



Το εγχειρίδιο του BlueJ

**Έκδοση 1.4
για το BlueJ 1.2.x**

Michael Kölling
Mærsk Institute
University of Southern Denmark

μετάφραση: Τάσος Ζερβός

1	Πρόλογος	4
1.1	<i>BlueJ</i>	4
1.2	<i>Πλαίσιο και αναγνωστικό κοινό</i>	4
1.3	<i>Πνευματικά δικαιώματα, εξουσιοδοτήσεις και αναδιανομή</i>	4
1.4	<i>Αντιδράσεις</i>	4
2	Εγκατάσταση	5
2.1	<i>Εγκατάσταση στα Windows</i>	5
2.2	<i>Εγκατάσταση σε Macintosh</i>	5
2.3	<i>Εγκατάσταση σε Linux/Unix και άλλα συστήματα</i>	6
2.4	<i>Προβλήματα εγκατάστασης</i>	6
3	Ξεκινώντας – διόρθωση / μεταγλώττιση / εκτέλεση (edit/compile/execute)	7
3.1	<i>Ξεκινώντας το BlueJ</i>	7
3.2	<i>Ανοίγοντας ένα πρόγραμμα (project)</i>	7
3.3	<i>Δημιουργώντας αντικείμενα</i>	8
3.4	<i>Εκτέλεση</i>	10
3.5	<i>Διορθώνοντας μία κλάση</i>	11
3.6	<i>Μεταγλώττιση</i>	12
3.7	<i>Βοήθεια με τα σφάλματα μεταγλωττιστών</i>	13
4	Κάνοντας κάτι παραπάνω	14
4.1	<i>Εξέταση</i>	14
4.2	<i>Μεταβιβάζοντας αντικείμενα ως παραμέτρους</i>	17
5	Δημιουργία ενός νέου προγράμματος	18
5.1	<i>Δημιουργία του καταλόγου αρχείων προγράμματος</i>	18
5.2	<i>Δημιουργώντας κλάσεις</i>	18
5.3	<i>Δημιουργώντας εξαρτήσεις</i>	18
5.4	<i>Αφαιρώντας στοιχεία</i>	19
6	Αποσφαλμάτωση	20
6.1	<i>Θέτοντας σημεία αναστολής</i>	20
6.2	<i>«Περπατώντας» μέσα στον κώδικα</i>	21
6.3	<i>Επιθεωρώντας μεταβλητές</i>	22
6.4	<i>Στάση και ολοκλήρωση (Halt and Terminate)</i>	22
7	Δημιουργώντας ολοκληρωμένες εφαρμογές	24
8	Δημιουργώντας applets (μίνι-εφαρμογές)	25
8.1	<i>«Τρέχοντας» ένα applet</i>	25
8.2	<i>Δημιουργώντας ένα applet</i>	26
8.3	<i>Εξετάζοντας το applet</i>	26

9	Άλλες διαδικασίες	27
9.1	Ανοίγοντας πακέτα μη ανήκοντα στο BlueJ με το BlueJ	27
9.2	Προσθέτοντας υπάρχουσες κλάσεις στο πρόγραμμά σας.....	27
9.3	Καλώντας την main και άλλες στατικές μεθόδους	27
9.4	Δημιουργώντας τεκμηρίωση (documentation)	27
9.5	Δουλεύοντας με βιβλιοθήκες.....	28
9.6	Δημιουργώντας αντικείμενα από κλάσεις βιβλιοθηκών	28
10	Οι περιλήψεις.....	29

1 Πρόλογος

1.1 BlueJ

Το εγχειρίδιο αυτό είναι μία εισαγωγή στη χρήση του προγραμματιστικού περιβάλλοντος BlueJ. BlueJ είναι ένα περιβάλλον ανάπτυξης Java™ ειδικά σχεδιασμένο για διδασκαλία σε εισαγωγικό επίπεδο. Σχεδιάστηκε και εφαρμόστηκε από την ομάδα BlueJ του πανεπιστημίου Monash, Μελβούρνη, Αυστραλία, και του πανεπιστημίου της Νότιας Δανίας, Odense.

Περισσότερες πληροφορίες για το BlueJ είναι διαθέσιμες στο: <http://www.bluej.org>.

1.2 Πλαίσιο και αναγνωστικό κοινό

Το εγχειρίδιο αυτό απευθύνεται σε αυτούς που θέλουν να εξοικειωθούν με τις δυνατότητες του περιβάλλοντος. Δεν εξηγεί τις αποφάσεις σχεδίασης της κατασκευής του περιβάλλοντος ή τα ερευνητικά ζητήματα πίσω από αυτή.

Το εγχειρίδιο αυτό δεν προσπαθεί να διδάξει Java. Συνιστούμε σε αρχάριους προγραμματιστές στη γλώσσα Java να μελετήσουν ένα εισαγωγικό βιβλίο Java ή να παρακολουθήσουν ένα σεμινάριο Java.

Αυτό το έγγραφο δεν είναι ένα εκτενές βιβλίο αναφοράς. Πολλές λεπτομέρειες δεν έχουν συμπεριληφθεί. Έμφαση έχει δοθεί στην σύντομη και συνοπτική εισαγωγή και όχι στην πλήρη κάλυψη των χαρακτηριστικών.

Κάθε ενότητα ξεκινά με μία μικρή περιληπτική πρόταση. Αυτό δίνει την δυνατότητα σε χρήστες εξοικειωμένους με μέρη του συστήματος να αποφασίσουν αν θέλουν να διαβάσουν ή να μεταπηδήσουν κάποιο τμήμα. Η ενότητα 10 επαναλαμβάνει μόνο τις περιλήψεις σαν ταχεία αναφορά.

1.3 Πνευματικά δικαιώματα, εξουσιοδοτήσεις και αναδιανομή

Το σύστημα BlueJ και αυτό το εγχειρίδιο είναι ελεύθερα διαθέσιμα σε καθένα για οποιαδήποτε χρήση. Το σύστημα και η τεκμηρίωση του μπορούν να αναδιανεμηθούν ελεύθερα.

Κανένα μέρος του συστήματος BlueJ και της τεκμηρίωσης του δεν επιτρέπεται να πωλειθούν κερδοφόρα ή να συμπεριληφθούν σε εφαρμογή (πακέτο) η οποία πωλείται κερδοφόρα χωρίς την γραπτή εξουσιοδότηση των συγγραφέων.

Τα πνευματικά δικαιώματα © για το BlueJ ανήκουν στους M.Kölling και J.Rosenberg.

Η μετάφραση έγινε από τον Τάσο Ζερβό (<http://www.javagrinder.com>).

1.4 Αντιδράσεις

Σχόλια, ερωτήσεις, διορθώσεις, επικρίσεις και κάθε είδους αντίδραση αναφορικά με το σύστημα BlueJ ή αυτό το εγχειρίδιο είναι πολύ ευπρόσδεκτες και ενεργά προτροπόμενες. Παρακαλούμε απευθυνθείτε στον Michael Kölling (mik@mip.sdu.dk) ή στον Τάσο Ζερβό (tzervos@hotmail.com) για την μετάφραση στα ελληνικά.

2 Εγκατάσταση

Το BlueJ διανέμεται σε τρεις διαφορετικές μορφές: Μία για το λειτουργικό σύστημα Windows, μία για MacOS, και μία για όλα τα άλλα συστήματα. Η εγκατάσταση είναι αρκετά απλή.

Απαιτούμενα

Πρέπει να έχετε J2SE v1.3 (JDK 1.3) ή μεταγενέστερο εγκατεστημένο στο σύστημα σας ώστε να χρησιμοποιήσετε το BlueJ. Αν δεν έχετε το JDK εγκατεστημένο μπορείτε να το «κατεβάσετε» από την ιστοσελίδα της Sun από: <http://java.sun.com/j2se/>. Στο MacOS X, μία πρόσφατη έκδοση του JDK είναι προεγκατεστημένη – δεν χρειάζεται να την εγκαταστήσετε εσείς. Αν βρείτε κάποια ιστοσελίδα που προσφέρει το “JRE” (Java Runtime Environment) και το ”SDK” (Software Development Kit), πρέπει να «κατεβάσετε» το ”SDK” – το “JRE” δεν είναι αρκετό.

2.1 Εγκατάσταση στα Windows

Το διανεμόμενο αρχείο για τα συστήματα Windows ονομάζεται: *bluejsetup-xxx.exe*, όπου xxx είναι ο αριθμός εκδόσεως. Για παράδειγμα, το διανεμόμενο αρχείο για το BlueJ έκδοση 1.2.0 ονομάζεται *bluejsetup-120.exe*. Μπορεί να έχετε αυτό το αρχείο στο δίσκο ή να το «κατεβάσετε» από την ιστοσελίδα του BlueJ: <http://www.bluej.org>. Εκτελεστε τον «εγκαταστάτη» (installer).

Ο εγκαταστάτης σας προτρέπει να επιλέξετε κατάλογο εγκατάστασης (directory). Επίσης θα σας προσφέρει την επιλογή να εγκαταστήσετε μια συντόμευση (shortcut) στον αρχικό κατάλογο (start menu) και στην επιφάνεια εργασίας (desktop).

Αφού ολοκληρωθεί η εγκατάσταση, θα βρήτε το πρόγραμμα bluej.exe στον κατάλογο εγκατάστασης του BlueJ.

Την πρώτη φορά που θα εκκινήσετε το BlueJ, θα αναζητήσει το σύστημα Java (JDK). Αν βρει περισσότερα από ένα κατάλληλα συστήματα Java (π.χ. έχετε JDK 1.3.1 και JDK 1.4 εγκατεστημένα), ένα παράθυρο θα σας επιτρέψει να επιλέξετε ποιο να χρησιμοποιήσετε. Αν δεν βρει κανένα, θα σας ζητηθεί να το εντοπίσετε μόνοι σας (αυτό μπορεί να συμβεί όταν ένα σύστημα JDK έχει εγκατασταθεί αλλά οι αντίστοιχες εγγραφές στις καταχωρήσεις (registry) έχουν αφαιρεθεί).

Ο εγκαταστάτης του BlueJ εγκαθιστά επίσης ένα αρχείο που ονομάζεται *vmselect.exe*. Χρησιμοποιώντας αυτό το αρχείο, μπορείτε αργότερα να αλλάξετε την έκδοση της Java που χρησιμοποιεί το BlueJ. Εκτελέστε *vmselect* για να εκκινήσετε το BlueJ με μια διαφορετική έκδοση της Java.

Η επιλογή του JDK είναι αποθηκευμένη για κάθε έκδοση του BlueJ. Αν έχετε διαφορετικές εκδόσεις του BlueJ εγκατεστημένες, μπορείτε να χρησιμοποιήσετε μία έκδοση του BlueJ με JDK 1.3.1 και μία άλλη έκδοση του BlueJ με JDK 1.4. Αλλάζοντας την έκδοση της Java για το BlueJ θα αλλάξει όλες τις εγκαταστάσεις του BlueJ της ίδιας έκδοσης για τον ίδιο χρήστη.

2.2 Εγκατάσταση σε Macintosh

Προσέξτε ότι το BlueJ τρέχει μόνο στο MacOS X.

Το διανεμόμενο αρχείο για MacOS X ονομάζεται: *BlueJ-xxx.sit*, όπου xxx είναι ο αριθμός εκδόσεως. Για παράδειγμα, το διανεμόμενο αρχείο για το BlueJ έκδοση 1.2.0 ονομάζεται *BlueJ-120.sit*. Μπορεί να έχετε αυτό το αρχείο στο δίσκο ή να το «κατεβάσετε» από την ιστοσελίδα του BlueJ: <http://www.bluej.org>.

Το αρχείο μπορεί να αποσυμπιεστεί με το *StuffIt Expander*. Πολλά προγράμματα πλοήγησης (browsers) θα αποσυμπιέσουν αυτό το αρχείο για σας. Αλλιώς, ένα διπλό πάτημα (double-click) θα το αποσυμπιέσει.

Μετά την αποσυμπίεση, θα βρείτε ένα κατάλογο με το όνομα BlueJ-xxx. Μεταφέρετε αυτό τον κατάλογο στον κατάλογο Εφαρμογών (Applications) (ή όπου θέλετε να τον κρατήσετε). Περαιτέρω εγκατάσταση δεν είναι απαραίτητη.

2.3 Εγκατάσταση σε Linux/Unix και άλλα συστήματα

Το γενικό διανεμόμενο αρχείο είναι ένα εκτελέσιμο αρχείο jar. Ονομάζεται: *bluej-xxx.jar*, όπου *xxx* είναι ο αριθμός εκδόσεως. Για παράδειγμα, το διανεμόμενο αρχείο για το BlueJ εκδοση 1.2.0 ονομάζεται *bluej-120.jar*. Μπορεί να έχετε αυτό το αρχείο στο δίσκο ή να το «κατεβάσετε» από την ιστοσελίδα του BlueJ: <http://www.bluej.org>.

‘Εκτελέστε’ τον εγκαταστάτη εκτελώντας την παρακάτω εντολή. ΠΡΟΣΟΧΗ: Για αυτό το παράδειγμα, χρησιμοποιώ το διανεμόμενο αρχείο *bluej-120.jar* – εσείς πρέπει να χρησιμοποιήσετε το αρχείο που έχετε (με το σωστό αριθμό εκδόσεως).

```
<jdk-path>/bin/java -jar bluej-120.jar
```

<jdk-path> είναι ο κατάλογος, όπου είναι εγκατεστημένο το JDK.

Ένα «παράθυρο» θα ανοίξει, που θα σας επιτρέψει να επιλέξετε τον κατάλογο εγκατάστασης και τον αριθμό εκδόσεως του JDK που θα χρησιμοποιηθούν από το BlueJ. Σημείωση: Το μονοπάτι (path) για το BlueJ (και κάθε γονικός (parent) κατάλογος) δεν πρέπει να περιέχει κενά.

Πατήστε *Install*. Αφού τελειώσει, το BlueJ πρέπει να έχει εγκατασταθεί.

2.4 Προβλήματα εγκατάστασης

Αν έχετε προβλήματα, ελέγξτε τις Συχνά Τιθέμενες Ερωτήσεις (FAQ) στην ιστοσελίδα του BlueJ (<http://www.bluej.org/help/faq.html>) και διαβάστε την ενότητα *Πως Να Ζητώ Βοήθεια* (<http://www.bluej.org/help/ask-help.html>).

3 Ξεκινώντας – διόρθωση / μεταγλώττιση / εκτέλεση (edit/compile/execute)

3.1 Ξεκινώντας το BlueJ

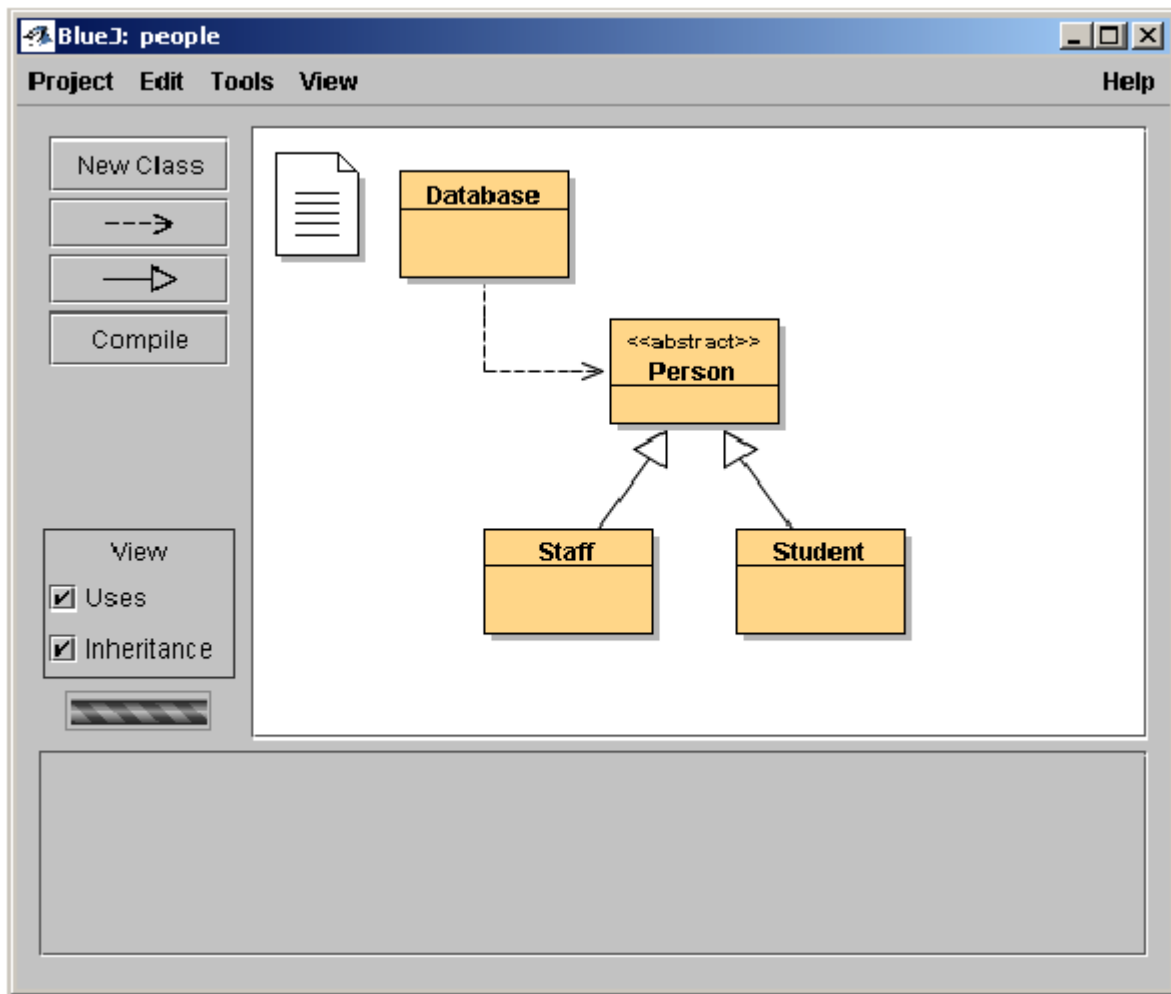
Στα Windows και στο MacOS, το πρόγραμμα *BlueJ* είναι εγκατεστημένο. Εκτελέστε το.

Στο Unix ο εγκαταστάτης εγκαθιστά ένα αρχείο εντολών (script) που ονομάζεται *bluej* στον κατάλογο εγκατάστασης. Από το γραφικό περιβάλλον (GUI), απλά κάντε διπλό πάτημα στο αρχείο. Από τη γραμμή εντολής μπορείτε να ξεκινήσετε το BlueJ με ή χωρίς κάποιο πρόγραμμα (project) ως όρισμα (argument):

```
$ bluej
```

ή

```
$ bluej examples/people
```



Εικόνα 1: Το κύριο παράθυρο του BlueJ

3.2 Ανοίγοντας ένα πρόγραμμα (project)

Περίληψη: Για να ανοίξετε ένα πρόγραμμα, επιλέξτε *Open* από τον κατάλογο (menu) *Project*.

Προγράμματα BlueJ, όπως τα πρότυπα πακέτα (standard packages) της Java, είναι κατάλογοι που περιέχουν αρχεία τα οποία περιλαμβάνονται στα προγράμματα.

Αφού ξεκινήσετε το BlueJ, χρησιμοποιείστε την εντολή *Project – Open...* για να επιλέξετε και να ανοίξετε ένα πρόγραμμα.

Μερικά παραδείγματα περιλαμβάνονται στην πρότυπη διανομή του BlueJ στον κατάλογο *examples*.

Για αυτή την ενότητα του εγχειριδίου, ανοίξτε το πρόγραμμα *people*, το οποίο περιλαμβάνεται σε αυτό τον κατάλογο. Θα βρείτε τον κατάλογο *examples* στον βασικό κατάλογο αρχείων (home directory) του BlueJ. Αφού ανοίξετε το πρόγραμμα θα πρέπει να δείτε κάτι παρόμοιο με το παράθυρο της Εικόνας 1. Το παράθυρο ίσως να μην φαίνεται ακριβώς το ίδιο στο σύστημα σας, αλλά οι διαφορές πρέπει να είναι ελάχιστες.

3.3 Δημιουργώντας αντικείμενα

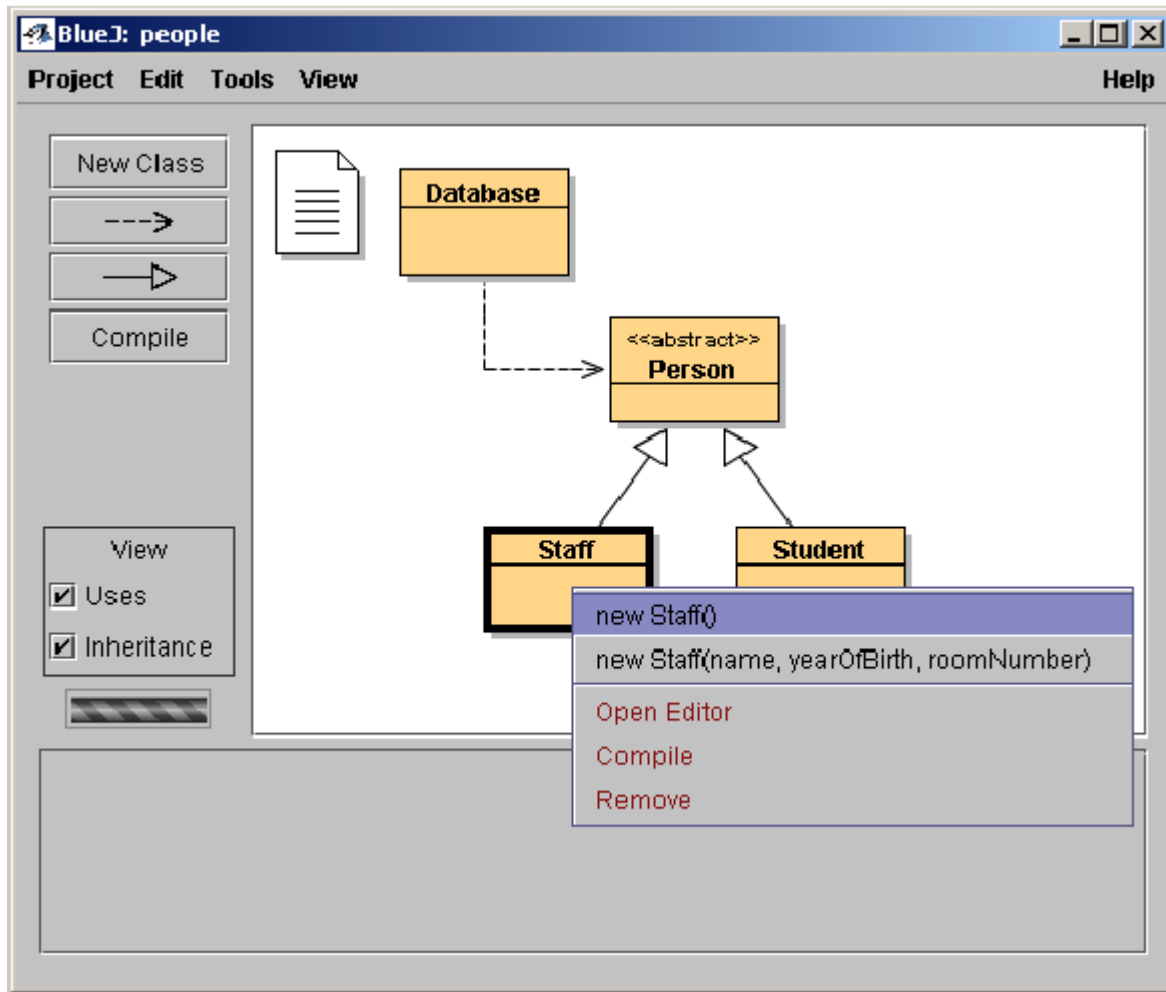
Περίληψη: Για να δημιουργήσετε ένα αντικείμενο, επιλέξτε ένα κατασκευαστή (*constructor*) από τον εμφανιζόμενο κατάλογο (*popup menu*) της κλάσης (*class*).

Ένα από τα θεμελιώδη χαρακτηριστικά του BlueJ είναι ότι όχι μόνο μπορείτε να εκτελέσετε μία πλήρη εφαρμογή, αλλά μπορείτε επίσης άμεσα να αλληλεπιδράσετε με ξεχωριστά αντικείμενα οποιασδήποτε κλάσης και να εκτελέσετε τις δημόσιες (*public*) μεθόδους τους. Μία εκτέλεση στο BlueJ συνήθως γίνεται δημιουργώντας ένα αντικείμενο και ύστερα να καλέσετε μία από τις μεθόδους του αντικειμένου. Αυτό είναι πολύ χρήσιμο κατά τη διάρκεια της εφαρμογής – μπορείτε να δοκιμάσετε τις κλάσεις ξεχωριστά μόλις γράφονται. Δεν είναι απαραίτητο να γράψετε μία πλήρη εφαρμογή πρώτα.

Υποσημείωση: Στατικές μέθοδοι μπορούν να εκτελεστούν άμεσα, χωρίς προηγουμένως να δημιουργηθεί αντικείμενο. Μία από τις στατικές μεθόδους ίσως να είναι η «*main()*», οπότε μπορείτε να κάνετε αυτό που συχνά συμβαίνει με τις εφαρμογές Java – να ξεκινήσετε μία εφαρμογή εκτελώντας απλά μία στατική «*main()*» μέθοδο. Θα επιστρέψουμε σε αυτό αργότερα. Πρώτα, θα κάνουμε κάποια άλλα, πιο ενδιαφέροντα πράγματα τα οποία δεν μπορούν να γίνουν κανονικά σε περιβάλλον Java.

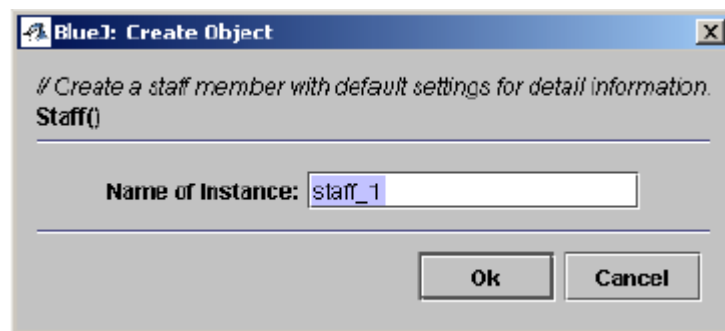
Τα τετράγωνα που βλέπετε στο κεντρικό μέρος του κυρίως παραθύρου (ονομαζόμενα *Database*, *Person*, *Staff* και *Student*) είναι εικονίδια που αντιπροσωπεύουν τις κλάσεις που συμμετέχουν στην εφαρμογή. Μπορείτε να δείτε ένα κατάλογο με τις εφαρμόσιμες διαδικασίες μίας κλάσης πατώντας πάνω στο εικονίδιο της κλάσης με το δεξί κουμπί του ποντικιού (Macintosh: *ctrl-click*¹) (Εικόνα 2). Οι διαδικασίες που εμφανίζονται είναι νέες διαδικασίες με κάθε ένα από τους ένα κατασκευαστές που καθορίζονται για αυτήν την κλάση (πρώτη) που ακολουθείται από μερικές διαδικασίες που παρέχονται από το περιβάλλον.

¹Όποτε αναφερόμαστε σε δεξί κουμπί σε αυτό το εγχειρίδιο, οι χρήστες Macintosh θα πρέπει να το διαβάζουν ως *ctrl-click*.



Εικόνα 2: Διαδικασίες κλάσης (εμφανιζόμενος κατάλογος)

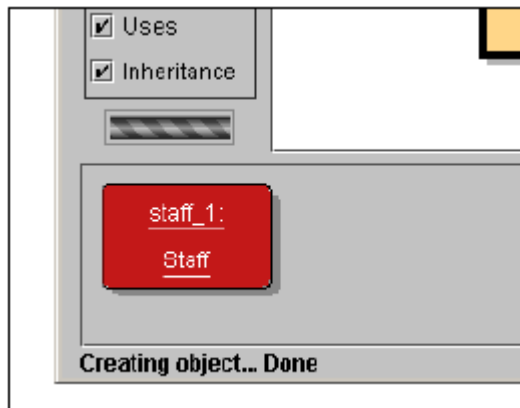
Θέλουμε να δημιουργήσουμε ένα αντικείμενο *Staff* (προσωπικού), έτσι εσείς πρέπει να κάνετε διπλό πατημα στο εικονίδιο *Staff* (που υπερεμφανίζει τον κατάλογο επιλογής που φαίνεται στην Εικόνα 2). Ο κατάλογος επιλογής εμφανίζει δύο κατασκευαστές για να δημιουργήσει ένα αντικείμενο *Staff*, ενός με παραμέτρους και ενός χωρίς. Πρώτα, επιλέξτε τον κατασκευαστή χωρίς παραμέτρους. Το παράθυρο διαλόγου (dialogue) που φαίνεται στην Εικόνα 3 εμφανίζεται.



Εικόνα 3: Δημιουργία κατασκευαστή χωρίς παραμέτρους

Αυτό το παράθυρο διαλόγου σας ρωτά το όνομα για το αντικείμενο που θα δημιουργηθεί. Συγχρόνως, ένα προκαθορισμένο όνομα (*staff_1*) προτείνεται. Αυτό το προκαθορισμένο όνομα είναι αρκετά καλό για τώρα, οπότε χτυπήστε *OK*. Ένα αντικείμενο *Staff* θα δημιουργηθεί.

Μόλις δημιουργηθεί το αντικείμενο τοποθετείται στον πάγκο (ή πλαίσιο) αντικειμένων (Εικόνα 4). Αυτό είναι ότι έχει να κάνει με τη δημιουργία αντικειμένων: Επιλέξτε ένα κατασκευαστή από τον κατάλογο επιλογής κλάσης, εκτελέστε το και έχετε το αντικείμενο στον πάγκο αντικειμένων.



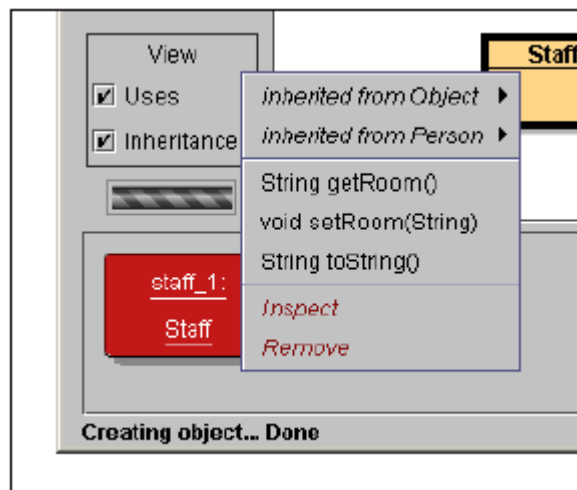
Εικόνα 4: Ένα αντικείμενο στον πάγκο αντικειμένων

Πιθανόν να έχετε παρατηρήσει ότι η κλάση *Person* (άτομο) ονομάζεται `<<abstract>>` (`<<αφηρημένη>>`) (είναι μια αφηρημένη κλάση). Θα παρατηρήσετε (εάν δοκιμάσετε) ότι δεν μπορείτε να δημιουργήσετε αντικείμενα αφηρημένων κλάσεων (όπως καθορίζει η προδιαγραφή της γλώσσας Java).

3.4 Εκτέλεση

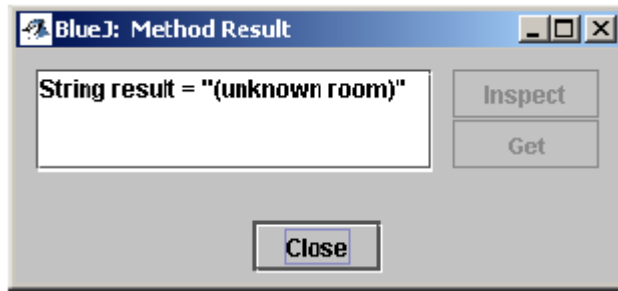
Περίληψη: Για να εκτελέσετε μια μέθοδο, επιλέξτε την από τον εμφανιζόμενο κατάλογο αντικειμένων.

Τώρα που έχετε δημιουργήσει ένα αντικείμενο, μπορείτε να εκτελέσετε τις δημόσιες διαδικασίες του. (Η Java καλεί τις διαδικασίες *μεθόδους*.) Πατήστε με το δεξί κουμπί του ποντικιού πάνω από το αντικείμενο και ένας κατάλογος επιλογής με τις διαδικασίες αντικειμένων θα εμφανιστεί (Εικόνα 5). Ο κατάλογος επιλογής εμφανίζει τις μεθόδους διαθέσιμες για αυτό το αντικείμενο και δύο ειδικές διαδικασίες που παρέχονται από το περιβάλλον (*Inspect* (επιθεωρήστε) και *Remove* (αφαιρέστε)). Θα συζητήσουμε γι' αυτές αργότερα. Πρώτα, ας συγκεντρώθουμε στις μεθόδους.



Εικόνα 5: Ο κατάλογος επιλογής αντικειμένων

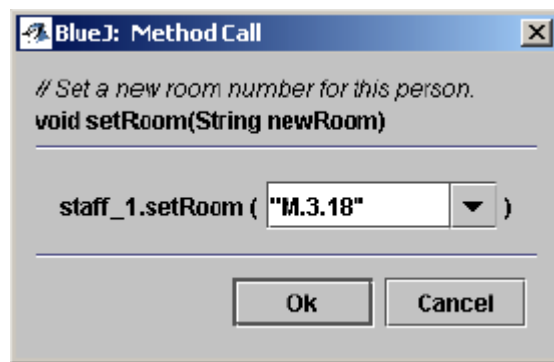
Βλέπετε ότι υπάρχουν μέθοδοι *getRoom* και *setRoom* που θέτουν και επιστρέφουν τον αριθμό δωματίου για αυτό το μέλος προσωπικού. Προσπαθήστε να καλέσετε την *getRoom*. Απλά επιλέξτε την από τον κατάλογο επιλογής του αντικειμένου και θα εκτελεσθεί. Ένα παράθυρο διαλόγου εμφανίζεται δείχνοντας σας το αποτέλεσμα της κλήσης (Εικόνα 6). Σε αυτήν την περίπτωση το όνομα λέει "(unknown room)" ("άγνωστο δωμάτιο") επειδή δεν προσδιορίσαμε ένα δωμάτιο για αυτό το άτομο.



Εικόνα 6: Παρουσίαση του αποτελέσματος τις μεθόδου

Οι μέθοδοι που κληρονομούνται από μία υπερκλάση (superclass) είναι διαθέσιμες μέσω ενός δευτερεύοντος καταλόγου επιλογής. Στην κορυφή του εμφανιζόμενου καταλόγου επιλογής του αντικειμένου υπάρχουν δύο δευτερεύοντες κατάλογοι επιλογής, ένας για τις μεθόδους που κληρονομούνται από την κλάση *Object* και ένας για εκείνους από την κλάση *Person* (Εικόνα 5). Μπορείτε να καλέσετε τις μεθόδους της κλάσης *Person* (όπως *getName*) από την επιλογή τους από τον δευτερεύοντα κατάλογο επιλογής. Δοκιμάστε το. Θα παρατηρήσετε ότι η απάντηση είναι (επίσης) εξίσου ασαφής: απαντά "(unknown name)" ("άγνωστο όνομα"), επειδή δεν έχουμε δώσει στο άτομο μας ένα όνομα.

Τώρα προσπαθήστε να προσδιορίσετε έναν αριθμό δωματίων. Αυτό θα σας δείξει πώς κάνετε μια κλήση που έχει παραμέτρους. (Οι κλήσεις σε *getRoom* και *getName* είχαν τιμές επιστροφής, αλλά καμία παράμετρο). Καλέστε τη μέθοδο *setRoom* επιλέγοντας την από τον κατάλογο επιλογής. Ένα παράθυρο διαλόγου εμφανίζεται προτρέποντας σας να εισάγετε μια παράμετρο (Εικόνα 7).



Εικόνα 7: Παράθυρο κλήσης μεθόδου με παραμέτρους

Στην κορυφή, αυτό το παράθυρο διαλόγου εμφανίζει τη διεπαφή (interface) της μεθόδου για να κληθεί (συμπεριλαμβανομένου του σχολίου και της υπογραφής). Κάτω από αυτόν είναι ένα πεδίο εισόδου κειμένου όπου μπορείτε να εισάγετε την παράμετρο. Η υπογραφή στην κορυφή μας λέει ότι μια παράμετρος τύπου *String* αναμένεται. Εισάγετε το νέο δωμάτιο σαν κείμενο (συμπεριλαμβανομένων των αποσπασμάτων (quotes)) στο πεδίο κειμένου και πατήστε *OK*.

Αυτό είναι όλο - δεδομένου ότι αυτή η μέθοδος δεν επιστρέφει καμία παράμετρο δεν υπάρχει κανένα αποτέλεσμα στο παράθυρο διαλόγου. Καλέστε ξανά την *getRoom* για να ελέγξετε ότι το δωμάτιο έχει πραγματικά αλλάξει.

Πειραματιστήτε με τη δημιουργία αντικειμένων και την κλήση μεθόδων για λίγο. Προσπαθήστε να καλέσετε ένα κατασκευαστή με ορίσματα και καλέστε μερικές ακόμα μεθόδους έως ότου είστε εξοικειωμένοι με αυτές τις διαδικασίες.

3.5 Διορθώνοντας μία κλάση

Περίληψη: Για να διορθώσετε τον πηγαίο κώδικα μίας κλάσης, κάντε διπλό πάτημα στο εικονίδιο της κλάσης.

Μέχρι τώρα, έχουμε ασχοληθεί μόνο με τη διεπαφή (interface) ενός αντικειμένου. Τώρα έφτασε η στιγμή να κοιτάξουμε βαθύτερα. Μπορείτε να δείτε την εφαρμογή μιας κλάσης επιλέγοντας *Open Editor* από τις διαδικασίες κλάσης. (Υπενθύμιση: το διπλό πάτημα του εικονιδίου κλάσης εμφανίζει τις διαδικασίες της

κλάσης.) Το διπλό χτυπημα του εικονιδίου κλάσης είναι ένας συντομότερος δρόμος στην ίδια λειτουργία. Ο διορθωτής δεν περιγράφεται με πολλή λεπτομέρεια σε αυτό το εγχειρίδιο, αλλά μάλλον θα το βρείτε πολύ απλό στη χρήση. Λεπτομέρειες για τον διορθωτή θα δοθούν ξεχωριστά, αργότερα. Για τώρα, ανοίξτε την εφαρμογή της κλάσης *Staff*. Βρείτε την εφαρμογή της μεθόδου *getRoom*. Επιστρέφει, όπως το όνομα υπαινίσσεται, τον αριθμό δωματίου του μέλους προσωπικού. Αλλάξτε τη μέθοδο με την προσθήκη του προθέματος "room" στο αποτέλεσμα λειτουργίας (έτσι ώστε η μέθοδος να επιστρέφει, για παράδειγμα, "room M.3.18" αντί του ακριβούς "M.3.18"). Μπορούμε να το κάνουμε αυτό με την αλλαγή της γραμμής

```
return room;
```

σε

```
return "room " + room;
```

Το BlueJ υποστηρίζει την πλήρη, χωρίς τροποποιήσεις, Java, έτσι δεν υπάρχει τίποτα το ιδιαίτερο για το πως εφαρμόζετε τις κλάσεις σας.

3.6 Μεταγλώττιση

Περίληψη: Για να μεταγλωττίσετε μια κλάση, χτυπήστε το κουμπί *Compile* στον διορθωτή. Για να μεταγλωττίσετε ένα πρόγραμμα, χτυπήστε το κουμπί *Compile* στο παράθυρο προγράμματος (*Project*).

Αφού εισάγετε το κείμενο (προτού να κάνετε τίποτ' άλλο), ελέγξτε την επισκόπηση προγράμματος (το βασικό παράθυρο). Θα παρατηρήσετε ότι το εικονίδιο κλάσης της κλάσης *Staff* έχει αλλάξει: είναι ριγωτό τώρα. Η ριγωτή εμφάνιση χαρακτηρίζει τις κλάσεις που δεν έχουν μεταγλωττιστεί από την τελευταία αλλαγή. Πίσω στον επεξεργαστή.

Υποσημείωση: Πιθανόν να αναρωτηθείτε γιατί τα εικονίδια κλάσης δεν ήταν ριγωτά όταν ανοίξατε αρχικά αυτό το πρόγραμμα. Αυτό γίνεται επειδή οι κλάσεις στο πρόγραμμα *people* διανεμήθηκαν ήδη μεταγλωττισμένες. Συχνά τα προγράμματα *BlueJ* διανέμονται χωρίς να έχουν μεταγλωττιστεί, έτσι αναμένετε να δείτε τα περισσότερα εικονίδια κλάσης ριγωτά όταν ανοίγετε αρχικά ένα πρόγραμμα από τώρα και στο εξής.

Στην εργαλειοθήκη (toolbar) στη κορυφή του διορθωτή υπάρχουν μερικά κουμπιά με τις συχνά χρησιμοποιούμενες λειτουργίες. Ένα από αυτά είναι το *Compile* (Μεταγλώττισε). Αυτή η λειτουργία σας αφήνει να μεταγλωττίσετε αυτήν την κλάση άμεσα, μέσα από τον διορθωτή. Πατήστε το κουμπί *Compile* τώρα. Εάν δεν κάνατε κανένα λάθος, ένα μήνυμα πρέπει να εμφανιστεί στην περιοχή πληροφοριών στο κατώτατο σημείο του διορθωτή που σας ειδοποιεί ότι η κλάση έχει μεταγλωττιστεί. Εάν κάνατε ένα λάθος που προέρχεται από ένα σφάλμα σύνταξης, η γραμμή του σφάλματος τονίζεται και ένα μήνυμα σφάλματος παρουσιάζεται στην περιοχή πληροφοριών. (Σε περίπτωση που η μεταγλώττιση ήταν επιτυχής την πρώτη φορά, προσπαθείστε να εισάγετε ένα σφάλμα σύνταξης - όπως μια ελλείπουσα άνω τελεία - και μεταγλωττίστε πάλι, ακριβώς για να δείτε πως φαίνεται).

Αφότου έχετε μεταγλωττίσει επιτυχώς την κλάση, κλείστε τον διορθωτή.

Υποσημείωση: Δεν είναι αναγκαίο να σώζετε ρητά τον πηγαίο κώδικα κλάσης. Ο κώδικας σώζεται αυτόματα όποτε θεωρήται απαραίτητο (π.χ. όταν κλείνει ο διορθωτής ή προτού να μεταγλωττιστεί μια κλάση). Μπορείτε να σώσετε εάν προτιμάτε (υπάρχει μια λειτουργία στον κατάλογο επιλογής κλάσης του διορθωτή (Class)), αλλά είναι απολύτως απαραίτητο μόνο εάν το σύστημα σας είναι πραγματικά ασταθές και κρεμάει (*crash*) συχνά και είστε ανήσυχος για την απώλεια της εργασίας σας.

Η εργαλειοθήκη του παραθύρου του προγράμματος έχει επίσης ένα κουμπί *Compile*. Αυτή η λειτουργία μεταγλώττισης μεταγλωττίζει ολόκληρο το πρόγραμμα. (Στην πραγματικότητα, καθορίζει ποιες κλάσεις χρειάζονται αναμεταγλώττιση και έπειτα τις αναμεταγλωττίζει στη σωστή σειρά.) Δοκιμάστε το αλλάζοντας δύο ή περισσότερες κλάσεις έτσι ώστε αυτές να εμφανίζονται ριγωτές στο διάγραμμα κλάσης και έπειτα πατήστε το κουμπί *Compile*. Εάν ανιχνευτεί κάποιο σφάλμα σε μια από τις μεταγλωττισμένες κλάσεις, θα ανοίξει ο διορθωτής και θα παρουσιαστούν η θέση και το μήνυμα σφάλματος.

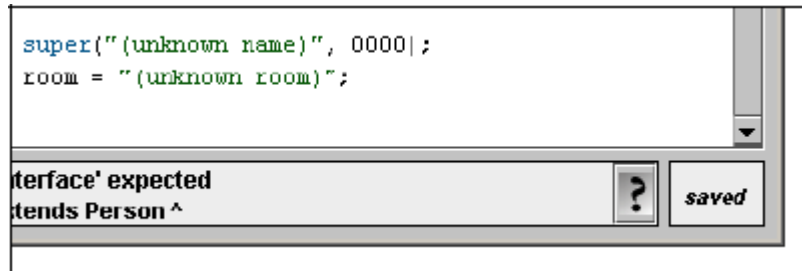
Θα παρατηρήσετε ότι ο πάγκος αντικειμένου είναι κενός πάλι. Τα αντικείμενα αφαιρούνται κάθε φορά που αλλάζει η εφαρμογή τους (implementation).

3.7 Βοήθεια με τα σφάλματα μεταγλωττιστών

Περίληψη: Για να δείτε τις οδηγίες για ένα μήνυμα σφάλματος μεταγλωττιστή, πατήστε το ερωτηματικό δίπλα στο μήνυμα σφάλματος.

Πολύ συχνά, αρχάριοι σπουδαστές έχουν δυσκολίες να καταλάβουν τα μηνύματα σφάλματος των μεταγλωττιστών. Θα προσπαθήσουμε να βοηθήσουμε με μερικές οδηγίες.

Ανοίξτε πάλι τον διορθωτή, εισάγετε ένα σφάλμα στο πηγαίο αρχείο, και μεταγλωττίστε. Ένα μήνυμα σφάλματος πρέπει να παρουσιαστεί στην περιοχή πληροφοριών του διορθωτή. Στο δεξί άκρο της περιοχής πληροφοριών ένα ερωτηματικό εμφανίζεται το οποίο μπορείτε να πατήσετε για να πάρετε περισσότερες πληροφορίες για αυτόν τον τύπο σφάλματος (Εικόνα 8).



Εικόνα 8: Ένα σφάλμα μεταγλωττιστή και το κουμπί *Help* (Βοήθεια)

Σε αυτή τη φάση, κείμενα οδηγιών δεν είναι διαθέσιμα για όλα τα μηνύματα σφάλματος. Κάποια κείμενα οδηγιών δεν έχουν γραφτεί ακόμα. Αλλά αξίζει να προσπαθήσετε - πολλά σφάλματα ήδη εξηγούνται. Τα υπόλοιπα θα γραφούν και θα συμπεριληφθούν σε κάποια μελλοντική έκδοση του BlueJ.

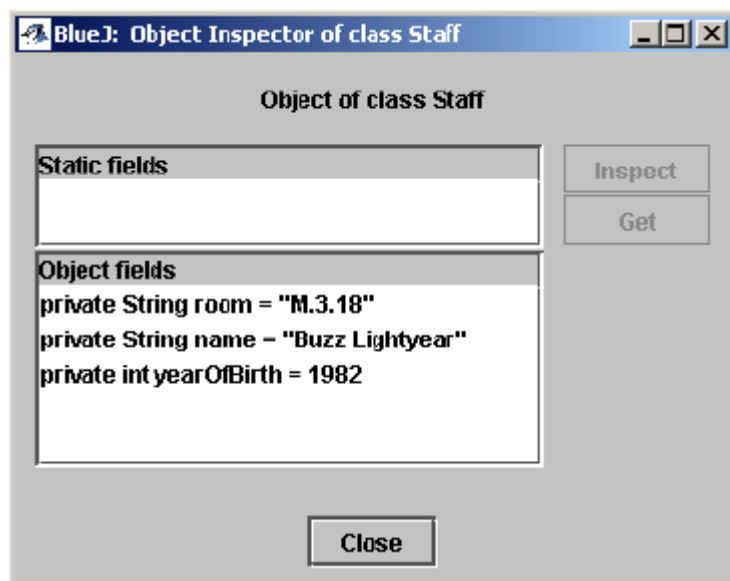
4 Κάνοντας κάτι παραπάνω...

Σε αυτό το κεφάλαιο, θα εξηγήσουμε κάποια επιπλέον πράγματα που μπορείτε να κάνετε στο περιβάλλον. Πράγματα που δεν είναι τόσο ουσιώδη, αλλά πολύ συχνά χρησιμοποιούμενα.

4.1 Εξέταση

Περίληψη: Η εξέταση αντικειμένων επιτρέπει κάποια απλή εξουδετέρωση σφαλμάτων εμφανίζοντας την εσωτερική κατάσταση ενός αντικειμένου.

Όταν εκτελέσατε τις μεθόδους ενός αντικειμένου, ίσως να παρατηρήσατε τη λειτουργία *Inspect* (Εξέταση) που είναι διαθέσιμη στα αντικείμενα εκτός από τις μεθόδους καθορισμένες από τον χρήστη (Εικόνα 5). Αυτή η λειτουργία επιτρέπει τον έλεγχο της κατάστασης των μεταβλητών δημιουργίας (instance variables) ("fields" ("πεδία")) των αντικειμένων. Δοκιμάστε να δημιουργήσετε ένα αντικείμενο με κάποιες τιμές καθορισμένες από τον χρήστη (π.χ. ένα αντικείμενο *Staff* με τον κατασκευαστή που έχει παραμέτρους). Κατόπιν επιλέξτε *Inspect* από τον κατάλογο επιλογής αντικειμένου. Ένα διαλογικό παράθυρο εμφανίζεται παρουσιάζοντας τα πεδία του αντικειμένου, τους τύπους τους και τις τιμές τους (Εικόνα 9).



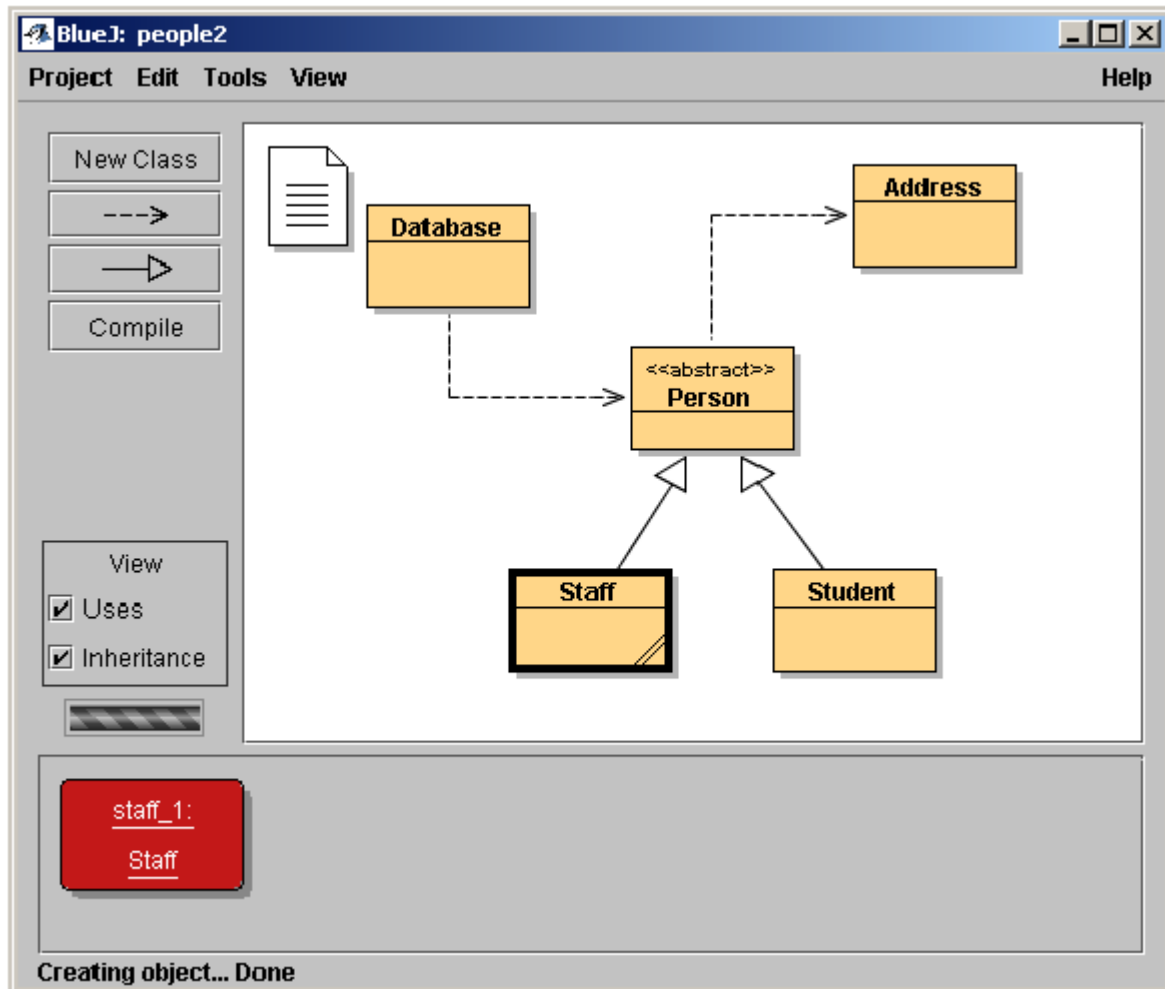
Εικόνα 9: Διαλογικό παράθυρο εξέτασης

Η εξέταση είναι χρήσιμη για να ελέγξουμε γρήγορα εάν μια λειτουργία θέσης (mutator) (μια λειτουργία που αλλάζει (θέτει) την κατάσταση του αντικειμένου) εκτελέστηκε σωστά. Κατά συνέπεια, η «εξέταση» είναι ένα απλό εργαλείο εξουδετέρωσης σφαλμάτων.

Στο παράδειγμα *Staff*, όλα τα πεδία είναι απλοί τύποι (είτε τύποι μη-αντικειμένου είτε συμβολοσειρές (*String*)). Η τιμή αυτών των τύπων μπορεί να εμφανιστεί άμεσα. Μπορείτε αμέσως να δείτε εάν ο κατασκευαστής έχει κάνει τις σωστές αναθέσεις (assignments).

Στις πιά σύνθετες περιπτώσεις, οι τιμές των πεδίων μπορεί να είναι αναφορές (references) στα καθορισμένα από τον χρήστη αντικείμενα. Για να εξετάσουμε ένα τέτοιο παράδειγμα θα χρησιμοποιήσουμε ένα άλλο πρόγραμμα.

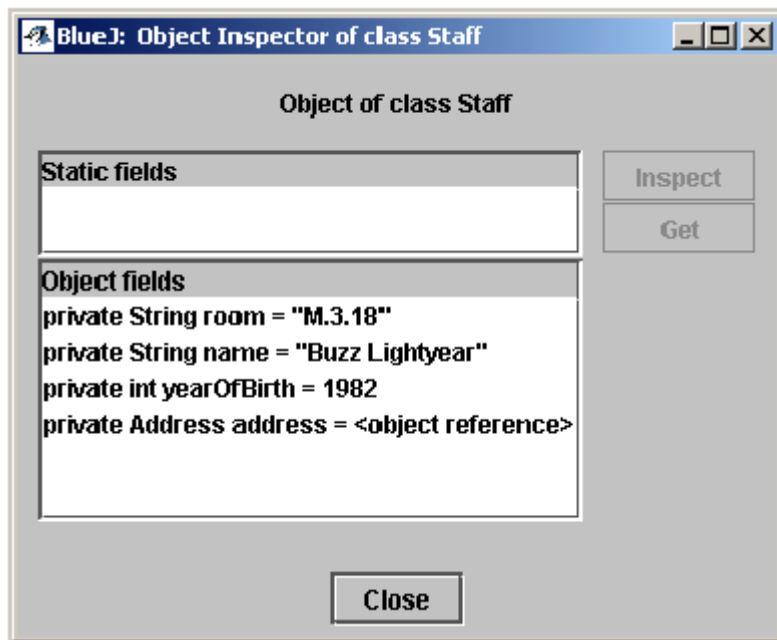
Ανοίξτε το πρόγραμμα *people2*, το οποίο επίσης συμπεριλαμβάνεται στην πρότυπη διανομή του BlueJ. Το πρόγραμμα *people2* φαίνεται στην Εικόνα 10. Όπως μπορείτε να δείτε, αυτό το δεύτερο παράδειγμα έχει μία κλάση *Address* εκτός από τις κλάσεις που είδαμε προηγουμένως. Ένα από τα πεδία της κλάσης *Person* είναι του καθορισμένου από το χρήστη τύπου *Address*.



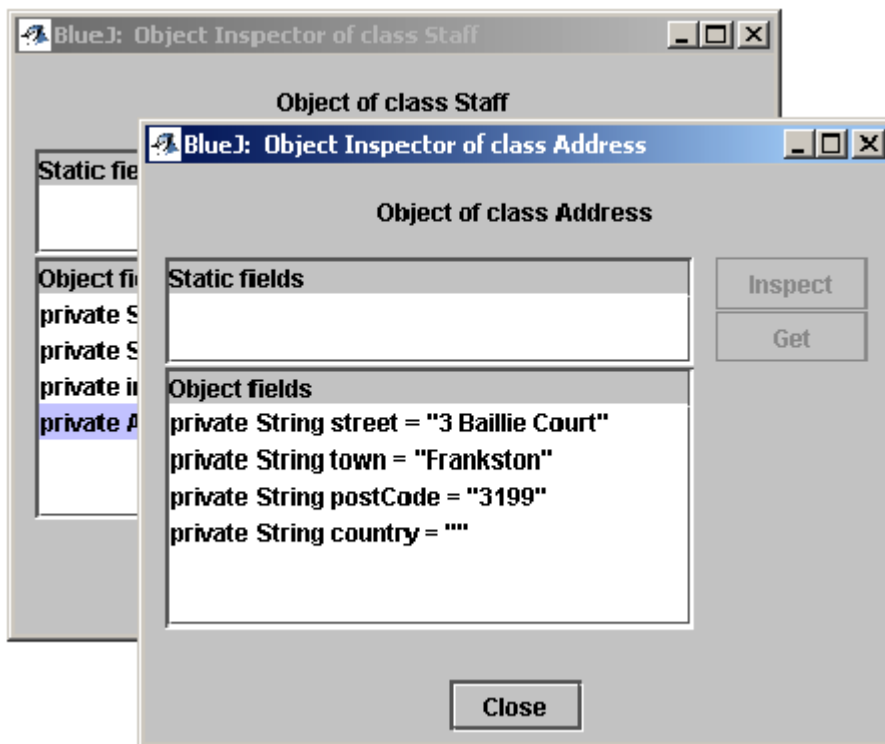
Εικόνα 10: Το παράθυρο προγράμματος *people2*

Για το επόμενο πράγμα που θέλουμε να δοκιμάσουμε (εξέταση με τα πεδία αντικειμένου) δημιουργήστε ένα αντικείμενο *Staff* και καλέστε έπειτα τη μέθοδο *setAddress* αυτού του αντικειμένου (θα την βρείτε στον δευτερεύοντα κατάλογο επιλογής *Person*). Εισάγετε μια διεύθυνση. Εσωτερικά, ο κώδικας του *Staff* δημιουργεί ένα αντικείμενο της κλάσης *Address* και το καταχωρεί στο πεδίο *address* (διεύθυνση).

Τώρα, εξετάστε το αντικείμενο *Staff*. Το διαλογικό παράθυρο επιθεώρησης που προκύπτει φαίνεται στην Εικόνα 11. Τώρα τα πεδία μέσα στο αντικείμενο *Staff* περιλαμβάνουν το πεδίο *address*. Όπως βλέπετε, η τιμή της εμφανίζεται σαν *<object reference>* (<αναφορά αντικειμένου>) - δεδομένου ότι αυτό είναι ένα σύνθετο, καθορισμένο από το χρήστη αντικείμενο, η τιμή της μπορεί να εμφανιστεί άμεσα σε αυτόν τον κατάλογο. Για να εξετάσετε τη διεύθυνση περαιτέρω, επιλέξτε το πεδίο *address* στον κατάλογο και πατήστε το κουμπί *Inspect* στο διαλογικό παράθυρο. (Μπορείτε επίσης να κάνετε διπλό πάτημα στο πεδίο *address*.) Ένα άλλο παράθυρο εξέτασης ανοίγει στη συνέχεια, εμφανίζοντας τις λεπτομέρειες του αντικειμένου *Address* (Εικόνα 12).



Εικόνα 11: Εξέταση με αναφορές αντικειμένων (object reference)



Εικόνα 12: Εξέταση εσωτερικού αντικειμένου

Εάν το επιλεγμένο πεδίο είναι δημόσιο (public) τότε, αντί να πατήσετε το κουμπί *Inspect*, θα μπορούσατε να επιλέξετε το πεδίο *address* και να πατήσετε το κουμπί *Get*. Αυτή η λειτουργία τοποθετεί το επιλεγμένο αντικείμενο στον πάγκο αντικειμένων. Εκεί μπορείτε να το εξετάσετε περαιτέρω κάνοντας κλήσεις στις μεθόδους του.

4.2 Μεταβιβάζοντας αντικείμενα ως παραμέτρους

Περίληψη: Ένα αντικείμενο μπορεί να μεταβιβαστεί ως παράμετρος σε μια κλήση μεθόδου με ένα πάτημα στο εικονίδιο αντικειμένου.

Τα αντικείμενα μπορούν να μεταβιβαστούν ως παράμετροι στις μεθόδους άλλων αντικειμένων. Ας δοκιμάσουμε ένα παράδειγμα. Δημιουργήστε ένα αντικείμενο της κλάσης *Database* (βάσης δεδομένων). (Θα παρατηρήσετε ότι η κλάση *Database* έχει μόνο έναν κατασκευαστή που δεν παίρνει καμία παράμετρο, έτσι η κατασκευή ενός αντικειμένου είναι απλή.) Το αντικείμενο *Database* έχει τη δυνατότητα να κρατήσει έναν κατάλογο προσώπων. Έχει τις διαδικασίες για να προσθέσει τα αντικείμενα προσώπων και για να παρουσιάσει όλα τα πρόσωπα που έχουν καταχωρηθεί. (Η κλήση του *Database* είναι λίγο υπερβολική!).

Εάν δεν έχετε ήδη ένα αντικείμενο *Staff* ή *Student* στον πάγκο αντικειμένων, δημιουργήστε ένα. Για τα ακόλουθα, χρειάζεστε ένα αντικείμενο *Database* και ένα αντικείμενο *Staff* ή *Student* στον πάγκο αντικειμένου συγχρόνως.

Τώρα καλέστε τη μέθοδο *addPerson* του αντικειμένου *Database*. Η υπογραφή (signature) σας λέει ότι αναμένεται μια παράμετρος του τύπου *Person*. (Θυμηθείτε: η κλάση *Person* είναι αφηρημένη, έτσι δεν υπάρχει κανένα αντικείμενο που είναι άμεσα τύπου *Person*. Αλλά, λόγω της δημιουργίας υπο-τύπων, αντικείμενα *Student* και *Staff* μπορούν να αντικατασταθούν με αντικείμενα *Person*. Έτσι είναι νόμιμο να μεταβιβάσει ένα *Student* ή ένα *Staff* όταν αναμένεται ένα *Person*) Για να μεταβιβάσετε το αντικείμενο, που έχετε στον πάγκο αντικειμένων σας, ως παράμετρο στην κλήση που κάνετε, θα μπορούσατε να εισάγετε το όνομά της στο πεδίο παραμέτρου ή, ως συντομότερο δρόμο, απλά να πατήσετε πάνω στο αντικείμενο. Αυτό εισάγει το όνομά του στο διαλογικό παράθυρο κλήσης μεθόδου. Πατήστε *OK* και η κλήση γίνεται. Δεδομένου ότι δεν υπάρχει καμία επιστρεφόμενη τιμή για αυτήν την μέθοδο, δεν βλέπουμε αμέσως κάποιο αποτέλεσμα. Μπορείτε να καλέσετε τη μέθοδο *listAll* στο αντικείμενο *Database* για να ελέγξετε ότι η λειτουργία πραγματικά εκτελέστηκε. Η λειτουργία *listAll* γράφει τις πληροφορίες του προσώπου στην πρότυπη έξοδο (standard output). Θα παρατηρήσετε ότι ένα τερματικό κειμένου (text terminal) ανοίγει αυτόματα για να παρουσιάσει το κείμενο.

Δοκιμάστε το πάλι εισάγοντας περισσότερα από ένα πρόσωπα στη "βάση δεδομένων".

5 Δημιουργία ενός νέου προγράμματος

Το κεφάλαιο αυτό σας κάνει μία περιήγηση της δημιουργίας ενός νέου προγράμματος.

5.1 Δημιουργία του καταλόγου αρχείων προγράμματος

Περίληψη: Για να δημιουργήσετε ένα πρόγραμμα, επιλέξτε New... από τον κατάλογο επιλογών Project.

Για να δημιουργήσει ένα νέο πρόγραμμα, επιλέξτε *Project - New...* από τον κατάλογο επιλογής. Ένα διαλογικό παράθυρο επιλογής αρχείων ανοίγει και σας επιτρέπει να προσδιορίσετε ένα όνομα και μια θέση για το νέο πρόγραμμα. Δοκιμάστε το τώρα. Μπορείτε να επιλέξετε οποιοδήποτε όνομα για το προγράμμα σας. Αφού πατήσετε *OK*, ένας κατάλογος αρχείων θα δημιουργηθεί με το όνομα που προσδιορίσατε, και το βασικό παράθυρο εμφανίζει το νέο, κενό πρόγραμμα.

5.2 Δημιουργώντας κλάσεις

Περίληψη: Για να δημιουργήσετε μία κλάση, πατήστε το κουμπί *New Class* και προσδιορίστε το όνομα της κλάσης.

Τώρα μπορείτε να δημιουργήσετε τις κλάσεις σας με το πάτημα του κουμπιού *New Class* στην εργαλειοθήκη του προγράμματος. Θα σας ζητηθεί να δώσετε ένα όνομα για την κλάση - το όνομα αυτό πρέπει να είναι ένα έγκυρο προσδιοριστικό (identifier) της Java.

Μπορείτε επίσης να επιλέξετε ανάμεσα από τέσσερις τύπους κλάσεων: αφηρημένη, διεπαφή, applet ή «συνήθης» (abstract, interface, applet or “standard”). Αυτή η επιλογή καθορίζει ποιος σκελετός κώδικα δημιουργείται αρχικά για την κλάση σας. Μπορείτε να αλλάξετε τον τύπο κλάσης αργότερα διορθώνοντας τον πηγαίο κώδικα (παραδείγματος χάριν, με την προσθήκη της λέξης κλειδιού "abstract" στον κώδικα).

Αφού δημιουργήσετε μια κλάση, στη συνέχεια αντιπροσωπεύεται από ένα εικονίδιο στο διάγραμμα. Εάν δεν είναι μια πρότυπη κλάση, ο τύπος (διεπαφή, αφηρημένη, ή applet) είναι υποδειγμένος στο εικονίδιο κλάσης. Όταν ανοίγετε τον διορθωτή για μια νέα κλάση, θα παρατηρήσετε ότι ένας προκαθορισμένος σκελετός κλάσης έχει δημιουργηθεί - αυτό πρέπει να σας βοηθήσει να ξεκινήσετε. Ο προκαθορισμένος κώδικας είναι συντακτικά σωστός. Μπορεί να μεταγλωττιστεί (αλλά δεν κάνει πολλά πράγματα). Προσπαθήστε να δημιουργήσετε μερικές κλάσεις και να τις μεταγλωττίσετε.

5.3 Δημιουργώντας εξαρτήσεις

Περίληψη: Για να δημιουργήσει ένα βέλος, πατήστε το κουμπί βελών και να σύρτε το βέλος στο διάγραμμα, ή απλά γράψτε τον πηγαίο κώδικα στον διορθωτή.

Το διάγραμμα κλάσης εμφανίζει τις εξαρτήσεις μεταξύ των κλάσεων υπό μορφή βελών. Οι σχέσεις κληρονομιάς («επεκτείνει» ή «εφαρμόζει» (“extends” or “implements”)) εμφανίζονται ως διπλά βέλη. Οι σχέσεις «χρήσεων» (“uses”) εμφανίζονται ως ενιαία (μονά) βέλη.

Μπορείτε να προσθέσετε τις εξαρτήσεις είτε γραφικά (άμεσα στο διάγραμμα) είτε κειμενικά στον πηγαίο κώδικα. Εάν προσθέτετε ένα βέλος γραφικά, ο κώδικας ενημερώνεται αυτόματα. Εάν προσθέτετε την εξάρτηση στον κώδικα, τότε το διάγραμμα ενημερώνεται.

Για να προσθέσετε ένα βέλος γραφικά, πατήστε το κατάλληλο κουμπί βελών (το διπλό βέλος για «επεκτείνει» ή «εφαρμόζει», ενιαίο βέλος για «χρησιμοποιεί») και σύρτε το βέλος από τη μια κλάση στην άλλη.

Η προσθήκη ενός βέλους κληρονομιάς εισάγει τον καθορισμό «επεκτείνει» ή «εφαρμόζει» στον πηγαίο κώδικα της κλάσης (ανάλογα με εάν ο στόχος είναι μια κλάση ή μια διεπαφή).

Η προσθήκη ενός βέλους «χρήσης» δεν αλλάζει αμέσως τον κώδικα (εκτός αν ο στόχος είναι μια κλάση από μια άλλο πακέτο (package)). Σε αυτή την περίπτωση δημιουργεί μια δήλωση «εισαγωγής» (import), αλλά δεν έχουμε δει αυτήν την περίπτωση ακόμα στα παραδείγματά μας. Έχοντας ένα βέλος χρήσης στο διάγραμμα να σκοπεύει μια κλάση που δεν χρησιμοποιείται πραγματικά στον πηγαίο κώδικα της θα παράγει μια προειδοποίηση δηλώνοντας ότι μια σχέση «χρήσης» σε μία κλάση δηλώθηκε αλλά η κλάση δεν χρησιμοποιείται ποτέ.

Η προσθήκη των βελών μέσω κειμένου είναι εύκολη: απλά πληκτρολογήστε τον κώδικα όπως θα κάνατε κανονικά. Μόλις η κλάση σωθεί, το διάγραμμα ενημερώνεται. (Και θυμηθείτε: το κλείσιμο του επεξεργαστή σώζει αυτόματα).

5.4 Αφαιρώντας στοιχεία

Περίληψη: Για να αφαιρέσετε μια κλάση, επιλέξτε τη λειτουργία *Remove* από τον εμφανιζόμενο κατάλογο επιλογής της. Για να αφαιρέσετε ένα βέλος, επιλέξτε *Remove* από το κατάλογο επιλογής *Edit* και πατήστε το βέλος.

Για να αφαιρέσετε μια κλάση από το διάγραμμα, επιλέξτε την κλάση και έπειτα επιλέξτε *Remove Class* από τον κατάλογο επιλογής *Edit*. Μπορείτε επίσης να επιλέξετε *Remove* από τον εμφανιζόμενο κατάλογο επιλογής της κλάσης. Για να αφαιρέσετε ένα βέλος, επιλέξτε *Remove Arrow* από τον κατάλογο επιλογής και έπειτα επιλέξτε το βέλος που θέλετε να αφαιρέσετε.

6 Αποσφαλμάτωση

Αυτό το κεφάλαιο παρουσιάζει τις σημαντικότερες πτυχές της λειτουργίας αποσφαλμάτωσης σε BlueJ. Μιλώντας σε δασκάλους προγραμματισμού, πολύ συχνά ακούμε το σχόλιο ότι η χρησιμοποίηση ενός αποσφαλματωτή στη διδασκαλία του πρώτου έτους θα ήταν χρήσιμη, αλλά δεν υπάρχει καθόλου χρόνος για να παρουσιασθεί. Οι σπουδαστές πασχίζουν με τον επεξεργαστή, το μεταγλωττιστή και την εκτέλεση και δεν υπάρχει καθόλου χρόνος για να παρουσιασθεί άλλο ένα περίπλοκο εργαλείο.

Γι' αυτό αποφασίσαμε να καταστήσουμε τον αποσφαλματωτή όσο το δυνατόν απλούστερο. Στόχος μας είναι να υπάρξει ένας αποσφαλματωτής που μπορείς να εξηγήσεις σε 15 λεπτά, και τον οποίο οι σπουδαστές μπορούν έπειτα να χρησιμοποιήσουν χωρίς περαιτέρω οδηγίες. Ας δούμε εάν έχουμε πετύχει.

Καταρχήν, έχουμε μειώσει τη λειτουργία των παραδοσιακών αποσφαλματωτών σε τρεις στοιχειώδεις εργασίες:

θέση σημείων αναστολής (breakpoints)

«περπάτημα» μέσα στον κώδικα

επιθεώρηση μεταβλητών

Σε αντάλλαγμα, κάθε μια από τις τρεις στοιχειώδεις εργασίες είναι πολύ απλή. Θα δοκιμάσουμε τώρα κάθε μία τους.

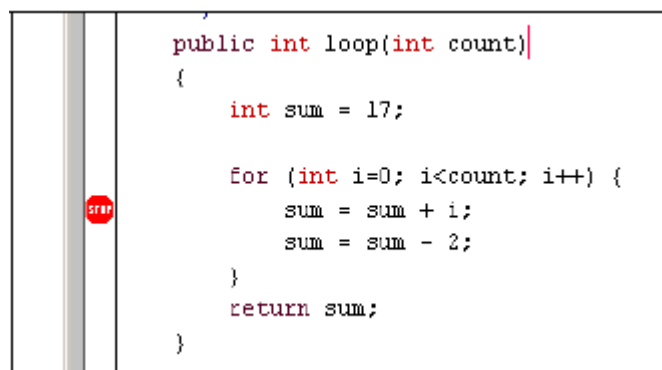
Για να αρχίσετε, ανοίξτε το προγράμμα *debugdemo*, το οποίο συμπεριλαμβάνεται στον κατάλογο *examples* στην διανομή. Αυτό το πρόγραμμα περιέχει μερικές κλάσεις που μοναδικό σκοπό έχουν την παρουσίαση των λειτουργιών του αποσφαλματωτή - ειδάλλως δεν έχουν κανένα νόημα.

6.1 Θέτοντας σημεία αναστολής

Περίληψη: Για να θέσετε ένα σημείο αναστολής, πατήστε στην περιοχή σημείου αναστολής αριστερά του κειμένου στον διορθωτή.

Η θέση ενός σημείου αναστολής σας επιτρέπει να διακόψετε την εκτέλεση σε ένα ορισμένο σημείο στον κώδικα. Όταν η εκτέλεση διακόπτεται, μπορείτε να ερευνήσετε την κατάσταση των αντικειμένων σας. Συχνά σας βοηθά να καταλάβετε τι συμβαίνει στον κώδικα σας.

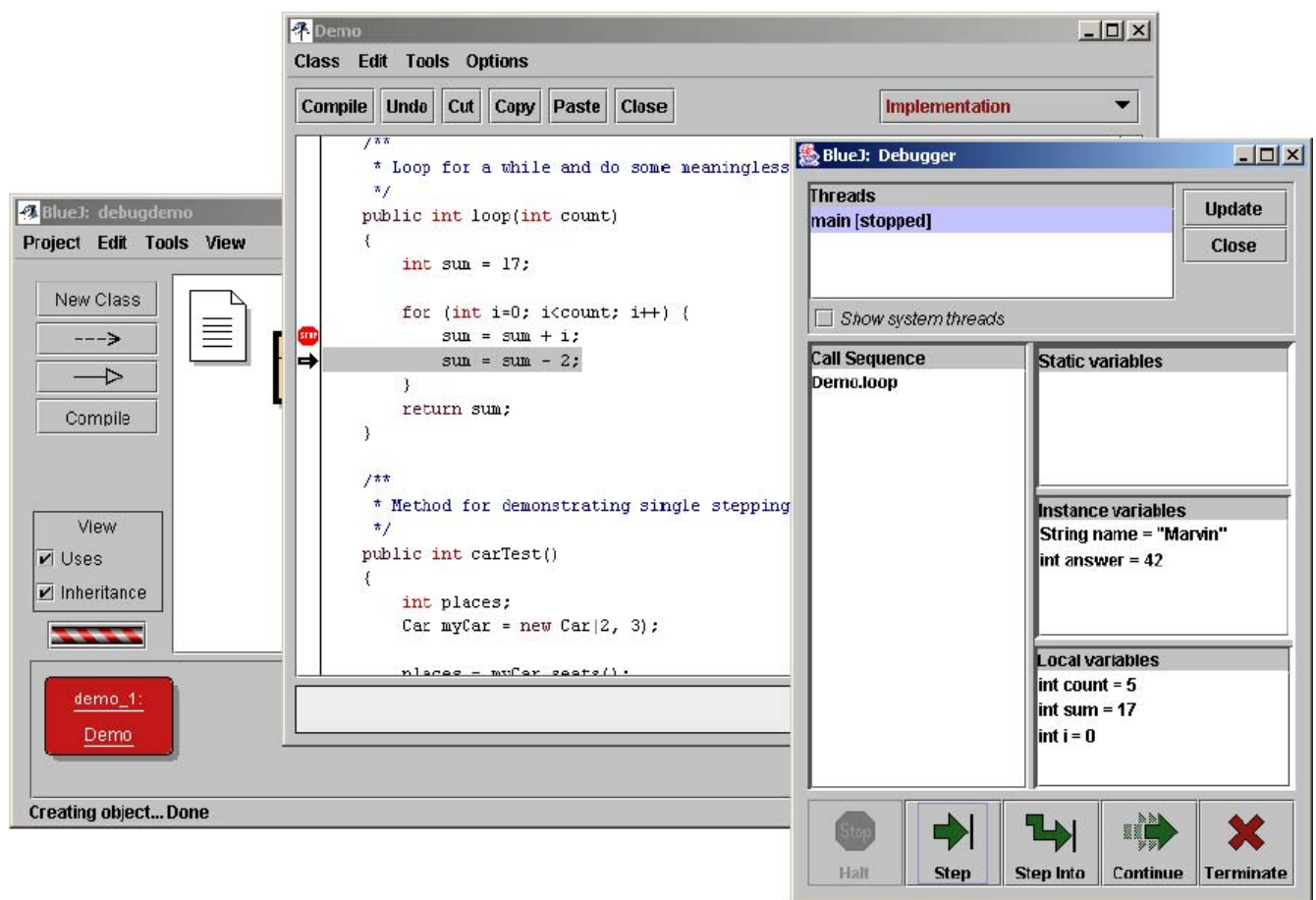
Στον διορθωτή, αριστερά του κειμένου, είναι η περιοχή σημείων αναστολής (Εικόνα 13). Μπορείτε να θέσετε ένα σημείο αναστολής πατώντας πάνω της. Ένα μικρό σημάδι στάσης (stop) εμφανίζεται για να σημαδεύσει το σημείο αναστολής. Δοκιμάστε αυτό τώρα. Ανοίξτε την κλάση *Demo*, βρείτε τη μέθοδο *loop*, και θέστε ένα σημείο αναστολής κάπου στο βρόχο *for*. Το σημάδι στάσης πρέπει να εμφανιστεί στον διορθωτή σας.



Εικόνα 13: Ένα σημείο αναστολής

Όταν φτάσει η γραμμή του κώδικα που έχει το σημείο αναστολής, η εκτέλεση θα διακοπεί. Ας το δοκιμάσουμε τώρα.

Δημιουργήστε ένα αντικείμενο της κλάσης *Demo* και καλέστε τη μέθοδο *loop* με μια παράμετρο, για παράδειγμα, 10. Μόλις φτάσει στο σημείο αναστολής, εμφανίζονται: το παράθυρο του διορθωτή, εμφανίζοντας τη τρέχουσα γραμμή κώδικα, και ένα παράθυρο αποσφαλματωτή. Μοιάζει κάπως σαν την Εικόνα 14.



Εικόνα 14: The debugger window

Η τονισμένη γραμμή στον διορθωτή δείχνει την επόμενη γραμμή που θα εκτελεσθεί (Η εκτέλεση σταματήσει *προτού* να εκτελεσθεί αυτή η γραμμή).

6.2 «Περπατώντας» μέσα στον κώδικα

Περίληψη: Για να κάνετε ένα βήμα μέσα στον κώδικα, χρησιμοποιήστε τα κουμπιά *Step* (βήμα) και *Step Into* (βήμα μέσα) στον αποσφαλματωτή.

Τώρα που έχουμε σταματήσει την εκτέλεση (που μας πείθει ότι η μέθοδος πραγματικά εκτελείται και αυτό το σημείο του κώδικα πραγματικά φτάνεται), μπορούμε να κάνουμε ένα βήμα μέσα στον κώδικα και να δούμε πώς η εκτέλεση προχωρεί. Για να κάνετε αυτό, χτυπήστε επανειλημμένα το κουμπί *Step* στο παράθυρο αποσφαλματωτή. Θα πρέπει να δείτε τη γραμμή πηγαίου κώδικα στον διορθωτή να αλλάζει (Η τονισμένη γραμμή μετακινείται μαζί με τη γραμμή που εκτελείται). Κάθε φορά που πατάτε το κουμπί *Step*, μια γραμμή κώδικα εκτελείται και η εκτέλεση σταματά πάλι. Σημειώστε επίσης ότι οι τιμές των μεταβλητών που παρουσιάζονται στο παράθυρο αποσφαλματωτή αλλάζουν (π.χ. η αξία του *sum* (σύνολο).) Έτσι μπορείτε να εκτελέσετε βαθμιαία και να παρατηρήσετε τι συμβαίνει. Μόλις κουραστείτε με αυτό, μπορείτε να πατήσετε στο σημείο αναστολής πάλι για να το αφαιρέσετε, και έπειτα να πατήσετε το κουμπί *Continue* στο αποσφαλματωτή για να επανεκκινήσετε την εκτέλεση και να συνεχίσετε κανονικά.

Ας το δοκιμάσουμε πάλι με μια άλλη μέθοδο. Θέστε ένα σημείο αναστολής στην κλάση *Demo*, μέθοδο *carTest()*, στη γραμμή που γράφει:

```
places = myCar.seats();
```

Καλέστε τη μέθοδο. Όταν φτάσει το σημείο αναστολής, είστε έτοιμοι να εκτελέσετε μια γραμμή που περιέχει μια κλήση στη μέθοδο `seats()` στην κλάση `Car`. Πατώντας το κουμπί `Step` θα περπατούσε πάνω από ολόκληρη τη γραμμή. Ας δοκιμάσουμε το κουμπί `StepInto` αυτή τη φορά. Εάν «περπατήσετε μέσα» (`step into`) σε μια κλήση μεθόδου, τότε μπαίνετε μέσα στη μέθοδο και εκτελείτε τη μέθοδο εκείνη, γραμμή προς γραμμή (όχι ως ένα βήμα). Σε αυτήν την περίπτωση, μπαίνετε μέσα στη μέθοδο `seats()` της κλάσης `Car`. Μπορείτε τώρα να «περπατήσετε» μέσω αυτής της μεθόδου έως ότου φθάσετε στο τέλος και επιστρέψετε στην μέθοδο κλήσης. Σημειώστε πώς η εμφάνιση του αποσφαλματωτή αλλάζει.

Η λειτουργία `Step` και `Step Into` συμπεριφέρονται όμοια εάν η τρέχουσα γραμμή κώδικα δεν περιέχει κάποια κλήση μεθόδου.

6.3 Επιθεωρώντας μεταβλητές

Περίληψη: Η επιθεώρηση μεταβλητών είναι απλή – εμφανίζονται αυτόματα στον αποσφαλματωτή .

Όταν αποσφαλματώνετε τον κώδικα σας, είναι σημαντικό να είστε σε θέση να επιθεωρήσετε την κατάσταση των αντικειμένων σας (τοπικές μεταβλητές (local) και μεταβλητές δημιουργίας (instance)).

Κάνοντας κάτι τέτοιο είναι απλό - έχετε ήδη δει πως γίνεται. Δεν χρειάζεστε ειδικές εντολές για να επιθεωρήσετε τις μεταβλητές. Οι στατικές μεταβλητές, οι μεταβλητές δημιουργίας του τρέχοντος αντικειμένου και οι τοπικές μεταβλητές της τρέχουσας μεθόδου πάντα παρουσιάζονται και ενημερώνονται αυτόματα.

Μπορείτε να επιλέξετε μεθόδους στην ακολουθία κλήσεων για να δείτε μεταβλητές άλλων ενεργών, αυτήν τη στιγμή, αντικειμένων και μεθόδων. Δοκιμάστε, παραδείγματος χάριν, ένα σημείο αναστολής στη μέθοδο `carTest()`. Στην αριστερή πλευρά του παραθύρου αποσφαλματωτή, βλέπετε την ακολουθία κλήσης. Αυτήν τη στιγμή εμφανίζει:

```
Car.seats
Demo.carTest
```

Αυτό δείχνει ότι η `Car.seats` κλήθηκε από την `Demo.carTest`. Μπορείτε να επιλέξετε την `Demo.carTest` σε αυτόν τον κατάλογο για να επιθεωρήσετε τον πηγαίο κώδικα και τις παρούσες τιμές μεταβλητών σε αυτήν την μέθοδο.

Εάν «περπατήσετε» πέρα από τη γραμμή που περιέχει την οδηγία `new Car(. . .)`, θα παρατηρήσετε ότι η τιμή της τοπικής μεταβλητής `myCar` εμφανίζεται σαν *<object reference>* (<αναφορά αντικειμένου>). Όλες οι τιμές των τύπων αντικειμένου (εκτός από `String`) εμφανίζονται κατ' αυτό τον τρόπο. Μπορείτε να επιθεωρήσετε αυτήν την μεταβλητή πατώντας δύο φορές (double-click) πάνω τους. Κάνοντας αυτό θα ανοίξει ένα παράθυρο επιθεώρησης αντικειμένου ίδιο με εκείνα που περιγράφηκαν νωρίτερα (Κεφάλαιο 4.1). Δεν υπάρχει καμία ουσιαστική διαφορά μεταξύ του να επιθεωρήσετε τα αντικείμενα εδώ, η να τα επιθεωρήσετε στον πάγκο αντικειμένων.

6.4 Στάση και ολοκλήρωση (Halt and Terminate)

Περίληψη: Το `Halt` και `Terminate` μπορούν να χρησιμοποιηθούν για να σταματήσει μια εκτέλεση προσωρινά ή μόνιμα.

Μερικές φορές ένα πρόγραμμα τρέχει για μεγάλο χρονικό διάστημα, και αναρωτιέστε εάν όλα είναι εντάξει. Ίσως υπάρχει ένας άπειρος βρόχος, ίσως απλά να χρειάζεται τόσο πολύ χρόνο. Μπορούμε να το ελέγξουμε. Καλέστε τη μέθοδο `longloop()` από την κλάση `Demo`. Αυτή τρέχει για αρκετό χρόνο.

Τώρα θέλουμε να μάθουμε τι γίνεται. Εμφανίστε το παράθυρο αποσφαλματωτή, εάν δεν είναι ήδη στην οθόνη. (Επ'ευκαιρία, πατώντας το συμβόλο στροφής που δείχνει ότι το μηχάνημα τρέχει κατά τη διάρκεια της εκτέλεσης είναι ένας σύντομος δρόμος για την εμφάνιση του διορθωτή).

Τώρα πατήστε το κουμπί `Halt` (στάση). Η εκτέλεση διακόπτεται ακριβώς σαν να είχαμε φτάσει ένα σημείο αναστολής. Τώρα μπορείτε να «περπατήσετε» μερικά βήματα, να παρατηρήσετε τις μεταβλητές, και να δείτε ότι είναι όλα εντάξει. Απλά χρειάζεται λίγο περισσότερο χρόνο για να ολοκληρωθεί. Μπορείτε να συνεχίσετε (`Continue`) και να σταματήσετε (`Halt`) αρκετές φορές για να δείτε πόσο γρήγορα μετρά.

Εάν δεν θέλετε να συνεχίσετε (π.χ., ανακαλύψατε ότι είστε σε ένα άπειρο βρόχο) μπορείτε να πατήσετε *Terminate* για να σταματήσετε την εκτέλεση. Το *Terminate* δεν πρέπει να χρησιμοποιείται πολύ συχνά. Μπορεί να αφήσετε τέλεια γραμμένα αντικείμενα σε μία ασταθή κατάσταση σταματώντας το μηχάνημα, γι' αυτό είναι ενδεδειγμένο να χρησιμοποιήτε το *Terminate* μόνο ως μηχανισμό έκτακτης ανάγκης.

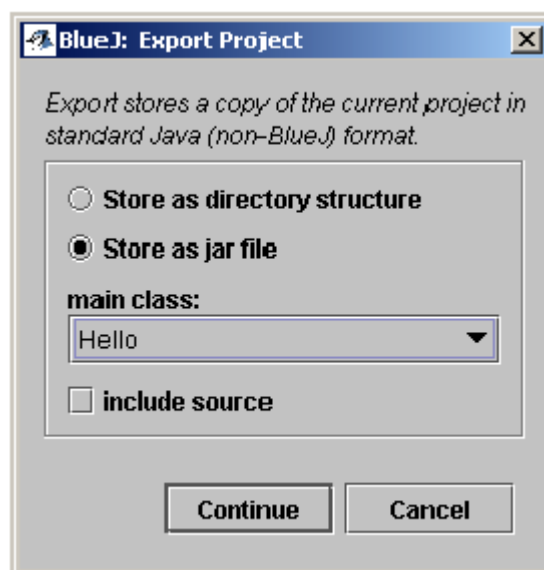
7 Δημιουργώντας ολοκληρωμένες εφαρμογές

Περίληψη: Για να δημιουργήσετε μία ολοκληρωμένη εφαρμογή, χρησιμοποιήστε Project - Export...

Το BlueJ μπορεί να δημιουργήσει εκτελέσιμα αρχεία jar. Τα εκτελέσιμα αρχεία jar μπορούν να εκτελεστούν σε μερικά συστήματα κάνοντας διπλό πάτημα στο αρχείο (για παράδειγμα Windows και MacOS X), ή εκτελώντας `java -jar <όνομα-αρχείου>.jar` (γραμμή εντολής Unix ή DOS).

Θα το δοκιμάσουμε με το πρόγραμμα-παράδειγμα *hello*. Ανοίξτε το (βρίσκεται στον κατάλογο αρχείων-παραδειγμάτων *examples*). Σιγουρευτείτε ότι το πρόγραμμα μεταγλωττίζεται. Επιλέξτε τη λειτουργία *Export* (εξαγωγή)... από τον κατάλογο *Project*.

Ένα διαλογικό παράθυρο ανοίγει και σας επιτρέπει να προσδιορίσετε τη μορφή αποθήκευσης (Εικόνα 15). Επιλέξτε το "jar file" για να δημιουργήσετε ένα εκτελέσιμο αρχείο jar. Για να κάνετε το αρχείο jar εκτελέσιμο, πρέπει επίσης να προσδιορίσετε μια βασική κλάση. Αυτή η κλάση πρέπει να έχει καθορισμένη μία έγκυρη μέθοδο `main` (με την υπογραφή `public static void main(String[] args)`).



Εικόνα 15: The "Export" dialogue

Στο παράδειγμά μας, επιλέγοντας τη βασική κλάση είναι εύκολο: υπάρχει μόνο μία κλάση. Επιλέξτε *Hello* από τον εμφανιζόμενο κατάλογο επιλογής. Εάν έχετε άλλα προγράμματα, επιλέξτε την κλάση που περιέχει τη μέθοδο *main* που θέλετε να εκτελέσετε.

Συνήθως, δεν περιλαμβάνετε τον πηγαίο κώδικα στα εκτελέσιμα αρχεία. Αλλά μπορείτε, εάν θελήσετε, να διανείμετε και τον πηγαίο κώδικα.

Πατήστε *Continue*. Έπειτα, βλέπετε ένα διαλογικό παράθυρο επιλογής αρχείων που σας επιτρέπει να προσδιορίσετε ένα όνομα για το αρχείο jar που θα δημιουργήσει. Πληκτρολογήστε *hello* και πατήστε *OK*. Η δημιουργία του εκτελέσιμου αρχείου jar ολοκληρώθηκε.

Μπορείτε να πατήσετε δύο φορές στο αρχείο jar μόνο εάν η εφαρμογή χρησιμοποιεί κάποια γραφική διεπαφή χρήστη (Graphical User Interface). Το παράδειγμά μας χρησιμοποιεί εισόδους/εξόδους κειμένου (text I/O), έτσι πρέπει να το αρχίσουμε από ένα τερματικό κειμένου (text terminal). Ας προσπαθήσουμε να «τρέξουμε» το αρχείο jar τώρα.

Ανοίξτε ένα τερματικό ή ένα παράθυρο DOS. Κατόπιν πηγαίνατε στον κατάλογο αρχείων όπου σώσατε το αρχείο jar σας (θα πρέπει να δείτε ένα αρχείο *hello.jar*). Υποθέτοντας ότι η Java είναι εγκατεστημένη σωστά στο σύστημα σας, θα πρέπει να είστε σε θέση να πληκτρολογήσετε:

```
java -jar hello.jar
```

για να εκτελέσετε το αρχείο.

8 Δημιουργώντας applets (μίνι-εφαρμογές)

8.1 «Τρέχοντας» ένα applet

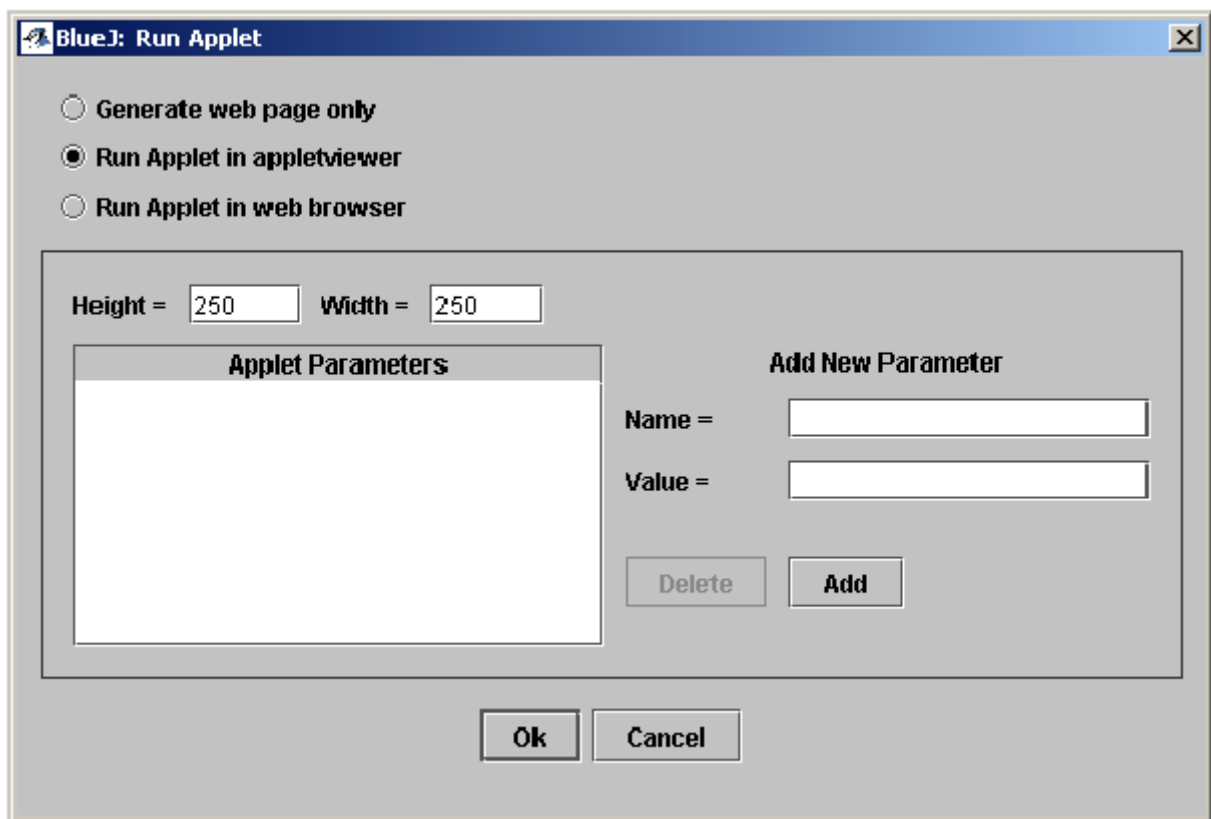
Περίληψη: Για να τρέξει ένα applet, επιλέξτε Run Applet από τον εμφανιζόμενο κατάλογο επιλογής του applet.

Το BlueJ επιτρέπει τη δημιουργία και εκτέλεση applets όπως και εφαρμογών. Έχουμε συμπεριλάβει ένα applet στον κατάλογο αρχείων παραδειγμάτων της διανομής. Αρχικά, θέλουμε να προσπαθήσουμε να εκτελέσουμε ένα applet. Ανοίξτε το πρόγραμμα *appletdemo* από τα παραδείγματα.

Θα δείτε ότι αυτό το πρόγραμμα έχει μόνο μια κλάση. Ονομάζεται *CaseConverter*. Το εικονίδιο κλάσης είναι χαρακτηρισμένο ως applet (με την ετικέτα <<applet>>). Αφού μεταγλωττίσετε, επιλέξτε την εντολή *Run Applet* από τον εμφανιζόμενο κατάλογο επιλογής της κλάσης.

A dialogue pops up that lets you make some selections (Figure 16).

Ένα διαλογικό παράθυρο εμφανίζεται και σας επιτρέπει να κάνετε μερικές επιλογές (Εικόνα 16).



Εικόνα 16: The "Run Applet" dialogue

Βλέπετε ότι έχετε την επιλογή να «τρέξετε» το applet σε ένα πρόγραμμα πλοήγησης (browser) ή σε ένα εικονοσκόπιο για applet (applet viewer) (ή απλά για να δημιουργήσει τη σελίδα ιστού χωρίς το τρέξιμο του). Αφήστε τις προκαθορισμένες τιμές των παραμέτρων και πατήστε *OK*. Μετά από μερικά δευτερόλεπτα, ένα εικονοσκόπιο για applet πρέπει να εμφανιστεί παρουσιάζοντας το *CaseConverter* applet.

Το εικονοσκόπιο για applet εγκαθίσταται μαζί με το JDK σας, έτσι είναι πάντα εγγυημένο ότι να είναι της ίδιας έκδοσης με το μεταγλωττιστή της Java σας. Γενικά, προκαλεί λιγότερα προβλήματα από οτι τα προγράμματα πλοήγησης. Το πρόγραμμα πλοήγησης ιστού σας μπορεί να «τρέξει» μια διαφορετική έκδοση της Java και, ανάλογα με ποια έκδοση ποίου προγράμματος πλοήγησης χρησιμοποιείτε, μπορεί να προκαλέσει προβλήματα. Με τα περισσότερα προγράμματα πλοήγησης όμως, πρέπει να λειτουργήσει σωστά.

Στα συστήματα Microsoft Windows και MacOS, το BlueJ χρησιμοποιεί το προκαθορισμένο πρόγραμμα πλοήγησης σας. Στα συστήματα Unix, το πρόγραμμα πλοήγησης καθορίζεται στις τιμές των παραμέτρων BlueJ.

8.2 Δημιουργώντας ένα applet

Περίληψη: Για να δημιουργήσετε ένα applet, πατήστε το κουμπί New Class και επιλέξτε Applet ως τον τύπο κλάσης.

Αφού είδαμε πώς να «τρέξουμε» ένα applet, θέλουμε να δημιουργήσουμε το δικό μας.

Δημιουργήστε μία νέα κλάση με *Applet* ως τύπο κλάσης (μπορείτε να επιλέξετε τον τύπο στο διαλογικό παράθυρο *New Class*). Μεταγλωττίστε, κατόπιν «τρέξτε» το applet. Αυτό ήταν! Δεν ήταν και άσχημα, έτσι δεν είναι;

Τα applets (όπως και οι άλλες κλάσεις) παράγονται με ένα προκαθορισμένο σκελετό κλάσης που περιέχει μερικό έγκυρο κώδικα. Για τα applets, αυτός ο κώδικας εμφανίζει ένα απλό applet με δύο γραμμές κειμένου. Μπορείτε τώρα να ανοίξετε τον διορθωτή και να διορθώσετε το applet ώστε να παρεμβάλετε τον κώδικα σας.

Θα δείτε ότι όλες οι κοινές μέθοδοι των applets βρίσκονται εκεί, κάθε μια με ένα σχόλιο που εξηγεί το σκοπό της. Ο κώδικας δειγμάτων είναι όλος στη μέθοδο *paint*.

8.3 Εξετάζοντας το applet

Σε μερικές καταστάσεις ίσως φανεί χρήσιμο να δημιουργήσετε ένα αντικείμενο applet στον πάγκο αντικειμένων (όπως για τις κανονικές κλάσεις). Μπορείτε να κάνετε κάτι τέτοιο. Ο κατασκευαστής εμφανίζεται στον εμφανιζόμενο κατάλογο επιλογής του applet. Από τον πάγκο αντικειμένων δεν μπορείτε να εκτελέσετε το πλήρες applet, αλλά μπορείτε να καλέσετε μερικές μεθόδους. Αυτό ίσως είναι χρήσιμο για να εξετάσετε μεθόδους που πιθανόν να είχατε γράψει ως τμήμα της εφαρμογής του applet σας.

9 Άλλες διαδικασίες

9.1 Ανοίγοντας πακέτα μη ανήκοντα στο BlueJ με το BlueJ

Περίληψη: Πακέτα μη ανήκοντα στο BlueJ μπορούν να ανοιχτούν με την εντολή Project: Open Non BlueJ...

Το BlueJ σας επιτρέπει να ανοίξετε υπάρχοντα πακέτα τα οποία δημιουργήθηκαν εκτός του BlueJ. Για να το κάνετε αυτό, επιλέξτε *Project – Open Non BlueJ...* από τον κατάλογο επιλογής. Επιλέξτε τον κατάλογο αρχείων που περιέχει τα αρχεία πηγαίου κώδικα της Java, κατόπιν πατήστε το κουμπί *Open in BlueJ*. Το σύστημα θα σας ζητήσει να επιβεβαιώσετε ότι θέλετε να ανοίξετε αυτόν τον κατάλογο αρχείων.

9.2 Προσθέτοντας υπάρχουσες κλάσεις στο πρόγραμμά σας

Περίληψη: Οι κλάσεις μπορούν να αντιγραφούν μέσα σε ένα πρόγραμμα με τη χρησιμοποίηση της εντολής Add Class from File...

Συχνά, θέλετε να χρησιμοποιήσετε μια κλάση την οποία βρήκατε κάπου αλλού μέσα στο πρόγραμμά σας στο BlueJ. Παραδείγματος χάριν, ένας δάσκαλος μπορεί να δώσει μια κλάση Java στους σπουδαστές για να χρησιμοποιηθεί σε ένα πρόγραμμα. Μπορείτε εύκολα να ενσωματώσετε μια υπάρχουσα κλάση στο πρόγραμμά σας επιλέγοντας *Edit – Add Class from File...* από τον κατάλογο επιλογής. Αυτό θα σας επιτρέψει να επιλέξετε ένα αρχείο πηγαίου κώδικα της Java (με ένα όνομα που τελειώνει στο *.java*) για να εισαχθεί.

Όταν η κλάση εισαχθεί στο πρόγραμμα, ένα αντίγραφο λαμβάνεται και καταχωρείται στον τρέχοντα κατάλογο αρχείων προγράμματος. Η επίδραση είναι ακριβώς η ίδια σαν να είχατε δημιουργήσει ακριβώς εκείνη την κλάση και είχατε γράψει όλο τον πηγαίο κώδικα της.

Ένας εναλλακτικός τρόπος είναι να προσθέσετε το αρχείο πηγαίου κώδικα της νέας κλάσης στον κατάλογο αρχείων προγράμματος έξω από το BlueJ. Την επόμενη φορά που θα ανοίξετε αυτό το πρόγραμμα, η κλάση θα έχει περιληφθεί στο διάγραμμα του προγράμματος.

9.3 Καλώντας την *main* και άλλες στατικές μεθόδους

Περίληψη: Οι στατικές μέθοδοι μπορούν να κληθούν από τον εμφανιζόμενο κατάλογο επιλογής της κλάσης.

Ανοίξτε το πρόγραμμα *hello* από τον κατάλογο αρχείων *examples*. Η μόνη κλάση στο πρόγραμμα (κλάση *hello*) καθορίζει μια πρότυπη μέθοδο *main*.

Κάντε δεξί πάτημα στην κλάση, και θα δείτε ότι ο κατάλογος επιλογής κλάσης περιλαμβάνει όχι μόνο τον κατασκευαστή της κλάσης, αλλά και τη στατική μέθοδο *main*. Μπορείτε τώρα να καλέσετε άμεσα τη *main* από αυτόν τον κατάλογο επιλογής (χωρίς πρώτα να δημιουργήσει ένα αντικείμενο, όπως θα αναμέναμε για μια στατική μέθοδο).

Όλες οι στατικές μέθοδοι μπορούν να κληθούν έτσι. Η πρότυπη μέθοδος *main* αναμένει ένα πίνακα (*array*) *String* ως όρισμα. Μπορείτε να μεταβιβάσετε ένα πίνακα *String* χρησιμοποιώντας την πρότυπη σύνταξη της Java για τις σταθερές πινάκων. Π.χ., θα μπορούσατε να μεταβιβάσετε:

```
{"one", "two", "three"}
```

(συμπεριλαμβάνοντας τις αγκύλες) στη μέθοδο. Δοκιμάστε το!

Υποσημείωση: Στην πρότυπη Java, οι σταθερές πινάκων δεν μπορούν να χρησιμοποιηθούν ως πραγματικά ορίσματα σε κλήσεις μεθόδων. Είναι δυνατόν να χρησιμοποιηθούν μόνο ως εκκινητές (*initializers*). Στο BlueJ, για να καταστήσουμε ικανές τις διαλογικές κλήσεις των πρότυπων μεθόδων *main*, επιτρέπουμε τη μεταβίβαση των σταθερών πινάκων ως παραμέτρους.

9.4 Δημιουργώντας τεκμηρίωση (*documentation*)

Περίληψη: Για να δημιουργήσετε τεκμηρίωση για ένα πρόγραμμα, επιλέξτε Project Documentation από τον κατάλογο επιλογής Tools.

Μπορείτε να δημιουργήσετε τεκμηρίωση για το πρόγραμμά σας στην πρότυπη μορφή *javadoc* μέσα από το BlueJ. Για να το κάνετε αυτό, επιλέξτε *Tools - Project Documentation* από τον κατάλογο επιλογής. Αυτή η λειτουργία θα δημιουργήσει την τεκμηρίωση για όλες τις κλάσεις σε ένα πρόγραμμα από τον πηγαίο κώδικα των κλάσεων και θα ανοίξει ένα πρόγραμμα πλοήγησης ιστού για να την παρουσιάσει.

Μπορείτε επίσης να δημιουργήσετε και να εμφανίσετε την τεκμηρίωση για μία κλάση μέσα στον διορθωτή του BlueJ. Για να το κάνετε αυτό, ανοίξτε τον διορθωτή και χρησιμοποιήστε τον εμφανιζόμενο κατάλογο επιλογής στη εργαλειοθήκη του διορθωτή. Αλλάξτε την επιλογή από *Implementation* σε *Interface*. Αυτό θα εμφανίσει την τεκμηρίωση με διαμόρφωση *javadoc* (η διεπαφή της κλάσης) στον διορθωτή.

9.5 Δουλεύοντας με βιβλιοθήκες

Περίληψη: Η διεπαφή προγραμματισμού εφαρμογής προτύπων κλάσεων της Java (Java standard class API) μπορεί να εμφανιστεί με την επιλογή Help – Java Standard Libraries.

Συχνά, όταν γράφετε ένα πρόγραμμα Java, πρέπει να αναφερθείτε στις πρότυπες βιβλιοθήκες της Java. Μπορείτε να ανοίξετε ένα πρόγραμμα πλοήγησης ιστού που εμφανίζει την τεκμηρίωση του JDK API με την επιλογή *Help - Java Standard Classes* από τον κατάλογο επιλογής (εάν είστε συνδεδεμένοι στο Διαδίκτυο).

Η τεκμηρίωση του JDK μπορεί επίσης να εγκατασταθεί και να χρησιμοποιηθεί τοπικά (χωρίς σύνδεση στο Διαδίκτυο). Λεπτομέρειες δίνονται στο τμήμα οδηγιών στην περιοχή ιστού του BlueJ.

9.6 Δημιουργώντας αντικείμενα από κλάσεις βιβλιοθηκών

Περίληψη: Για να δημιουργήσετε αντικείμενα από κλάσεις βιβλιοθηκών, χρησιμοποιήστε: Tools – Use Library Class.

Το BlueJ προσφέρει επίσης μια λειτουργία για να δημιουργήσετε αντικείμενα από τις κλάσεις που δεν είναι μέρος του προγράμματός σας, αλλά καθορισμένες στη βιβλιοθήκη. Μπορείτε, π.χ., να δημιουργήσετε τα αντικείμενα της κλάσης *String* ή *ArrayList*. Αυτό μπορεί να είναι πολύ χρήσιμο για το γρήγορο πειραματισμό με αυτά τα αντικείμενα βιβλιοθηκών.

Δύνασθε να δημιουργήσετε ένα αντικείμενο βιβλιοθηκών επιλέγοντας *Tools – Use Library Class...* από τον κατάλογο επιλογής. Ένα διαλογικό παράθυρο θα εμφανιστεί παροτρύνοντας σας να εισάγετε ένα πλήρως κατάλληλο όνομα κλάσης, όπως *java.lang.String*. (Σημειώστε ότι πρέπει να πληκτρολογήσετε το πλήρως κατάλληλο όνομα, το οποίο είναι το όνομα συμπεριλαμβανομένων των ονομάτων συσκευασίας που περιέχουν την κλάση).

Το πεδίο εισόδων κειμένου έχει έναν συνεργαζόμενο εμφανιζόμενο κατάλογο επιλογής που εμφανίζει πρόσφατα χρησιμοποιημένες κλάσεις. Μόλις εισαχθεί ένα όνομα κλάσης, πατώντας *Enter* θα παρουσιάσει όλους τους κατασκευαστές και τις στατικές μεθόδους εκείνης της κλάσης σε έναν κατάλογο στο διαλογικό παράθυρο. Οποιοδήποτε από αυτούς τους κατασκευαστές ή στατικές μεθόδους μπορούν να κληθούν τώρα επιλέγοντας τους από αυτόν τον κατάλογο.

Η επίκληση (invocation) προχωρά σαν κάθε άλλο κατασκευαστή ή κλήση μεθόδου.

10 Οι περιλήψεις...

Ξεκινώντας – διόρθωση / μεταγλώττιση / εκτέλεση (edit/compile/execute)

1. Για να ανοίξετε ένα πρόγραμμα, επιλέξτε *Open* από τον κατάλογο (menu) *Project*.
2. Για να δημιουργήσετε ένα αντικείμενο, επιλέξτε ένα κατασκευαστή (constructor) από τον εμφανιζόμενο κατάλογο (popup menu) της κλάσης (class).
3. Για να εκτελέσετε μια μέθοδο, επιλέξτε την από τον εμφανιζόμενο κατάλογο αντικειμένων.
4. Για να διορθώσετε τον πηγαίο κώδικα μίας κλάσης, κάντε διπλό πάτημα στο εικονίδιο της κλάσης.
5. Για να μεταγλωττίσετε μια κλάση, χτυπήστε το κουμπί *Compile* στον διορθωτή. Για να μεταγλωττίσετε ένα πρόγραμμα, χτυπήστε το κουμπί *Compile* στο παράθυρο προγράμματος (*Project*).
6. Για να δείτε τις οδηγίες για ένα μήνυμα σφάλματος μεταγλωττιστή, πατήστε το ερωτηματικό δίπλα στο μήνυμα σφάλματος.

Κάνοντας κάτι παραπάνω...

7. Η εξέταση αντικειμένων επιτρέπει κάποια απλή εξουδετέρωση σφαλμάτων εμφανίζοντας την εσωτερική κατάσταση ενός αντικειμένου.
8. Ένα αντικείμενο μπορεί να μεταβιβαστεί ως παράμετρος σε μια κλήση μεθόδου με ένα πάτημα στο εικονίδιο αντικειμένου.

Δημιουργία ενός νέου προγράμματος

9. Για να δημιουργήσετε ένα πρόγραμμα, επιλέξτε *New...* από τον κατάλογο επιλογών *Project*.
10. Για να δημιουργήσετε μία κλάση, πατήστε το κουμπί *New Class* και προσδιορίστε το όνομα της κλάσης.
11. Για να δημιουργήσει ένα βέλος, πατήστε το κουμπί βελών και να σύρετε το βέλος στο διάγραμμα, ή απλά γράψτε τον πηγαίο κώδικα στον διορθωτή.
12. Για να αφαιρέσετε μια κλάση, επιλέξτε τη λειτουργία *Remove* από τον εμφανιζόμενο κατάλογο επιλογής της.
13. Για να αφαιρέσετε ένα βέλος, επιλέξτε *Remove* από το κατάλογο επιλογής *Edit* και πατήστε το βέλος.

Αποσφαλμάτωση

14. Για να θέσετε ένα σημείο αναστολής, πατήστε στην περιοχή σημείου αναστολής αριστερά του κειμένου στον διορθωτή.
15. Για να κάνετε ένα βήμα μέσα στον κώδικα, χρησιμοποιήστε τα κουμπιά *Step* (βήμα) και *Step Into* (βήμα μέσα) στον αποσφαλματωτή.
16. Η επιθεώρηση μεταβλητών είναι απλή – εμφανίζονται αυτόματα στον αποσφαλματωτή.
17. Το *Halt* και *Terminate* μπορούν να χρησιμοποιηθούν για να σταματήσει μια εκτέλεση προσωρινά ή μόνιμα.

Δημιουργώντας ολοκληρωμένες εφαρμογές

18. Για να δημιουργήσετε μία ολοκληρωμένη εφαρμογή, χρησιμοποιήστε *Project - Export...*

Δημιουργώντας applets (μίνι-εφαρμογές)

19. Για να τρέξει ένα applet, επιλέξτε *Run Applet* από τον εμφανιζόμενο κατάλογο επιλογής του applet.
20. Για να δημιουργήσετε ένα applet, πατήστε το κουμπί *New Class* και επιλέξτε *Applet* ως τον τύπο κλάσης.

Άλλες διαδικασίες

21. Πακέτα μη ανήκοντα στο BlueJ μπορούν να ανοιχτούν με την εντολή *Project: Open Non BlueJ...*
22. Οι κλάσεις μπορούν να αντιγραφούν μέσα σε ένα πρόγραμμα με τη χρησιμοποίηση της εντολής *Add Class from File...*
23. Οι στατικές μέθοδοι μπορούν να κληθούν από τον εμφανιζόμενο κατάλογο επιλογής της κλάσης.
24. Για να δημιουργήσετε τεκμηρίωση για ένα πρόγραμμα, επιλέξτε *Project Documentation* από τον κατάλογο επιλογής *Tools*.
25. Η διεπαφή προγραμματισμού εφαρμογής προτύπων κλάσεων της Java (Java standard class API) μπορεί να εμφανιστεί με την επιλογή *Help – Java Standard Libraries*.
26. Για να δημιουργήσετε αντικείμενα από κλάσεις βιβλιοθηκών, χρησιμοποιήστε: *Tools – Use Library Class*.