**הערה חשובה** למשתמשים בגרסה הישנה של ה-ECLIPSE

אם במחשב שלכם הותקנה גרסת ה-ECLIPSE מאתר של הספר "עיצוב תוכנה מבוסס עצמים" של האוניברסיטה העברית

<http://sites.huji.ac.il/science/stc/center/groups/OOP2006/Java/EclipseInstall4Unit4/EclipseInstaller.htm>

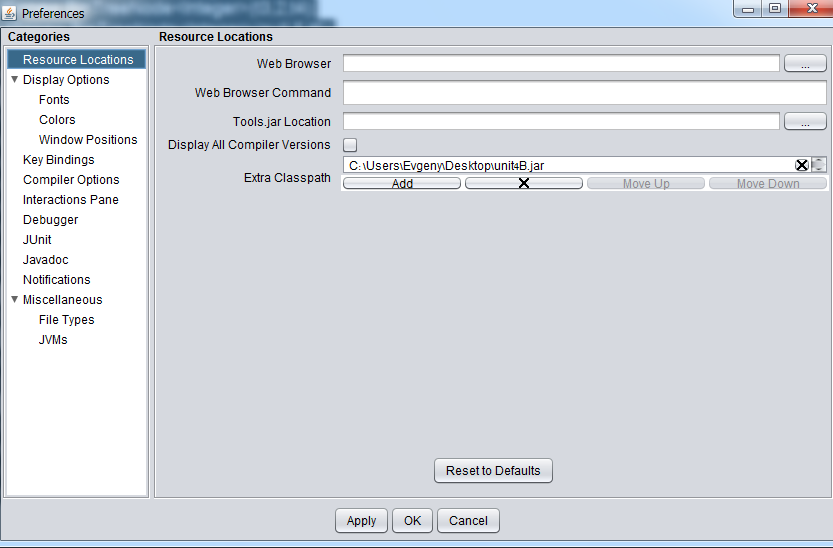
אתם תצטרכו או להתקין גרסה חדשה יותר (מומלץ!) או לבצע פעולות הבאות:

1. למחוק את הקובץ Unit4.jar שנמצא בתקיה HelpFiles.
2. להעתיק את הקובץ Unit4C.jar לתיקיה HelpFiles
3. לשנות את שם הקובץ ל-Unit4.jar

### הוספת ה-UNIT4B.jar

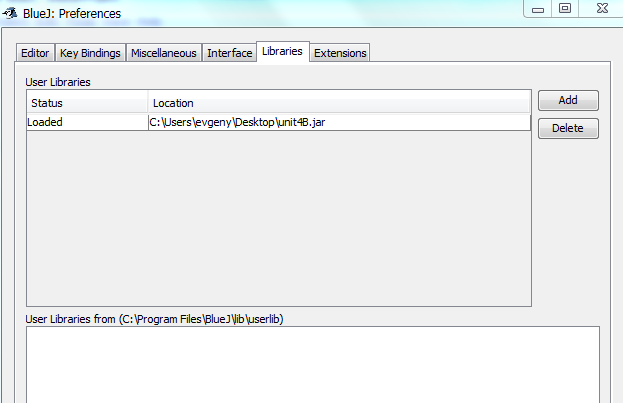
**DrJAVA**

Edit 🡪 Preferences 🡪Resource Locations 🡪 Add Extra Path

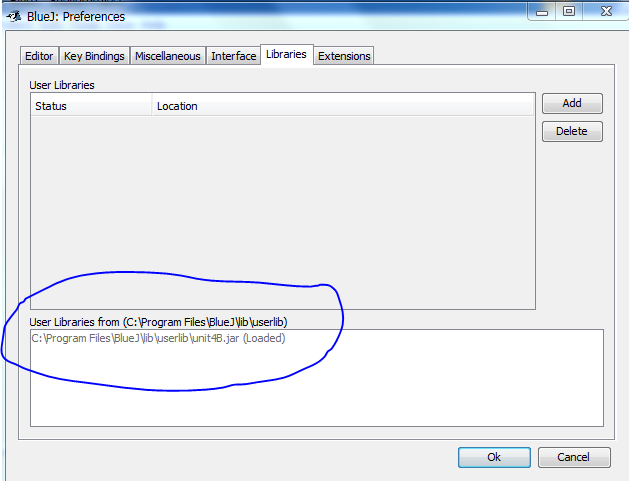


**BlueJ**

**Tools 🡪Preferences 🡪 Libraries 🡪Add**

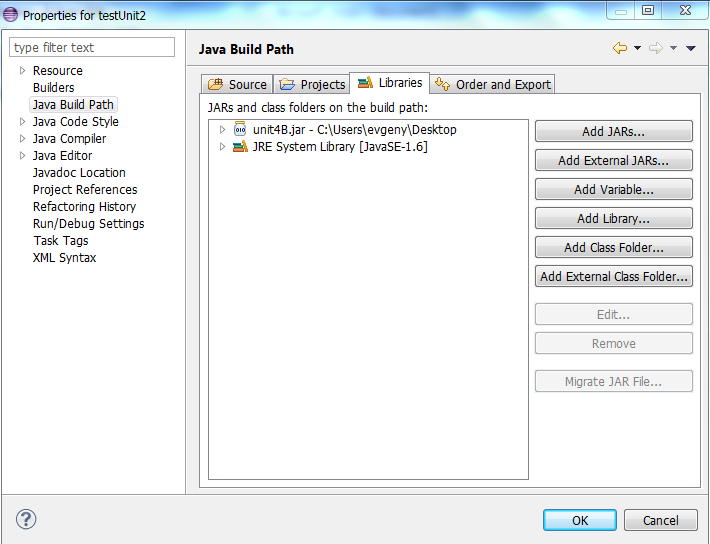


דרך השנייה – להעתיק את הקובץ UNIT4B.jar לתיקיה…\BlueJ\lib\userlib\



**Eclipse**

**Project 🡪 Properties 🡪 Java Build Path 🡪 Libraries 🡪 Add External JARs**



### שימוש ב-UNIT4B.jar

הקובץ Unit4B מיועד להחליף את הקובץ המקורי Unit4.jar אשר פותח על – ידי צוות פיתוח של ספר "עיצוב תכנה מבוסס עצמים". הוא מאפשר להשתמש בפעולות ממשק של תכנית הלימודים "עיצוב תכנה" (תכנית ישנה) וגם בפעולות ממשק של תכנית לימודים "מבנה נתונים" (תכנית חדשה)

על מנת להשתמש במחלקות   
של תכנית הישנה:  
 Node, List, Stack, Queue, BinTreeNode

וגם של תכנית החדשה:  
Node, Stack, Queue, BinNode

**import unit4.collectionsLib.\*;**

**import unit4.utilsLib.\*;**

class Test1{

על מנת להשתמש במחלקות עזר של עצים בינריים   
של תכנית הישנה: **BinTreeUtils**

וגם של תכנית החדשה: **TreeUtils**

public static void main(String[] args) {

BinTreeNode<Integer> t1=new BinTreeNode<Integer>(1);

BinTreeNode<Integer> t2=new BinTreeNode<Integer>(7);

BinTreeNode<Integer> t3=new BinTreeNode<Integer>(2);

בניית עץ בינארי לפי פקודות של **תכנית הישנה**  והצגתו

BinTreeNode<Integer> t4=new BinTreeNode<Integer>(7);

BinTreeNode<Integer> t5=new BinTreeNode<Integer>(t1,1,t2);

BinTreeNode<Integer> t6=new BinTreeNode<Integer>(t3,2,t4);

BinTreeNode<Integer> t7=new BinTreeNode<Integer>(null,5,t6);

BinTreeNode<Integer> t8=new BinTreeNode<Integer>(t5,7,t7);

**BinTreeUtils.showTree(t8,"OLD ");**

BinNode<Integer> tt1=new BinNode<Integer>(1);

BinNode<Integer> tt2=new BinNode<Integer>(7);

בניית עץ בינארי לפי פקודות של **תכנית החדשה**  והצגתו

BinNode<Integer> tt3=new BinNode<Integer>(2);

BinNode<Integer> tt4=new BinNode<Integer>(7);

BinNode<Integer> tt5=new BinNode<Integer>(tt1,1,tt2);

BinNode<Integer> tt6=new BinNode<Integer>(tt3,2,tt4);

BinNode<Integer> tt7=new BinNode<Integer>(null,5,tt6);

BinNode<Integer> tt8=new BinNode<Integer>(tt5,7,tt7);

**TreeUtils.showTree(tt8," NEW");**

///

List<Integer> l=new List<Integer>();

בניית רשימה לפי **תכנית הישנה**

l.insert(null,2); l.insert(null,20); l.insert(null,200);

System.out.println(l);

Node<Integer> q=l.getFirst();

System.out.println(q.getInfo());

בחוליה (Node ) יש אפשרות להשתמש גם פעולות של **תכנית הישנה**:  
 get/setInfo(), get/setNext()

וגם בפעולות של **תכנית החדשה**:

get/setValue, get/setNext, hasNext

Node<Integer> p=new Node<Integer>(10);

System.out.println(**p.getInfo()**+" "+p.**getValue**());

}

}