

קשיים של תלמידים ביחידה חמישית יחידת תכנות מונחה עצמים

במה עוסקת היחידה?

חשיפת התלמידים הלומדים מדעי המחשב ברמה מוגברת לגישה של עבודה עם מחלקות ועצמים הכוללת קשרים ביניהם. יחס של הכלה, ירושה או תפקוד משותף כלומר ממשקים. התלמיד נחשף לעקרונות והמנגנונים של התכנות מונחה עצמים: הכמסה או הסתרת מידע, הורשה וממשקים, פולימורפיזם, דריסת פעולות לעומת העמסת פעולות, המרות, הרשאות גישה. התלמיד עוסק במידול בעיה ויצירת מחלקות לפתרון הבעיה תוך שימוש בעקרונות והמנגנונים שהוזכרו לעיל.

קשיים ובעיות של תלמידים

- 1- למה צריך את זה? בשביל מה ירושה? בשביל מה ממשקים?
- 2- ניתוח בעיה וקביעת המחלקות והיחסים ביניהן לפתרון הבעיה.
- 3- הימנעות משיפול קוד.
- 4- הבחנה בין תכנות מופע של עצם לעומת תכנות מחלקה.
- 5- מה עובר בירושה ומה לא ומה הקשר להרשאות הגישה.
- 6- המרות. מתי אפשר? מתי צריך מפורש ומתי לא?
- 7- מתי המרה כלפי מעלה ומתי כלפי מטה?
- 8- הבחנה בין יחס הכלה ליחס ירושה.
- 9- נושא הפולימורפיזם ודריסת פעולות.
- 10- בפרט, מעקב אחר קוד הכולל יחסי ירושה בין המחלקות ודריסת פעולות במחלקות הבת.
- 11- מתי ניתן להפעיל פעולה ומתי לא? כלומר הפעלת פעולה על עצם שנוצר ממחלקת הבת האם יביא לשגיאת תחביר או לא? אם לא, מה איזו פעולה יבצע?
- 12- זיהוי שגיאות והבחנה בין שגיאת הידור לשגיאת ריצה.
- 13- מתי יש דריסת פעולות ומתי העמסת פעולות.
- 14- ממשקים וההבדל ביניהם לבין ירושה בין מחלקות.
- 15- פולימורפיזם בממשקים.
- 16- למה צריך מחלקה מופשטת ומה ההבדל בינה לבין ממשק.
- 17- קיום או אי קיום פעולה בונה במחלקה.
- 18- הפניות לעצמים, הבחנה בין הפעלת פעולה בונה מעתיקה לפעולה בונה רגילה בהקשר של עצמים.
- 19- שימוש ב `instanceof` לא מספיק, כדי להפעיל פעולה שקיימת במחלקת הבת, יש לבצע קודם המרה מפורשת למחלקת הבת ואז להפעיל את הפעולה. בדיקה שנוצר ממחלקת הבת, לא מספיקה.

דוגמאות

דוגמה 1-א: קיום פעולה בונה ריקה כברירת מחדל ופולימורפיזם

<pre>public class A { public String toString() { return "A"; } }</pre>	<pre>public class B extends A { public String toString() { return "B, kind of "+ super.toString(); } }</pre>	<pre>A a = new A(); B b = new B(); System.out.println(a); System.out.println(b);</pre>
--	--	--

אין בעיה בקוד המחלקות, הקוד תקין, קיימות במחלקות פעולה בונה ריקה כברירת מחדל.

פלט הקוד:

A
B, kind of A

דוגמה 1-ב: אי קיום פעולה בונה ריקה כברירת מחדל

<pre>public class A { protected int x; public A(int x) { this.x = x; } public String toString() { return "A"; } }</pre>	<pre>public class B extends A { public String toString() { return "B, kind of "+ super.toString(); } }</pre>	<pre>A a = new A(); B b = new B(); System.out.println(a); System.out.println(b);</pre>
---	--	--

הוספנו למחלקה A, תכונה ופעולה בונה המקבלת פרמטר.

שאלה: מה קרה ?

תשובה: הופיעה שגיאת תחביר במחלקה B.

שאלה: מדוע ?

תשובה: ברגע שיצרנו פעולה בונה עם פרמטר, הפעולה הבונה ברירת המחדל הריקה לא קיימת יותר,

לכן, במחלקה B, מחלקת הבת, אי אפשר

להפעיל יותר את הפעולה הבונה הריקה שהייתה

קיימת במחלקה A .

הצעות לפתרון:

(1) להוסיף פעולה בונה ריקה במחלקה A

(2) להוסיף פעולה בונה עם פרמטר במחלקה B

(3) להוסיף פעולה ריקה במחלקה B ובתוכה לזמן את (מספר) super

4) במידה ולא מוסיפים פעולה בונה ריקה במחלקה A, יש לזמן בפעולה הראשית את הפעולה הבונה עם פרמטר.

דוגמה 2-א: הפעלת הבנאי הריק של מחלקת העל גם אם לא כתוב בצורה מפורשת.

<pre>public class C { public C() { System.out.print("C"); } }</pre>	<pre>public class D extends C { public D() { System.out.print("D"); } }</pre>	<pre>(1) C c = new C(); (2) D d = new D();</pre>
---	---	--

שאלה:

האם הקוד תקין? במידה וכן מה הפלט עבור ההוראות בפעולה הראשית? אם לא ציין את השגיאות והצע תיקון.

תשובה:

הקוד תקין. הפעולה הבונה של מחלקה B מפעילה את הפעולה הבונה של מחלקת העל, כלומר את **super()** ולכן

הפלט יהיה: **CCD**

דוגמה 2-ב: הוספת השורות הבאות לפעולה הראשית:

```
(3) C c1 = d;
(4) D d1 = c;
```

שאלה:

קבע עבור כל הוראה האם תקינה או לא. נמק תשובתך.

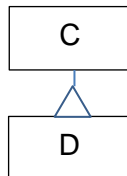
תשובה:

לפני שעונים על השאלה, רצוי לצייר את היררכיית המחלקות.

העצם **d** הוא סוג של **C**, ולכן ההצבה נכונה בהוראה (3). יש כאן המרה

מפורשת כלפי מעלה ממחלקה **D** למחלקה **C**.

הוראה (4) שגויה, יש כאן **שגיאת תחביר**. אין התאמה בטיפוסים. המחלקה **D** לא "מכירה" את מחלקה **C**.



דוגמה 3: מתי זימון פולימורפי של פעולות ומתי שגיאת תחביר?

<pre>public class A { public void p1() { System.out.println("A: p1"); } }</pre>	<pre>public class B extends A { public void p1() { System.out.println("B: p1"); } public void p2() { System.out.println("B: p2"); } }</pre>	<ol style="list-style-type: none"> 1) A a1 = new A(); 2) A a2 = new B(); 3) B b1 = new B(); 4) a1.p1(); 5) a2.p1(); 6) b1.p1(); 7) a2.p2(); 8) b1.p2(); 9) B b2 = a2; 10) ((B)a1).p2();
---	---	---

שאלה:

לפניך המחלקות **A, B** והוראות בפעולה הראשית.

א- עקוב אחר ההוראות (6) – (1) ורשום את הפלט.

ב- קבע עבור כל הוראה האם היא נכונה או לא. במידה ויש שגיאה, קבע אם שגיאת הידור או ריצה.

הצע תיקון לשגיאות.

תשובה:

א- הפלט עבור הוראות (6) - (1) הוא:

A: p1 B: p1 B: p1

במחלקה **B** יש דריסת הפעולה **p1**.

מי שמפעיל את הפעולה הוא העצם ממנו נוצר המופע.

ב- בשורה (7) יש שגיאת הידור. העצם **a2** אמנם נוצר ממחלקה **B**, אך מתסכלים עליו מנקודת מבט של

המחלקה **A**. במחלקה **A** לא קיימת הפעולה **p2** ולכן תופיע שגיאת תחביר.

תיקון: נבצע המרה מפורשת כלפי מטה אל המחלקה **B** ואז נפעיל את הפעולה **p2** ((B)a2).p2()

בשורה (9) יש שגיאת הידור, אין התאמה בטיפוסים. אי אפשר להציב בעצם מטיפוס המחלקה **B** עצם מטיפוס

המחלקה **A**.

תיקון: נבצע המרה מפורשת כלפי מטה מטיפוס המחלקה **A** לטיפוס המחלקה **B**. ניתן ואפשרי כי העצם **a2**

נוצר כ **B**. B b2 = (B)a2;

בשורה (10) יש שגיאת ריצה. יש ניסיון להמיר עצם שנוצר ממחלקה **A** לעצם מטיפוס המחלקה **B**.

המרות מתבצעות בזמן ריצה. בזמן ריצה התכנית תגלה שאי אפשר לבצע המרה זו.

המלצות

- 1- אין טוב ממראה עיניים.
רצוי להקליד קוד קטן, לפני שמריצים לשאול את התלמידים מה לדעתם יהיה הפלט או האם יש שגיאת הידור? שגיאת ריצה בקוד?
התלמידים עונים ורק אז המורה מריץ את הקוד לבדיקה האם צדקו.
במידה ושגו, לנתח ביחד עם התלמידים ולהבין למה זה כך ולא כפי שחשבו.
- 2- לבקש לשרטט את היררכית המחלקות גם אם לא נדרשו בשאלה.
- 3- במעקב אחר קוד, יש לשרטט תרשים עצמים הכולל הפנייה לעצם, בתוך המלבן המתאר את העצם, יש לרשום:
נותרת המחלקה, שם התכונה והתוכן שלה בעבור כל תכונה שהיא תכונת מופע של עצם.
את התכונות של המחלקה לכתוב ליד המלבן וכל פעם שיש בהן שינוי, יש לציין את העדכון, מחיקת הערך הקודם וקביעת הערך החדש.
- 4- בכתיבת מחלקות לפתרון בעיה, יש לזהות תכונות ופעולות משותפות וליצור עבורם מחלקת על, ממנה ירשו המחלקות עם התכונות והפעולות הנוספות.



בהצלחה למורים ולתלמידים, EVI