

חומרים שהוכנו על-ידי משתתפי קורס מורים מובילים תשע"א

ניתן להשתמש בחומרים לצורך הוראה בלבד.

לא ניתן לפרסם את החומרים או לעשות בהם כל שימוש מסחרי

ללא קבלת אישור מראש מצוות הפיתוח

מחלקה עוטפת לעבודה עם קבצים – מדריך למורה

מותאם לסביבת ג'אווה

כתיבה ועריכה:

רוני אלנקרי

סימה בורנשטיין

ענת שלוס-סגל

עבודה עם קבצים ב-java

השימוש במחלקות `FileReader` ו-`File Writer` מאפשרות לנו לעבוד עם קבצים באופן פשוט ונוח.
עבודה עם קבצים תאפשר לתלמידים להריץ תוכניות עם קלט מוכן מראש בקלות וביעילות.
השיטות לקריאה וכתובה על גבי קובץ נכתבו כך שתוכלו לשלבן בתוכניות של התלמידים בצורה פשוטה.

יש להבחין בין כתיבה לקובץ ובין קריאה ממנו .

על מנת לעבוד עם קבצים עליכם להעתיק את המחלקות `FileReader.java` ו-`FileReader.java` לתיקיה בה הינכם עובדים .

המחלקה **FileWriter** מאפשרת לכתוב את נתוני התוכנית לקבצים ויכולה לשמש ל :

- פרויקטים בהם יש צורך לשמור את הנתונים (תוצאות משחקים וכד.).
- במקרים בהם פלט התוכנית מכיל מספר נתונים רב, כתיבת פלט זה לקובץ מאפשר הן למורה והן לתלמיד לבדוק את נכונותה ביתר קלות

המחלקה **FileReader** מאפשרת לקרוא נתונים מקובץ טקסט ויכולה לשמש ל :

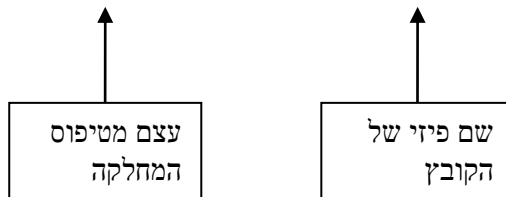
- הכנת קלטים שונים עבור תלמידים
- שילוב מקרי קצה בקלט
- הרצת התוכנית על קלטים גדולים(מערכים, מטריצות, עצמים ועוד).

כתיבה לקובץ

כללים לשימוש בשיטות המוגדרות במחלקה FileWriter

- יש להגדיר וליצור עצם מהמחלקה FileWriter כך שנוכל להשתמש בפעולות המחלקה.

```
static FileWriter writer = new FileWriter("test.txt")
```



- בסיום תוכנית העושה שימוש בקבצים יש לסגור את הקובץ ע"י שימוש בפקודה `close()` גם אם לא תבצעו פעולה זו תתנהל התוכנית באופן תקין והקבצים לא יינזקו (המחלקה FileWriter מטפלת במקרה זה), אך מנכון יותר מבחינת ניהול נכון של משאבים לבצע פעולה זו.

הטבלה שלהלן מפרטת את הפעולות המוגדרות ב **FileWriter** :

דוגמאות	תאור הפעולה	הפעולה
<code>writer.writeString("good morning");</code>	פעולה הכותבת מחרוזת לקובץ	<code>writeString(st)</code>
<code>writer.writeInt(3);</code> <code>int num=5;</code> <code>writer.writeInt(num);</code>	פעולה הכותבת מספר שלם לקובץ	<code>writeInt(num)</code>
<code>writer.writeDouble(3.6);</code> <code>double num=5.0;</code> <code>writer.writeDouble(num);</code>	פעולה הכותבת מספר ממשי לקובץ	<code>writeDouble(num)</code>
<code>writer.writeBoolean(true);</code> <code>writer.writeBoolean(a>b);</code>	פעולה הכותבת ערך בוליאני לקובץ	<code>writeBoolean(f)</code>
<code>writer.writeChar('#');</code>	פעולה הכותבת תו לקובץ	<code>writeChar(c)</code>
<code>int [] arr1 = { 1,2,3,4,5,6,7,8,9};</code> <code>writer.writeIntArray(arr);</code> לא ניתן לבצע כתיבת מערך באופן הבא : <code>writer.writeIntArray({ 1,2,3,4,5,6,7});</code>	פעולה הכותבת מערך של מספרים שלמים לקובץ	<code>writeIntArray(arr)</code>
	פעולה הכותבת מערך של מספרים ממשיים לקובץ	<code>writeDoubleArray(arr)</code>
	פעולה הכותבת מערך של מחרוזות לקובץ	<code>writeStringArray(arr)</code>
	פעולה הכותבת מטריצה של שלמים לקובץ	<code>writeIntMatrix(mat)</code>
<code>writer.close()</code>	פעולה הסוגרת את הקובץ עם סיום כתיבת הנתונים אליו.	<code>close()</code>

דוגמא מספר 1 – כתיבה לקובץ

```
public class TestFileWriter
{
    static FileWriter writer = new FileWriter(".\\t.txt");
    public static void main(String args[])
    {
        int[] arr1 = { 1, 2, 3, 4, 5, 6, 7, 8, 9 };
        double[] arr2 = { 1, 2, 3, 4, 5, 6, 7, 8, 9 };
        String[] arr3 = { "aba", "abba", "abbba" };
        int[][] mat = { { 1, 2, 3 }, { 4, 5, 6 }, { 7, 8, 9 } };

        writer.writeString("Shalom kita alef");
        writer.writeInt(3);
        writer.writeDouble(5.81);
        writer.writeBoolean(false);
        writer.writeChar('#');
        writer.writeIntArray(arr1);
        writer.writeDoubleArray(arr2);
        writer.writeStringArray(arr3);
        writer.writeIntMatrix(mat);
        writer.close();
    }
}
```

לאחר ההרצה יתקבל קובץ בשם t.txt (ניתן לפתוח אותו לצפייה) שתוכנו :

```
Shalom kita alef
3
5.81
false
#
[ 1 2 3 4 5 6 7 8 9 ]
[ 1.0 2.0 3.0 4.0 5.0 6.0 7.0 8.0 9.0 ]
[ aba abba abbba ]
[ [ 1 2 3 ] [ 4 5 6 ] [ 7 8 9 ] ]
```

♥ שימו לב

**המחלקה אינה מאפשרת לדעת עד מתי ניתן לקרוא מהקובץ.
על מנת למנוע גלישה עליכם להחליט מראש את אופן הקריאה מהקובץ
לדוגמא:**

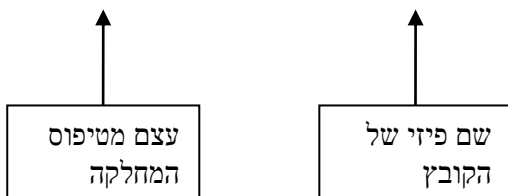
- ← הנתון הראשון בקובץ יהיה מספר שיציין את מספר הנתונים בו.
- ← הנתון האחרון בקובץ יהיה זקיף שיצביע על סיום הנתונים.

קריאה מקובץ

כללים לשימוש בשיטות המוגדרות במחלקה FileReader

- יש להגדיר וליצור עצם מהמחלקה FileReader כך שנוכל להשתמש בפעולות המחלקה.

```
static FileReader reader = new FileWriter("test.txt")
```



- בסיום תוכנית העושה שימוש בקבצים יש לסגור את הקובץ ע"י שימוש בפקודה `close()` גם אם לא תבצעו פעולה זו תתנהל התוכנית באופן תקין והקבצים לא יינזקו (המחלקה `FileReader` מטפלת במקרה זה), אך מנכון יותר מבחינת ניהול נכון של משאבים לבצע פעולה זו.

ניתן להכין מראש קבצי `txt` עבור התלמידים, אותם הם יוכלו לקרוא כפי שיוסבר להלן.

את הקבצים ניתן להכין מראש על ידי הפעולות שהוזכרו לעיל או באופן ישיר:

File ← New ← File ← בחרו את הפרויקט ← תנו שם לקובץ עם סיומת `txt`
← הכניסו נתונים לקובץ כרצונכם

שימו לב ♥

הפרדה בין הנתונים הינה על-ידי רווח או מעבר שורה.
עליכם להתחשב בכך בהכנת קבצי הקריאה וכן בזמן קריאת הקבצים.

הטבלה שלהלן מפרטת את הפעולות המוגדרות ב **FileReader** :

דוגמאות	תאור הפעולה	הפעולה
<pre>String st = reader.readString(); System.out.println(reader.readString())</pre>	<p>פעולה הקוראת מחרוזת <u>אחת</u> מקובץ. מחרוזת מסתיימת ברווח או בסיום שורה. פעולה זו מקבילה לפעולה <code>next()</code> המשמשת לקריאת מחרוזת קלט מהמשתמש.</p>	<code>readString()</code>
<pre>int num = reader.readInt(); System.out.println(reader.readInt())</pre>	פעולה הקוראת מספר שלם מקובץ	<code>readInt()</code>
	פעולה הקוראת מספר ממשי מקובץ	<code>readDouble()</code>
	פעולה הקוראת ערך בוליאני מקובץ	<code>readBoolean()</code>
	פעולה הקוראת תו אחד מקובץ. <u>התו הנקרא הינו התו הראשון המופיע בתחילת שורה או אחרי רווח.</u>	<code>readChar()</code>
	פעולה הקוראת שורה שלמה מהקובץ. שורה יכולה להכיל מספר מחרוזות, מספרים וכו'...	<code>readLine()</code>
<pre>int [] arr1 = reader.readIntArray();</pre>	פעולה הקוראת מערך של מספרים שלמים לקובץ	<code>readIntArray()</code>
	פעולה הקוראת מערך של מספרים ממשיים לקובץ	<code>readDoubleArray()</code>
	פעולה הקוראת מטריצה של מחרוזות מקובץ	<code>readStringArray()</code>
	פעולה הקוראת מטריצה של שלמים מקובץ	<code>readIntMatrix(mat)</code>
<pre>reader.close()</pre>	פעולה הסוגרת את הקובץ עם סיום קריאת הנתונים ממנו	<code>close()</code>

דוגמא מספר 2 – קריאה מקובץ טקסט

ראשית נכין קובץ טקסט כפי שהוסבר לעיל (← File ← New ← ...)

לדוגמא הקובץ exampleFile.txt

```
5 2 4 6 8 10
5 2 4 6 8 10
3 abc def ghi
[11 12 13 14 15 16 17 18]
[ [ 11 12 13 14 ] [ 21 22 23 24 ] [ 31 32 33 34 ] [ 41
42 43 44 ] ]
```

שימו לב לקיום רווח בין כל אחד מהנתונים (כולל לפני ואחרי הסוגריים המרובעים)

```
public class Exmple
{
    static FileReader reader = new FileReader("ExmpleFile.txt");
    public static void main(String[] args)
    {
        // קריאת השורה הראשונה בשלמותה
        System.out.println(reader.readLine());

        // קריאת מספרים שלמים בשורה השנייה
        int n1 = reader.readInt(); // המספר הראשון מציין את מספר הנתונים אחריו
        for (int i=1 ;i<=n1 ; i++)
            System.out.println(reader.readInt());

        // קריאת הנתונים בשורה השלישית בקובץ הנתונים

        int n2 = reader.readInt() // המספר הראשון מציין את מספר הנתונים אחריו
        for (int i=1 ;i<=n2 ; i++)
            System.out.println(reader.readChar());
    }
}
```

// קריאת המערך המופיע בשורה הרביעית בקובץ הנתונים

```
int[] arr1 = reader.readIntArray() ;  
System.out.println("arr1=");  
for (int i=0 ; i<arr1.length ; i++)  
    System.out.print(arr1[i] + " " );  
System.out.println();
```

// קריאת המטריצה המופיעה בשורה החמישית בקובץ הנתונים

```
int[][] mat = reader.readIntMatrix()  
System.out.println("mat=");  
for (int i=0 ; i<mat.length ; i++)  
{   for(int j=0 ; j< mat[0].length ; j++)  
        System.out.print(mat[i][j] + " " );  
    System.out.println()  
}  
reader.close();  
}  
}
```

יתקבל הפלט הבא :

```
5 2 4 6 8 10  
2  
4  
6  
8  
10  
a  
d  
g
```

המספרים המופיעים בשורת הקובץ השנייה

3 התווים הראשונים בכל אחת מ-3 המחרוזות בשורה

```
arr1=  
12 13 14 15 16 17  
mat=  
11 12 13 14  
21 22 23 24  
31 32 33 34  
41 42 43 44
```

דוגמא פתורה מספר 3 – שימוש במחרוזות, char ופקודת switch

קלט: מחרוזת שבה האותיות בעברית נקלטו במקלדת כאותיות קטנות באנגלית.

לדוגמא המחרוזת "סוס" נקלטה כ "xix"

פלט: המחרוזת בעברית .

יש להכין מראש קובץ עבור התלמידים המכיל את הטקסט שיש להחליף בו את האותיות.
(את הקובץ ניתן להכין כפי שהוסבר לעיל)

ניתן להרחיב את התרגיל :

- הטקסט יכול גם אותיות גדולות באנגלית
- הטקסט יכול יותר משורת טקסט אחת

חומרי עזר שהוכנו ע"י משתתפי קורס מורים מובילים תשע"א
ניתן להשתמש בחומרים לצורך הוראה בלבד. אסור לפרסם את החומרים או לעשות בהם שימוש מסחרי כלשהו ללא קבלת אישור מראש מצוות הפיתוח

```
{    public class TestString
static FileReader reader = new FileReader("testString.txt");
public static void main(String args[])
{
    String str1 = reader.readLine(); // קריאת המחרוזת שיש לפענח
    String str2=""; // המחרוזת המפוענחת
    char tav2=' ';
    for (int i=0;i<str1.length();i++)
    {
        switch (str1.charAt(i))
        {
            case 'a' : tav2='ש';break;
            case 'b' : tav2='נ';break;
            case 'c' : tav2='ב';break;
            case 'd' : tav2='ג';break;
            case 'e' : tav2='ק';break;
            case 'f' : tav2='כ';break;
            case 'g' : tav2='ע';break;
            case 'h' : tav2='י';break;
            case 'i' : tav2='ך';break;
            case 'j' : tav2='ח';break;
            case 'k' : tav2='ל';break;
            case 'l' : tav2='ך';break;
            case 'm' : tav2='צ';break;
            case 'n' : tav2='מ';break;
            case 'o' : tav2='ם';break;
            case 'p' : tav2='פ';break;
            case 'q' : tav2='/' ;break;
            case 'r' : tav2='ר';break;
            case 's' : tav2='ד';break;
            case 't' : tav2='א';break;
            case 'u' : tav2='ו';break;
            case 'v' : tav2='ה';break;
            case 'w' : tav2=' ' ;break;
            case 'x' : tav2='ס';break;
            case 'y' : tav2='ט';break;
            case 'z' : tav2='ז';break;
            case ',' : tav2='ת';break;
            case '.' : tav2='ץ';break;
            case '/' : tav2='.';break;
            case ';' : tav2='ף';break;
            case '' : tav2=' ' ;break;
        }

        str2=str2+tav2;
    }
    System.out.println(str2);
    reader.close();
}
}
```

ניתן להשתמש בחומרים לצורך הוראה בלבד.
לא ניתן לפרסם את החומרים או לעשות בהם כל שימוש מסחרי ללא קבלת אישור מראש מצוות הפיתוח

דוגמא פתורה מספר 4 – מערך עצמים

במספר דקות נוכל להכין קובץ לקריאה עם עצמים.

נגדיר מחלקה Student

```
public class Student
{
    private String name // שם התלמיד ;
    private int [] mark // ציוני התלמיד ;
    private int grade // כיתה ;
    public Student (String name ,int numMark, int grade)
    {
        this.name = name;
        this.grade = grade;
        this.mark = new int[numMark];
        for (int i=0 ; i< numMark; i++)
            mark[i] = 0;
    }
    public Student (String name ,int[] mark, int grade)
    {
        this.name = name;
        this.grade = grade;
        this.mark = mark;
    }
    // פעולות get ו-set לכל אחת מהתכונות
    public String getName() {return this.name;}
    public int getGrade() {return this.grade ; }
    public int getMark(int n) {return this.mark[n]; }
    public void setName(String name) { this.name = name; }
    public void setGrade(int grade) { this.grade = grade;}
    public void setMark(int mark, int n) { this.mark[n] = mark;}
    public String toString ()
    {
        String st= "name="+this.name+" grade;
        switch (this.grade)
        {
            case 10 : st = st + " כיתה י' " + break;
            case 11 : st = st + " כיתה יא' " + break;
            case 12 : st = st + " כיתה יב' " + break;
            default : st = st + " מספר לא מזוהה "
        }

        st = st + "\n" + "mark=";
        for (int i=0 ; i<this.mark.length; i++ )
            st=st+this.mark[i]+" ";
        return st;
    }
}
```

נבנה את קובץ הטקסט הבא : (הקובץ מכיל 12 תלמידים)

```
david 3 12
eli [ 90 100 70 90 100 ] 11
gil [ 60 100 78 60 90] 12
gadi [ 80 90 87 ] 11
dina [ 100 100 100 100 100 100 100 ] 10
rina [ 90 80 70 90 90 99 ] 10
yael [ 80 99 87 67 ] 12
dor [ 76 65 74 87 65 77 ] 10
hila [ 77 68 77 98 88 90 100] 10
liat [ 90 90 90 90 90 90 90 ] 10
boaz [ 90 87 66 77 55 55 55 ] 10
dana [ 100 90 88 99 77 100 100] 10
```

```
public class TestSudent
{

    static FileReader reader = new FileReader("studentFile.txt");

    public static void main(String[] args)
    {

        // שימוש בפעולה הבונה הראשונה המקבלת את שם התלמיד, מספר כיתה ומספר ציוניו (קריאת השורה הראשונה מהקובץ)
        Student t1 = new Student(reader.readString(),reader.readInt(),reader.readInt());
        System.out.println(t1);

        // שימוש בפעולה הבונה השנייה המקבלת את שם התלמיד, מערך ציוניו ומספר כיתה (קריאת השורה השנייה מהקובץ)
        Student t2 = new Student(reader.readString(),reader.readIntArray(),reader.readInt());
        System.out.println(t2);

        // הגדרת מערך מטיפוס Student
        // וקריאת נתונים מהקובץ לתוכו (קריאת 10 שורות מהקובץ)

        Student [] st = new Student[10] ;
        for(int i=0; i<st.length ; i++)
        {
            st[i] = new Student(reader.readString(),reader.readIntArray(),reader.readInt());
            System.out.println(st[i]);
        }
        reader.close();

    }

}
```