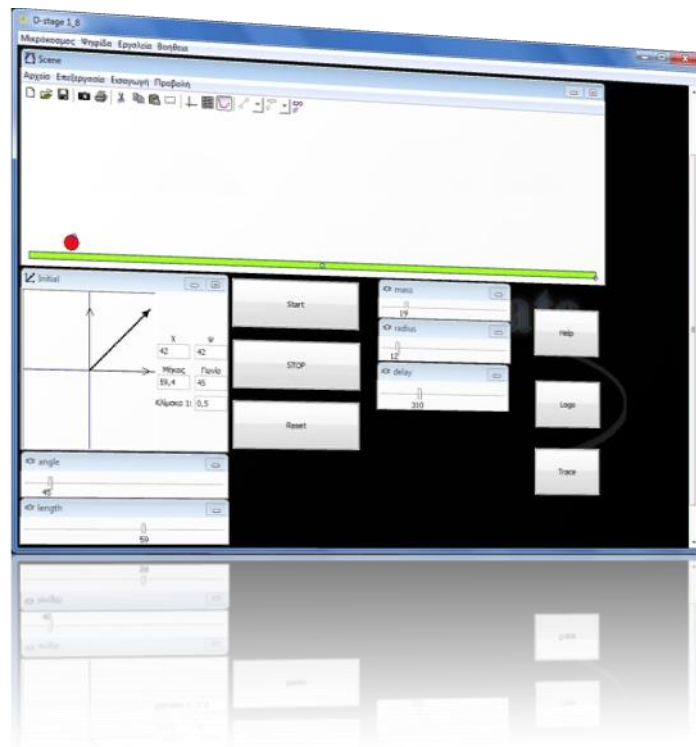




Εργαστήριο Εκπαιδευτικής Τεχνολογίας

Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών
Φιλοσοφική Σχολή
Τμήμα Φ.Π.Ψ., Τομέας Παιδαγωγικής
Διευθυντής: Καθ. Χ. Κυνηγός



Εγχειρίδιο Χρήσης του «D-stage»

Έκδοση: 3.0

Περιεχόμενα

Περιεχόμενα.....	1
1. Εισαγωγή.....	2
1.1 Το κιτ D-stage	2
2. Περιγραφή.....	3
2.1 Το Περιβάλλον του D-stage	3
2.2 Περιγραφή Κατηγοριών Ψηφίδων	3
3. Δομή και λειτουργία του D-stage.....	6
3.1. Επέκταση: Κατασκευή γραφικής παράστασης	11
4. Σύνδεση ψηφίδων.....	12
4.1. Σύνδεση ψηφίδων μέσω «διαχείρισης συνδέσμων»	12
4.2. Σύνδεση ψηφίδων μέσω προγράμματος.....	13
5. Συνοπτική παρουσίαση και ανάλυση κώδικα Logo	15

1. Εισαγωγή

Το παρόν εγχειρίδιο απευθύνεται στο φοιτητή, εκπαιδευτικό ή ερευνητή που επιθυμεί αξιοποιώντας το λογισμικό «D-stage», να κατασκευάσει εκπαιδευτικά λογισμικά, χωρίς να έχει βαθιά γνώση προγραμματισμού.

1.1 Το κιτ D-stage

Το λογισμικό (κιτ) D-stage (ή αλλιώς «Δυναμοσκηνή») ανήκει στην κατηγορία «Μικρόκοσμων», λογισμικών που κατασκευάζονται με τη χρήση του συγγραφικού εργαλείου «Αβάκιο». Ο μικρόκοσμος D-stage αποτελεί πρότυπο για κατασκευή νέων μικρόκοσμων Φυσικής με έμφαση στον τομέα της Μηχανικής. Αρχικός στόχος του μικρόκοσμου είναι η μελέτη της πλάγιας βολής.

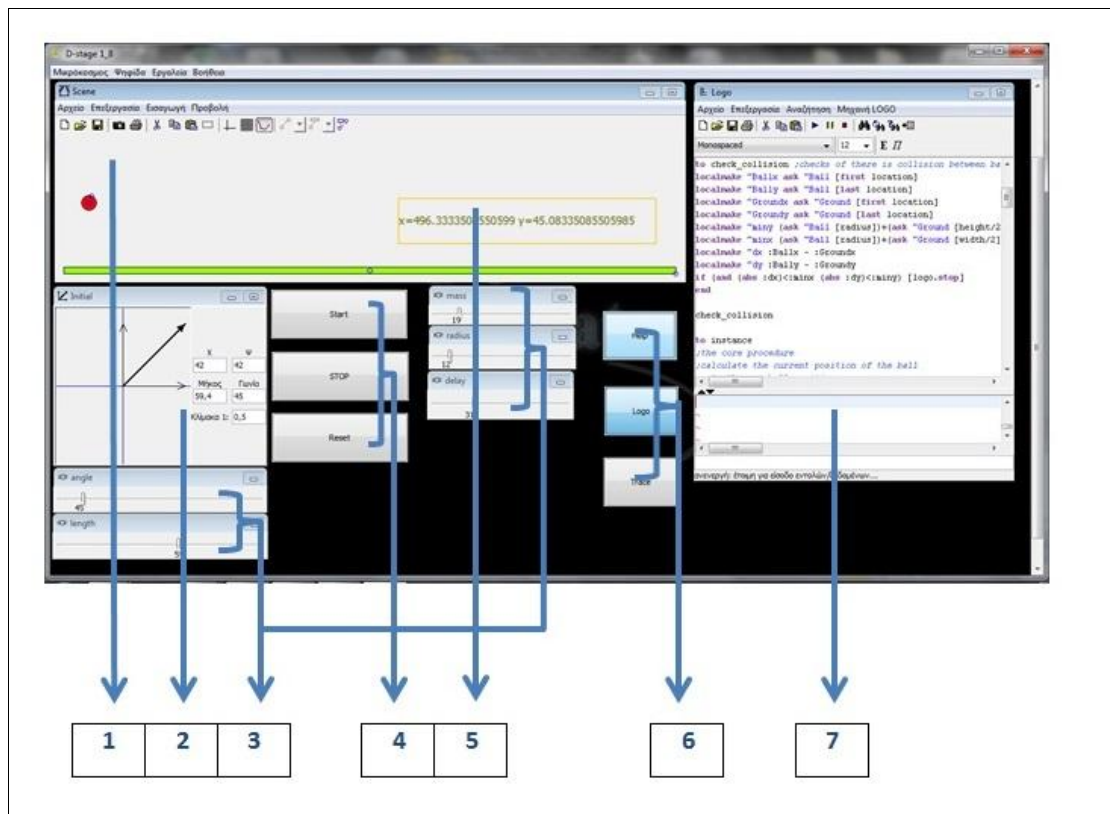
Προσαρμόζοντας κατάλληλα τα στοιχεία που αποτελούν τον μικρόκοσμο αλλά και προσθέτοντας νέες λειτουργίες γίνεται δυνατή η μελέτη και προσομοίωση πληθώρας φυσικών φαινομένων.

2. Περιγραφή

2.1 Το Περιβάλλον του D-stage

Το περιβάλλον του D-stage αποτελείται από 19 στοιχεία, το καθένα από τα οποία εκτελεί συγκεκριμένες λειτουργίες. Τα στοιχεία ονομάζονται «ψηφίδες» και παρέχονται ως μια βιβλιοθήκη έτοιμων υπολογιστικών αντικειμένων στο «Αβάκιο», ειδικά σχεδιασμένων για εκπαιδευτική χρήση. Οι ψηφίδες αποκτούν λειτουργικότητες είτε με κατάλληλη γραφική σύνδεση μεταξύ τους, είτε με τη χρήση ειδικού συντάκτη εντολών.

Στο σχήμα που ακολουθεί φαίνονται οι ψηφίδες του D-stage ομαδοποιημένες σε κατηγορίες («τύπους») με όμοια λειτουργία:



Εικ. 1 Κατηγορίες Ψηφίδων

2.2 Περιγραφή Κατηγοριών Ψηφίδων

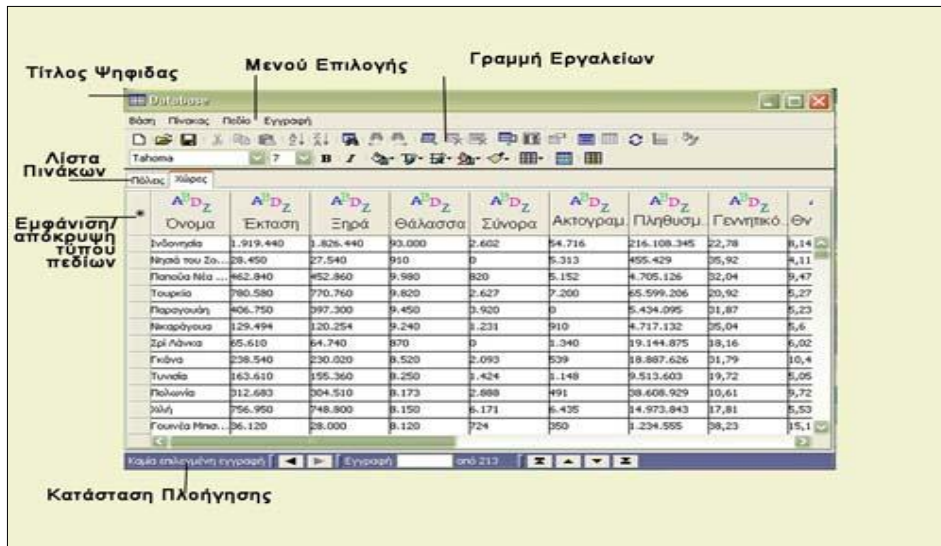
1. **Ψηφίδα Σκηνή:** Η Σκηνή παρέχει ένα περιβάλλον δημιουργίας «σκηνικών» με αντικείμενα όπως κουτιά, μπάλες, ελατήρια, σκοινιά κλπ. Τα φυσικά

χαρακτηριστικά των αντικειμένων, όπως το μέγεθος, το χρώμα και η θέση, προσδιορίζονται είτε με άμεσο χειρισμό με το ποντίκι, είτε μέσα από αντίστοιχες εντολές στο Συντάκτη. Με αυτόν τον τρόπο στη Σκηνή μπορούν να πραγματοποιηθούν προσομοιώσεις και πειράματα Φυσικής, η λογική των οποίων (π.χ. κίνηση των σωμάτων σε σχέση με το χρόνο) ορίζεται μέσα από προγράμματα που γράφονται στο Συντάκτη.

2. **Ψηφίδα Διάνυσμα:** Η ψηφίδα παρέχει τη δυνατότητα ορισμού και διαχείρισης διανυσματικών μεγεθών με άμεσο τρόπο. Η τιμή του διανύσματος προσδιορίζεται είτε με βάση τις αριθμητικές συνιστώσες του, είτε σύροντας το ποντίκι στο επιθυμητό τέλος του διανύσματος στην περιοχή γραφικής αναπαράστασης.
3. **Ψηφίδα Ολισθητής:** Ο ολισθητής μεταβάλλει με γραφικό τρόπο τις τιμές μιας αριθμητικής μεταβλητής. Συνδεδεμένος με άλλες ψηφίδες παρέχει τις τιμές αυτές για πρόσθετη χρήση και επεξεργασία.
4. **Ψηφίδα Κουμπί:** Το κουμπί αποτελεί ένα γραφικό αντικείμενο, το οποίο, κάθε φορά που πατιέται μπορεί να εκτελεί μια προκαθορισμένη λειτουργία.
5. **Ψηφίδα Ετικέτα:** Η ψηφίδα χρησιμοποιείται για την απλή αναγραφή κειμένου.
6. **Ψηφίδα Διακόπτης:** Η ψηφίδα αυτή είναι ουσιαστικά μια ψηφίδα «Κουμπί» δύο θέσεων: ενεργοποιημένο και απενεργοποιημένο. Χρησιμοποιείται ώστε να προκαλεί εναλλαγή καταστάσεων (όπως ένας διακόπτης on/off).
7. **Ψηφίδα Logo:** Πρόκειται για την ψηφίδα «Συντάκτη» των προγραμμάτων που δίνουν λειτουργικότητες στον μικρόκοσμο. Η γλώσσα προγραμματισμού που χρησιμοποιείται είναι η LOGO.

Οι παρακάτω ψηφίδες κατά το άνοιγμα του D-stage δε φαίνονται αλλά ενεργοποιούνται έπειτα από κάποιο «γεγονός», όπως για παράδειγμα μετά το πάτημα κάποιου κουμπιού :

- 1) **Ψηφίδα Κείμενο:** Η ψηφίδα Κείμενο παρέχει βασικές λειτουργίες σύνταξης και εμφάνισης μορφοποιημένου κειμένου. Μπορεί να δεχτεί κείμενο από άλλες ψηφίδες (μέσα από κατάλληλη διασύνδεση) για εμφάνιση ή για επεξεργασία.
- 2) **Ψηφίδα Επεξεργαστής Βάσεων:** Ο Επεξεργαστής Βάσεων Δεδομένων παρέχει τη δυνατότητα δημιουργίας και γενικής διαχείρισης (αποθήκευσης, ταξινόμησης, κατηγοριοποίησης, κλπ) δεδομένων στη μορφή ενός ή περισσότερων πινάκων. Τα δεδομένα μπορούν είτε να δημιουργηθούν εξ αρχής, είτε να εισαχθούν από άλλη συνδεδεμένη ψηφίδα, είτε να ενσωματωθούν από άλλες εφαρμογές (π.χ. Microsoft Excel, Microsoft Access) μέσω ενδιάμεσων αρχείων.



Εικ. 2 Η ψηφίδα "Επεξεργαστής Βάσης Δεδομένων"

- 3) **Ψηφίδα Καμβάς:** Ο καμβάς αποτελεί ένα περιβάλλον όπου μπορούν να γίνουν μία σειρά από διαφορετικές σχεδιαστικές λειτουργίες σε ξεχωριστές «σελίδες ζωγραφικής». Τα σχέδια (π.χ. γραφικές παραστάσεις) που δημιουργούνται πάνω στον καμβά μπορούν να γίνουν είτε με χρήση έτοιμης βιβλιοθήκης, είτε δυναμικά με χρήση συνδεδεμένης «χελώνας», ενός γραφικού αντικειμένου που δημιουργεί ίχνος στον καμβά ανάλογα με το πρόγραμμα που έχει γραφεί στον Συντάκτη.



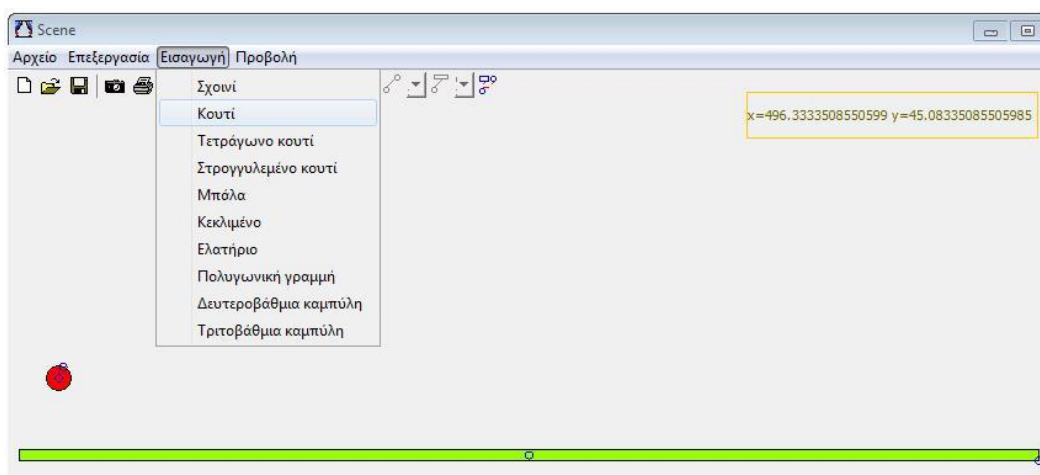
Εικ. 3 Οι ψηφίδες «Καμβάς» και «Χελώνα»

- 4) **Ψηφίδα Χελώνα:** Η ψηφίδα αυτή αναπαριστά την κλασική «LOGO-Χελώνα» η οποία ζωγραφίζει το ίχνος της στην ψηφίδα Καμβάς (εφόσον έχει συνδεθεί μαζί της).

Το σύνολο των ψηφίδων που χρησιμοποιεί ένας μικρόκοσμος φαίνεται στην επιλογή «Ψηφίδες» του μενού «Ψηφίδα». Ο καθορισμός των ρυθμίσεων των ψηφίδων γίνεται από το μενού [Εργαλεία]>[Διαχείριση Ψηφίδων]. Με τη χρήση της επιλογής [Εργαλεία]>[Γραμμή Εργασίας] εμφανίζονται όλες οι ψηφίδες του μικρόκοσμου σε γραμμή εργασίας.

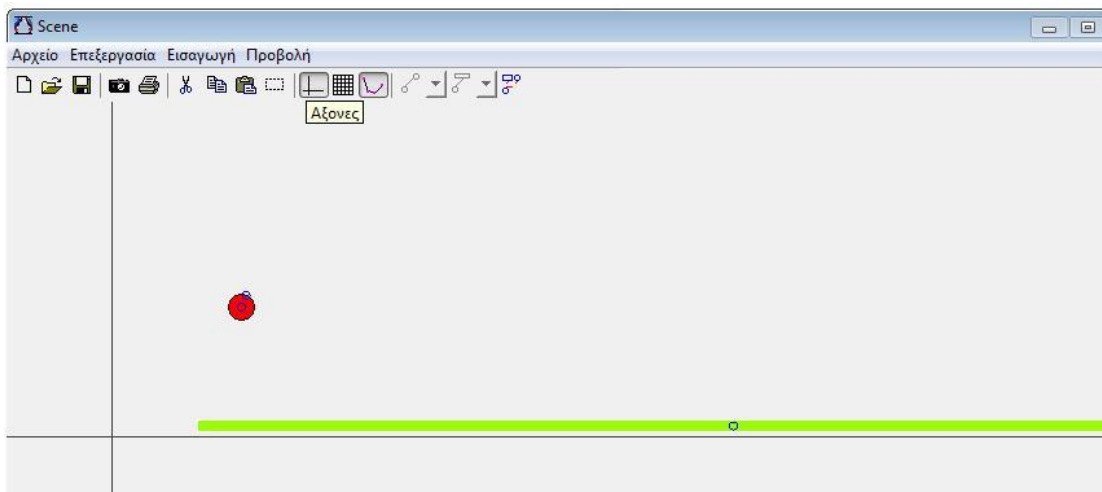
3. Δομή και λειτουργία του D-stage

Κεντρικό στοιχείο του D-stage είναι η Σκηνή (με όνομα «Scene»). Στην ψηφίδα «Scene» φαίνεται η μπάλα και η βάση (γρασίδι) πάνω στην οποία θα προσγειωθεί μετά την εκτέλεση της Πλάγιας Βολής. Τα δυο αυτά αντικείμενα εισάγονται στη Σκηνή από το μενού [Εισαγωγή] > [Κουτί] της ψηφίδας. Στη συνέχεια πατώντας δεξί κλικ πάνω στο αντικείμενο «Κουτί» και επιλέγοντας «ιδιότητες», είναι δυνατή η ρύθμιση των ιδιοτήτων του (χρώμα, θέση μέσα στη Σκηνή, μάζα, κλπ.). Ακολουθώντας την ίδια διαδικασία, τοποθετείται και η μπάλα μέσα στη Σκηνή και ρυθμίζονται οι ιδιότητές της. Με τον ίδιο τρόπο, μπορούν να τοποθετηθούν και να ρυθμιστούν όσα αντικείμενα απαιτούνται.



Εικ. 4 Εισαγωγή αντικειμένου

Πάνω στην επιφάνεια της σκηνής, στο πάνω δεξιό μέρος, έχει τοποθετηθεί μια ψηφίδα, μια ετικέτα. Η ετικέτα αυτή εμφανίζει (μέσω προγράμματος στον Συντάκτη) τις συντεταγμένες x και y της μπάλας στη σκηνή. Η αρχή των αξόνων της σκηνής μπορεί να αλλάξει από την επιλογή [Προβολή]>[Άξονες] αν σύρουμε το ποντίκι κρατώντας πατημένο το αριστερό κουμπί.



Εικ. 5 Αλλαγή της αρχής των αξόνων

Ο καθορισμός των χαρακτηριστικών της μπάλας και της κίνησης γίνεται από τους ολισθητές: "mass", "radius", "delay".

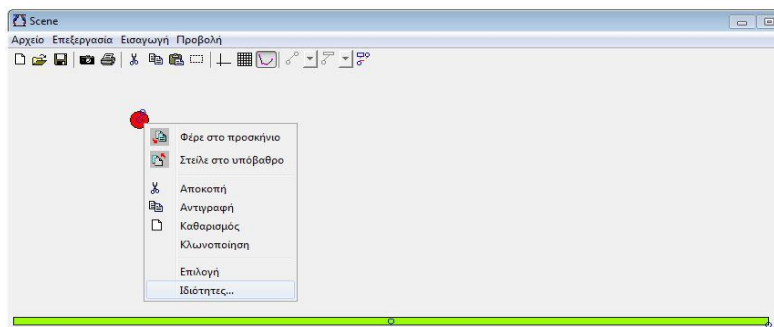


Εικ. 6 Οι ολισθητές

Ο ολισθητής "mass", ρυθμίζει τη μάζα της μπάλας και ο ολισθητής "radius" ρυθμίζει την ακτίνα της μπάλας. Ο "delay" προκαλεί καθυστέρηση στην εξέλιξη της προσομοίωσης, κάνοντας έτσι ποιο εύληπτη την κίνηση της μπάλας.

Σημείωση: Η μάζα της μπάλας δε χρησιμοποιείται στους υπολογισμούς του D-stage. Ο ολισθητής της μάζας παραμένει για υλοποίηση εναλλακτικών σεναρίων στο θέμα της Πλάγιας Βολής.

Τα -αρχικά -χαρακτηριστικά ενός αντικειμένου (π.χ. της μπάλας) μπορούν να μεταβληθούν και μέσα από τις ιδιότητές τους πάνω στη σκηνή κάνοντας δεξί κλικ πάνω στο αντικείμενο.



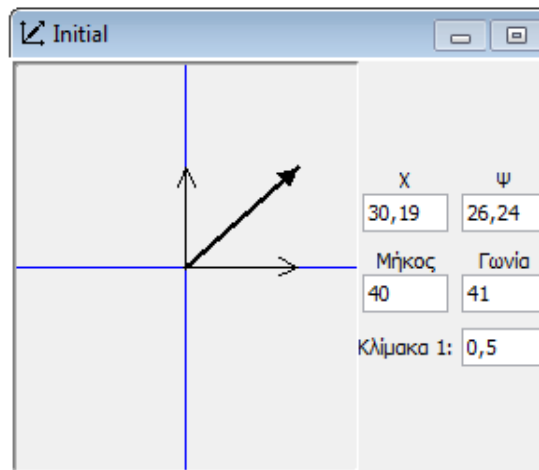
Εικ. 7 Επιλογές με δεξί κλικ στο αντικείμενο

Μπορούν να μεταβληθούν μια σειρά από ιδιότητες, από το όνομα και τη μορφή του ως σκηνικό αντικείμενο, έως και φυσικά χαρακτηριστικά του, όπως είναι η ακτίνα του, η μάζα του, αλλά και στοιχεία που χαρακτηρίζουν την θέση του στο χώρο και την κίνησή του, όπως η δύναμη που του ασκείται, η επιτάχυνσή του, η ταχύτητά του.



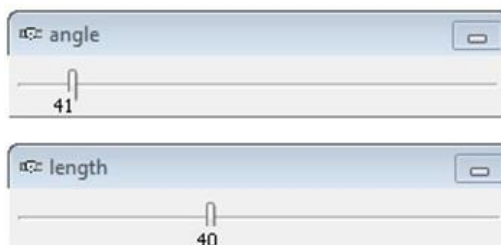
Εικ. 8 Ιδιότητες αντικειμένου

Η ψηφίδα «Initial» είναι μια ψηφίδα «Διάνυσμα». Απεικονίζει το διάνυσμα της αρχικής ταχύτητας της μπάλας, δηλαδή το μέτρο της και τη γωνία βολής της. Η αλλαγή της αρχικής ταχύτητας είναι δυνατή με απευθείας μεταβολή των δεδομένων της ψηφίδας.



Εικ. 9 Το διάνυσμα της αρχικής ταχύτητας

Η αρχική ταχύτητα της βολής μπορεί να καθοριστεί εναλλακτικά από τους ολισθητές «Angle», «Length» μεταβάλλοντας την γωνία βολής και το μέτρο της ταχύτητας αντίστοιχα.



Εκ. 10 Οι ολισθητές Angle, Length

Τα κουμπιά «Start», «Stop» και «Reset» αποτελούν τα κουμπιά χειρισμού της προσομοίωσης της κίνησης.

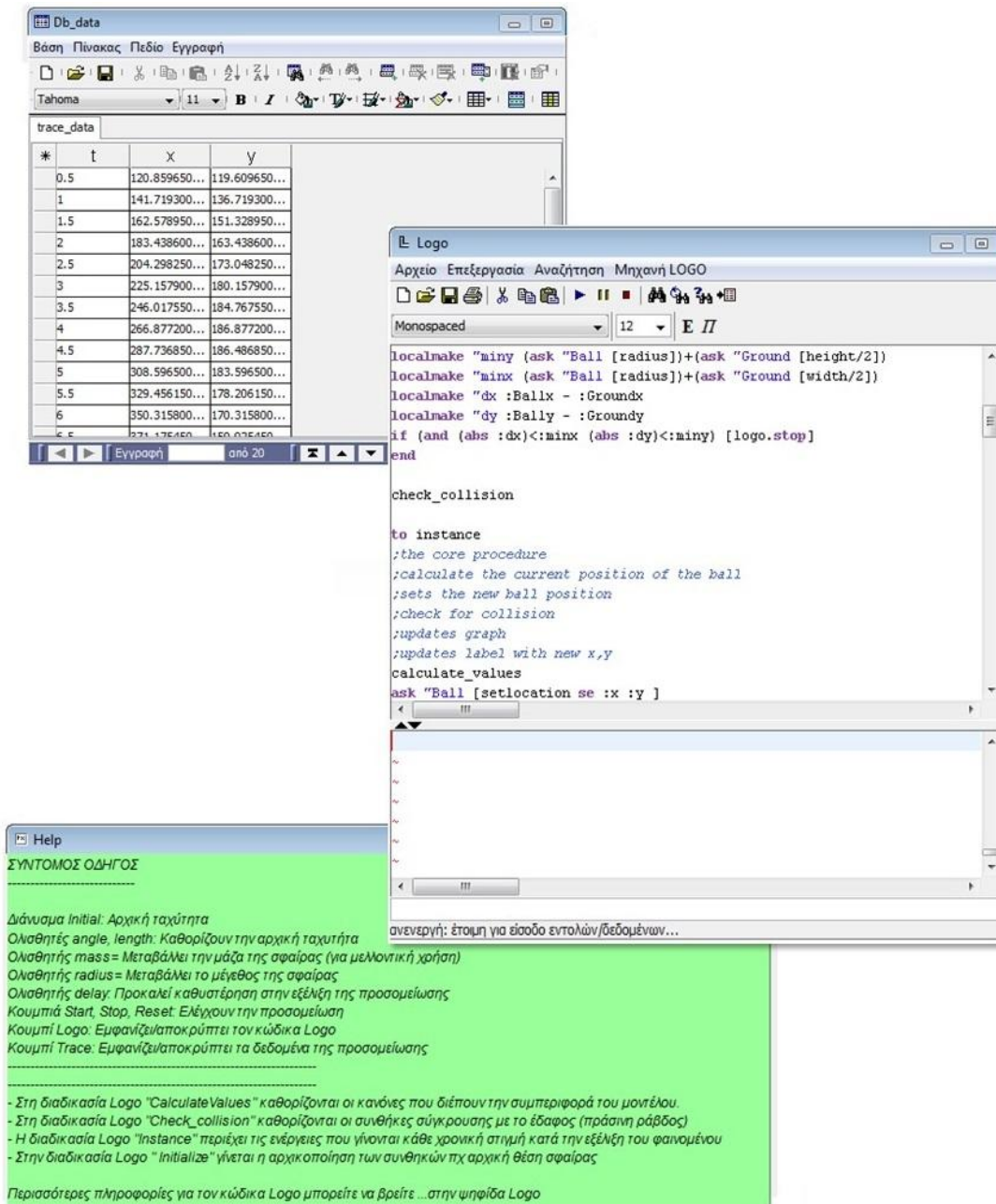


Με το κουμπί «Start» ξεκινά η προσομοίωση, με το «Stop» σταματά και με το «Reset» επαναφέρεται η μπάλα στην αρχική της θέση (χωρίς να επαναφέρονται και οι αρχικές τιμές του διανύσματος της ταχύτητας). Εάν κατά την εξέλιξη της προσομοίωσης πατηθεί το «Stop», δίνεται η δυνατότητα να επανακαθοριστούν οι αρχικές συνθήκες και η μπάλα να συνεχίσει την κίνηση με τις νέες τιμές.

Τέλος, στο περιβάλλον του D-stage τα κουμπιά-διακόπτες «Logo», «Trace», «Help» εμφανίζουν και αποκρύπτουν αντίστοιχες ψηφίδες.



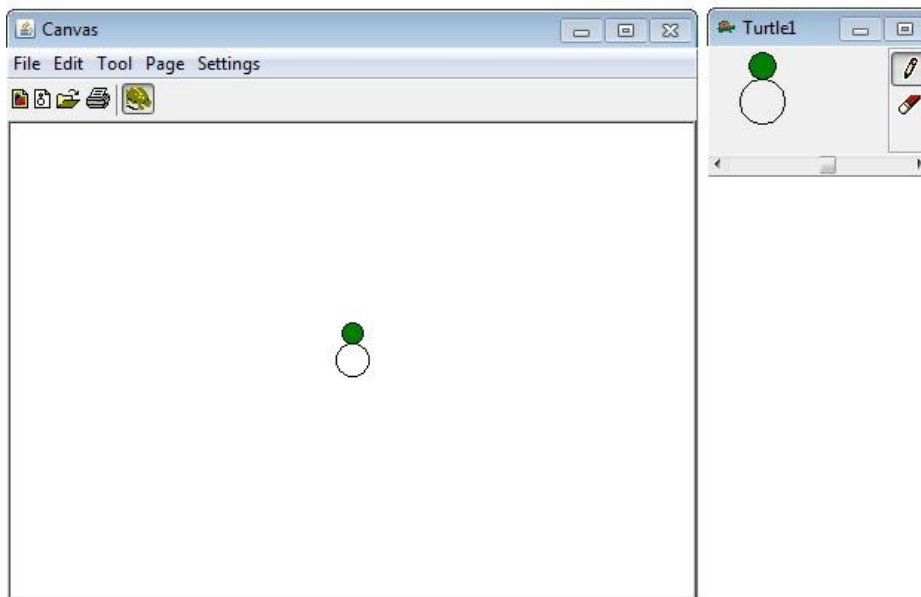
Στην ψηφίδα «Logo» συντάσσεται το πρόγραμμα που προσδίδει λειτουργικότητες στα στοιχεία του μικρόκοσμου. Το κουμπί «Trace» εμφανίζει και αποκρύπτει μια βάση δεδομένων η οποία σε κάθε εκτέλεση της προσομοίωσης γεμίζει με τα δεδομένα της κίνησης της μπάλας, δηλαδή με τα σημεία (x,y) από τα οποία περνάει η μπάλα καθώς περνάει ο χρόνος t. Τα δεδομένα αυτά μπορούν να χρησιμοποιηθούν για περαιτέρω μελέτη-αξιοποίηση του φαινομένου. Τέλος, το κουμπί «Help» εμφανίζει ένα σύντομο οδηγό χρήσης του Μικρόκοσμου.



Εικ. 11 Οι ψηφίδες που εμφανίζονται από τα αντίστοιχα κουμπιά

3.1. Επέκταση: Κατασκευή γραφικής παράστασης

Στο κιτ D-stage υπάρχει πρόβλεψη για δημιουργία δυναμικής γραφικής παράστασης ενός μεγέθους καθώς εξελίσσεται η προσομοίωση. Για το σκοπό αυτό έχουν ενσωματωθεί μια ψηφίδα «Καμβάς» και μια διασυνδεδεμένη «Χελώνα». Η χελώνα μέσω προγράμματος στο συντάκτη LOGO μπορεί να «χαράξει» τους άξονες και στη συνέχεια να «ζωγραφίζει» τα ενδιάμεσα σημεία της γραφικής παράστασης κατά την εξέλιξη του φαινομένου.



Εικ. 12 Οι ψηφίδες Καμβάς, Χελώνα

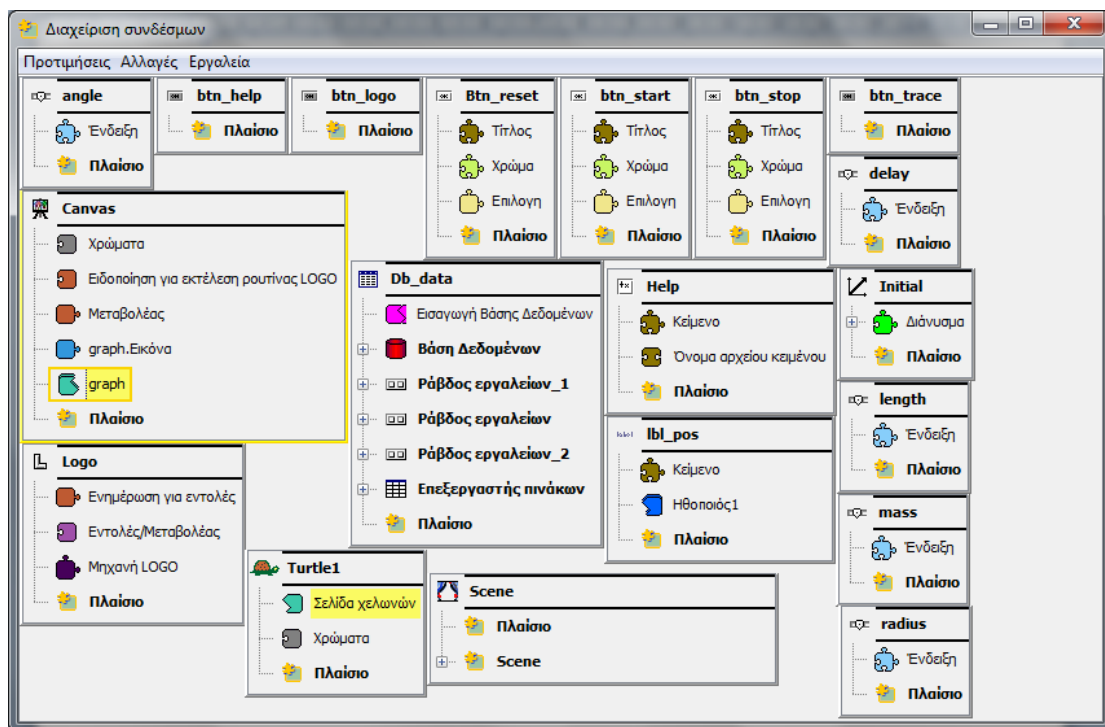
4. Σύνδεση ψηφίδων

Η διασύνδεση των ψηφίδων και η απόδοση λειτουργικότητας σε αυτές γίνεται με δύο τρόπους: (α) μέσω της «διαχείρισης συνδέσμων» και (β) μέσω προγραμμάτων στο Συντάκτη της LOGO.

4.1. Σύνδεση ψηφίδων μέσω «διαχείρισης συνδέσμων»

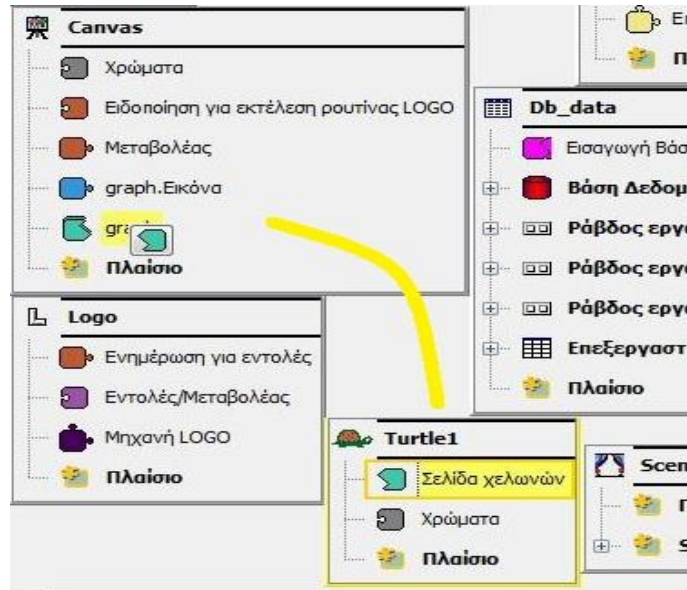
Οι ψηφίδες έχουν τη δυνατότητα να ανταλλάσσουν μεταξύ τους δεδομένα μέσω διαφόρων τύπων συνδέσμων και να έχουν έτσι μια εύκολη και «αυτοματοποιημένη» λειτουργία.

Η προβολή των υπαρχουσών συνδέσμων αλλά και η δημιουργία νέων γίνεται από την επιλογή [Εργαλεία]>[Διαχείριση Συνδέσμων]



Εικ. 13 Διαχείριση Συνδέσμων

Σε κάθε πλαίσιο-ψηφίδα παρουσιάζονται οι ιδιότητες που μπορούν να χρησιμοποιηθούν σε συνδέσεις με άλλες ψηφίδες. Χαρακτηριστικά παραδείγματα είναι η σύνδεση χελώνας «Turtle1» με τον καμβά «Canvas» ώστε να η χελώνα αν «ζωγραφίζει» πάνω στον καμβά. Κάνοντας κλικ σε μια ιδιότητα μιας ψηφίδας, αυτόματα μαρκάρονται όλες οι συμβατές ιδιότητες από τις υπόλοιπες ψηφίδες με τις οποίες μπορεί να γίνει σύνδεση. Η σύνδεση γίνεται, επιλέγοντας και σέρνοντας την ιδιότητα προς την άλλη ψηφίδα.



Εικ. 14 Σύνδεση ψηφίδων

4.2. Σύνδεση ψηφίδων μέσω προγράμματος

Η πλήρης παραμετροποίηση και προσαρμογή της λειτουργίας ενός μικρόκοσμου μπορεί να επιτευχθεί με τη χρήση προγραμμάτων LOGO που εισάγονται στην αντίστοιχη ψηφίδα.

Παράδειγμα: Για να τοποθετηθεί η μπάλα σε συγκεκριμένη θέση της σκηνής με το πάτημα ενός κουμπιού χρειάζεται:

- A) Να οριστεί μια διαδικασία (δηλαδή ένα μικρό πρόγραμμα) στο Συντάκτη LOGO.
- B) Να εισαχθεί το όνομα της διαδικασίας στο κατάλληλο «γεγονός» που ενεργοποιείται με το πάτημα του κουμπιού

Με τον τρόπο αυτό το κουμπί «συνδέεται» με την κατάλληλη ενέργεια που είναι καθορισμένη στο LOGO πρόγραμμα.

Ειδικότερα: Για να μετακινείται σε αρχική θέση η μπάλα όταν πατιέται το κουμπί «Reset»:

Στο Συντάκτη LOGO εισάγεται ο παρακάτω κώδικας:

```
to reset
ask "Ball [setlocation se 100 100]
cs
ask "lbl_pos [hide]
end
```

Η διαδικασία (που ξεκινάει πάντα με το «to» και τελειώνει με «end») ονομάζεται «reset» και για να είναι έγκυρη χρειάζεται να «οριστεί» μαρκάροντας την και πατώντας το πλήκτρο [INSERT]. Σύμφωνα με τις εντολές που περιλαμβάνει, η μπάλα «Ball» τοποθετείται στη θέση [100,100] και αποκρύπτεται η ετικέτα θέσης.

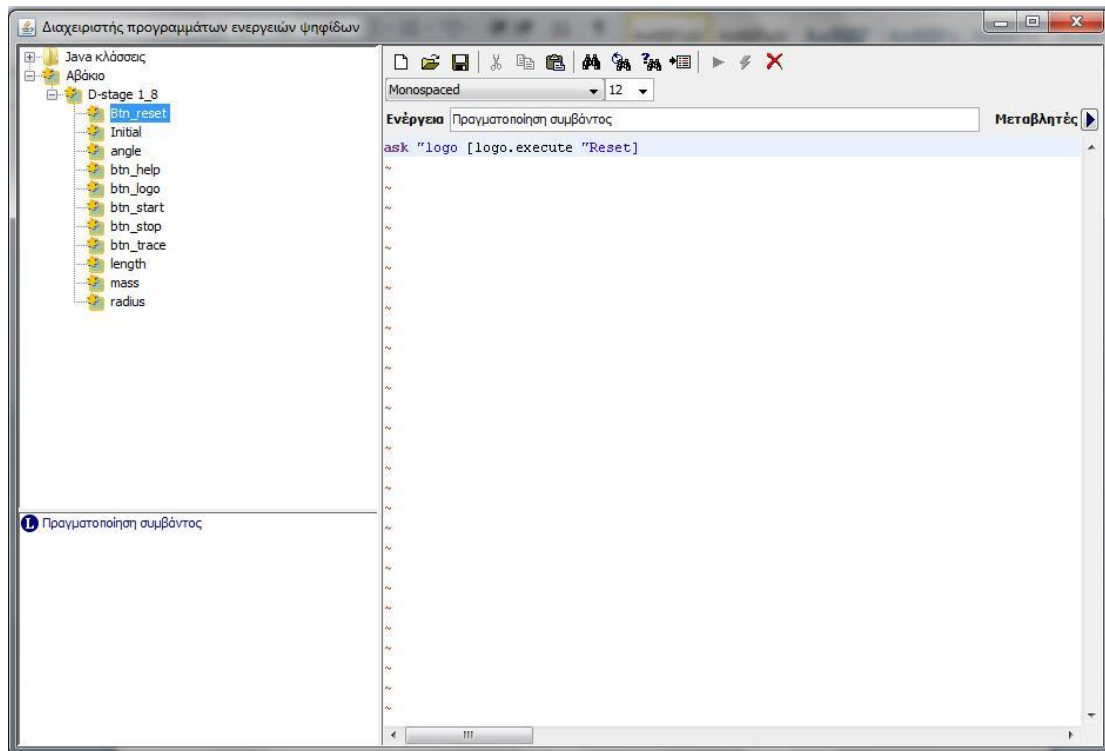
Για να γίνει η σύνδεση με την ψηφίδα-κουμπί «Reset» χρειάζεται να οριστεί στο γεγονός «Πραγματοποίηση συμβάντος» με τον εξής τρόπο:

Α) Επιλέγεται η διαχείριση ψηφίδων ([Εργαλεία]>[Διαχείριση Ψηφίδων])

Β) Στο παράθυρο που εμφανίζεται επιλέγεται από την αναδυόμενη λίστα η ψηφίδα «Btn_reset» η οποία είναι η ψηφίδα-κουμπί

Γ) Στην καρτέλα «Γεγονότα» επιλέγεται το κουμπί «L» δίπλα στο «Πραγματοποίηση συμβάντος»

Δ) Στο παράθυρο «Διαχειριστής προγραμμάτων ενεργειών ψηφίδων» που εμφανίζεται καταχωρείται η εντολή: `ask "Logo [Logo.execute "reset]` και επιλέγεται το κουμπί με το σύμβολο του κεραυνού. Με την εντολή αυτή όταν ενεργοποιηθεί το συγκεκριμένο γεγονός (με το κλικ του ποντικιού πάνω στο κουμπί) εκτελείται αυτόματα η διαδικασία που είναι γραμμένη στη LOGO και έχει όνομα «reset»



Εικ. 15 Εισαγωγή εντολών σε "γεγονός"

5. Συνοπτική παρουσίαση και ανάλυση κώδικα Logo

Παρακάτω παρουσιάζονται οι διαδικασίες που δίνουν λειτουργικότητες στον μικρόκοσμο και βρίσκονται στο Συντάκτη LOGO.

Αρχικά παρουσιάζονται στον κώδικα κάποιες οδηγίες που έγραψε ο κατασκευαστής του μικρόκοσμου. Εάν μπροστά από οποιαδήποτε εντολή εισαχθεί το ελληνικό ερωτηματικό, η εντολή απενεργοποιείται και θεωρείται σχόλιο.

```
; HINTS-TIPS for editing logo code
;-----
;1) When the button "start" is pressed the proc "Start" is called
;2) if there is not any collision with the ground the simulation never stops
;3) The time scale is set in "start" and default value is "0.5"
;4) The proc "draw_graph" is for future use and could represent any variable / time combination
;5) The proc "draw_axes" draws the axes on the graph
;6) The physics rules are set in "calculate_values".
;7) The value of gravity acceleration is set to "10" in "calculate_values"
;8) The recognition accuracy of the collision point depends on the time scale
```

Στη συνέχεια θέτονται οι προϋποθέσεις με τις οποίες σταματά η κίνηση της μπάλας. Ελέγχεται η θέση της μπάλας σε σχέση με τη θέση του επιπέδου. Εάν η μπάλα έχει φτάσει στο επίπεδο του δαπέδου, τότε ζητά να σταματήσει η προσομοίωση.

```
to check_collision ;checks of there is collision between ball and ground
localmake "Ballx ask "Ball [first location]
localmake "Bally ask "Ball [last location]
localmake "Groundx ask "Ground [first location]
localmake "Groundy ask "Ground [last location]
localmake "miny (ask "Ball [radius])+(ask "Ground [height/2])
localmake "minx (ask "Ball [radius])+(ask "Ground [width/2])
localmake "dx :Ballx - :Groundx
localmake "dy :Bally - :Groundy
if (and (abs :dx)<:minx (abs :dy)<:miny) [logo.stop]
end
```

Η διαδικασία «instance» είναι η «κεντρική» διαδικασία της προσομοίωσης πραγματοποιεί όλες τις ενέργειες που απαιτούνται σε κάθε χρονική στιγμή. Έτσι, υπολογίζει τη θέση της μπάλας (μέσω ξεχωριστής διαδικασίας), θέτει τη νέα θέση, ελέγχει αν η μπάλα ακουμπάει στο έδαφος, σχεδιάζει τη γραφική παράσταση (μελλοντική χρήση), ανανεώνει την ετικέτα με τις συντεταγμένες της μπάλας.


```

to instance
;the core procedure
;calculate the current position of the ball
;sets the new ball position
;check for collision
;updates graph
;updates label with new x,y
calculate_values
ask "Ball [setlocation se :x :y ]
check_collision
Draw_graph
localmake "lblx word "x= :x
localmake "lbly word "y= :y
localmake "lbly word "| | :lbly
ask "lbl_pos [label.settext word :lblx :lbly]
end

```

Ο υπολογισμός της θέσης της μπάλας για λόγους εύκολης συντήρησης βρίσκεται σε διαφορετική διαδικασία. Εδώ βρίσκεται το «κομμάτι» της Φυσικής, όπου θέτονται οι σχέσεις για τη θέση και την ταχύτητα και στους δύο άξονες. Ορίζονται οι απαραίτητες σταθερές ($g=10\text{m/s}^2$) και ανακτώνται οι αρχικές τιμές: η γωνία θ , η αρχική ταχύτητα. Σε ξεχωριστή διαδικασία («update_db») γίνεται και η ενημέρωση της Βάσης Δεδομένων με τα δεδομένα της κίνησης.

```

to calculate_values
;calculates position and velocity in each position
;  $x=u_0\cos\theta t$      $y=u_0\sin\theta t-(gt^2)/2$ 
localmake "g 10
localmake "theta round (ask "initial [vectorangle])
localmake "u0 round (ask "initial [vectorlength])
make "x :x0 + :u0* (cos :theta)*:t
make "y :y0 + (:u0*(sin :theta)*:t) - (:g * (power :t 2) )/2
make "uy :u0 * (sin :theta) + :g * :t
make "ux :u0
update_db :t :x :y
end

to update_db :time :curX :curY ;updates simulation data
localmake "recordcount ask "Db_data [(db.recordcount "trace_data )]
localmake "recordcount :recordcount + 1
ask "Db_data [(db.addrecord "trace_data )]
ask "Db_data [(db.setcell "trace_data :recordcount "t :time )]
ask "Db_data [(db.setcell "trace_data :recordcount "x :curX )]
ask "Db_data [(db.setcell "trace_data :recordcount "y :curY )]
end

```

Η έναρξη της προσομοίωσης γίνεται στη διαδικασία «start».

```

to start
;everything starts here - by button press
wait 100
initialize
while [l=1] [
make "t :t + 0.5
print se "t= :t
wait ask "delay [slider.value]
instance
]
end

```

Οι διαδικασίες `define_mass` και `define_radius` ενημερώνουν το σκηνικό αντικείμενο με τις ιδιότητες από τους αντίστοιχους ολισθητές. Οι διαδικασίες αυτές καλούνται όταν αλλάζει τιμή σε έναν ολισθητή.

```

to define_mass ;updates the mass fo every slider change
localmake "mass_value ask "mass [slider.value]
ask "Ball [setmass :mass_value]
end

to define_radius ;updates the radius fo every slider change
localmake "radius_value ask "radius [slider.value]
ask "Ball [setradius :radius_value]
end

```

Στη διαδικασία «initialize» θέτονται οι αρχικές τιμές κίνησης και ταχύτητας. Ακόμη σχεδιάζονται οι άξονες της γραφικής παράστασης (μελλοντική χρήση) και διαγράφονται δεδομένα προηγούμενης προσομοίωσης.

```

to initialize
  ;prepares variables and objects
  ;sets the initial ball position
  ;gets the initial velocity
  ;draws the axes
  ;changes the pen color
  ;makes the position label visible
  make "t 0
  make "x 100
  make "y 100
  make "x0 first ask "Ball [location]
  make "y0 last ask "Ball [location]
  make "ux0 first ask "Initial [vector]
  make "uy0 last ask "Initial [vector]
  ask "turtle1 [setpencolor (list 0 0 0)]
  Draw_axes
  ask "turtle1 [setpencolor (list 0 128 0)]
  ask "turtle1 [pu]
  ask "turtle1 [setpos list (5*:ux) (5*:uy)]
  ask "turtle1 [pd]
  ask "lbl_pos [restore]
  wait 10
  localmake "recordcount ask "Db_data [(db.recordcount "trace_data )]
  repeat :recordcount [ask "Db_data [(db.remove-record "trace_data 1) ]]
  end

```

Οι παρακάτω διαδικασίες ενημερώνουν τους ολισθητές με τυχόν αλλαγές στο διάνυσμα και αντίστροφα το διάνυσμα με νέα τιμή από τους ολισθητές. Καλούνται από αντίστοιχα «γεγονότα» των ψηφίδων και παρέχουν αμφίδρομη επικοινωνία μεταξύ ολισθητή και διανύσματος.

```

to loadangle
  ;updates initial vector after slider change
  ask "Initial [setvectorangle (ask "angle [slider.value])]
  end

to loadlength
  ;updates initial vector after slider change
  ask "Initial [setvectorlength (ask "length [slider.value])]
  end

to initial_updated
  ;updates sliders after vector change
  ask "angle [slider.setvalue round (ask "initial [vectorangle])]
  ask "length [slider.setvalue round (ask "initial [vectorlength])]
  end

```

Οι διαδικασίες «draw_axes» και «draw_graph» σχηματίζουν τους άξονες και ζωγραφίζουν τη γραφική παράσταση.

```
to draw_axes
cs
ask "turtle1 [pu setpos list 0 0 pd rt 90 fd 600 lt 180 fd 1200 rt 180 fd 600 lt 90 fd 500 lt 180 fd 1000 lt 180 fd 500]
end
draw_axes

to draw_graph
ask "turtle1 [setpos list (10*:t) (5*:uy)]
end
```