


Java - שאלות תרגול #6

נושאים נוספים ב-OOP: Casting, Abstract, Interface

1. כיצד נוכל במחלקה היורשת לדרוס פונקציה שנכתבה במקור במחלקת האב? למה שנרצה לעשות זאת?
2. (א) צור מחלקה בשם Fruit (פרי) ובתוכה פונקציה בשם eat (לאכול) שמדפיסה: "...eating the fruit".
(ב) כעת צור מחלקה בשם Orange (תפוז) היורשת מ-Fruit. בתוכה דרוס את הפונקציה eat של מחלקת האב. הפונקציה תדפיס "...peeling the orange and eating".
3. (א) צור מחלקה בשם Apartment (דירה) ובתוכה שדה address ושדה size. הוסף למחלקה פונקציית toString שמחזירה string המכיל את שדות המחלקה.
(ב) כעת צור מחלקה בשם StudioApartment (דירת סטודיו) היורשת מ-Apartment, ובתוכה שדה number_of_rooms. הוסף למחלקה פונקציית toString שמדפיסה את שדה(ות) המחלקה וגם קוראת ל- toString של מחלקת האב.
4. (אתגר) מהו Polymorphism? מה הוא מאפשר לנו לעשות?

5. * צור מחלקה בשם Teacher (מורה) ובתוכה שדה name ושדה id_number.
* צור מחלקה בשם MathTeacher (מורה למתמטיקה) היורשת מ-Teacher ובתוכה שדה math_skills.
(int).
* צור מחלקה בשם HistoryTeacher (מורה להיסטוריה) היורשת מ-Teacher ובתוכה שדה field_of_profession (תחום התמחות - string).
* שים לב - כל השדות במחלקות לעיל צריכים להיות protected.
כעת ב-Main צור שני משתנים מטיפוס **Teacher**, כאשר הראשון יצביע על מופע חדש של MathTeacher והשני יצביע על מופע חדש של HistoryTeacher.
6. (א) בהמשך לשאלה הקודמת - נסה כעת לפנות לשדה math_skills של MathTeacher באמצעות המופע שיצרת ב-Main. האם הצלחת? מדוע?
(ב) צור פונקציה ב-Main בשם IdentifyTeacher שמקבלת כפרמטר Teacher ומדפיסה האם הוא מורה למתמטיקה או מורה להיסטוריה. (רמז - השתמש ב-instance of)
(ג) כעת שדרג את הפונקציה שכתבת בסעיף ב' - אם זה מורה למתמטיקה הדפס את השדה math_skills שלו, אם זה מורה להיסטוריה הדפס את השדה field_of_profession שלו.
(רמז - בצע casting)
7. (א) מה היא מחלקה אבסטרקטית?
(ג) האם מחלקה אבסטרקטית יכולה לרשת ממחלקה אבסטרקטית אחרת?
8. (א) מהי פונקציה אבסטרקטית?
(ב) האם פונקציה אבסטרקטית (במחלקה אבסטרקטית) יכולה להיות private?
(ג) האם לפונקציה אבסטרקטית ניתן להגדיר גם גוף או חתימה בלבד?
9. (א) האם במחלקה אבסטרקטית כל הפונקציות חייבות להיות אבסטרקטיות?
(ב) האם תיתכן מחלקה אבסטרקטית ובה פונקציות רגילות בלבד?



10. * צור מחלקה אבסטרקטית בשם Mechanic (מכונאי) ובתוכה שדה name ושדה years_of_experience (int). הוסף למחלקה:

- פונקציה אבסטרקטית בשם repair.
- פונקציה רגילה בשם useScrewdriver שמדפיסה למסך: "...Using screwdriver to open this screw".

* צור מחלקה בשם CarsMechanic (מכונאי רכב) היורשת מ-Mechanic ובתוכה שדה (מערך של string) בשם car_types_i_can_repair. הוסף למחלקה:
- מימוש של הפונקציה repair של מחלקת האב, שמדפיסה: "...repairing car"
- (*אתגר) בפונקציה בדוק אם השעה הנוכחית היא אחרי 17:00 או לפני 7:00, אם כן הדפס - "garage is ...closed"

פונקציה בשם replaceWheel המדפיסה: "...replacing the damaged wheel"



* צור מחלקה בשם AirplanesMechanic (מכונאי מטוסים) היורשת מ-Mechanic ובתוכה שדה rank (int). הוסף למחלקה:
- מימוש של הפונקציה repair של מחלקת האב, שמדפיסה: "...repairing airplane"
- פונקציה בשם checkEngine המדפיסה: "...checking the engine"

11. * צור מחלקה אבסטרקטית בשם ElectricDevice (מכשיר חשמלי) ובתוכה שדה בשם voltage (float) ושדה בשם worksOnBatteries (boolean) ושדה בשם isFaulty (boolean). הוסף למחלקה:

- בנאי שמקבל כפרמטר את worksOnBatteries + מאתחל את הערך של isFaulty ל-true.
- פונקציה בשם turnOn שמדפיסה: "...Turning device on"
- פונקציה מסוג final בשם insertNewBatteries. פעולת הפונקציה: אם השדה worksOnBatteries הוא true, הפונקציה תדפיס: "Inserting new batteries to the device", אם לא, הפונקציה תדפיס: "This device doesn't use batteries"
- פונקציה אבסטרקטית בשם useDevice.
- פונקציה אבסטרקטית בשם fixDevice.



* צור מחלקה בשם TV (טלוויזיה) היורשת מ-ElectricDevice, ובתוכה שדה screen_size. הוסף למחלקה:
- בנאי שמקבל את גודל המסך, קורא לבנאי האבא ושולח לו את הערך של worksOnBatteries כ-true.
- מימוש של הפונקציה useDevice של מחלקת האב, שמדפיסה: "...watching tv"
- מימוש של הפונקציה fixDevice של מחלקת האב, שבודקת האם הערך של isFaulty הוא true, אם כן היא תדפיס: "...fixing tv" ותעביר את isFaulty ל-true, אם לא היא תדפיס: "this tv is working properly"

* צור מחלקה בשם Cellphone (טלפון סלולרי) היורשת מ-ElectricDevice ובתוכה שדה brand. הוסף למחלקה:
- בנאי שמקבל את יצרן הטלפון, קורא לבנאי האבא ושולח לו כפרמטר את הערך של worksOnBatteries כ-true.
- מימוש של הפונקציה useDevice של מחלקת האב, שמדפיסה: "...talking on the phone"
- מימוש של הפונקציה fixDevice של מחלקת האב, שבודקת האם הערך של isFaulty הוא true, אם כן היא תדפיס: "...fixing cellphone" ותעביר את isFaulty ל-true, אם לא היא תדפיס: "this cellphone is working properly"
- נסה לדרוס את הפונקציה insertNewBatteries, האם הצלחת? מדוע?

המשך בעמוד הבא...

* בעת ב-Main צור שני משתנים מטיפוס ElectricDevice, כאשר הראשון יצביע על מופע חדש של TV והשני יצביע על מופע חדש של Cellphone.
צור פונקציה בשם RepairElectricDevice ובתוכה:

- קבל כפרמטר ElectricDevice.
- בדוק האם מדובר במכשיר שעובד על סוללות (worksOnBatteries == true), אם כן הפעל את הפונקציה insertBatteries.
- בדוק האם המכשיר תקין (isFaulty == false), אם כן צא מן הפונקציה באמצעות return.
- קרא לפונקציה fixDevice כדי לתקן את המכשיר.
- בדוק האם המכשיר הוא טלוויזיה (TV) או טלפון סלולרי (Cellphone). בצע casting בהתאם.
- אם מדובר בטלוויזיה הדפס את גודל הטלוויזיה, אם מדובר בטלפון סלולרי הדפס את שם היצרן.
- הדפס למסך: "fixing is done"!

12. (א) האם ניתן לרשת ממחלקה מסוג final?
(ב) תן דוגמא למחלקה שכדאי יהיה להגדיר אותה כ-final.

13. (א) מהו interface? באילו מקרים נשתמש ב-interface?

14. (א) האם ניתן להוסיף access modifier ל-interface? לפונקציות שבתוך ה-interface?
(ב) (*אתגר) האם אפשר לממש את הפונקציות שב-interface?
(ג) האם מחלקה יכולה לרשת מיותר מ-interface אחד?
(ג) (*אתגר) האם ניתן להגדיר שדות בתוך interface?

15. (*אתגר) כיצד נהוג לקרוא ל-interface ריק? מתי נשתמש בו?

16. מה ההבדל בין interface ל-abstract class? מתי נבחר בכל אחד מהם?

17. * צור interface בשם IPlayer (נגן) ובתוכו שתי פונקציות מסוג void: compose ו-void: play.

* צור מחלקה אבסטרקטית בשם Player המממשת את IPlayer. הוסף לה שדה בשם name ושדה בשם favoriteGenre (שניהם string).

* צור מחלקה בשם GuitarPlayer (נגן גיטרה) היורשת מ-Player.
במימוש הפונקציה play הדפס למסך: "...Playing Hotel California"
במימוש הפונקציה compose הדפס למסך: "...composing guitar melody"
הוסף פונקציה בשם rockAndRoll שמדפיסה למסך: "Let the show begin"!

* צור מחלקה בשם PianoPlayer (נגן פסנתר) היורשת מ-Player.
במימוש הפונקציה play הדפס למסך: "...Playing Bach"
במימוש הפונקציה compose הדפס למסך: "...composing piano melody"
הוסף פונקציה בשם practice שמדפיסה למסך: "practice makes perfect"!
הוסף פונקציה בשם playJazz שמדפיסה למסך: "Playing I Got Rhythm by George Gershwin"

* במחלקת Main צור שני משתנים מטיפוס IPlayer. כאשר הראשון יצביע על מופע חדש של GuitarPlayer והשני יצביע על מופע חדש של PianoPlayer.
כתוב פונקציה בשם startMusicShow שמקבלת כפרמטרים נגן (Player). בתוך הפונקציה:
- הפעל את הפונקציה play.
- בדוק איזה נגן קיבלת ובצע casting.
- אם הוא נגן גיטרה - הפעל את הפונקציה rockAndRoll.
- אם הוא נגן פסנתר - הפעל את הפונקציה playJazz.

18. מהו ArrayList? מה ההבדל בינו לבין מערך רגיל?



19. א) צור רשימה מסוג `ArrayList<Integer>` והוסף אליה שלושה מספרים: 5,7,9.
ב) הדפס את כל איברי הרשימה (אין צורך בלולאה לשם כך).
ג) הדפס את המספר הראשון ברשימה.
ג) מחק מן הרשימה את המספר השני ברשימה.
ד) הדפס את מיקומו של המספר 9 ברשימה (רמז: השתמש ב-`indexOf`)

20. (*אתגר) מדוע מגדירים `ArrayList` של `Integer` ולא של `int`?

21. א) מתי נשתמש בסימון `Test@` מעל שם הפונקציה?
ב) מה עושה הפונקציה `assertEquals`?

22. צור מחלקה בשם `RoundZero` ובה אוסף של פונקציות (סטטיות) המקבלות מספר בין 0 ל-1 ומעגלות אותו:
- `RoundUp` - אם התקבל 0 הוא יחזיר 0, אחרת הוא יעגל למעלה ל-1.
- `RoundDown` - אם התקבל 1 הוא יחזיר 1, אחרת יעגל למטה ל-0.
- `Round` - מעגל למספר הקרוב ביותר (0.5 יעוגל כלפי מעלה)
אם אחת הפונקציות שלעיל תקבל כפרמטר מספר הקטן מ-0 או גדול מ-1 היא תחזיר -1

צור מחלקת `RoundTest` וממש בתוכה פונקציות `test` הבודקות את כל אפשרויות העיגול, כולל מקרי קצה וכולל הפלט -1.

בהצלחה!

