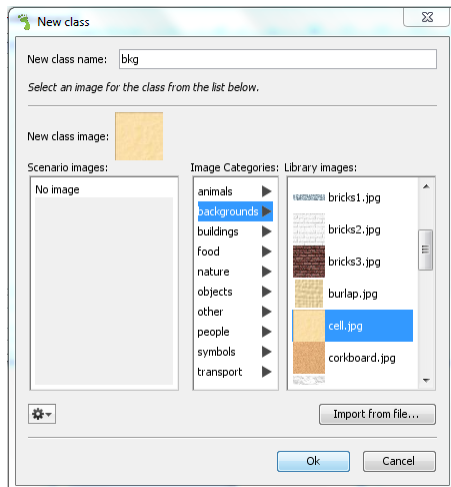


4. Τυχαία κίνηση



Στο κεφάλαιο αυτό μια μορφή (**Actor**) του προγράμματός μας θα κινείται με τυχαίο τρόπο μέσα στο υπόβαθρο. Δημιουργήστε ένα νέο υπόβαθρο (**background**) και χρησιμοποιείστε την εικόνα **cell.jpg** όπως φαίνεται στο παρακάτω σχήμα

Η εικόνα **cell.jpg** έχει μέγεθος 60x60 εικονοστοιχεία (pixel) και με αυτή μπορούμε να δημιουργήσουμε ένα πλέγμα με μεγάλα τετράγωνα.

Ανοίξτε τον editor για το υπόβαθρο (εντολή «**Open Editor**») και αλλάξτε τις παραμέτρους της εντολής **super**. Η εντολή **super** δημιουργεί το πλέγμα στο οποίο κινούνται οι μορφές.

Η σύνταξη της εντολής αυτής είναι η εξής:

super (αριθ_κελιών_οριζόντια, αριθ_κελιών_κάθετα, μέγεθος)

Αν το μέγεθος κελιού είναι 1 όπως μέχρι τώρα τότε τα αντικείμενα μπορούν να κινηθούν ελεύθερα μέσα στο πλέγμα.

Επειδή χρησιμοποιούμε την εικόνα **cell.jpg** θα αλλάξουμε το μέγεθος του κελιού σε 60. Για αυτό τον λόγο θα χρησιμοποιήσουμε 8 κελιά στον οριζόντιο άξονα και 8 κελιά στον κάθετο άξονα. Θα πρέπει να αλλάξετε και το κείμενο που βρίσκεται στα σχόλια ώστε να συμφωνεί με την αλλαγή που κάνατε.

Κώδικας 4.1:

```
super (8, 8, 60) ;
```

Τώρα μπορούμε να εισάγουμε ένα αντικείμενο μέσα στο πλέγμα. Δημιουργήστε μια νέα υποκλάση της κλάσης **Actor** με την εικόνα ενός ποντικιού, ονομάστε τη **mouse**.

Έχουμε μάθει με ποιον τρόπο μπορεί μια μορφή (**Actor**) να κινηθεί μέσα στο πλέγμα. Δεδομένου ότι το πλέγμα μας έχει μέγεθος 8x8 κελιά θα χρησιμοποιήσουμε μικρές τιμές για την παράμετρο της εντολής **move**. Δοκιμάστε την εντολή **move(1)**; Θα δείτε ότι το ποντίκι κινείται από κελί σε κελί και δεν κάνει μικρές κινήσεις όπως στα έργα που έχετε υλοποιήσει μέχρι τώρα.



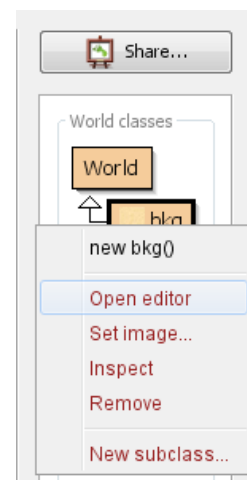
Επειδή το ποντίκι δεν στρίβει, σύντομα φτάνει στο τελευταίο κελί του πλέγματος και σταματάει. Μπορούμε να κάνουμε το ποντίκι να στρίβει με την εντολή **turn**.

Αν χρησιμοποιήσουμε τις εντολές

```
move (1) ;
```

```
turn (90) ;
```

το ποντίκι καλύπτει μόνο μια μικρή περιοχή του πλέγματος.



Μπορούμε να κάνουμε το ποντίκι να μην στρίβει κάθε φορά, αλλά να στρίβει μόνο τις μισές φορές και μάλιστα με τυχαίο τρόπο. Για να το καταφέρουμε θα χρειαστούμε την εντολή:

Greenfoot.getRandomNumber (όριο)

Η εντολή αυτή επιστρέφει έναν τυχαίο αριθμό από το 0 μέχρι το όριο. Το όριο είναι ένας ακέραιος αριθμός (int). Μπορείτε να δείτε πώς λειτουργεί η εντολή αυτή χρησιμοποιώντας την εντολή

System.out.println

Κώδικας 4.2:

```
move(1) ;  
System.out.println(Greenfoot.getRandomNumber(100)) ;
```

Αν εκτελέσετε τον παραπάνω κώδικα εμφανίζεται ένα νέο παράθυρο με τίτλο «Greenfoot: Terminal Window». Κάθε φορά που πατάτε **Act** μια νέα τυχαία τιμή εμφανίζεται σε αυτό το παράθυρο. Η τιμή αυτή είναι το αποτέλεσμα της μεθόδου **Greenfoot.getRandomNumber**. Επειδή το όριο είναι 100 οι αριθμοί αυτοί μπορεί να είναι από 0 έως 99.

Μπορούμε να προγραμματίσουμε το ποντίκι να στρίβει μόνο αν ο αριθμός αυτός είναι μεγαλύτερος από το 50. Για να το κάνουμε αυτό θα χρειαστούμε την εντολή **if (εάν)**.

Κώδικας 4.3:

```
move(1) ;  
if (Greenfoot.getRandomNumber(100)>50)  
{  
    turn(90) ;  
}
```

Το ποντίκι στρίβει μόνο αν ο τυχαίος αριθμός είναι μεγαλύτερος του 50. Όμως αυτή δεν ξέρουμε πότε θα συμβεί δεδομένου ότι η μέθοδος **Greenfoot.getRandomNumber** παράγει έναν τυχαίο αριθμό κάθε φορά. Έτσι το ποντίκι κάνει μια τυχαία κίνηση μέσα στο πλέγμα.

Επέκταση 4.1:



Μπορείτε να προγραμματίσετε το ποντίκι ώστε να μην στρίβει τόσο συχνά; Ποια παράμετρο του προγράμματός σας θα αλλάξετε;

Επέκταση 4.2:



Μπορείτε να βάλετε παραπάνω από ένα αντικείμενα να κινούνται με διαφορετικούς τρόπους (με διαφορετικές παραμέτρους στις τιμές move, turn αλλά και με διαφορετική πιθανότητα περιστροφής;