

יסודות מדעי המחשב 1

מדריך מעבדה לסביבת העבודה

DrJava

כתב: אבי ברנשטיין

מהדורת עיצוב

תשס"ו 2006

אוניברסיטת תל-אביב החוג להוראת המדעים

מטה מל"מ המרכז הישראלי להוראת המדעים ע"ש עמוס דה-שליט

משרד החינוך האגף לתכנון ולפיתוח תכניות לימודים



אין לשכפל, להעתיק, לצלם, לתרגם או לאחסן במאגר מידע כל חלק שהוא מחומר הלימוד של ספר זה. שימוש מסחרי מכל סוג שהוא בחומר הכלול בספר זה אסור בהחלט, אלא ברשות מפורשת בכתב מהגורמים המפורטים להלן.



כל הזכויות שמורות
אוניברסיטת תל-אביב ומשרד החינוך

מדריך מעבדה לסביבת DrJava

5	1. הקדמה.....
8	2. סביבת העבודה.....
9	3. התוכנית הראשונה שלי.....
13	4. חלונות סביבת העבודה.....
14	5. הפעלה בסיסית.....
14	5.1 כיצד להפעיל את DrJava?.....
14	5.2 פתיחת ויצירת קבצים.....
15	5.3 שמירת קבצים.....
15	5.4 הידור (קומפילציה) של קובץ.....
17	5.5 הרצת תוכנית.....
19	5.6 חלון הדו-שיח (Interaction Pane).....
21	5.6.1 פעולות שכיחות בחלון הדו-שיח.....
23	5.6.2 הרצת הפעולה הראשית בחלון הדו-שיח.....
24	5.7 הזנת קלט.....
27	5.8 התמצאות ואפשרויות חיפוש.....
29	6. פונקציות מתקדמות.....
29	6.1 ניפוי שגיאות ריצה (Debug Mode).....
32	6.2 תצורה של סביבת העבודה (Preference Definitions).....
36	7. נספחים.....
36	7.1 דרישות המערכת.....
36	7.2 התקנת סביבת העבודה.....
42	7.3 מה חשוב לוודא?.....
43	7.4 שגיאות התקנה.....
44	7.5 שגיאות תפעול.....
45	7.6 שגיאות הידור.....
46	7.7 רישיון שימוש בתוכנה.....

1. הקדמה

מטרת מדריך זה היא להנחות וללוות אותך התלמיד, במהלך עבודתך המעשית במעבדת המחשבים, במסגרת לימודי היחידה יסודות מדעי המחשב 1. בטרם נרד לפרטים ובטרם נדקדק ונקפיד במיקומו של כל תו ופסיק, ננסה ליצור שפה אחידה שתשמש אותנו כמכנה משותף בעבודתנו המעשית.

ראשית, עלינו להבין מהו המחשב (למרות שעל פניו הוא מהווה חלק בלתי נפרד מחיינו...), מהי תוכנית מחשב, למה הכוונה כאשר אנו נדרשים להדר (לקמפל) תוכנית ומה פרוש להריץ תוכנית. מושגים אלה מוצגים בפרק 1 של ספר הלימוד וכאן נתייחס אליהם על קצה המזלג.

המחשב הוא מערכת אלקטרונית היכולה לקרוא נתונים ממקור קלט כלשהו, לעבד אותו ולהציג כפלט את תוצאת העיבוד.

תוכנית מחשב היא סדרה של הוראות מדויקות לביצוע משימה נתונה, כאשר **הוראה** היא מילה או רצף מילים שהמחשב "מבין".

תוכנית מחשב כתובה בדרך כלל במה שקרוי **שפה עילית**. היא כתובה באופן טקסטואלי וכוללת הוראות מובנות יחסית לקריאת נתונים מהקלט, הגדרות משתנים בזיכרון, ביצוע פעולות חשבוניות ולוגיות, הצגת נתונים לפלט וכו'. תוכנית נכתבת בעזרת **עורך (editor)** וניתנת לשמירה כקובץ במחשב.

כדי שהמחשב "יבין" תוכנית מחשב, יש לתרגמה **לשפת מכונה (machine language)**. הוראות בשפת מכונה הן למעשה רצפים של אפסים ואחדות ולכן שפת מכונה היא קשה מאוד להבנה לבני אדם. למשל, הנה שורה אפשרית בתוכנית הכתובה בשפת מכונה: 1100 1010 0000. התהליך בו מתורגמת תוכנית משפה עילית לתוכנית בשפת מכונה נקרא **הידור. המהדר (compiler)** הוא גם תוכנית מחשב, הקוראת תוכנית שכתובה בשפה עילית ומתרגמת אותה לשפת מכונה כדי שניתן יהיה להריץ אותה.

Java היא שפה עילית, אבל, להבדיל משפות עיליות אחרות, כגון C או C++, תוכניות הכתובות בשפת Java מומרות על ידי המהדר קודם כל לשפת ביניים הנקראת bytecode, שהיא עדיין אינה שפת מכונה. כלומר, ישנו שלב נוסף שהוא שלב **הפרשנות (interpretation)**. **המפרש (interpreter)** של Java הוא תוכנית מחשב בעלת תפקיד כפול: **תרגום והרצה**. במהלך פעולתו לוקח המפרש חלקים קטנים מה-bytecode, מתרגם אותם לשפה מכונה ומיד מריץ אותם.

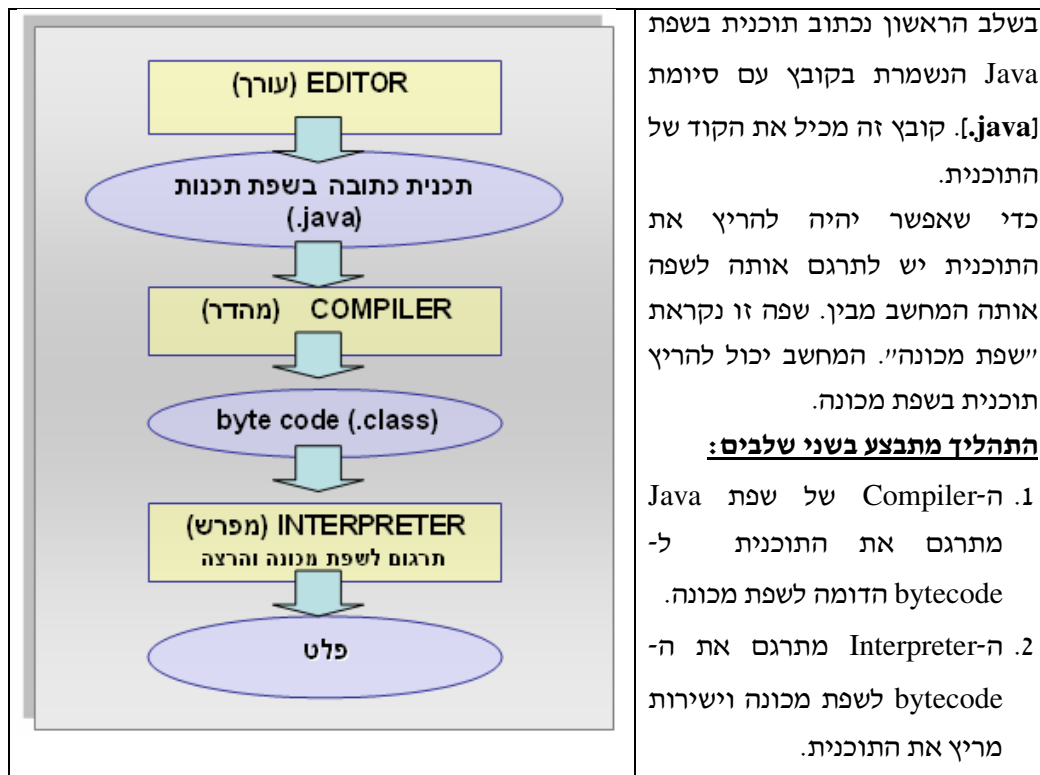
מהן בעצם הפעולות הנדרשות מאתנו כדי שנוכל לכתוב תוכנית מחשב ולהריצה? כדי לכתוב ולהריץ תוכנית בסביבה מבוססת Java יש צורך בשלוש תוכניות, כאשר לכל תוכנית תפקיד שונה:

1. מעבד תמלילים (למשל Notepad) – עורך.

2. מהדר – Compiler

3. מפרש – Interpreter

תמונה 1 מציגה בצורת תרשים את רצף הפעילויות הנדרשות להגיע לשלב ההרצה:



תמונה 1

אנחנו נעבוד עם כלי שנקרא DrJava שפותח באוניברסיטת Rice בארה"ב ומאגד תחת קורת גג אחת את כל הפעולות הללו, כלומר, DrJava משמש כסביבת פיתוח לתוכניות הכתובות ב-java. בעזרתו נכתוב ונערוך את התוכניות, נהדר (נקמפל) אותן, ולבסוף גם נריץ.

DrJava היא סביבת פיתוח עבור שפת Java, שמכוונת לסייע לתלמידים בתכנון הפיתוח ללא צורך להתמודד עם סביבות פיתוח מורכבות. הרעיון המרכזי שעמד בפני המפתחים הוא לספק סביבת עבודה בעלת ממשק משתמש קל, נוח ופשוט לתפעול מצד אחד, שהיא כלי פיתוח בעל עוצמה עם מגוון יכולות, מצד שני. **DrJava** מספקת גם יכולות ומאפיינים מתקדמים למשתמש המנוסה. עיקר הפונקציות המתקדמות מבוססות על חלון הדו-שיח (Interaction Pane), המאפשר לבצע חישובי ביטויים, ואותו נכיר בסעיף 5.6.

אם כך, מה בעצם מאפשרת לנו סביבת **DrJava**? אל סביבת העבודה אנחנו מגיעים לאחר שפיתחנו פתרון לבעיה אלגוריתמית, ויישמנו אותו כתוכנית בשפת Java, ואנו רוצים להשלים את תהליך הפתרון בכך שנריץ את התוכנית על דוגמאות קלט שונות ונבחן את הפלט.

- סביבת **DrJava** מאפשרת לנו לכתוב את קוד המקור של תוכנית המחשב. כאשר אנו מדברים על כתיבת תוכנית בסביבת הפיתוח כוונתנו למכלול פעולות עריכת הקוד, שכוללות את כתיבת הקוד עצמו, יכולת ביצוע שינויים, הזזה, הוספה, מחיקת שורות וכו'.

קוד המקור ייכתב בחלון עריכה ראשי, במסגרת חלון כתיבה המאפשר עריכה קלה ונוחה של קוד התוכנית.

- לאחר הכתיבה ניתן יהיה להדר (לקמפל) את הקוד ולתקן שגיאות הידור במידת הצורך. שגיאת הידור מתרחשת כאשר אנו כותבים הוראה שאינה תואמת את כללי התחביר (syntax בלעז) הנכון של השפה. בסביבת **DrJava** שגיאת הידור בולטת בתצוגתה ובפרטים הנלווים כדי לסייע לנו באיתור מקור הבעיה.

- כאשר בדינו תוכנית שעברה בהצלחה את שלב ההידור, ניתן להריץ את התוכנית ולוודא שהפלט המתקבל הוא אכן הפלט הנדרש (מאחורי הקלעים מתבצע גם תהליך הפירוש והתרגום כפי שהוזכר לעיל).

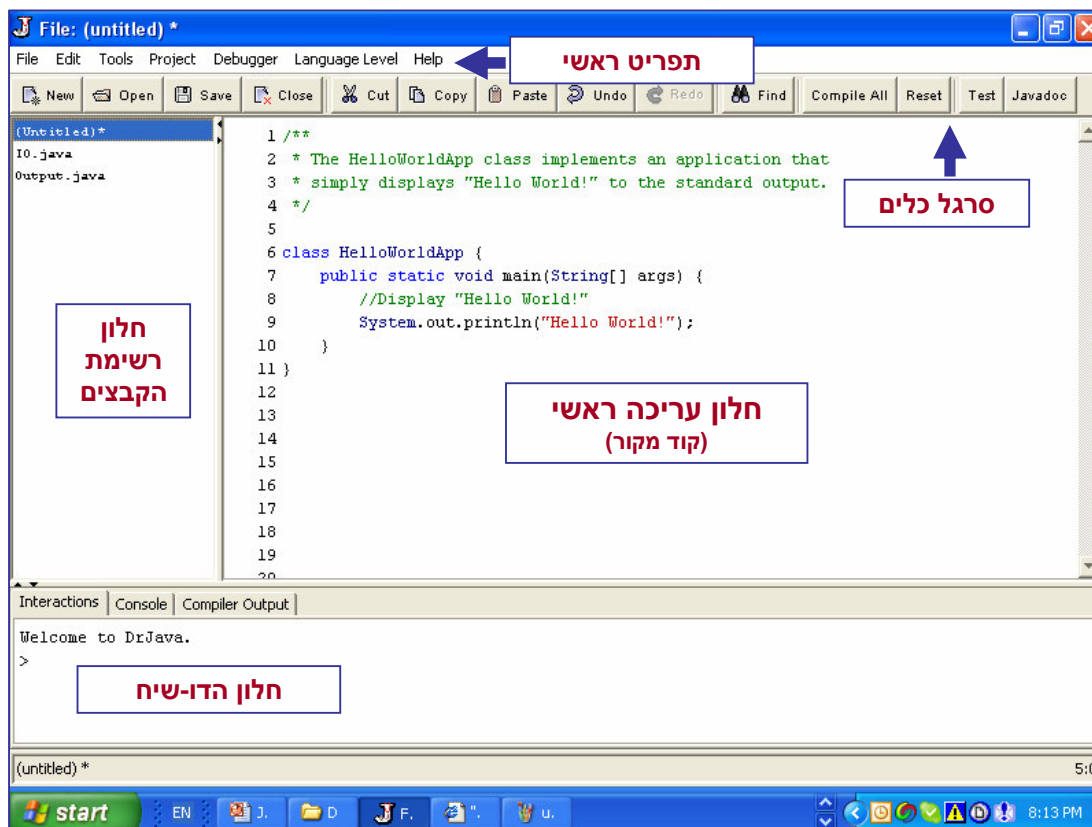
ואת כל אלה (והרבה יותר...), סביבת **DrJava** מאפשרת לנו לבצע.

ומה תפקידו של מדריך המעבדה?

תפקידו של מדריך זה פשוט: לכוון, להדריך ולסייע. לאפשר לכם הלומדים להבין ולהכיר את סביבת הפיתוח וכיצד תוכלו בעזרתה לכתוב תוכניות מחשב בשפת Java, לוודא את נכונותן ולהריצן כדי לקבל תוצאות, ולדעת להתמודד עם בעיות שיכולות לצוץ בסביבה זו.

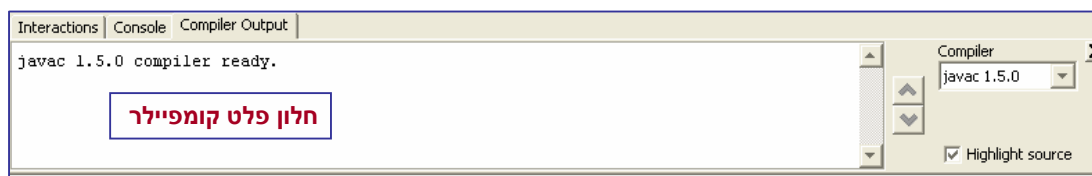
2. סביבת העבודה

סביבת העבודה מורכבת מכמה רכיבים מרכזיים, כפי שמוצג ומודגש בתמונה 2: (א) תפריט ראשי (ב) סרגל כלים (ג) חלון עריכת קוד המקור (ד) חלון רשימת הקבצים. בחלקו התחתון של העורך, ישנן מספר לשוניות שמאפשרות לפתוח חלונות נוספים: (ה) חלון הדו-שיח (Interaction Pane) (ו) חלון פלט הידור (קומפילציה – Compiler Output) (ז) חלון קונסול (Console).



תמונה 2

כך נראה חלון פלט הקומפיילר (ראו תמונה 3) לאחר שלחצנו על הלשונית המתאימה או שהפעלנו את תהליך ההידור (הקומפילציה – Compilation)



תמונה 3

לעיתים, נפתחים חלונות נוספים תוך כדי עבודה, בהתאם לבחירה בפעולות שונות, כפי שנראה בהמשך.

3. התוכנית הראשונה שלי

מטרת פרק זה היא להנחות אתכם הלומדים, בכתיבת תוכנית פשוטה וקלה כדי שנוכל לתרגל את שלבי הכתיבה השונים עד לקבלת פלט התוכנית, לאחר ששמרנו את קוד המקור בקובץ, הידרנו והרצנו את התוכנית.

התוכנית שנכתוב נקראת HelloWorldApp (צר לי על חוסר המקוריות, אך זו המסורת...). בעקבות הרצתה, תציג תוכנית זו כפלט על המסך את המילים Hello World.

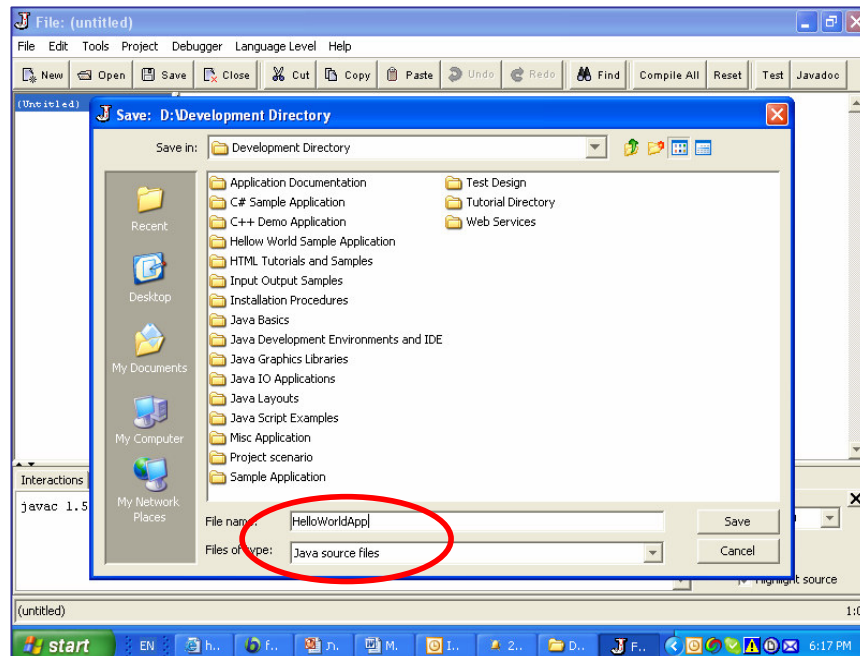
1. הפעילו את **DrJava**: על ידי בחירה בתפריטים הבאים (לפי הסדר, משמאל לימין):

Start → Java Environment → **DrJava**

או פשוט לחצו לחיצה כפולה (Double Click) על הצלמית (Icon) של **DrJava** בתיקה של C:\Yesodot\ (או על קיצור הדרך להפעלתו). בעקבות ההפעלה נפתח עבורנו (כברירת מחדל) קובץ חדש ששמו בשלב זה הוא untitled. קובץ זה אינו שמור עדיין בזיכרון המחשב.

2. לפתיחת קובץ חדש ישנן שתי אפשרויות: האחת, להקיש על כפתור New בסרגל הכלים והשנייה לבחור באופציה New בתפריט File, File → New: מתקבל המסך שראינו בתמונה מס. 2.

3. שמרו את הקובץ (פעולה זו ניתנת לביצוע גם לאחר כתיבת הקוד). לשם כך בחרו באופציה Save בתפריט File, כלומר File → Save (או כפתור Save בסרגל הכלים). נקבל את המסך הבא:



תמונה 4

הקלידו את שם הקובץ, HelloWorldApp, ובחרו בתיקיה בה אתם מעוניינים לשמור אותו. הקובץ שיישמר לאחר לחיצה על כפתור Save יהיה קובץ עם סיומת [java]. (ראו תמונה 4). כדאי לוודא כי לקובץ שנשמר אכן יש את הסיומת הדרושה.

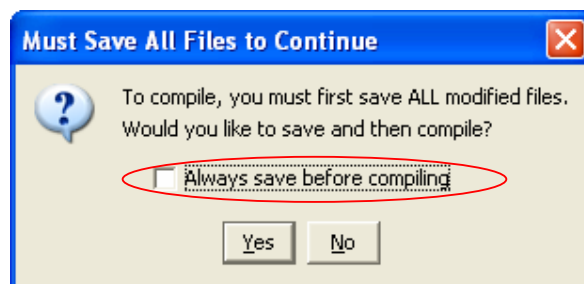
4. הקלידו את התוכנית הבאה (הקפידו על סימני פיסוק ואותיות גדולות וקטנות):

```
/*  
Output: the sentence "Hello World!"  
*/  
  
class HelloWorldApp  
{  
    public static void main (String[] args)  
    {  
  
        // Display message "Hello World!"  
        System.out.println ("Hello World!");  
    }  
}
```

5. שמרו את הקובץ תחת השם HelloWorldApp.java. שם הקובץ צריך להיות כשם המחלקה שהקובץ מכיל, וזו ברירת המחדל שתוצע לכם כאשר תלחצו על כפתור Save בסרגל הכלים, או תבחרו בתפריט את: File → Save.

6. **הדרו (קמפלו) את התוכנית:**

בשורת התפריט בחרו: Tools → Compile Current Document. אם לא נעשתה טעות הקלדה ייווצר קובץ חדש בשם HelloWorldApp.class, באותה תיקיה בה נשמר הקובץ HelloWorldApp.java (וודאו שכך הדבר). בחלון פלט המהדר (Compiler Output) תתקבל ההודעה: Last compilation completed successfully. אם לא שמרתם את הקובץ בשלב הקודם (סעיף 5), תידרשו לשמור אותו לפני ביצוע ההידור, (ראו תמונה 5). ההידור יכול להתבצע רק על קוד שמור בקובץ.



תמונה 5

ניתן למנוע הישנות הודעה זו אם נבחר את האפשרות Always save before compiling. כלומר, אנו מבקשים לשמור באופן אוטומטי את הגרסה העדכנית (על פי שם הקובץ שניתן לפני כן) לפני כל פעולת הידור.

7. הריצו את התוכנית:

בשורת התפריט בחרו Tools → Run Document's Main Method.

פלט התוכנית, המילים Hello Word, יוצג בחלון הקונסול (Console) ויופיע גם בחלון הדו-שיח כפי שניתן לראות בתמונה 6.



תמונה 6

8. כעת ננסה להתחכם... (אולי כדי לראות מדוע לא כדאי...): נכתוב במקום `System.out.println` את הרצף `System.out.println` וננסה להדר (לקמפל) את התוכנית. בדקו מה תהא תגובת המהדר.

כעת, לאחר שהבנו את תהליך הכתיבה וההרצה של תוכנית מחשב, נדגים את תהליך על תוכנית נוספת שכל תפקידה לקלוט מספר שלם מהמשתמש. הקלידו את התוכנית כפי שהיא מוצגת כאן, שמרו אותה תחת השם `GetNumber.java`, הדרו והריצו אותה.

```
/*
Input: an integer number, typed in by the user
Output: The given number
*/

class GetNumber
{
    public static void main (String args[])
    {
        // Read in an Integer number
        int input_num = In.readInt ("Insert an Integer Number:");

        // Print the number typed by the user
        System.out.println(input_num);
    }
}
```

שימו ♥ לחלון הקלט בו עליכם להקליד את המספר השלם, והיכן מופיעה בקשת הקלט. בסביבת **DrJava** הדבר מתבצע בשני חלונות נפרדים, והסיבה לכך היא התפקיד השונה של כל חלון: הבקשה לקלט, או ההנחיה על טיב הקלט המבוקש (Insert an Integer Number) מוצגת בחלון הדו-שיח (Interaction Pane) משום שזהו החלון בו מתבצע הדו-שיח עם המשתמש. את הקלט עצמו יש להקליד בחלון הקלט הסטנדרטי של **DrJava**, אשר מופיע מעצמו לצורך הזנת הקלט.

שימו ♥ : התוכנית עוצרת בהגיעה לשורת הקלט וממתינה לקלט. לאחר הכנסת הקלט בחלון הקלט יש ללחוץ על מקש Enter או על כפתור Done ורק אז תמשיך ריצת התוכנית!

לאחר שקיבלנו את התחושה הבסיסית של כתיבת תוכנית ב-Java והבנו את התהליך המתבצע כדי להריץ תוכנית, נרחיב ונעמיק, בהמשכו של מדריך זה, את בסיס הידע וההבנה שלנו ונדון בסביבת העבודה על מרכיביה השונים.

4. חלונות סביבת העבודה

כדי לאפשר סביבת עבודה נוחה, קלה וידידותית כוללת תצוגת סביבת העבודה כמה חלונות מרכזיים, וכולם יחד מספקים תפעול נוח וקל.

החלון המרכזי הוא ה-Text Editor, **חלון עריכה ראשי (Definitions Pane)**. חלון זה מאפשר את כתיבת הקוד ב-Java, תוך צביעה אוטומטית של מילים שונות בקוד, לצורך בהירות וקריאות הקוד, מספור אוטומטי של שורות והזחה (אינדנטציה - Indentation) מתאימה.

החלון השמאלי, **חלון רשימת הקבצים**, מציג את רשימת הקבצים הפתוחים בכל רגע. בחירת קובץ מסוים בחלון זה תציג את תוכנו של הקובץ בחלון העריכה הראשי. קובץ חדש שנפתח, אך עדיין לא נשמר, יופיע בחלון זה תחת הכותרת untitled.

בחלקה התחתון של סביבת העבודה ישנן שלוש לשוניות, ולחיצה על כל אחת מהן תציג את החלון המתאים לה. חלון אחד הוא חלון הדו-שיח (Interaction Pane), החלון השני מציג את פלט תהליך ההידור, הקומפילציה (Compiler Output) והחלון השלישי הוא חלון הקונסול (Console):

חלון הדו-שיח (Interaction Pane) הוא אחד מיתרונותיה (הטובים יותר) של סביבת ה-DrJava. חלון זה מקל על עבודת הלומד המתחיל ותרומתו רבה גם למשתמש המנוסה. החלון מאפשר בצורה זריזה לנסות קטעי קוד, מבלי לדרוש לכתוב מעטפת מורכבת.

למשל, ניתן לבצע חישובים מספריים או השמת ערך במשתנה מסוים או במשתנים מסוימים וביצוע חישוב עם משתנים אלה. באופן דומה ניתן לבצע פעולות מורכבות יותר בהתבסס על מבנים פנימיים. תמונה 7 מראה חישוב פשוט שנוכל לבצע בחלון. פעולות נוספות יפורטו בתת פרק 5.6.



תמונה 7

חלון ההידור, הקומפילציה (Compiler Output) מציג את הפלט של תהליך ההידור והשגיאות שאותרו (אם היו כאלו) או הודעה על הצלחת ההידור (קומפילציה).

בחלון הקונסול (Console) מתבצע ומוצג קלט התוכנית וגם מוצג הפלט (בנוסף להצגה בחלון הדו-שיח).

חלונות נוספים מתווספים תוך כדי עבודה, בהתאם לפעולות המתבצעות. למשל, כאשר אנחנו מבצעים פעולת חיפוש או החלפה של טקסט בתוכנית, פעולת Find / Replace, מתווספת לשונית נוספת שמספקת אפשרויות נוספות של חיפוש (ראו סעיף 5.7, התמצאות וחיפוש).

5. הפעלה בסיסית

פרק זה דן בכל הפעולות הבסיסיות הנדרשות לעבודה בסביבת העבודה של **DrJava** בכלל, ובעבודה עם קבצים בסביבת העבודה של **DrJava**, בפרט. נלמד כיצד להפעיל את **DrJava**, ליצור קובץ חדש, לפתוח קובץ קיים, להדר (לקמפל) קובץ וכמובן להריץ תוכנית שפיתחנו. בנוסף נלמד כיצד להגדיר את הסביבה וכיצד להריץ תוכנית ב-Debug Mode, לצורך ניפוי שגיאות ריצה.

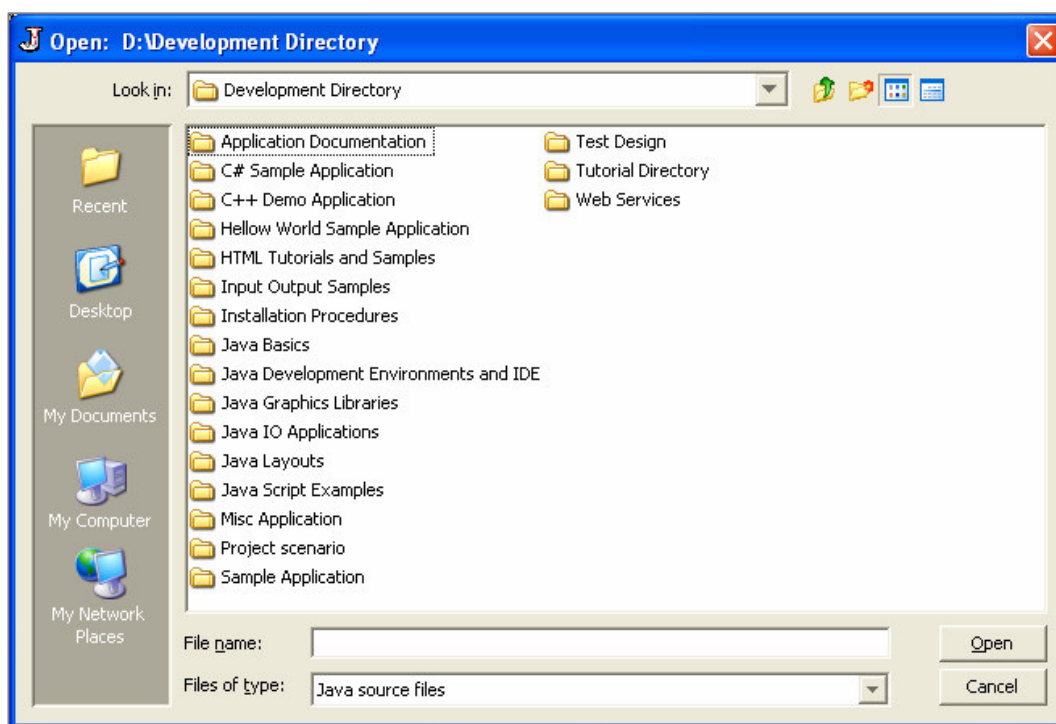
5.1. כיצד להפעיל את DrJava ?

הפעלת **DrJava** היא פשוטה וקלה. כל שנדרש הוא ללחוץ לחיצה כפולה (Double Click) על תוכנית ההפעלה של **DrJava**, המיוצגת על ידי הצלמית (Icon) שצורתה כאות "J". ניתן לשנות את שם התוכנית, למשל ל-drjava.exe (השם שבו אנו נשתמש) ולהגדיר קיצור דרך לתוכנית זו.



5.2. פתיחת ויצירת קבצים

כדי ליצור קובץ חדש בחרו New מתפריט File או לחצו על כפתור New בסרגל הכלים. קובץ חדש נוצר עבורכם וניתן להתחיל להקליד את תוכנו. לפתיחת קובץ קיים בחרו Open מתפריט File או לחצו על כפתור Open בסרגל הכלים. ייפתח חלון בחירת קובץ. שם התיקיה מופיע בכותרת חלון הבחירה וניתן לנווט בין התיקיות כדי להגיע לקובץ הנדרש בתיקיה המתאימה (ראו תמונה 8).

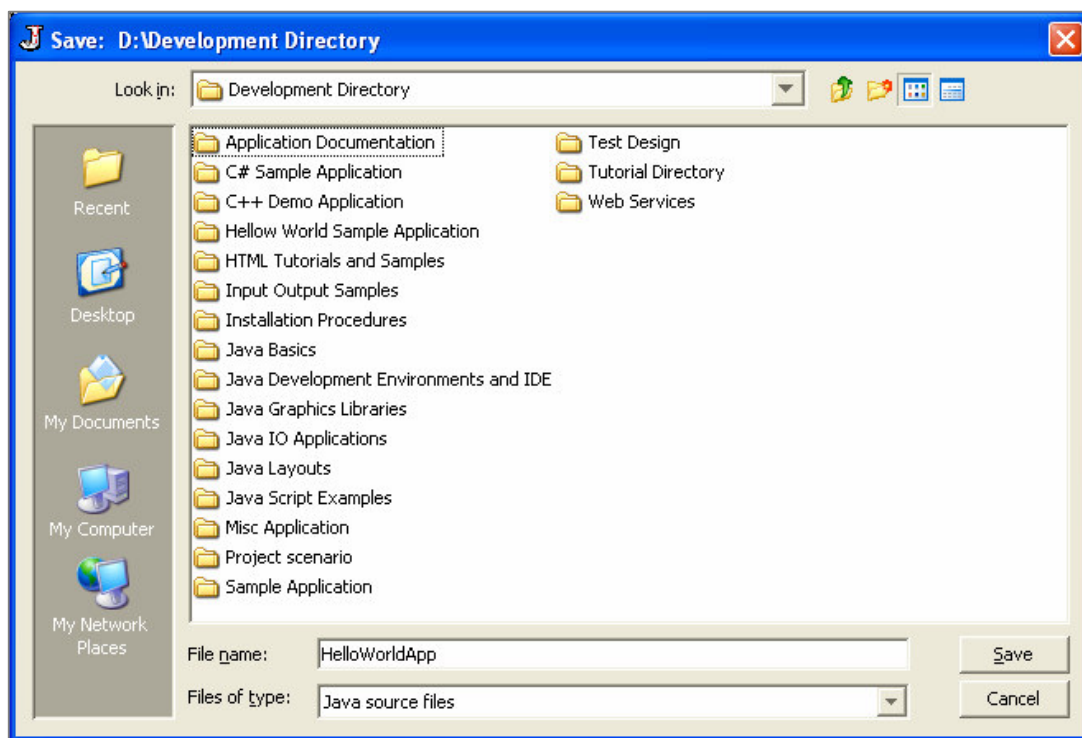


תמונה מס. 8

ברגע שיצרנו קובץ חדש או פתחנו קובץ קיים, שם הקובץ יצורף לרשימת הקבצים שנמצאת בחלון רשימת הקבצים וטקסט הקובץ יופיע בחלון העריכה הראשי. כאשר נבחר קובץ אחר מתוך רשימת הקבצים, יסומן הקובץ כקובץ עכשווי ותוכנו יוצג בחלון העריכה הראשי.

5.3. שמירת קבצים

כדי לשמור קובץ בחרו Save מתפריט File או לחצו על כפתור Save בסרגל הכלים. אם הקובץ כבר נשמר בעבר, תעודכן גרסת הקובץ השמור. אם הקובץ מעולם לא נשמר, תישאלו היכן ברצונכם לשמור קובץ זה. תוצג בפניכם תיבת דו-שיח, ושם התיקיה הנוכחית יוצג בכותרתה. מתיקיה זו, ניתן לנווט לתיקיה הרצויה כדי לשמור את הקובץ במקומו הרצוי. ניתן לשמור את הקובץ תחת שם שונה על ידי בחירת האפשרות Save As (שמירה בשם...). בתפריט File או לשמור את כל הקבצים הפתוחים על ידי בחירת האפשרות Save All בתפריט File (ראו תמונה 9).

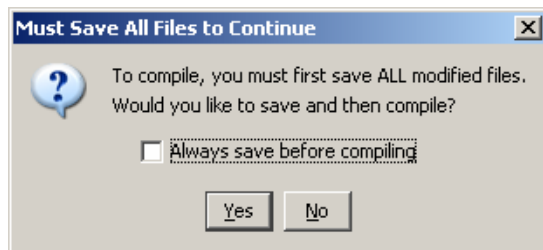


תמונה 9

5.4. הידור (קומפילציה) של קובץ

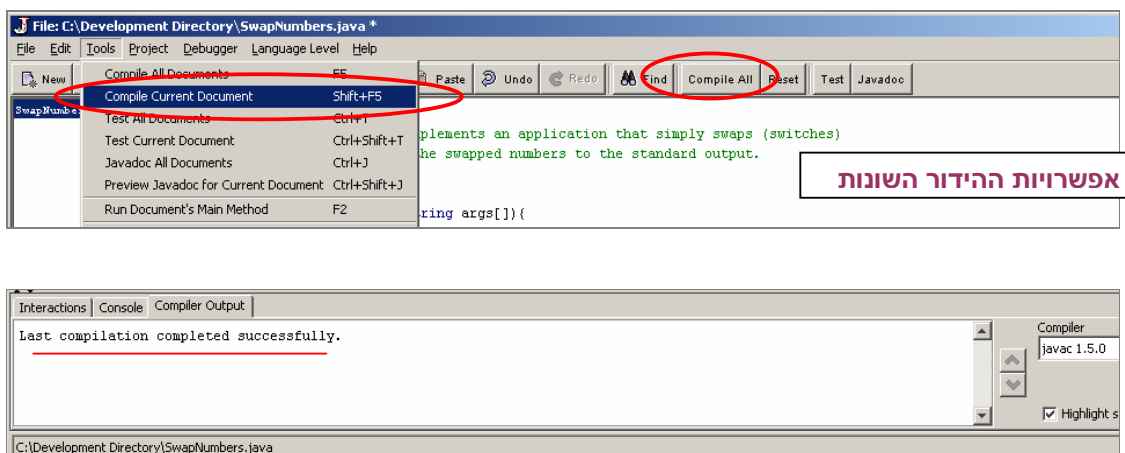
ניתן להדר (לקמפל) קובץ בודד או את כל הקבצים הפתוחים בנקודת זמן מסוימת. כדי להדר את הקובץ הנוכחי (הפעיל), יש לבחור את האופציה Compile Current Document בתפריט Tools (קיצור דרך Shift+F5). כדי להדר את כל הקבצים הפתוחים יש ללחוץ על הכפתור Compile All Documents או לבחור את האופציה Compile All בתפריט Tools (קיצור דרך F5). קובץ העבודה חייב להיות שמור על הדיסק (אופציית Save בתפריט Files) כולל כל השינויים.

האחרונים שבוצעו, בטרם ניתן יהיה להדרו. אם הקובץ אינו שמור, תתקבל הודעת השגיאה המופיעה בתמונה 10.



תמונה 10

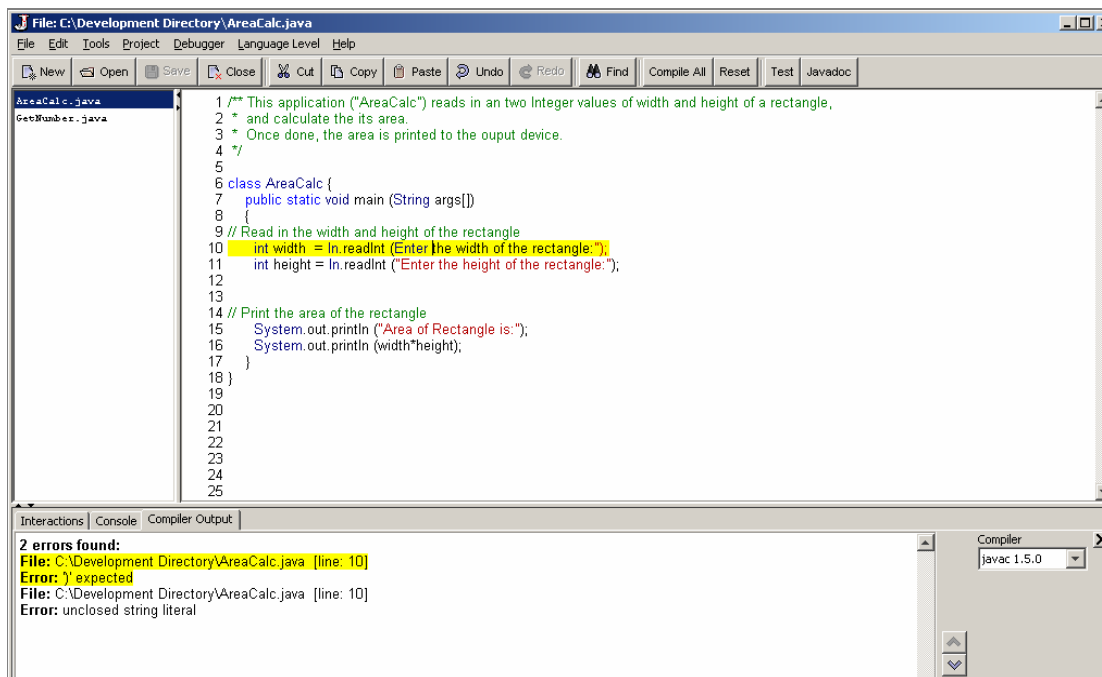
כאשר תהליך ההידור מסתיים מוצגות תוצאותיו בחלון פלט המהדר (Compiler Output). כאשר תהליך ההידור מצליח (כלומר, מסתיים ללא שגיאות הידור) מתקבלת ההודעה הבאה בחלון פלט ההידור: Last compilation completed successfully. (ראו תמונה 11, בחלקה התחתון).



תמונה 11

כאשר תהליך ההידור נכשל, מוצגת רשימת השגיאות. אם בתהליך ההידור נמצאו שגיאות (Errors) וגם אזהרות (Warnings), מוצגות שתי הרשימות אחת אחרי השנייה, כאשר רשימת השגיאות מופיעה ראשונה. נוכל ללחוץ באמצעות העכבר על שורת שגיאה בחלון פלט ההידור, ואז תסומן בצבע צהוב השורה השגויה בקוד המקור. ניתן להשתמש בחיצים (מעלה ומטה) כדי לנווט בין הודעות השגיאה. בכל שלב תודגש בצהוב השגיאה בחלון פלט ההידור ובמקביל יודגשו בחלון העריכה הראשי, גם כן בצהוב, השורות שגרמו להודעת השגיאה. אלה השורות שעלינו לבדוק כדי לזהות את מקור השגיאה ולתקנה (ראו תמונה 12).

הערה: ניתן לשנות את צבע הסימון מצהוב לצבע אחר, על ידי הגדרת הצבע המתאים בקובץ ההגדרות (הקונפיגורציה), כפי שמוסבר בסעיף 7.2.

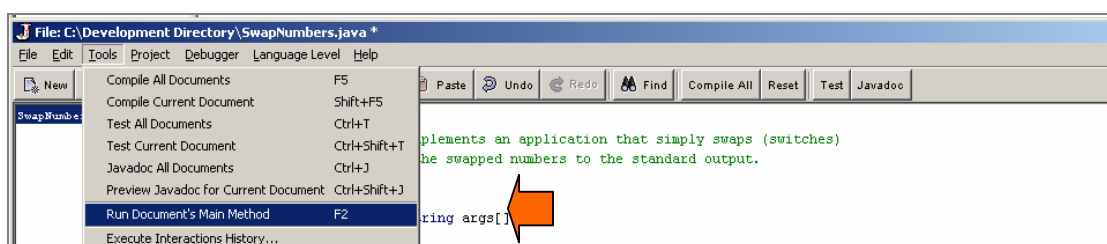


תמונה 12

5.5. הרצת תוכנית

בסביבת **Dr.Java** ניתן להריץ תוכנית Java רק לאחר שעברה בהצלחה את תהליך ההידור, ונוצר קובץ עם סיומת [.class]. בתיקייה המתאימה.

כדי להריץ את התוכנית, יש לבחור באופציה Run Document's Main Method (קיצור דרך F2) בתפריט Tools (ראו תמונה 13). פעולה זו תטען את הקובץ הנוכחי ותריץ את הפעולה הראשית שלו. אין כפתור בסרגל הכלים שמייצג פעולה זו, ולכן מומלץ להשתמש במקש F2.

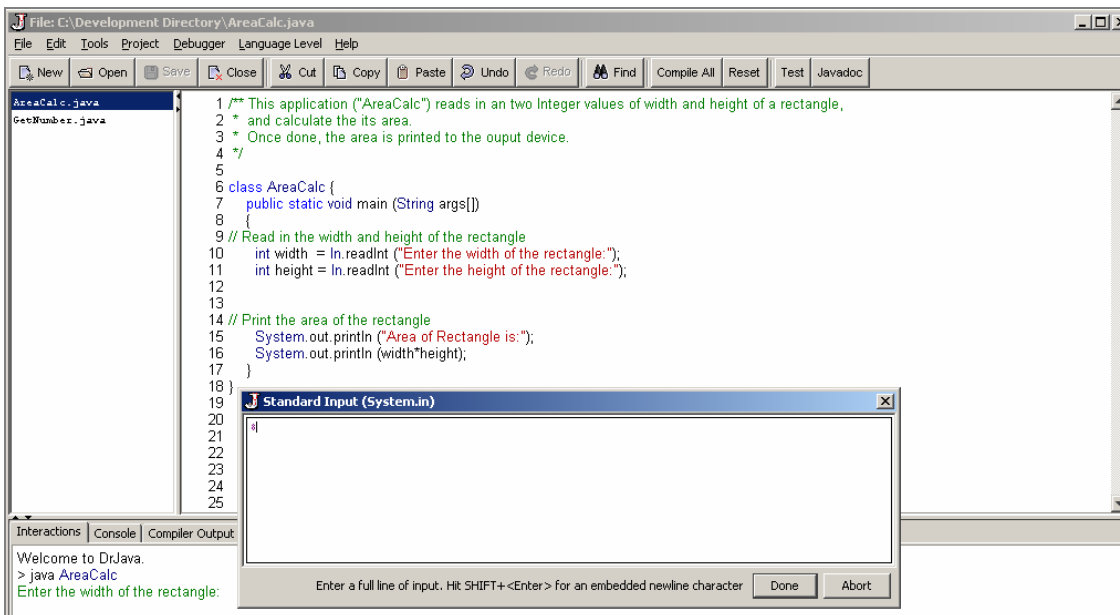


תמונה 13

שימו: ♥ תמיד תורץ הגרסה התקינה האחרונה של התוכנית. כלומר, אם ערכנו שינויים בקובץ הנוכחי אך שינויים אלה לא עברו בהצלחה את תהליך ההידור, וקיים (בעקבות הידור קודם) קובץ ישן בשם זה עם סיומת [.class], יורץ הקובץ הישן ותוצג הודעה על חוסר תאימות.

במקרים רבים, במהלך הרצת תוכנית נדרש המשתמש להזין קלט לתוכנית (הנושא יורחב בסעיף 5.7). למשל, אם נבחן את התוכנית שמופיעה בתמונה 14 (חישוב שטח מלבן) נראה שהתוכנית

מדפיסה הנחיות למשתמש (איזה קלט עליו להקליד, במקרה זה גובה ורוחב מלבן). ההנחיות מוצגות בחלון הדו-שיח, בעוד שאת הקלט עצמו יש להקליד בחלון הקלט הסטנדרטי:



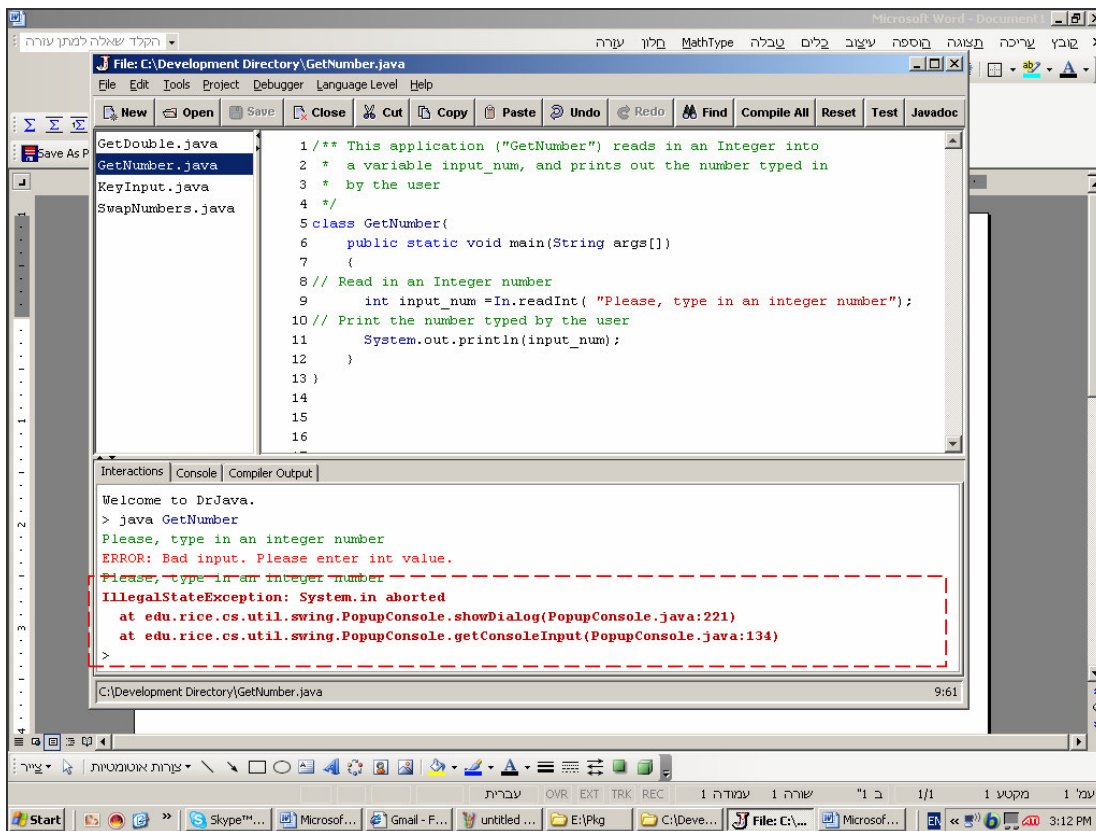
תמונה 14

אם נרצה להפסיק את הרצת התוכנית, גם בזמן המתנתה לקלט, נוכל להשתמש בכפתור Abort (ראו תמונה 15א):



תמונה 15א'

הלחיצה על כפתור זה מפסיקה את הריצה, וחלון הדו-שיח יראה אוסף של הודעות. אין צורך להתרגש, רק להבין שהן נובעות מהפסקה פתאומית של תהליך ההרצה (ראו המסגרת המלבנית המקווקוות בתמונה 15ב').



תמונה 15'

5.6. חלון הדו-שיח (Interaction Pane)

סביבת **DrJava** מציעה את חלון הדו-שיח המאפשר למשתמש להקליד ולערוך הוראות וביטויים ב-Java. זו צורת עבודה שימושית ונוחה מאוד הן למשתמש המתחיל, שלא צריך לשם כך להכיר את כל הכללים של כתיבת תוכנית בשפת Java, והן למשתמש המנוסה. מההיבט הלימודי, חלון זה מאפשר להתנסות בהוראות שפת הפיתוח בצורה קלה יחסית. אין צורך להכיר את תחביר השפה במלואו כדי ללמוד ולחוש באופן מעשי את יכולות השפה.

אז כיצד נשתמש? חלון הדו-שיח תומך בביצוע של כל הוראות Java חוקית, כולל חישוב של ביטויים. לכן, נוכל להצהיר בו על משתנים, כפי שהיינו עושים בעת כתיבת תוכנית מלאה, להשים בהם ערכים, ולכתוב הוראות אחרות לביצוע בשפה. תוצאת הביצוע של כל הוראה שלא נכתב בסופה הסימן נקודה-פסיק (;) תוצג בחלון הדו-שיח, בעוד שהוראה שנכתב בסופה הסימן נקודה-פסיק, תתבצע, אך תוצאת ביצועה לא תוצג. כל תוצאת ביצוע המוצגת בחלון הדו-שיח, מוצגת בצבע ירוק (וגם בחלון ה-Console). הודעות שגיאה של המערכת מוצגות בצבע אדום (צבעים אלה הם צבעי ברירת המחדל וניתן לשנותם).

הנה כמה דוגמאות לשימוש בחלון הדו-שיח:

1. כתיבת תוכנית הדוגמה הראשונה, HelloWorldApp, בחלון הדו-שיח (ראו תמונה 16) מצריכה שורה אחת בלבד.

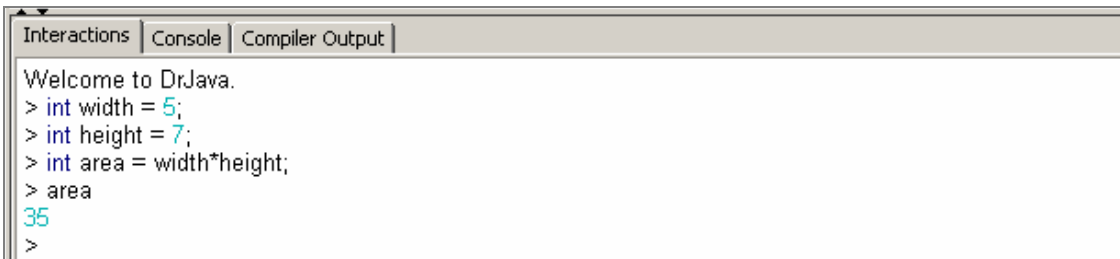


```
Interactions | Console | Compiler Output |
Welcome to DrJava.
> System.out.println ("Hello World");

Hello World
>
D:\Development Directory\HelloWorldApp.java 1:0
```

תמונה 16

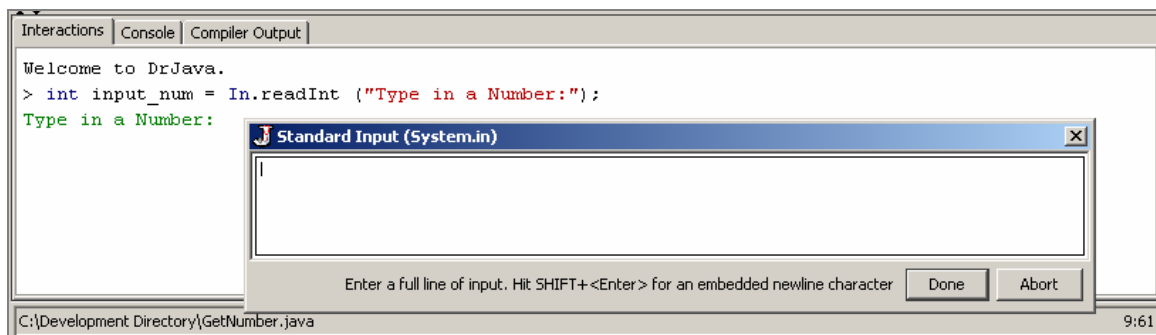
2. דוגמה מורכבת יותר ובעלת אופי מעט שונה, שמאפשרת להבין את יכולות חלון הדו-שיח מוצגת בתמונה 17. נכתבה סדרה של שלוש הוראות ב-java (לחישוב שטח מלבן) שמסתיימות בנקודה-פסיק. עבור שתי ההוראות הראשונה, התצוגה אינה משתנה, אך ההוראה מתבצעת. בפעולה השלישית מתבצע החישוב, אך התוצאה תוצג רק כאשר נבקש זאת באופן מפורש על ידי הקשת שם המשתנה:



```
Interactions | Console | Compiler Output |
Welcome to DrJava.
> int width = 5;
> int height = 7;
> int area = width*height;
> area
35
>
```

תמונה 17

3. נראה דוגמה נוספת, ובה נכתבת בחלון הדו-שיח תוכנית המבקשת קלט מהמשתמש. בעקבות ביצוע פעולת הקלט יופיע חלון הקלט הסטנדרטי של **DrJava**, שכותרתו: Standard Input (ראו תמונה 18). ההנחיה הנלווית לפעולת הקלט מופיעה בחלון הדו-שיח.



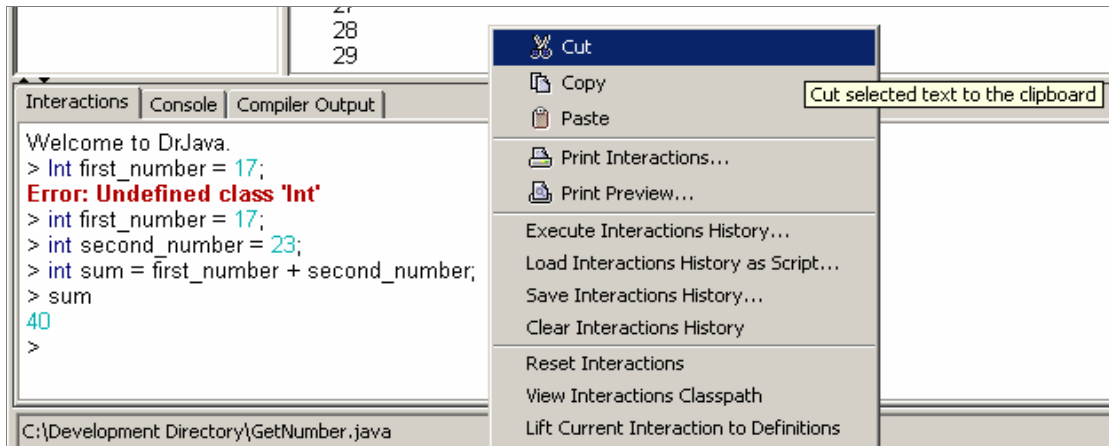
```
Interactions | Console | Compiler Output |
Welcome to DrJava.
> int input_num = In.readInt ("Type in a Number:");
Type in a Number:
Standard Input (System.in)
Enter a full line of input. Hit SHIFT+<Enter> for an embedded newline character
Done Abort
C:\Development Directory\GetNumber.java 9:61
```

תמונה 18

- אם נקיש קלט שגוי, למשל את האות A במקום מספר שלם, תתקבל הודעת השגיאה בחלון הדו-שיח. (וגם בחלון הקונסול – Console).

5.6.1 פעולות שכיחות בחלון הדו-שיח

ישנן לא מעט פעולות שעושות את תהליך העבודה עם חלון הדו-שיח לפשוט, קצר ויעיל יותר. פעולות אלה מופעלות באחת משתי הצורות: מתפריט Tools או על ידי הקשה על הכפתור הימני של העכבר בחלון הדו-שיח, הקשה הגורמת להופעת התפריט המוצג בתמונה 19.

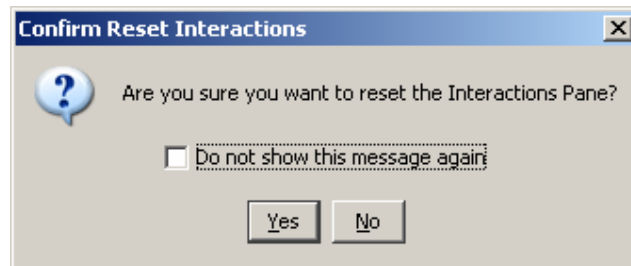


תמונה 19

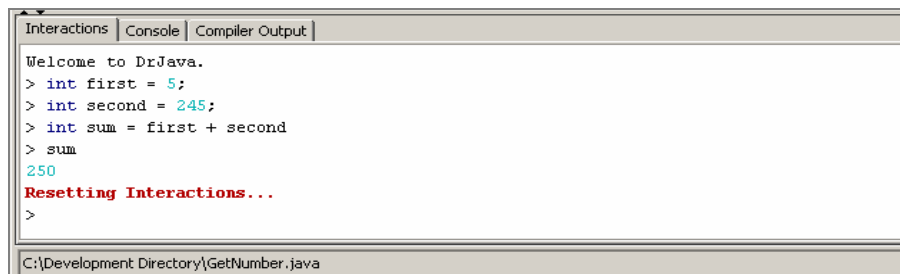
הפעולות השכיחות של העתקה, גזירה והדבקה (Copy, Cut, Paste) הן בעלות אותה משמעות בכלל יישום חלונות אחר.

• איפוס (ניקוי) הדו-שיח: Reset Interactions

איפוס הדו-שיח מאפשר התחלה נקייה, מנקה משתנים ומסיים פעולה של כל פעולה או פעולות שרצו עד נקודת זמן זו. כדי לנקות (למחוק את תוכנו) של הדו-שיח יש לאשר פעולה זו, על ידי לחיצה על מקש Yes. ניתן למנוע הופעת חלון זה בפעולות איפוס עתידיות ואז ברירת המחדל היא שתוכן הדו-שיח יימחק ללא צורך באישור של המשתמש (ראו תמונה 20). בזמן מחיקת הדו-שיח (האורכת כמה שניות) תופיע הודעה על ביצוע האיפוס בחלון הדו-שיח עצמו (ראו תמונה 21).



תמונה 20



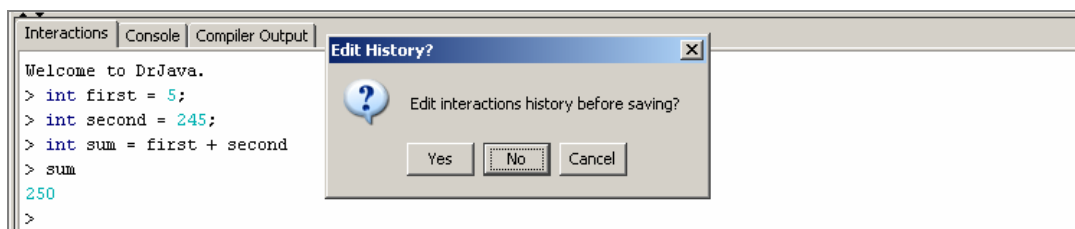
תמונה 21

• שמירת ההיסטוריה של חלון הדו-שיח: Save Interactions History

אופציה זו מאפשרת לשמור את תוכן הדו-שיח בקובץ חיצוני. היא גם מאפשרת עריכת התוכן, בחלון נפרד, לפני ביצוע השמירה. חלון ההיסטוריה מוגבל ל-500 פקודות דו-שיח, אך מספר זה ניתן לשינוי (תמונה 22).

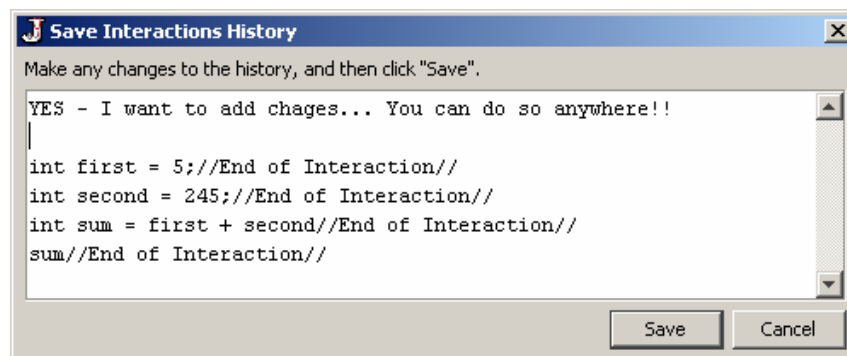
השמירה מאפשרת טעינה של קובץ ההיסטוריה בשלב מאוחר יותר (כפי שיוסבר בהמשך) והיא בדרך כלל מתאימה לביצוע סדרות של פעולות שחוזרות על עצמן. לאחר שקובץ ההיסטוריה נטען אפשר להריצו.

תמונה 22 מראה כיצד ניתן לבחור בעריכת קובץ לפני שמירתו.



תמונה 22

אם נענה Yes, ייפתח לנו חלון עריכה בו נוכל לבצע שינויים או תוספות (תמונה 23), כמו למשל התוספות שאני הכנסתי. בעזרת תוספות כאלו אפשר לתעד את הדו-שיח, כדי להקל על שימוש עתידי בו.

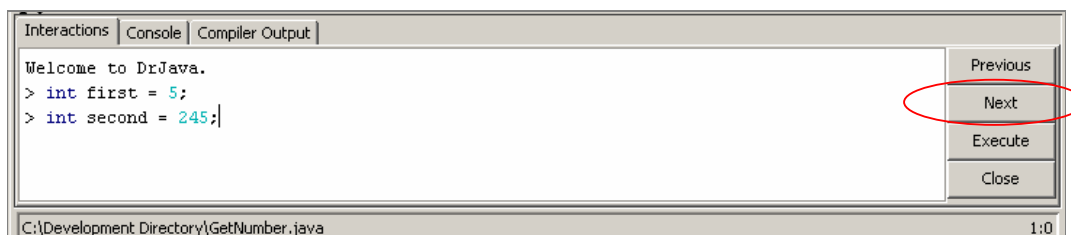


תמונה 23

אם נענה No (כלומר, איננו רוצים לערוך את הקובץ) תיפתח תיבת דו-שיח שתאפשר שמירת קובץ ההיסטוריה של הדו-שיח (עם סיומת hist). כאמור, שמירה זו מאפשרת הפעלה והרצה עתידית לפי הצורך.

- **טעינת היסטוריה של דו-שיח: Load Interactions History as a Script**

טעינת ההיסטוריה מאפשרת קריאת תוכן קובץ שנשמר (להזכירכם, עם סיומת hist). אל תוך חלון הדו-שיח והפעלתו פקודה אחת פקודה (כלומר, זו אינה טעינה של כל פקודות הקובץ בבת אחת) על ידי כפתורי Previous ו-Next (ראו תמונה 24). כפתור Execute מבצע את השורה.



תמונה 24

- **הרצת היסטוריה של דו-שיח: Execute Interactions History...**

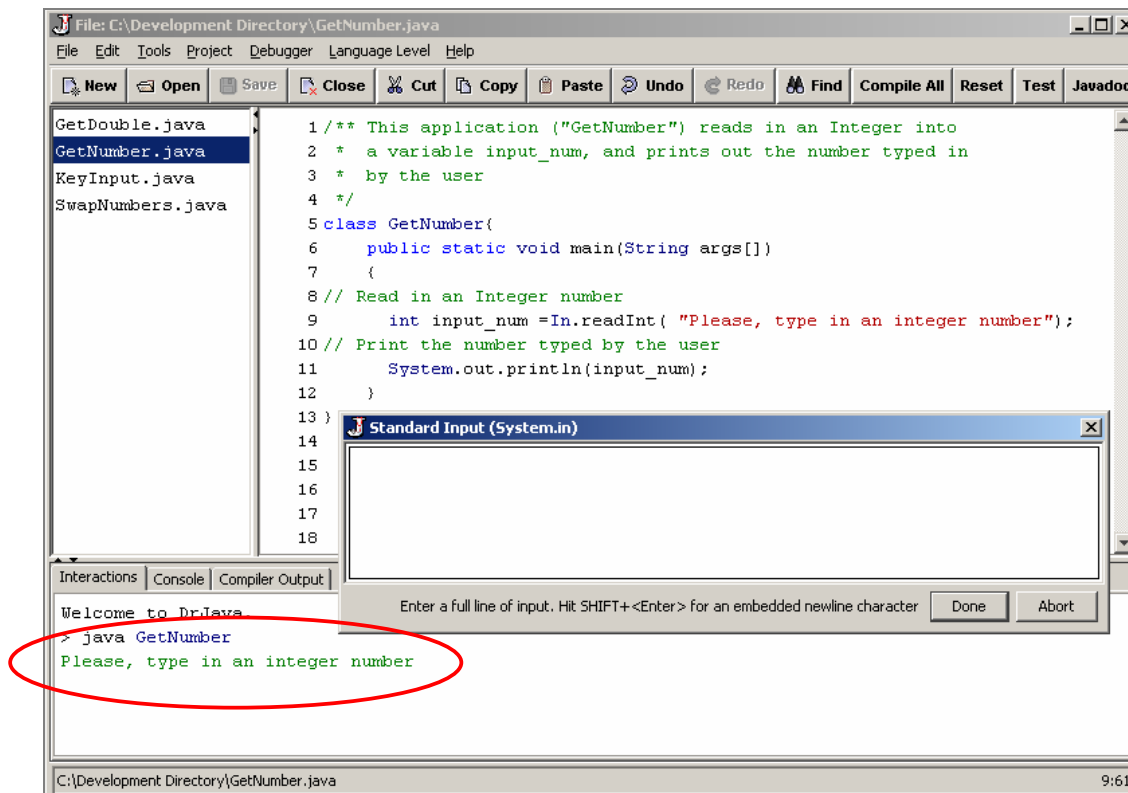
פעולה זאת מאפשרת פתיחת קובץ היסטוריה שנשמר קודם לכן והרצתו. כלומר, תוכנו של הקובץ הופך להיות פעיל בחלון הדו-שיח. פעולה זו מתאימה לצורך אתחול דו-שיח.

- ◆ **העתקת תוכן הדו-שיח: Lift current Interactions to Definitions**

אופציה שימושית מאד, המאפשרת לנו לנסות הוראות ב-Java בחלון הדו-שיח, ובסיום ההתנסות והבדיקה להעביר את הקוד מחלון הדו-שיח לחלון העריכה כחלק מקוד המקור של התוכנית.

5.6.2 הרצת הפעולה הראשית בחלון הדו-שיח

לנוחות המשתמש, תומכת סביבת **DrJava** בהרצת הפעולה הראשית (Main) בתוך חלון הדו-שיח. בדוגמה המוצגת בתמונה 25 אנו מריצים את התוכנית GetNumber בחלון הדו-שיח על ידי כתיבת המילה Java ואז שם התוכנית (שימו לב שמשפט הקלט מופיע בחלון הדו-שיח):



תמונה 25

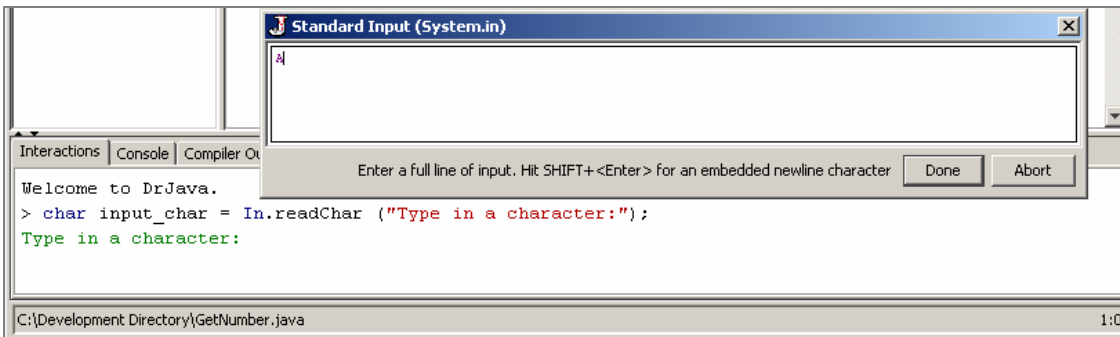
5.7. הזנת קלט

פרק זה יבהיר לנו הלומדים, את צורת העבודה עם קלט בסביבת **DrJava**, באמצעות מחלקת הקלט `In`, המשמשת אותנו במהלך לימוד "יסודות מדעי המחשב". יש לוודא שקובץ ההגדרות מכיל התייחסות למחלקת הקלט הנדרשת, כפי שמוסבר בסעיף 7.2, תצורת סביבת העבודה. כפי שצוין קודם לכן, בסביבת **DrJava** פעולת הזנת הקלט, מתבצעת בחלון נפרד, חלון הקלט הסטנדרטי, `Standard Input`.

פעולות הקלט בהן נשתמש הן `readInt`, לקליטת מספר שלם, `readDouble`, לקליטת מספר ממשי, `readChar`, לקליטת תו בודד, ו-`readString`, לקליטת מחרוזת.

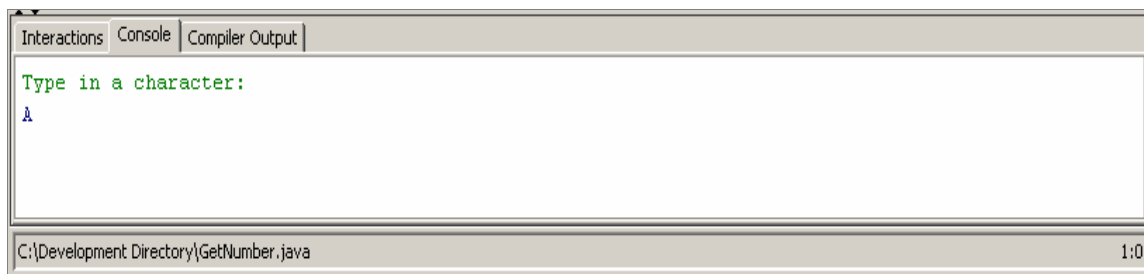
ניתן להפעיל את הפעולות האלו בשתי צורות:

1. אפשר לכתוב בחלון הדו-שיח הוראות המשתמשות בפעולות הקלט. הוראות אלה יפעילו את חלון הקלט הסטנדרטי של **DrJava** (ראו תמונה 26).



תמונה 26

את הקלט עצמו יש להקליד בחלון הקלט הסטנדרטי, והבקשה לביצוע (במקרה זה, 'Type in a character') תופיע בחלון הדו-שיח. לאחר הקלדת קלט מתאים, במקרה זה תו בודד, יש להקיש על מקש ה-Enter או על הכפתור Done בחלון הקלט. את כל תהליך הדו-שיח שבוצע ניתן לראות גם בחלון ה-Console בצורה כוללת, בנוסף לתצוגה בחלון הדו-שיח (שראינו לעיל). בתמונה 27 ניתן לראות את הנחיית הקלט והקלט עצמו בחלון ה-Console.



תמונה 27

שימו ♥ : בתחתית החלון מופיע שם הקובץ הפעיל (הפתוח) בנקודת הזמן בו הקלדנו נתונים בחלון הדו-שיח, אך קובץ זה אינו בהכרח קשור לפעולה העכשווית שמתבצעת.

2. ניתן כמובן לשלב פעולות קלט בתוך תוכנית מלאה ב-Java. תהליך הקלט שיתבצע יהיה זהה לתהליך שתואר לעיל, פרט לעובדה שהתהליך מתבצע כתוצאה מהרצת תוכנית ולא מהקלדת הוראות קוד בחלון הדו-שיח.

נדגים זאת על ידי תוכנית פשוטה, GetNumber, המשתמשת בפעולת הקלט של מספר שלם (readInt) שמטרתה לקלוט מספר שלם ולהדפיס את הקלט הנקלט מהמשתמש :

```

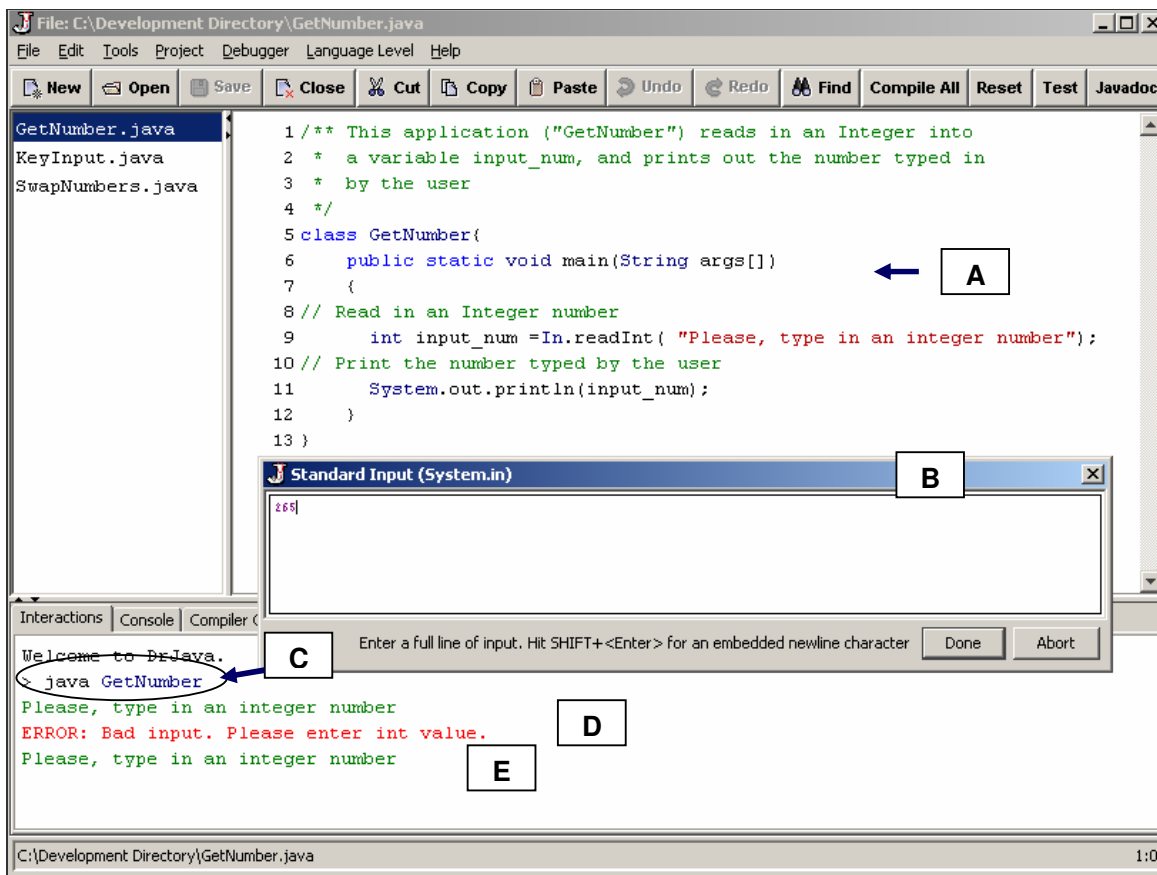
/** This application ("GetNumber") reads in an Integer saved
 * in a variable input_num and prints out the number typed in by
 * the user
 */

class GetNumber
{
    public static void main (String args[])
    {
        // Read in an Integer number
        int input_num = In.readInt ("type in an Integer Number");

        // Print the number typed by the user
        System.out.println (input_num);
    }
}

```

בתמונה מספר 28 ניתן לראות את השלב בהרצת התוכנית בו נפתח חלון הקלט הסטנדרטי. תמונה זו מדגימה גם כמה דברים נוספים, אליהם נתייחס מיד.



תמונה 28

(A) מראה את חלון העריכה וקוד התוכנית של GetNumber.java.

- (B) מראה את חלון הקלט הסטנדרטי של **DrJava** שמופיע עם הרצת התוכנית `GetNumber`.
- (C) שתי צורות להריץ את התוכנית: האחת, כתיבת המילה `Java` ואחריה שם התוכנית (במקרה זה `GetNumber`) בחלון הדו-שיח. האפשרות השנייה, לחיצה על מקש `F2` שמשמעותה הרצת הפעולה הראשית (`Main`), כלומר, הרצת התוכנית המלאה.
- (D) הודעת שגיאה על הקלדת ערך שגוי. למשל, הקלדת האות `A` כאשר התוכנית מצפה לקלט של מספר שלם. ההודעה מופיעה בחלון הדו-שיח וגם בחלון הקונסול.
- (E) הודעת ההנחיה לקלט שעל המשתמש להזין. הודעה זו מוצגת בחלון הדו-שיח, להבדיל מהקלט עצמו שמתבצע בחלון הקלט הסטנדרטי. ניתן לראות הקלדות אלה גם בחלון הקונסול.
- כדי להשלים את התמונה, נוכל לכתוב תוכניות דומות שעושות שימוש בפעולות הנוספות, `readDouble`, לקריאת מספר ממשי, `readChar`, לקריאת תו בודד ו-`readString`, לקריאת רצף של תווים. בכל מקרה נצטרך להזין קלט מתאים (מספר ממשי, תו בודד או רצף תווים, בהתאמה), ולסיים בהקשה על מקש `Enter` או על הכפתור `Done`. בשלב זה, מחלקת הקלט אינה תומכת בקלט בעברית.

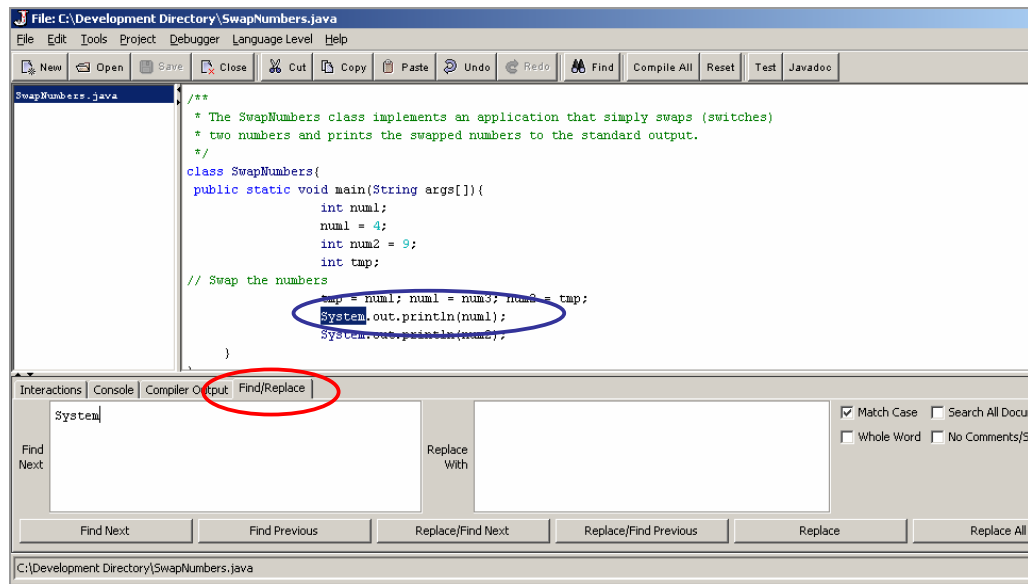
5.8. התמצאות ואפשרויות חיפוש

ישנן כמה אפשרויות חיפוש בסביבת **DrJava**. הן מייצגות דרכים שימושיות לחיפוש ומציאת טקסט מסוים בקובץ (בשני הכיוונים, קדימה ואחורה), כולל החלפתו בטקסט אחר, במידת הצורך.

האפשרויות הללו מוצעות תחת התפריט `Edit` ומאפשרות את החיפושים כדלהלן:

- מצא והחלף (`Find/Replace...`) – מקש קיצור `Ctrl-F`
- חפש את הבא (`Find Next`) – מקש קיצור `F3`
- חפש את הקודם (`Find Previous`) – מקש קיצור `Shift+F3`
- גש לשורה... (`Go to Line...`) – מקש קיצור `Ctrl+G`

בחירה של כל אחת מהאפשרויות הללו תוסיף לשונית נוספת בשם `Find/Replace`, לצד הלשוניות הקיימות בתחתית העורך (ראו תמונה 29):



תמונה 29

כעת יש להקיש את הטקסט המבוקש במלבן שצמוד לכותרת Find Next (בדוגמה לעיל הטקסט המבוקש הוא System). כדי להחליף בתוכנית טקסט זה בטקסט אחר, יש להקליד את הטקסט להחלפה במלבן הצמוד לכותרת Replace With, לבחור אחת מארבע האפשרויות בצד ימין, ואז, להקיש על אחד הכפתורים בתחתית הלשונית (למשל, Replace לביצוע החלפת הטקסט).

האפשרויות בצד ימין מנחות את החיפוש. אנו נזכיר את השתיים העיקריות מביניהן: **התאמה גם בגודל אות** (Match Case) – אם בתוכנית מופיעה המילה SYSTEM (באותיות גדולות) ואנו נבקש לחפש את הטקסט system (באותיות קטנות), החיפוש לא יביא אותנו אל המופע SYSTEM (וגם לא למופעים כמו System או sYstem), אלא אם אפשרות ה-Match Case כבויה. **התאמת מילה שלמה** (Whole Word) – יתבצע חיפוש רק אחר מילים התחומות ברווחים משני צידיהן, להבדיל מקטעי מילים (שאינם תחומים ברווחים משני הצדדים).

אם ברצוננו רק למצוא טקסט מסוים, נקליד את הטקסט לחיפוש במסגרת שלצד הכותרת Find Next, ואחר כך נלחץ על הכפתור המתאים על פי כיוון התנועה: Find Next לחיפוש המופע הבא של הטקסט (בכיוון קדימה, לכיוון סוף הקובץ) ו-Find Previous לחיפוש המופע הקודם (כלומר, אחורה, לכיוון תחילת הקובץ).

6. פונקציות מתקדמות

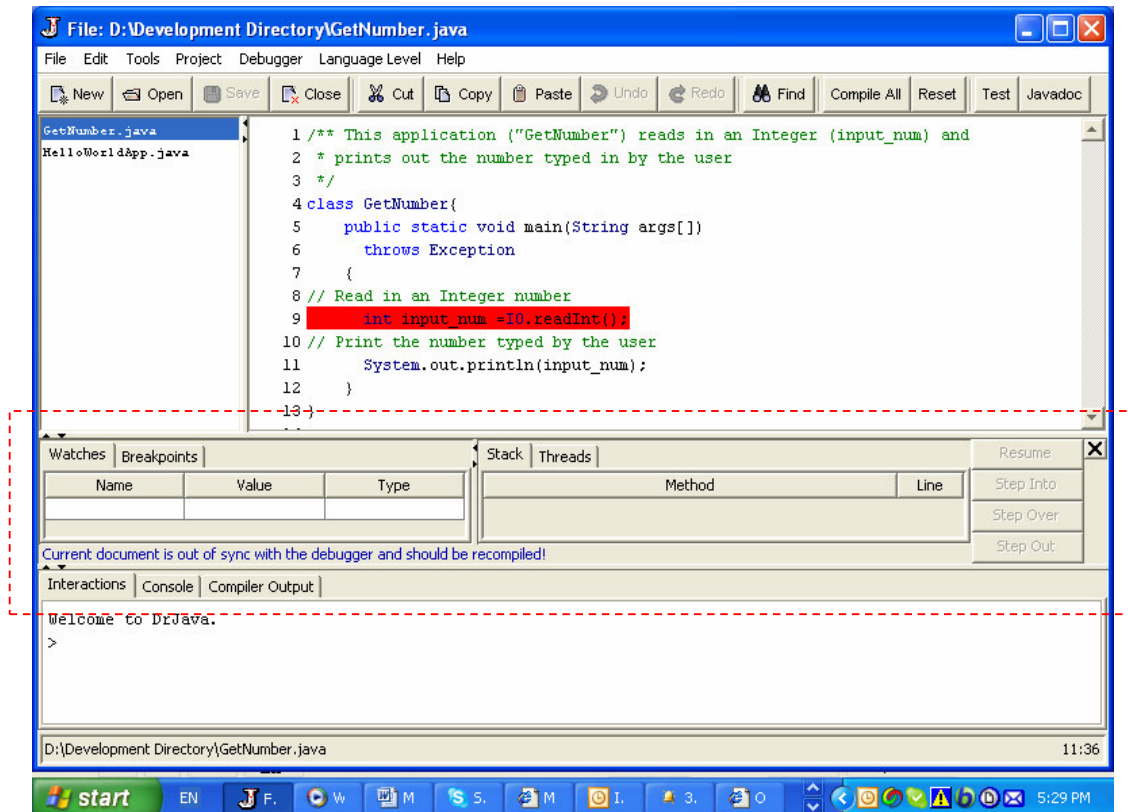
6.1 ניפוי שגיאות ריצה (Debug Mode)

סביבת **DrJava** מספקת כלים מתקדמים לאיתור שגיאות בזמן ריצה של תוכנית. איתור השגיאות מתבצע ע"י דו-שיח בחלון הדו-שיח, בכפוף להגדרות שהגדרנו טרם הרצת התוכנית. לניפוי השגיאות (debugging), יש להריץ את התוכנית באופן עבודה מיוחד, Debug Mode.

מה מאפשרת צורת הרצה זו? ניתן להגדיר נקודות עצירה (break points) בקוד התוכנית, ולהריץ את התוכנית עד לנקודת עצירה. כאשר הרצת התוכנית מגיעה לנקודת עצירה, היא מושהה, והמשתמש יכול לבחון את ערכי המשתנים בנקודה זו. המשתמש יכול לאפשר לתוכנית להמשיך את ריצתה עד נקודת העצירה הבאה או להמשיך בריצה מבוקרת, כלומר, התקדמות בקצב של שורה בכל פעם.

כיצד נשתמש ב-Debugger? יש לבחור באופציה Debug Mode בתפריט ה-Debugger. ייפתח חלון נוסף, בין חלון העריכה לחלון הדו-שיח, (ראו בתמונה 30 את החלון המסומן בקו אדום מקווקו), ובנוסף תתאפשר גישה לאופציות נוספות בתפריט ה-Debugger שיפורטו בהמשך.

שאר הלשוניות של החלונות שהכרנו עד כה זמינות גם בתצורת עבודה זו.



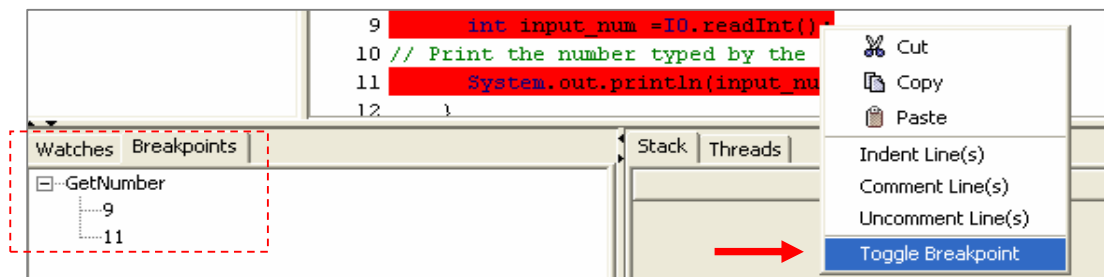
תמונה 30

שימו ♥ : אם נבצע שינויים בקוד התוכנית תוך כדי ריצה בתצורת Debug לא תהיה להם כל השפעה. כדי לבחון שינויים כאלה יש לשמור את קובץ התוכנית, להדר את התוכנית מחדש ולהריצה שוב בתצורת Debug. במקרה כזה, סביבת **DrJava** תתריע באמצעות הודעת אזהרה, כפי שניתן לראות בתמונה 30.

שינויים כאלה בזמן ריצה אינם מומלצים משום שהם גורמים לשינוי במספור השורות ולפיכך גם לחוסר תאימות בין השורות המופיעות בחלון העריכה לשורות המבוצעות. מצב כזה של חוסר התאימות מלווה בהודעה: *Current Document is out of sync with its class file*.

נקודות עצירה (Breakpoints):

לאחר שבחרנו באופציית ה-Debug Mode יש ביכולתנו להגדיר נקודות עצירה, כמעט בכל שורת קוד בחלון העריכה, על ידי בחירת באופציה *Toggle Breakpoint in Current Line*. אפשרות נוספת היא לבצע זאת בחלון העריכה על ידי שימוש במקש הימני של העכבר ובחירת האופציה *Toggle Breakpoint*. כאשר נגדיר שורת עצירה, היא תיצבע בצבע אדום ותתווסף לרשימת נקודות העצירה שמופיעה בלשונית ה-Breakpoints בחלון ה-Debugger, מתחת לחלון העריכה (ראו תמונה 31).



תמונה 31

בדוגמה לעיל, ניתן לראות בבירור נקודות עצירה בשורות 9 ו-11 במחלקה `GetNumber`.

כאשר הרצת התוכנית מגיעה לנקודת עצירה, השורה בה נמצאת נקודת העצירה צבועה בצבע כחול בהיר ומוצגת הודעה מתאימה בחלון הדו-שיח. לאחר מכן מופיע סמן (Prompt) בחלון הדו-שיח שיאפשר לנו "דו-שיח" עם התוכנית בנקודה זו.

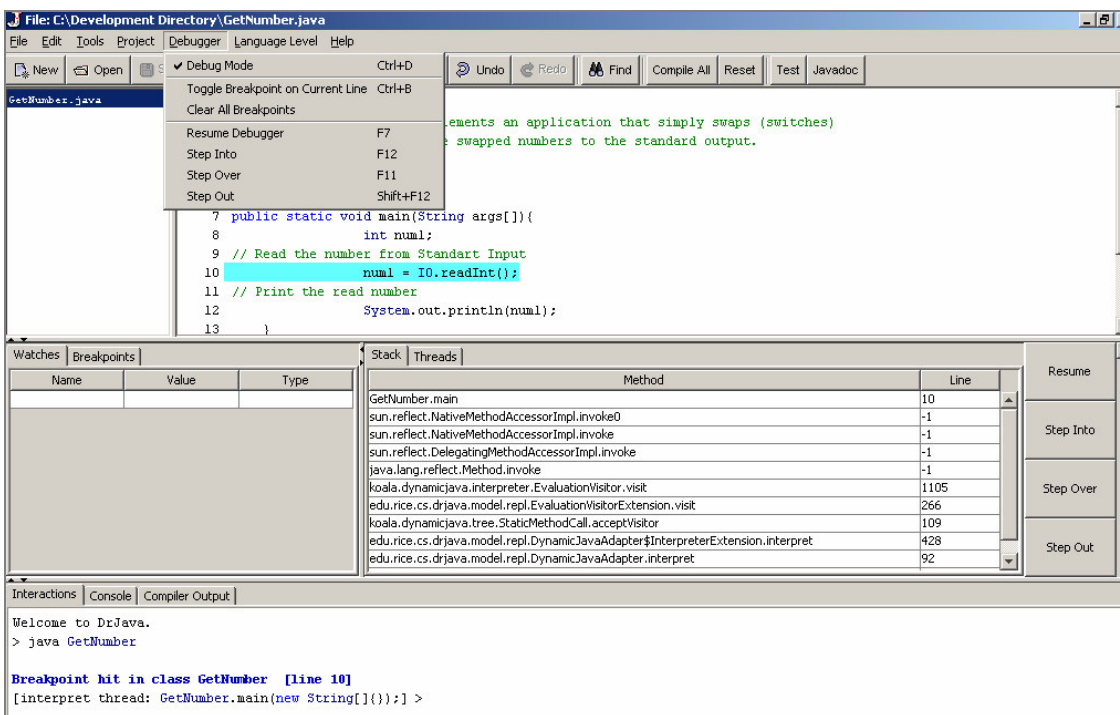
שימו ♥ : יש להגדיר נקודת עצירה בשורה שיש בה הוראת ביצוע. למשל, אין טעם להגדיר נקודת עצירה בשורה שיש בה רק הצהרה על משתנה, או סוגר/פותר מסולסל. בפרט, אין להגדיר נקודת עצירה בשורה ריקה או בשורת הערה. בזמן ריצה תוכנית לא תעצור בנקודות אלה, למרות שהן נצבעות באדום ונראות, לכאורה, נקודות עצירה "כשרות".

תמונה 32 מראה את **תצורת DrJava** בהגעה לנקודת עצירה. בנקודה זו התוכנית נמצאת במצב של *שהייה (Suspension)* וניתן לבחון את ערכי כל המשתנים. ניתן לבחון ערך של משתנה, להשים בו ערך חדש ולהמשיך את התוכנית (*Resume*) עם הערך החדש של המשתנה. נדגיש שוב כי אין לשנות שורות קוד בתוכנית בזמן ההשהיה.

מנקודת העצירה אליה הגענו ניתן להמשיך בשני אופנים :

- **חידוש הרצה - Resume :** פעולת Resume נגישה דרך תפריט ה-Debugger, בחירת האופציה Resume Debugger או מקש קיצור F7 (כפי שניתן לראות בתפריט עצמו). האופציה קיימת גם ככפתור בחלון ה-Debugger. הפעלתה תמשיך את ריצת התוכנית (עם ההשמות החדשות, אם הגדרנו כאלו) ותעצור בנקודת העצירה הבאה, אם ישנה כזאת.
- **התקדמות בצעדים – Stepping :** התקדמות בצעדים, שורה אחר שורה, שעבורה כמה פקודות מתאימות :
 - ♦ **Step Into –** פקודה זו תיכנס לפעולה שמופיעה בשורת הקוד בה עצרנו, גם אם הדבר כרוך במעבר לקובץ אחר בפעולה של מחלקה אחרת.
 - ♦ **Step Over –** פקודה זו תבצע את שורת הקוד הנוכחית ותעבור לשורת הקוד הבאה מבלי להיכנס לפעולה בשורת ההוראה עליה הפעלנו את הפקודה.
 - ♦ **Step Out –** פקודה זו תסיים את הרצתה של הפעולה הנוכחית אליה נכנסנו ותעצור בשורת הקוד הבאה לאחר השורה שהפעילה את הפעולה הנוכחית.

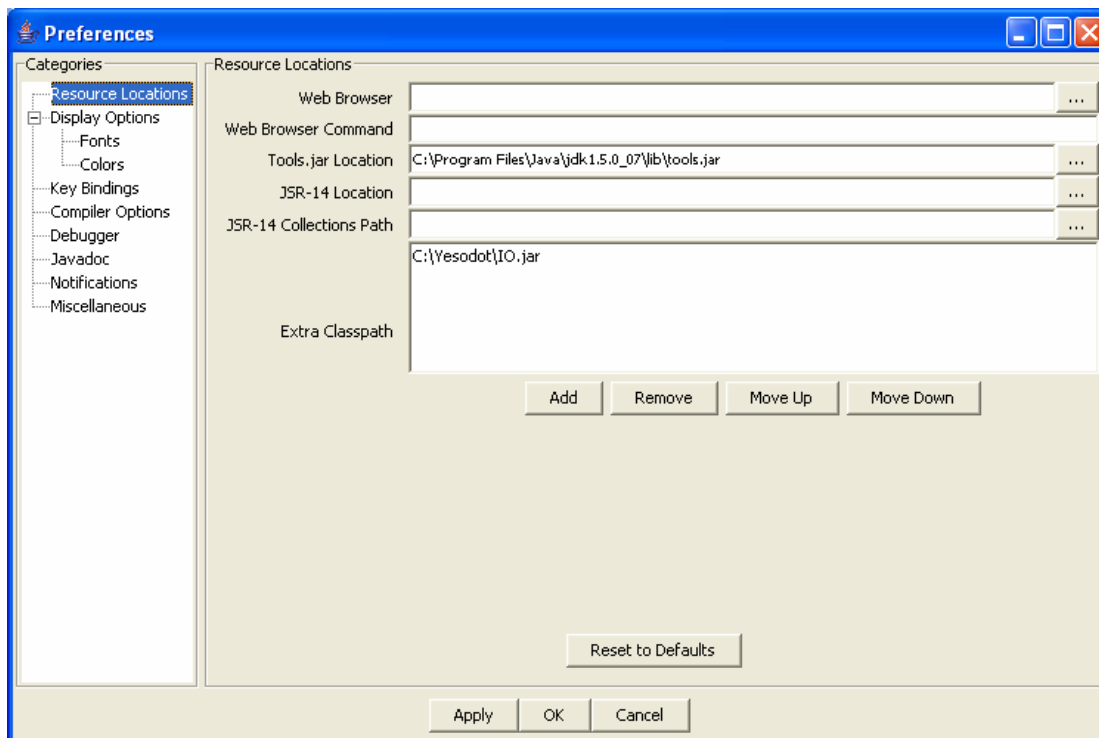
פקודות אלה ניתנות להפעלה דרך תפריט ה-Debugger, או על ידי לחיצה על אחד הכפתורים בחלון ה-Debugger שמתחת לחלון העריכה (ראו תמונה 32).



תמונה 32

6.2 תצורה של סביבת העבודה (Preference Definitions)

הגדרת סביבת העבודה מתבצעת ע"י בחירה באופציה Preferences בתפריט Edit. המסך המתקבל מוצג בתמונה 33.



תמונה 33

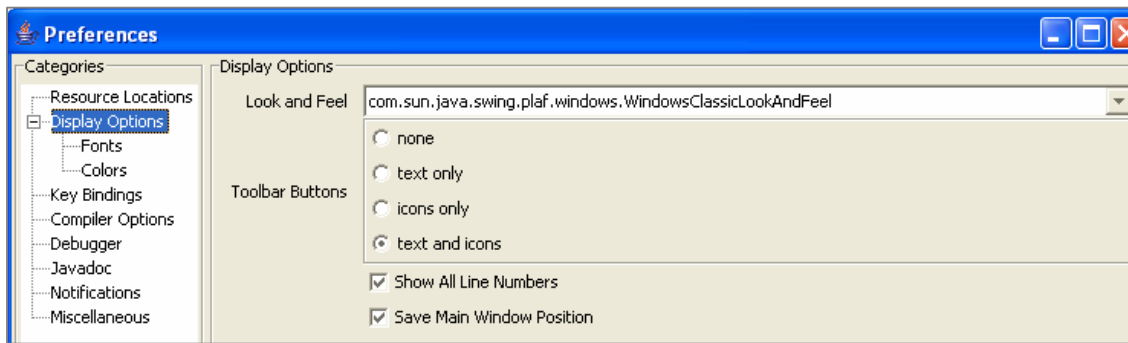
אפשרויות הגדרת תצורת סביבת העבודה של **DrJava** הינן רבות ומגוונות. אנו נתייחס לכמה נקודות בלבד. הנחיות ועוזרים נוספים ניתן למצוא בקובץ העזרה (Help) של התוכנה.

Resource Location - מיקום המשאבים:

מיקום הקובץ Tools.Jar נקבע כחלק מההתקנה. אם קובץ זה אינו מוגדר, או שמיקומו שגוי תהליך ההידור יכשל. יש להגדירו (החיפוש מתאפשר ע"י לחיצה על הכפתור שעליו שלוש נקודות (...)).
הגדרה זו מאפשרת גישה לספריות נוספות לשימוש בסביבת העבודה. למשל, מיקומה של ספריית הקלט מוגדר תחת תיקיית Yesodot בכוון C:. ברור שאם הספרייה אינה נמצאת במקומה, או שלא הוגדרה כלל, לא נוכל לכתוב תוכניות שכוללות פעולות השייכות אליה

Display Options - שליטה על תצוגה:

היכולת שלנו לשלוט על אלמנטים שונים בתצוגת סביבת העבודה מתאפשרת דרך Display Options, כפי שניתן לראות בתמונה מס. 34.



תמונה 34

נדגיש שלוש אפשרויות תצוגה בעלות משמעות:

א. Look and Feel – מראה סביבת העבודה, ארבע אפשרויות לבחירה

- Classic Metal – תצורה מתכתית
- Windows Classic – גרסת חלונות קלאסית
- Windows – תצורת חלונות רגילה
- Motif – תצורת Motif (יוניקס)

ב. Toolbar Buttons – סרגל כלים, ארבע אפשרויות לבחירה

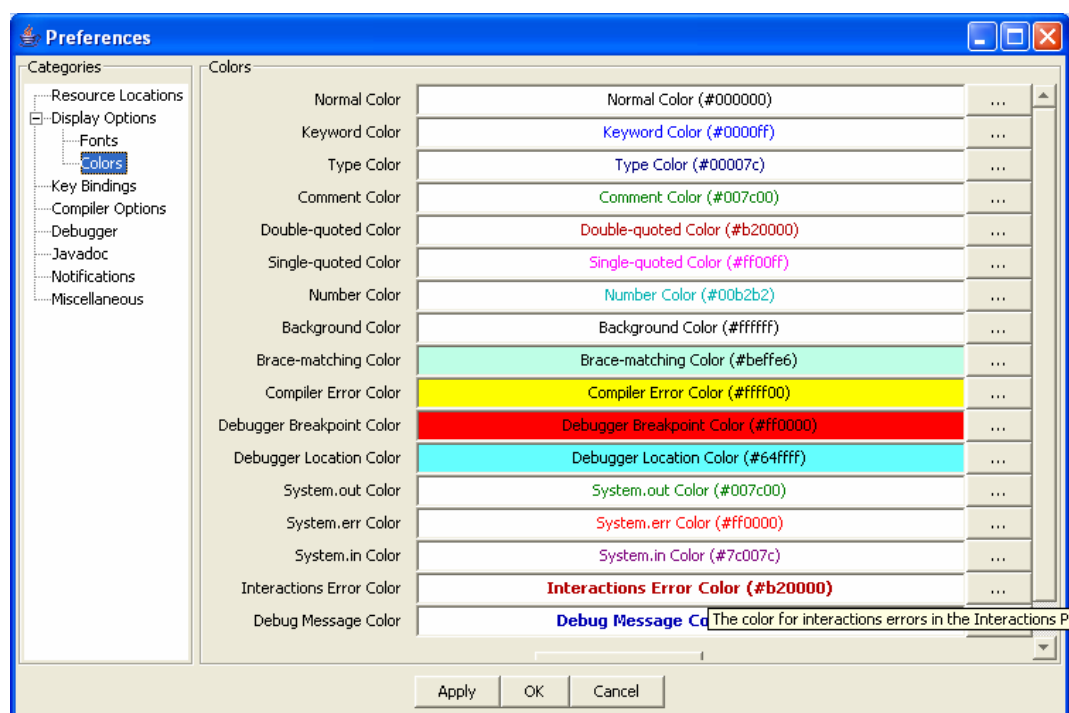
- None – ללא סרגל כלים
- Text Only – טקסט בלבד בכפתורי סרגל הכלים
- Icons Only – צלמיות בלבד בכפתורי סרגל הכלים
- Text and Icons – טקסט וצלמיות בכפתורי סרגל הכלים

ג. Show All Line Number – הראה או הסתר את מספר השורות בעורך. מומלץ לעבוד

כאשר אופציה זו מופעלת, לפחות בשלבי העבודה הראשונים.

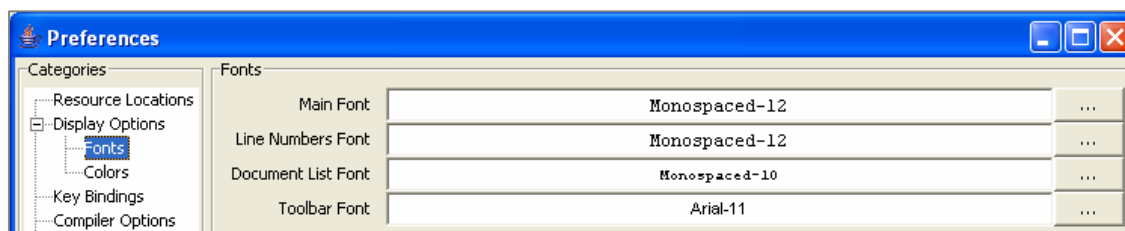
ישנן שתי אפשרויות נוספות שדרכן ניתן להגדיר את תצורת סביבת העבודה ויש להן תרומה מבחינת קריאות ובהירות התוכנית: הגדרת הגופנים השונים (Fonts) והגדרת צבעי טקסט העריכה, הודעות השגיאה, משתנים וכי' (Colors) (ראו תמונות 35 ו-36).

הגדרת צבעי עריכה:



תמונה 35

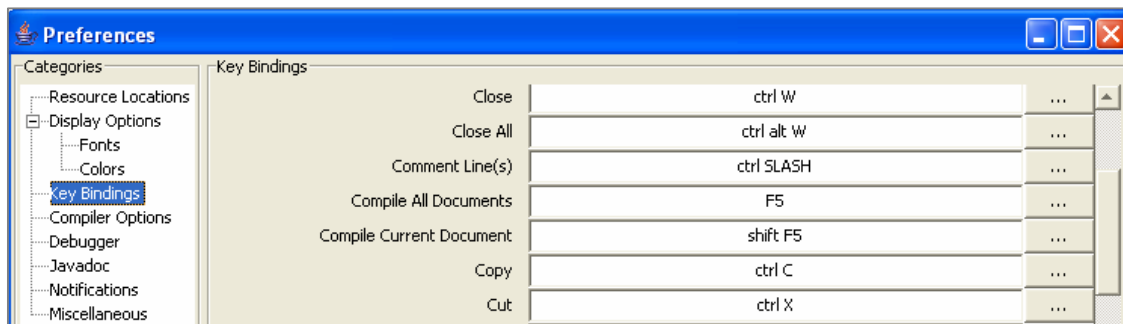
הגדרת גופנים:



תמונה 36

Key Bindings – מיפוי מקשים

תמונה 37 מראה כמה אפשרויות מתוך המגוון הקיים. למשל, המקש F5 משמש להידור הקבצים הפתוחים, ו-Shift F5 משמש להידור קובץ נוכחי. המקשים Ctrl X ו-Ctrl C משמשים לפעולות גזירה והעתקה בהתאמה. הרשימה ארוכה, ומאפשרת מיפוי מקשים בצורה מאד רחבה. מומלץ לעבור עליהם באופן עצמאי.



תמונה 37

נספחים

7.1 דרישות המערכת

סביבת **DrJava** דורשת גרסת התקנה של Java 2 v 1.3 Virtual Machine או גרסה מתקדמת יותר.
(הערה – ה-JDK ולא ה-JRE חייב להיות מותקן כדי לאפשר הידור (קומפילציה)). התקנת ה-JDK מכילה בתוכה התקנה של ה-JRE.
בכל פעם שמתעורר הצורך בהורדת גרסת JVM (Java Virtual Machine), מומלץ להוריד את גרסת Sun's JDK 1.4 עבור מערכת ההפעלה של Windows.
בנוסף, כדי לאפשר עבודה תקינה של **DrJava**, חייבים להיות מותקנים התקנים (דרייברים) של TCP/IP.
שטח הדיסק הפנוי הנדרש להתקנה – 264MB.

7.2 התקנת סביבת העבודה

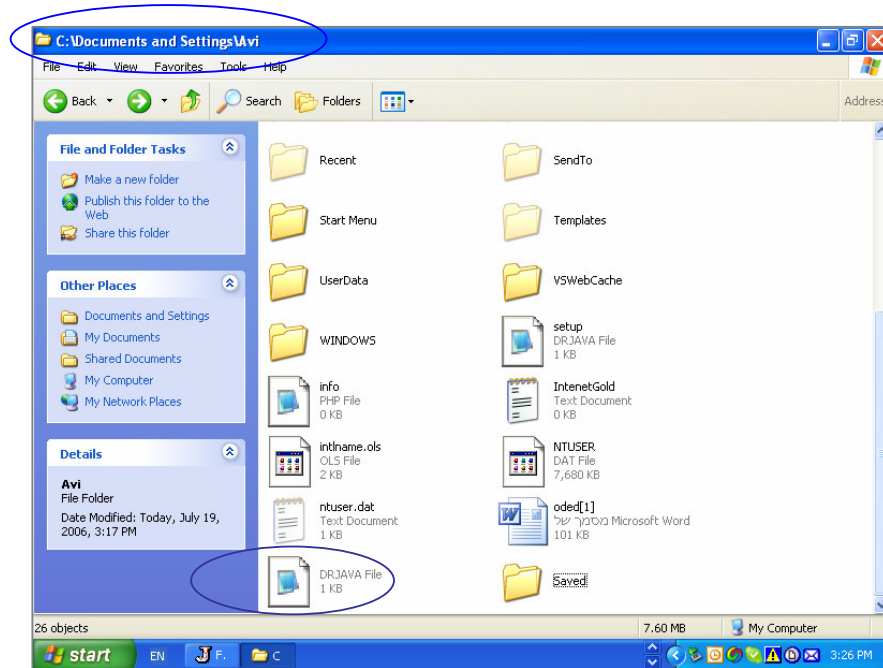
את סביבת העבודה אפשר להתקין מקובץ ההתקנה, שאותו ניתן להוריד מאתר קבוצת מדעי המחשב בחוג להוראת המדעים של אוניברסיטת ת"א, לפי ההנחיות בדף <http://www.tau.ac.il/~csedu/yesodot.html>.
כדי להפעיל את ההתקנה יש להוריד את הקובץ DJPackage.Zip, לשמור (לבצע Save) את הקובץ בתיקיה כלשהי בכונן במחשב שלכם, ואז :

(א) פתחו את הקובץ ובצעו פעולת Extract לתיקיית הכונן
(ב) לחצו לחיצה כפולה עם העכבר על קובץ SetupDJ.Exe
הערה: ניתן להפעיל את תכנית ההתקנה גם בפתיחת חלון DOS.

אלה השלבים המתבצעים בפעולת ההתקנה :

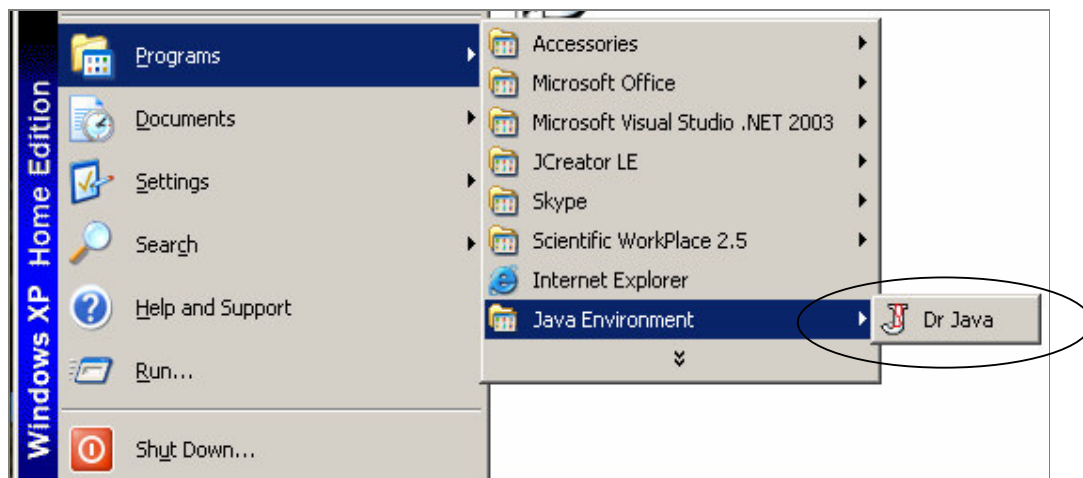
1. יצירת תיקיה בשם Yesodot תחת כונן C:.
2. העתקת קובץ ההפעלה של סביבת העבודה (DrJava.Exe) לתיקיה שנוצרה.
3. העתקת קובץ ספריית הקלט (In.Jar) לתיקיית יסודות - C:\Yesodot.
4. העתקת קובץ ההגדרות של סביבת העבודה של DrJava (שימו לב, לקובץ אין שם אלא סיומת drjava. בלבד) לתיקיית המשתמש הרשום (Logged In) **מתחת** לתיקיית Documents and Settings בכונן C:.

הנה דוגמה של מבנה התיקיה (תמונה 38) עבור משתמש בשם Avi :



תמונה 38

5. יצירת "צלמית הפעלה" - **Menu Entry** עבור הפעלת **DrJava**. הכניסה בתפריט נקראת **"Java Environment"** ותחת תיקייה זו נוסף קישור להפעלה, כפי שניתן לראות בתמונה 39 :



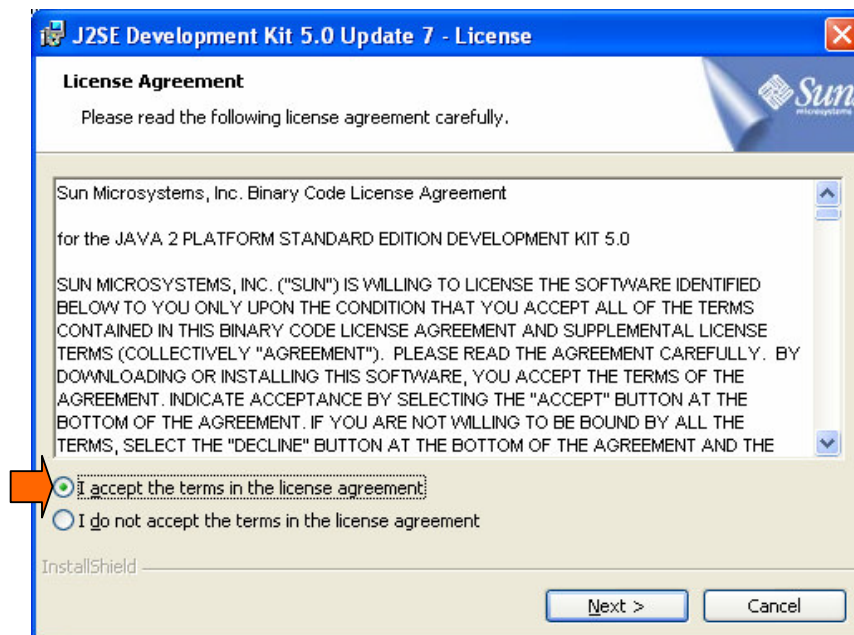
תמונה 39

6. הפעלת תוכנת ההתקנה של ה-JDK, שהועתקה בשלב קודם, על ידי לחיצה כפולה על הקובץ :

jdk-1_5_0_07-windows-i586p

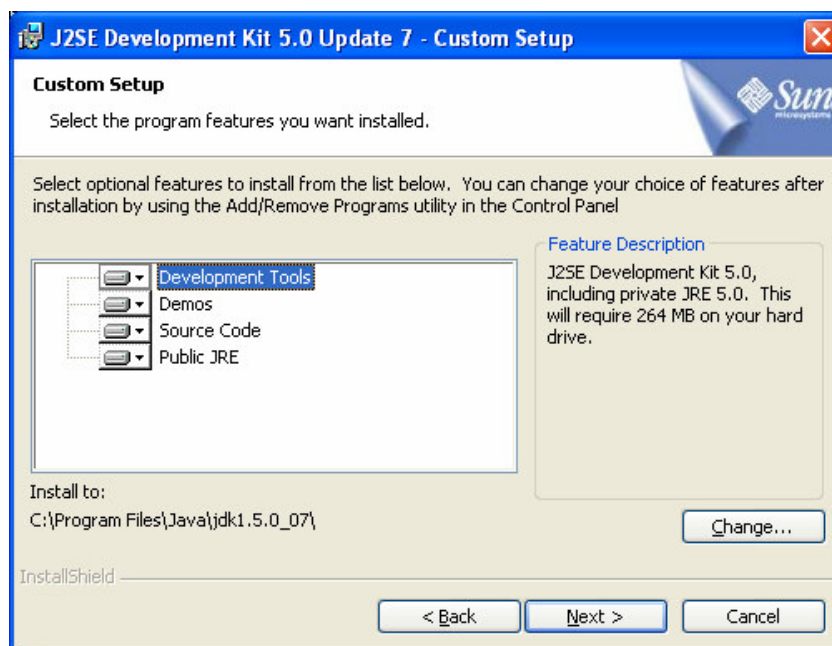
תקבל סדרת המסכים הבאה :

- I. מסך רישיון – License (תמונה 40). כדי להמשיך יש לבחור ב- "I accept the terms..."
לחצו על הכפתור Next להתחלת ההתקנה.



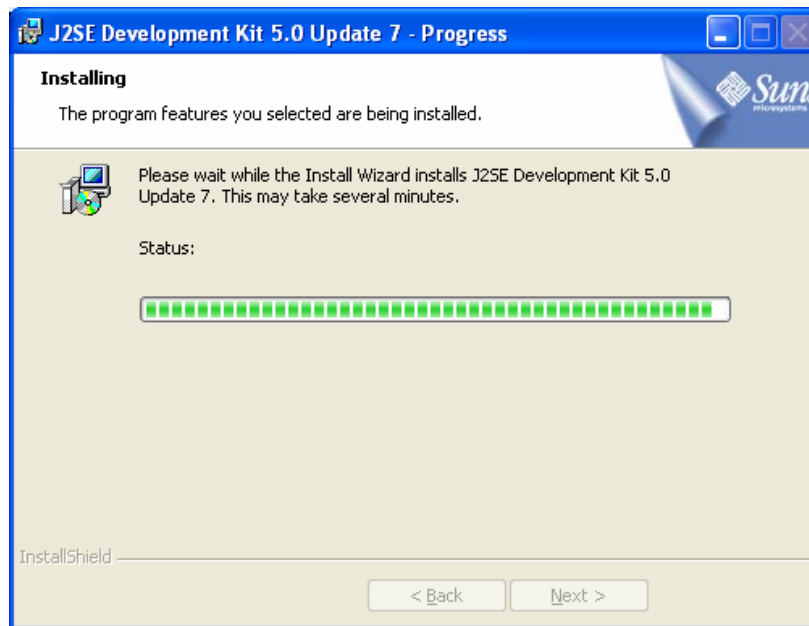
תמונה 40

- II. התקנת ה-JDK עצמו (תמונה 41). ההתקנה גם מתקינה את ה-JRE (Java Runtime Environment) מומלץ לא לשנות את מיקום ההתקנה המוצע. לחצו על הכפתור Next כדי להמשיך.



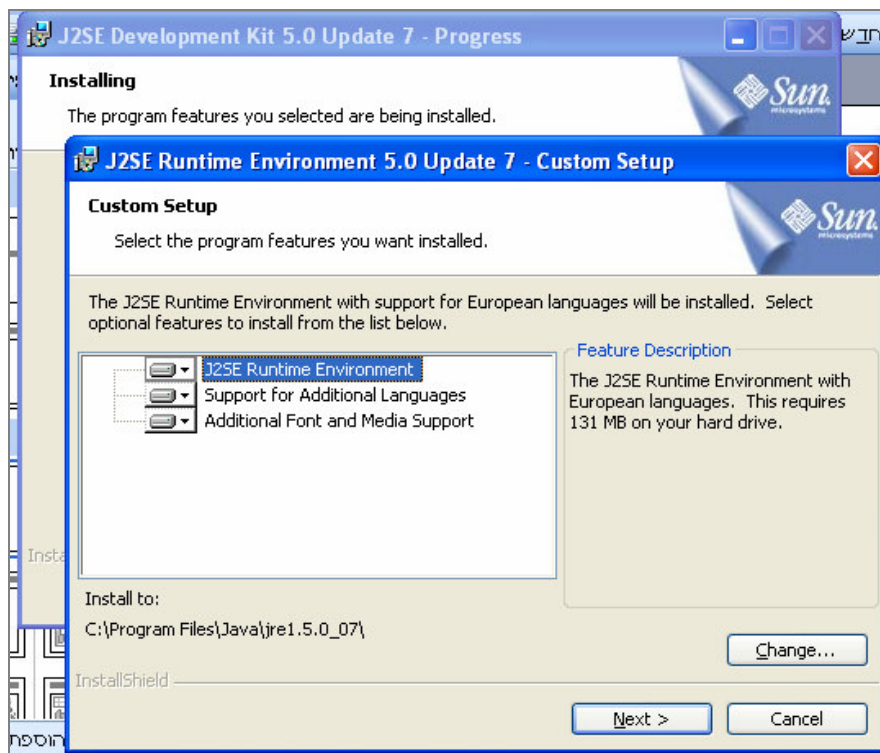
תמונה 41

III. התקנה בפועל – העתקת קבצים, שינוי הגדרות וכו' (תמונה 42). צריך סבלנות! התהליך לוקח מספר דקות. הכפתור Next יידלק עם סיום ההתקנה של ה-JDK.



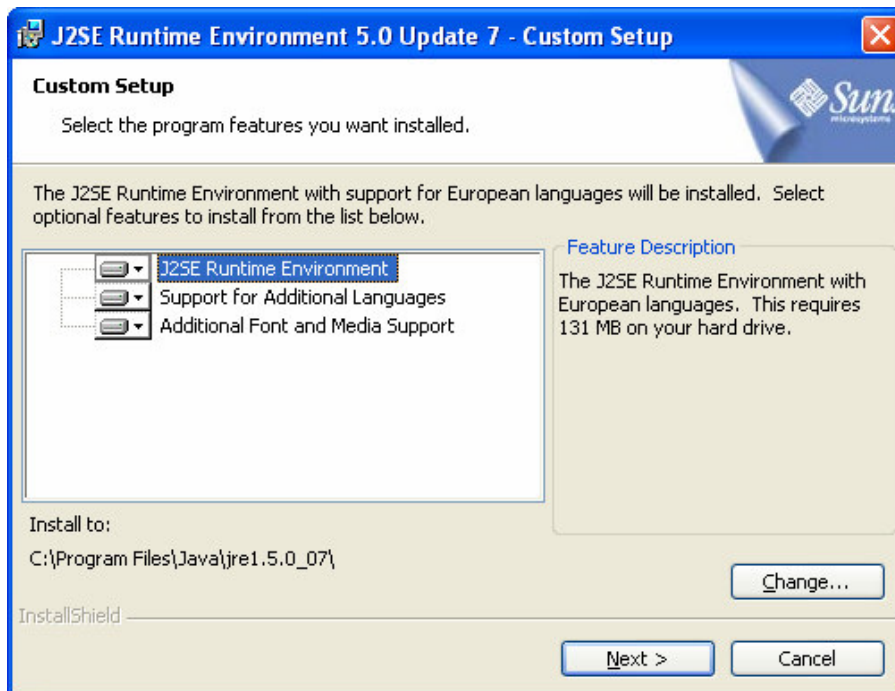
תמונה 42

IV. הפעלת התקנת ה-JRE מתוך התקנת ה-JDK (תמונה 43). שימו לב: ישנם שני מסכי התקנה. מומלץ לא לשנות את שם תיקיית ההתקנה. לחצו על הכפתור Next להמשך.



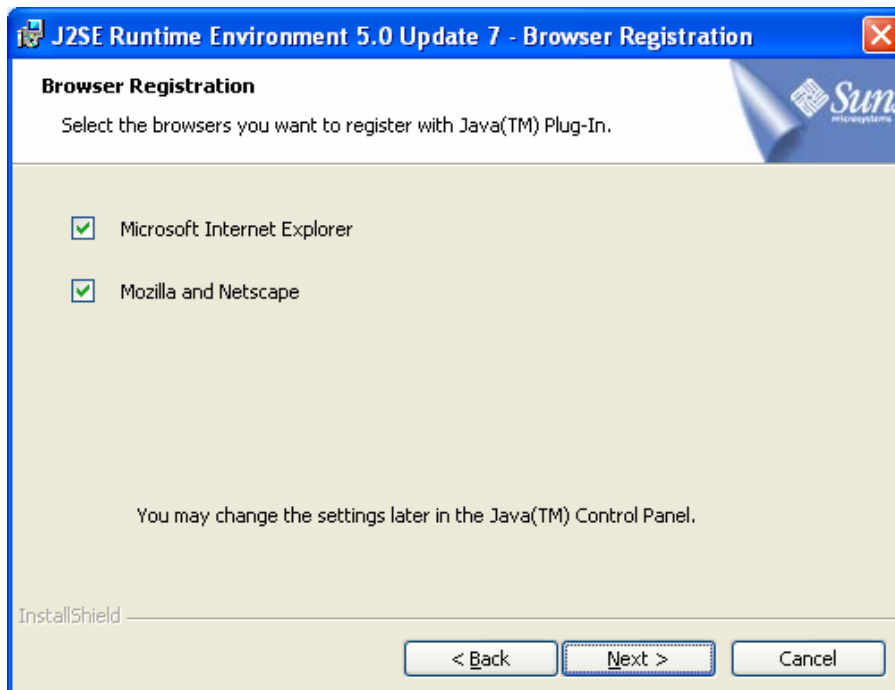
תמונה 43

V. התקנת ה-JRE (סביבת ה-Java Runtime) (תמונה 44).



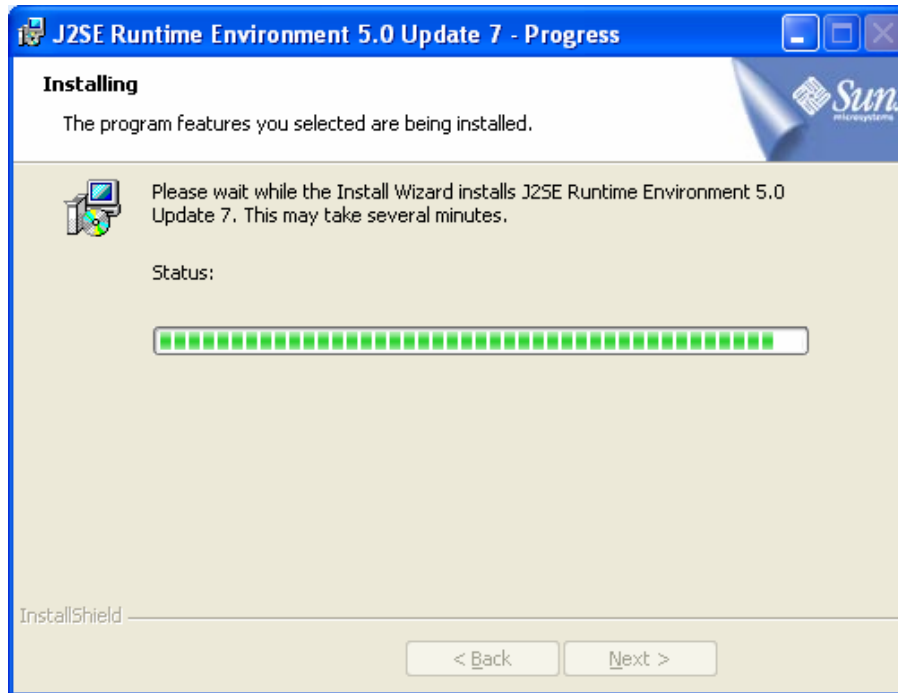
תמונה 44

VI. בחירת הדפדפן המומלץ (תמונה 45). מומלץ להשאיר את שתי הבחירות. בכל מקרה אין להסיר את דפדפן ה-Explorer. לחצו על הכפתור Next כדי להמשיך.



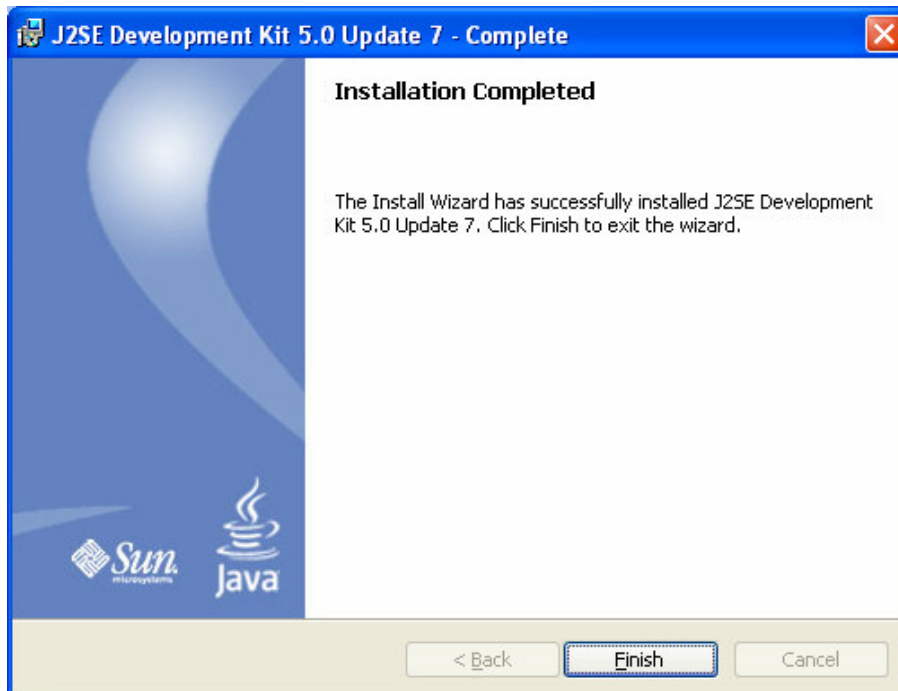
תמונה 45

VII. תהליך ההתקנה בפועל של סביבת ה-Java Runtime (תמונה 46) במהלכו מועתקים הקבצים הנדרשים ומבוצעות ההגדרות הנדרשות.



תמונה 46

VIII. סיום ההתקנה (תמונה 47) לחצו על הכפתור Finish לסיום.



תמונה 47

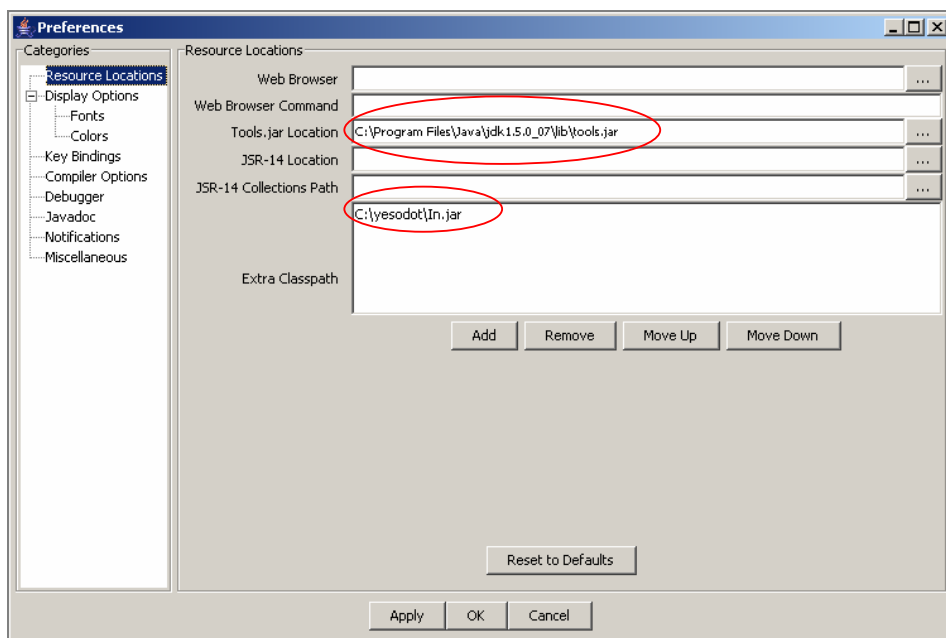
7.3 מה חשוב לוודא לאחר ההתקנה?

ראשית, יש לוודא שכל הקבצים הנדרשים נמצאים בתיקיות המתאימות (לפי המפורט בטבלה למטה), כדי לוודא עבודה תקינה. חלק מהקבצים הם קבצי הפעלה, קבצי ספרייה וקבצי הגדרות. וודאו קיומם כדי שכל רכיבי סביבת העבודה בהקשר של יסודות מדעי המחשב ובכלל יתפקדו כשורה:

שם קובץ	מאפייני הקובץ	שם תיקייה
DrJava.Exe	קובץ הפעלה ראשי של תוכנת DrJava	C:\Yesodot
In.Jar	קובץ מחלקת הקלט	C:\Yesodot
.DrJava	קובץ הגדרות של תוכנת DrJava	תיקית המשתמש הרשום
Jdk-1_5_0_07-Windows-i586-p.Exe	קובץ התקנה של גרסת JDK עדכנית. קובץ זה משמש כגיבוי להתקנה נוספת.	C:\Yesodot

יש לוודא כי הקובץ Tools.Jar מותקן, דרך התפריט Edit → Preferences (ראו תמונה 48).

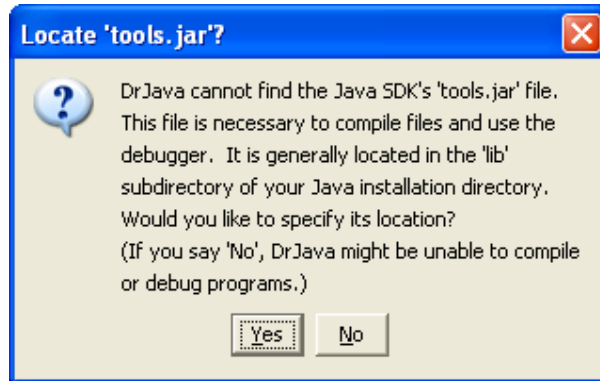
יש לוודא כי הקובץ In.Jar מותקן, דרך התפריט Edit → Preferences (ראו תמונה 48).



תמונה 48

7.4 שגיאות התקנה

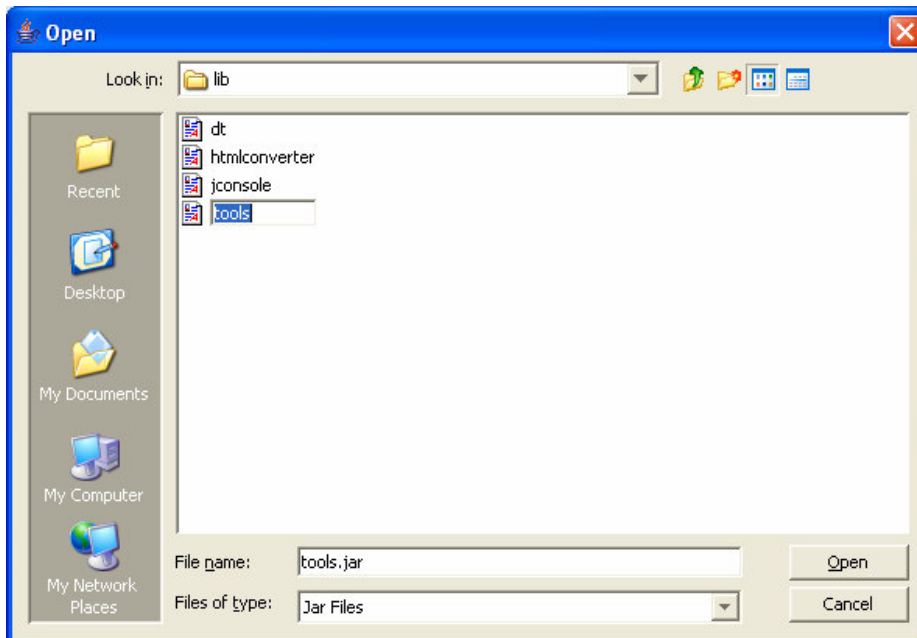
השגיאה המהותית ביותר, שלא תאפשר עבודה תקינה בסביבת **DrJava** תופיע עם הפעלת **DrJava**, כך (תמונה 49):



תמונה 49

שגיאה זו מציינת את היעדרו של קובץ הספרייה Tools.Jar שמהווה חלק מהתקנת ה-JDK. מוסבר היכן צריך למוצאו וניתנת האפשרות לבצע זאת. לחיצה על הכפתור No תשאיר את סביבת העבודה במצב לא תקין.

בהנחה שלא שינינו את מיקום תיקיית ההתקנה, נמצא את הקובץ Tools.Jar (תמונה 50) בתיקייה C:\Program Files\Java\jdk1.5.0_07\lib.



תמונה 50

שגיאת הפעלה – Executor Error: הודעה על שגיאה זו (תמונה 51) תופיע כאשר נפעיל את **DrJava** ותוכנת ה-JDK אינה מותקנת (או אינה מותקנת כראוי) מסיבה כלשהי.

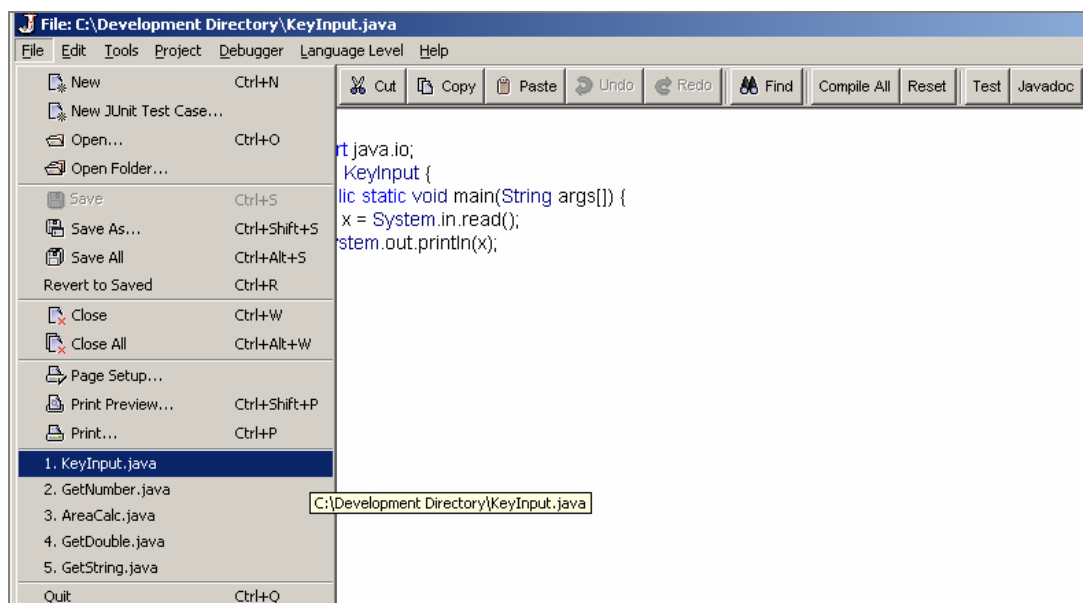


תמונה 51

פתרון הבעיה: התקינו את תוכנת ה-JDK ע"י לחיצה כפולה על הקובץ `C:\Yesodot\jdk-1_5_0_07-windows-i586p.Exe` בתיקיית יסודות בכונן C.

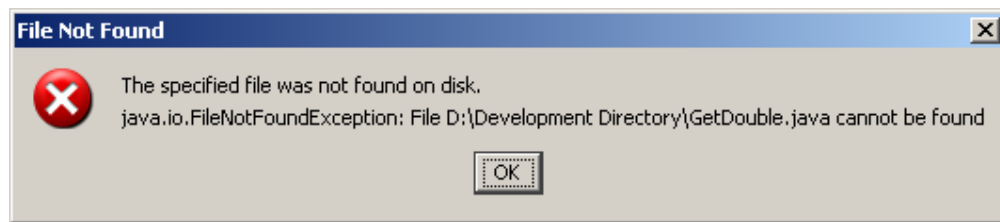
7.5 שגיאות תפעול

שגיאות אלו נובעות בדרך כלל מהיעדר קובץ בכונן מסוים, למרות שהתוכנה מצפה שהוא יהיה שם. למשל, בתמונה 50 אנו רואים את רשימת הקבצים שנפתחו לאחרונה (תחת תפריט File).



תמונה 51

הקובץ `GetNumber.java` אינו נמצא בניסיון פתיחה של הקובץ (ראו את הודעת השגיאה בתמונה 52) למרות שהוא מופיע ברשימה לעיל. הסיבה לכך הוא שכונן D אינו מחובר כעת. אותה בעיה הייתה צצה לו מחקנו את הקובץ.

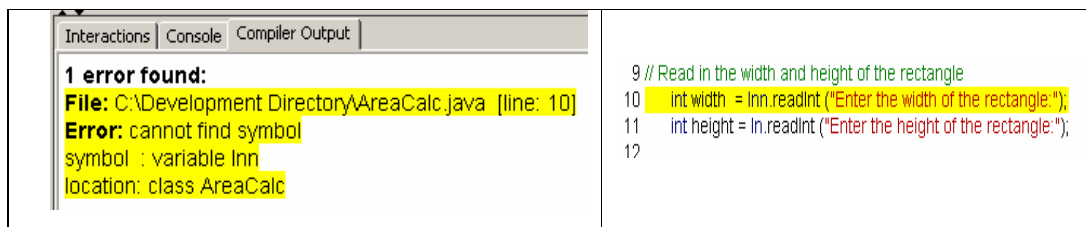


תמונה 52

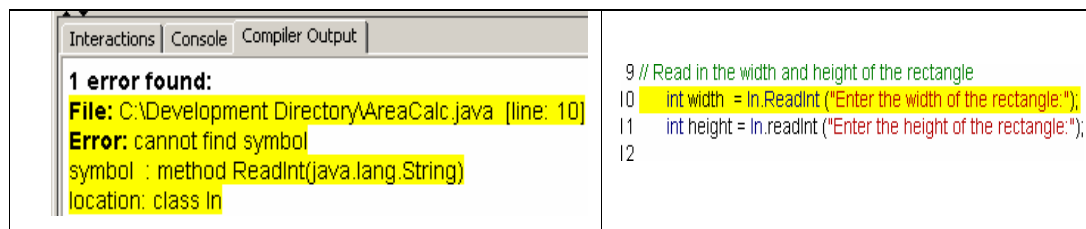
7.6 שגיאות הידור

שגיאות ההידור הנפוצות ביותר נוגעות לשגיאות תחביר (Syntax).

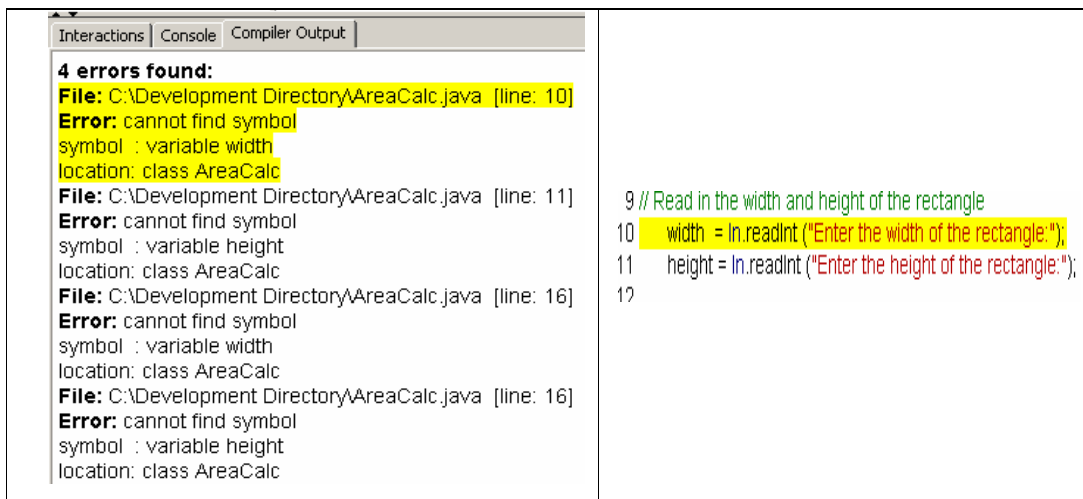
למשל, שם מחלקת הקלט הוא In אך בקוד עצמו כתבנו Inn. המהדר יתייחס לשם Inn כאל שמו של משתנה לא מוכר



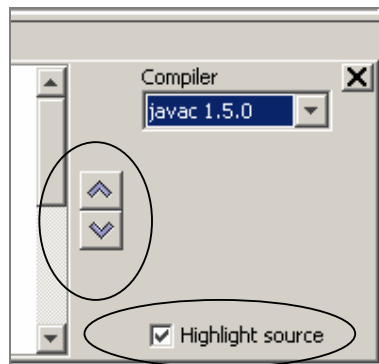
הנה דוגמה נוספת: שם מחלקת הקלט הוא In ובה מוגדרת פעולה בשם readInt. בקוד עצמו שם הפעולה נרשמה עם R גדולה. ReadInt. במקרה זה המהדר יודיע כי זו כפעולה לא מוכרת במחלקה In.



והנה דוגמה אחרונה: אנו משתמשים בשני משתנים width ו-height. אך שכחנו להצהיר על המשתנים וכתוצאה מכך קיבלנו ארבע שגיאות הידור, בכל מקום בו אנו משתמשים במשתנים אלה.



בכל המקרים, הודעת השגיאה נותנת אינדיקציה ברורה למדי על מהות הבעיה ומיקומה. ניתן לעבור משגיאה לשגיאה (על ידי שימוש בחיצים שעל לוח המקשים או על ידי החיצים שמופיעים בתמונה 53) ושורת הקוד השגויה תסומן גם בחלון פלט המהדר וגם בחלון העריכה. באפשרותנו להוריד את סימון שורת השגיאה בקוד עצמו על ידי אי-סימון את האפשרות Highlight source (תמונה 53).



תמונה 53

7.7 רישיון הפעלת התוכנה

הנה ההגדרה הרשמית לגבי שימוש בתוכנת **DrJava**. כל המשתמש בתוכנה ובמדריך זה במסגרת לימודי יסודות מדעי המחשב מתחייב לשמור על הכללים הבאים:

Copyright (C) 2001-2005 JavaPLT group at Rice University (javaplt@rice.edu)
All rights reserved.

DrJava Open Source License

Developed by: Java Programming Languages Team, Rice University
<http://www.cs.rice.edu/~javaplt/>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal with the

Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimers.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimers in the documentation and/or other materials provided with the distribution.
- Neither the names of DrJava, the JavaPLT, Rice University, nor the names of its contributors may be used to endorse or promote products derived from this Software without specific prior written permission.
- Products derived from this software may not be called "DrJava" nor use the term "DrJava" as part of their names without prior written permission from the JavaPLT group. For permission, write to javaplt@rice.edu.