

## חזרה בנושא אוסף ספציפי

# ספר טלפונים

### מטרות

כתיבת מחלקה המגדירה אוסף ספציפי.  
בחירת ייצוג לאור תיאור בעיה ושיקולי יעילות.

### רמת השאלה

מעבר לשאלת בגרות. על אופן ההתאמה לרמת בגרות ראו בהנחיות המיוחדות.

### רקע

ספר טלפונים (Phone Book) הוא אוסף דינמי של אנשי קשר. לכל איש קשר יש שם-מלא ומספר-טלפון. בספר הטלפונים לא קיימים שני אנשי קשר עם אותו שם.

להלן ממשק המחלקה PhoneBook המגדירה את ספר הטלפונים:

PhoneBook()	הפעולה בונה ספר טלפונים ריק
<b>void</b> addContact (String name,String phone)	הפעולה מוסיפה לספר הטלפונים איש קשר חדש ששמו name ומספר הטלפון שלו phone. <b>הנחה</b> : לא קיים איש קשר בשם name בספר הטלפונים
<b>void</b> delContact (String name)	הפעולה מוחקת מספר הטלפונים את איש הקשר ששמו name. <b>הנחה</b> : קיים איש קשר בשם name בספר הטלפונים
String[] getAllContactsNames()	הפעולה מחזירה את אוסף השמות של אנשי הקשר המופיעים בספר הטלפונים
String getPhone (String name)	הפעולה מחזירה את מספר הטלפון של איש הקשר ששמו name. אם לא קיים איש קשר בשם name, הפעולה תחזיר את הערך <b>null</b>
<b>void</b> setPhone (String name, String phone)	הפעולה מעדכנת את מספר הטלפון של איש הקשר ששמו name, במספר החדש phone. <b>הנחה</b> : קיים איש קשר בשם name בספר הטלפונים
String toString()	הפעולה מחזירה מחרוזת המתארת את ספר הטלפונים במבנה הבא: <name> <phone> <name> <phone> : <name> <phone> אנשי הקשר יופיעו בשורות נפרדות בסדר אלפביתי עולה של שמם

## מה עליכם לעשות?

1. להלן קטע תכנית הבונה ספר טלפונים בשם pb, ושומר בו שישה אנשי קשר:

```
// יצירת ספר טלפונים
PhoneBook pb = new PhoneBook();

// הוספת אנשי קשר לספר הטלפונים
pb.addContact("Galit Israel", "03-9089730");
pb.addContact("Avner Chohen", "02-7474747");
pb.addContact("Gershon Avraham", "02-8900011");
pb.addContact("Daniela Yariv", "04-5677708");
pb.addContact("Alice Marlo", "04-5699300");
pb.addContact("Bob Denver", "04-5699300");

// הדפסת ספר הטלפונים
System.out.println(pb);
```

א. כתבו את הפלט המתקבל בסיום ביצוע קטע התכנית.

ב. כעת נניח שנוספו לספר הטלפונים pb עוד מספר לא ידוע של אנשי קשר. כתבו שורת קוד אחת שתדפיס את מספר אנשי הקשר הקיימים בספר הטלפונים pb.

2. כתבו פעולה המקבלת שני ספרי טלפון ומחזירה ספר טלפון חדש שהוא מיזוג של השניים. הספר החדש שיוחזר לא יכיל כפילויות של אנשי קשר. (אם אותו איש קשר מופיע בשני ספרי הטלפון המקוריים, הוא יישמר פעם אחת בלבד בספר החדש. אנו מניחים שאין שוני בין הנתונים השמורים עבור שמות זהים, כלומר אין משמעות של עדכון נתונים כאשר יש שני מופעים של אותו איש קשר).

3. כתבו את המחלקה PhoneBook. במידה והשתמשתם במחלקת עזר נוספות – כתבו אותן.

### הנחיות מיוחדות

- שאלה זו בודקת את יכולת התלמיד להבין ולהשתמש בפעולת ממשק של מחלקה נתונה. בנוסף נדרש התלמיד לכתוב את המחלקה על פי ההגדרה והממשק הנתון, תהליך הבנוי משני שלבים: בחירת ייצוג מתאים ומימוש הפעולות על פי הייצוג הנבחר.
- ברור שיש להשתמש במחלקת עזר שמגדירה איש קשר; מחלקה בה יש שתי תכונות: שם ומספר טלפון.
- זכרו שכשיש הנחות בממשק אין צורך לבדוק אותן בתוך המימוש של הפעולה.
- התלבטויות בבחירת הייצוג של ספר הטלפונים:
  - ספר הטלפונים הוא אוסף ספציפי ודינמי של אנשי קשר. לכן, ניתן לייצג אותו בעזרת רשימה או שרשרת חוליות של אנשי קשר.
  - מאחר ולא ייתכן שיהיו שני אנשי קשר עם אותו שם בספר הטלפונים, ולכל שם איש קשר "מוצמד" מספר טלפון, ניתן לייצג את ספר הטלפונים בעזרת מפה.
  - הפעולה toString() מחזירה מחרוזת המתארת את ספר הטלפונים כרשימה של אנשי הקשר המסודרים על פי הסדר האלפביתי של שמם. דרישה זו של מיון אנשי הקשר על פי שמם, מעלה את הדילמה הבאה: האם ספר הטלפונים יהיה ממוין כל הזמן? או שנמיון אותו רק כאשר תזומן הפעולה toString()?

אם נבחר באפשרות הראשונה, אז הפעולה `addContact(...)` צריכה לדאוג להכנסת איש הקשר החדש במקום המתאים כך שספר הטלפונים יישאר ממוין, ואז מימוש הפעולה `toString()` יהיה פשוט ומייד. אם נבחר באפשרות האחרת, נצטרך לבצע את המיון בתוך הפעולה `toString()`. אז מה כדאי? התשובה לכך תלויה בתדירות השימוש בפעולות `addContact(...)` ו-`toString()`.

- כדי להתאים את היקף השאלה לבגרות, ניתן לוותר על מימוש חלק מפעולות הממשק של ספר הטלפונים. מומלץ כן לממש את הפעולות `addContact(...)` ו-`toString()`, המבטאות את הדילמה המתוארת לעיל והנידונה גם בספר הלימוד בפרק 7: **מבני נתונים דינמיים לייצוג אוספים**.
- שאלה זו היא שאלת חזרה שהעונה עליה כראוי מפגין שליטה בנושאים רבים ומרכזיים הנלמדים ביחידה. היא משלבת מידול ובחירת ייצוג המושפע משיקולי יעילות, משלבת עצמים מורכבים וטיפול באוספים. בהיקף מתאים היא תבנית לשאלת בגרות טובה.

### פתרון

מצורפים קבצי המקור הבאים:

שם הקובץ	תיאור
PhoneBook.java	מחלקה המגדירה את ספר הטלפונים, המיוצג ע"י רשימה ונשמר ממוין כל הזמן.
Contact.java	מחלקת עזר המגדירה איש קשר, לו יש שתי תכונות שם ומספר טלפון.
TestPhoneBook1.java	מחלקת בדיקה ראשונה לספר הטלפונים: תפריט בחירה לביצוע פעולות.
TestPhoneBook2.java	מחלקת בדיקה שנייה לספר הטלפונים, הכוללת פתרון לסעיף 2 של השאלה.

### מקור השאלה

צוות הפיתוח.