלבד, ללא רווחים, ב upper case. לא יכול להיות מחרוזת ריקה. רכיב חובה. |
| 3 | Element | rse-company-name | מתאר את שם החברה לה שייכת המנייה. מלל חופשי יכול להכיל רווחים. רכיב חובה |
| 4 | Element | rse-price | מתאר את מחיר המנייה בעת טעינת המידע. מספר שלם חיובי בלבד. |

## סכמת קובץ XML – גרסה II

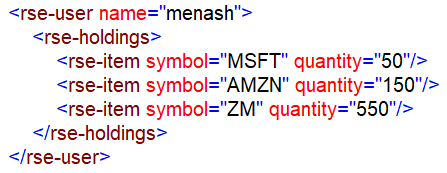
עדכונים מגרסה קודמת:

* 1. נוסף אלמנט חובה חדש (ראשי) המגדיר את הלקוחות הרשומים במע'
  2. עבור כל לקוח מתוארות אחזקות המניות שלו

|  |  |  |  |
| --- | --- | --- | --- |
| # | סוג | שם | מהות |
| 1 | Element | rse-users | מבנה המתאר את המשתמשים הרשומים במע' |



**rse-user**

****

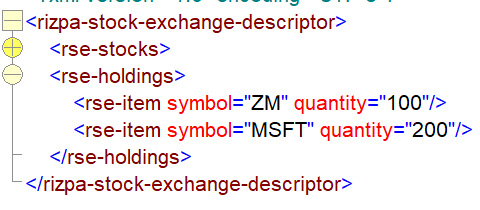
## 

|  |  |  |  |
| --- | --- | --- | --- |
| # | סוג | שם | מהות |
| 1 | Element | rse-user | מבנה המתאר משתמש אחד במע' |
| 2 | Attribute | name | מתאר את שם המשתמש. השם הוא ייחודי ומורכב ממלל חופשי באנגלית. יכול לכלול רווחים.  שדה חובה |
| 3 | Element | rse-holdings | מבנה המתאר את אחזקותיו של המשתמש הנ"ל |
| 4 | Element | rse-item | מבנה מתאר אחזקה של מנייה אחת ע"י המשתמש |
| 5 | Attribute | symbol | מתאר את סמל המנייה המוחזקת ע"י משתמש זה |
| 6 | Attrobute | Quantity | מתאר את הכמות המוחזקת עבור המנייה המדוברת אצל המשתמש המדובר. מספר חיובי שלם גדול מ 0. |

## סכמת קובץ XML – גרסה III

עדכונים מגרסה קודמת:

* + 1. אין הגדרת לקוחות.
    2. השדה rse-holdings המתואר כחלק מסכמה מס' 2 הופך להיות מבנה עצמאי (לא בהקשר של משתמש מסוים). הגדרותיו נותרות כשם שהיו בתרגיל 2

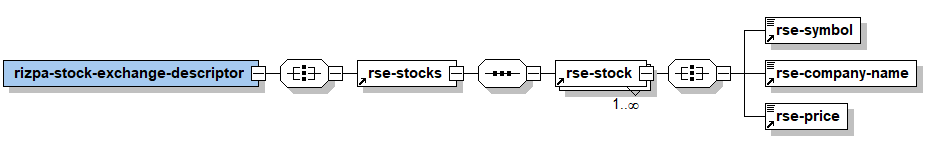
****

נספח ו' – קבצי סכמה לוידוא קובץ ה XML

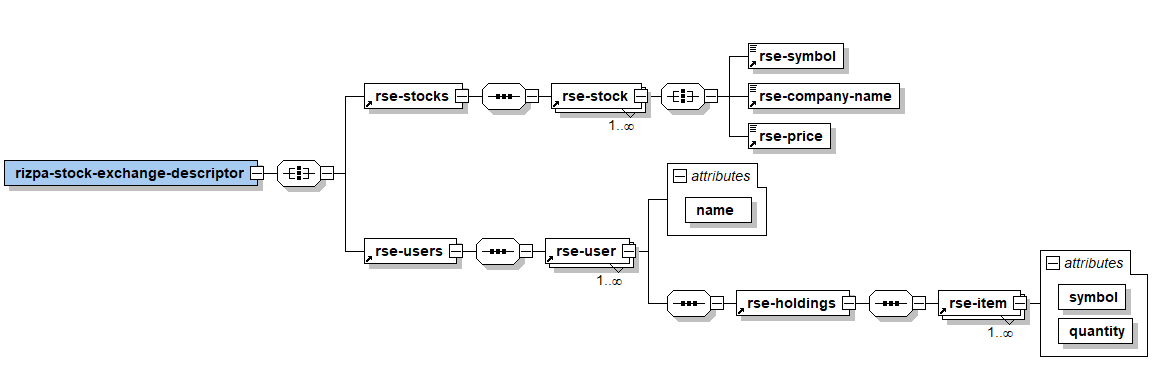
קובץ ה XML מקושר לסכמה שמגדירה את חלקיו השונים ואת חוקיותם.

להלן תרשימים המתארים את מבנה הסכמות בגרסאותיהם השונות:

### סכמה – גרסה 1



### סכמה – גרסה 2



### סכמה – גרסה 3

