

App Inventor – 2^ο μάθημα (η 2^η μου εφαρμογή - Ζάρια)

- ✓ Εγκατάσταση
- ✓ Δημιουργία έργου
- ✓ Αντικείμενα
- ✓ Γεγονότα
- ✓ Πακετάρισμα και διαμοιρασμός

Η ανάγκη για εύκολη ανάπτυξη εφαρμογών για φορητές συσκευές με Android οδήγησε αρχικά τα εργαστήρια της Google στη δημιουργία του App Inventor, ενός ελεύθερου, διαδικτυακού και οπτικού προγραμματιστικού περιβάλλοντος με πλακίδια (blocks), όπως και το Scratch. Στη συνέχεια, το γνωστό κορυφαίο αμερικάνικο πανεπιστήμιο MIT (Massachusetts Institute of Technology) ανέλαβε την ανάπτυξη και συντήρηση αυτού.

Βήμα 1: Εγκατάσταση MIT AI2 Companion

Η ανάπτυξη και ο ταυτόχρονος έλεγχος μιας εφαρμογής μπορεί να γίνει ακόμα κι αν δεν διαθέτετε συσκευή Android, καθώς το App Inventor παρέχει έναν εξομοιωτή Android (emulator). Έτσι, ο έλεγχος της εφαρμογής μπορεί να γίνεται επιτόπου στην οθόνη του υπολογιστή, στην εικονική συσκευή που θα υφίσταται σε ένα ξεχωριστό παράθυρο. Πέρα όμως, του προφανούς μειονεκτήματος ότι δεν θα μπορείτε να αξιολογήσετε όλα τα χαρακτηριστικά μιας πραγματικής συσκευής, ο εξομοιωτής είναι σχετικά αργός, ενώ απαιτείται να εγκαταστήσετε στον υπολογιστή σας έξτρα λογισμικό.

Για τους παραπάνω λόγους, απεναντίας, συνιστάται η επιτόπου σύνδεση της Android συσκευής σας με τον Η/Υ μέσω ασύρματου δικτύου Wi-Fi. Κατ' αυτό τον τρόπο, ο έλεγχος της εφαρμογής μπορεί να γίνεται επιτόπου στη συσκευή σας, εφόσον Η/Υ και συσκευή είναι συνδεδεμένα στο ίδιο δίκτυο. Απαραίτητη προϋπόθεση για τη σύνδεση μέσω Wi-Fi είναι η εγκατάσταση στην Android συσκευή της εφαρμογής **MIT AI2 Companion**.

1. Με ενεργοποιημένο το Wi-Fi του κινητού σας μπειτε στο **Play Store**, αναζητήστε την παραπάνω εφαρμογή και ακολουθήστε τα βήματα εγκατάστασης.

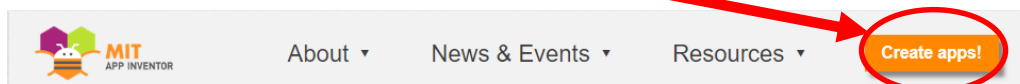


Βήμα 2: Σύνδεση και δημιουργία νέου έργου στο App Inventor

Στόχος της πρώτης μας υλοποίησης είναι μια εφαρμογή που θα δίνει τη δυνατότητα στον χρήστη της να ρίχνει δύο ζάρια με το πάτημα ενός κουμπιού. Το περιβάλλον της εφαρμογής μας θα μοιάζει με αυτό της διπλανής εικόνας.

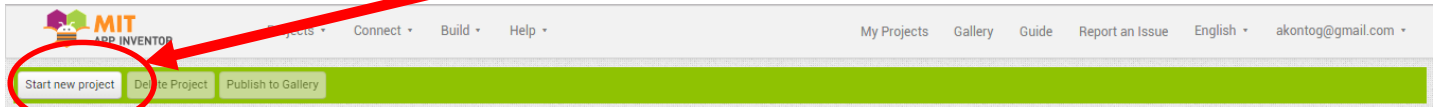
Επισκεπτόμαστε τη σελίδα του MIT για το App Inventor <http://appinventor.mit.edu/>. Θα εμφανιστεί η σελίδα για σύνδεση στο λογαριασμό που έχουμε στη Google. Δίνουμε το όνομα χρήστη και το συνθηματικό του λογαριασμού μας και πατάμε **Sign In** (Σύνδεση). Θα μας ζητηθεί η άδειά μας για την πρόσβαση της εφαρμογής MIT App Inventor στον λογαριασμό μας. Επιλέγουμε **Να επιτρέπεται** και κάνουμε **Accept** στους όρους χρήσης.

2. Κάνουμε κλικ στο **Create** στην άνω δεξιά γωνία της σελίδας

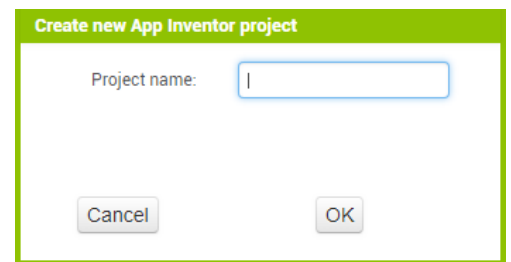


Εναλλακτικά για απευθείας μετάβαση στο περιβάλλον δημιουργίας εφαρμογών πληκτρολογούμε <http://ai2.appinventor.mit.edu/>.

3. Στη σελίδα που εμφανίζεται, για να δημιουργήσουμε μια νέα εφαρμογή στο περιβάλλον του App Inventor, κάνουμε κλικ στο κουμπί **Start New Project** (Νέο έργο).



Θα εμφανιστεί ένα παράθυρο για να καταχωρήσουμε το όνομα που θα δώσουμε στην εφαρμογή μας.

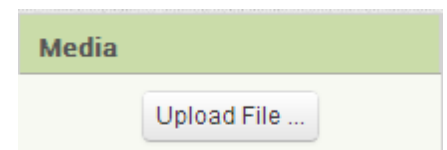


4. Φροντίζουμε το όνομα που θα δώσουμε να αποτελείται από λατινικούς χαρακτήρες και να μην περιέχει κενά ή άλλα σημεία στίξης εκτός από την κάτω παύλα '_'. Προτείνεται το όνομα **RollDice**. Στη συνέχεια πατάμε το κουμπί **OK**.

Αρχικά, βρισκόμαστε στην ενότητα **Designer**, στην οποία σχεδιάζουμε τη *διεπαφή* (interface) της εφαρμογής μας, προσθέτοντας τα απαραίτητα *αντικείμενα-συστατικά* (components) και ορίζοντας *ιδιότητες* (properties) για αυτά.

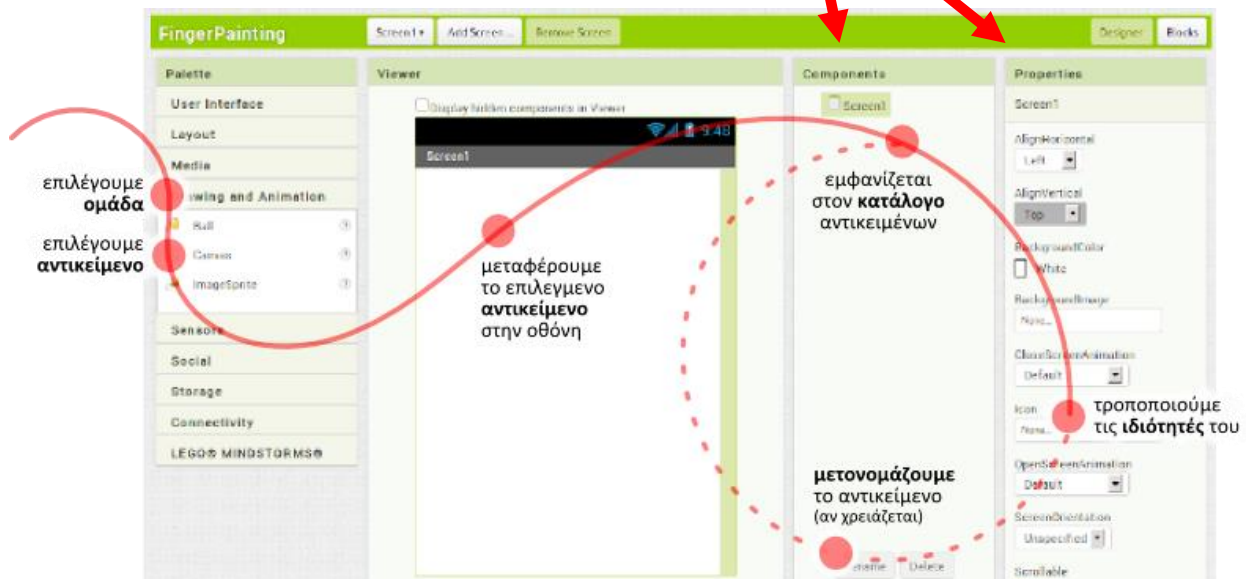
Βήμα 3: Μεταφόρτωση αρχείων στο App Inventor

5. Θα χρειαστούμε κάποιες εικόνες για τα ζάρια και έναν ήχο για το ρίξιμο των ζαριών. Στο φάκελο που βρίσκονται οι παρούσες οδηγίες θα βρείτε εικόνες ζαριών, **1.png** έως **6.png**, και τον σχετικό ήχο ρίψης τους, **RollDiceSound.mp3**. Κάνουμε κλικ στο κουμπί **Upload File...** στην περιοχή **Media**, κάτω δεξιά στο περιβάλλον. Στο παράθυρο διαλόγου που εμφανίζεται κάνουμε κλικ στο Επιλογή Αρχείου και επιλέγουμε ένα ένα τα αρχεία των εικόνων και του ήχου για να τα μεταφορτώσουμε στο App Inventor. Δυστυχώς το περιβάλλον δεν μας επιτρέπει την εισαγωγή πολλών αρχείων ταυτόχρονα.



Βήμα 4: Εισαγωγή αντικειμένων στην περιοχή σχεδίασης

6. Μέχρι στιγμής, το μοναδικό αντικείμενο, στην περιοχή **Components** της εφαρμογής είναι η οθόνη, **Screen1**. Προτού προσθέσουμε στο έργο μας τα απαραίτητα αρχεία ήχου και εικόνων, καθώς και επιπλέον αντικείμενα που θα χρειαστούμε, ας τροποποιήσουμε ορισμένες ιδιότητες της οθόνης, που με επιλεγμένο το **Screen1** βρίσκονται στο πλαίσιο **Properties**. Στο πεδίο **Title** βάζουμε τον τίτλο **Ζάρια** και στο πεδίο **Background Color** το χρώμα **Dark Gray**.



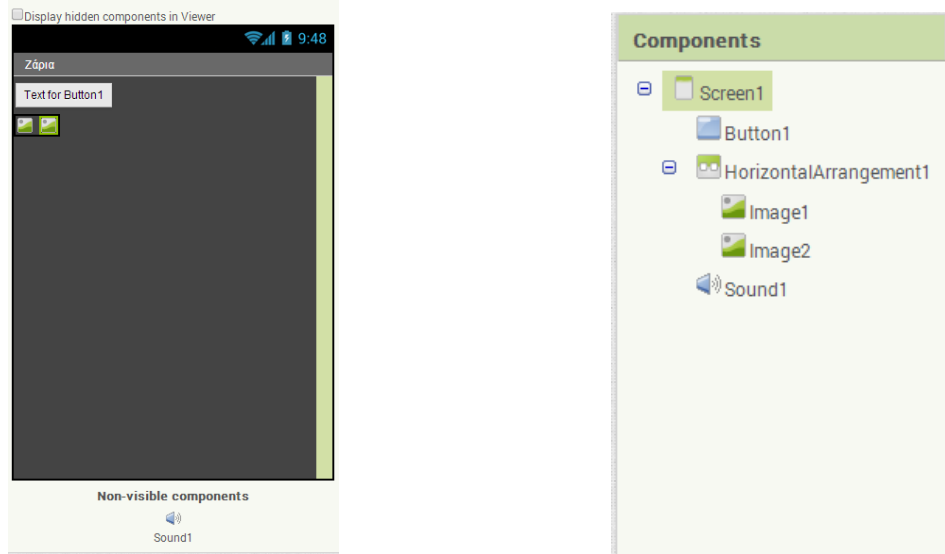
7. Στη συνέχεια θα εισάγουμε στο έργο μας τα απαραίτητα αντικείμενα. Η διαδικασία που πρέπει να ακολουθήσουμε, είναι να επιλέγουμε τα αντικείμενα που μας χρειάζονται από την εκάστοτε ομάδα της παλέτας, **Palette**, αριστερά και να τα αποθέτουμε στην οθόνη, όπως ενδεικτικά περιγράφεται στην παραπάνω εικόνα.

Η εφαρμογή που θα φτιάξουμε θα αποτελείται από τα παρακάτω αντικείμενα:

- από την ομάδα **User Interface**, ένα αντικείμενο **Button**
- από την ομάδα **Layout**, ένα αντικείμενο **Horizontal Arrangement**

- από την ομάδα **User Interface**, δύο αντικείμενα **Image**
- από την ομάδα **Media**, ένα αντικείμενο **Sound**

Τα τοποθετούμε με την παραπάνω σειρά στην περιοχή σχεδίασης και φροντίζουμε τα αντικείμενα **Image** να τοποθετηθούν μέσα στο αντικείμενο **Horizontal Arrangement**. Σημειώνεται ότι, ο ήχος δεν θα βρίσκεται στην οθόνη κινητού αλλά ως μη ορατό αντικείμενο θα βρίσκεται ακριβώς από κάτω. Όταν ολοκληρωθεί αυτό το βήμα, η περιοχή σχεδίασης πρέπει να είναι όπως παρακάτω:

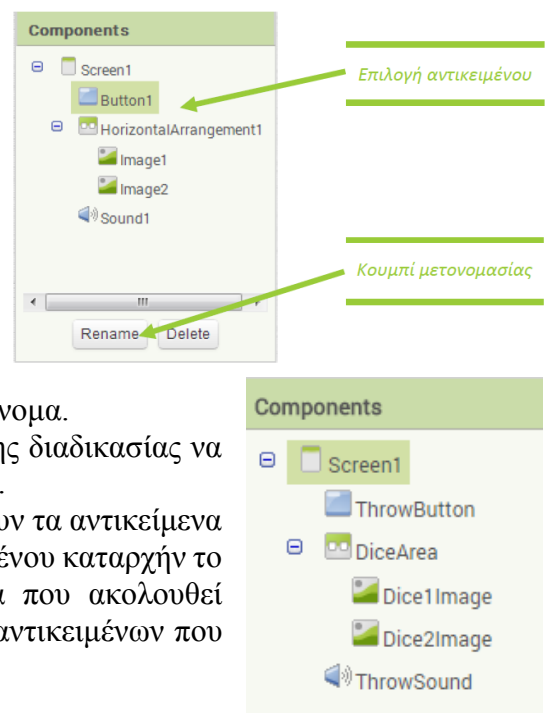


Βήμα 5: Καθορισμός ιδιοτήτων αντικειμένων

- Η πρώτη μας κίνηση είναι να μετονομάσουμε τα αντικείμενα που έχουμε εισάγει στην εφαρμογή μας, ώστε να έχουν χαρακτηριστικά ονόματα και όχι button1, image1 κ.λπ., ώστε να τα αναγνωρίζουμε ευκολότερα. Για να μετονομάσουμε ένα αντικείμενο το επιλέγουμε από την περιοχή των αντικειμένων (**Components**) και στη συνέχεια κάνουμε κλικ στο κουμπί **Rename**, όπως φαίνεται στην διπλανή εικόνα.

Προσοχή: Τα ονόματα αποτελούνται από λατινικούς χαρακτήρες, αριθμούς και κάτω παύλες. Δεν μπορούμε, έτσι, να χρησιμοποιήσουμε ελληνικά ή κενά στα ονόματά μας. Επιπλέον δεν μπορούμε να έχουμε δύο αντικείμενα με το ίδιο όνομα. Μετονομάζουμε τα αντικείμενα ώστε μετά την ολοκλήρωση της διαδικασίας να έχουμε την διπλανή εικόνα στην περιοχή των αντικειμένων μας.

- Στη συνέχεια θα καθορίσουμε κάποιες ιδιότητες που θα έχουν τα αντικείμενα της εφαρμογής. Για να δούμε τις ιδιότητες κάποιου αντικειμένου καταρχήν το επιλέγουμε από την περιοχή **Components**. Στον πίνακα που ακολουθεί φαίνονται αναλυτικά τα νέα ονόματα και οι ιδιότητες των αντικειμένων που θα πρέπει να θέσετε.





από την
ομάδα

μεταφέρουμε το
αντικείμενο

του δίνουμε το
όνομα

μεταβάλλουμε τις
ιδιότητες

UserInterface

Button

ThrowButton

Background Color : Orange

Font Size : 30

Shape : rounded

Text : Ρίξε τα ζάρια

Width : Fill Parent

Height : 80 (pixels)

Layout

HorizontalAlignment

DiceArea

AlignHorizontal : Centre

Width : Fill Parent

Height : 100 pixels

UserInterface

Image

Dice1Image

Picture : 1.png

UserInterface

Image

Dice2Image

Picture : 1.png

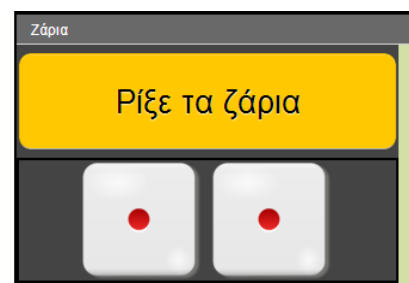
Media

Sound

ThrowSound

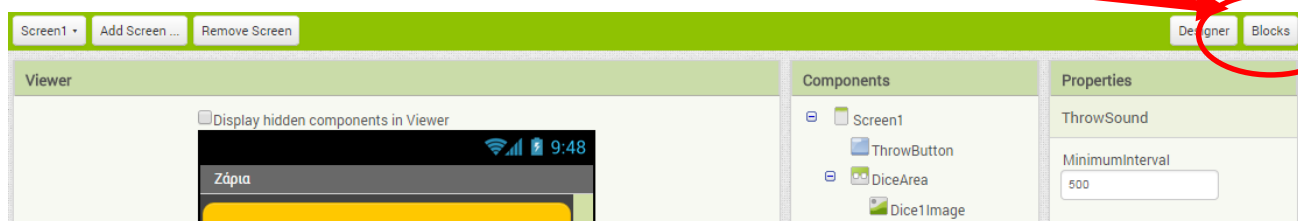
Source : RollDiceSound.mp3

Σημείωση: Στο *DiceArea* για να έχει νόημα η ιδιότητα **AlignHorizontal: Centre**, θα πρέπει πρώτα να έχετε αλλάξει το **Width** σε *Fill Parent* που δίνει οριζόντια όσο χώρο έχει η οθόνη του κινητού στο αντικείμενο. Η οθόνη της εφαρμογής, μετά και από τις παραπάνω αλλαγές, έχει διαμορφωθεί όπως στην διπλανή εικόνα.



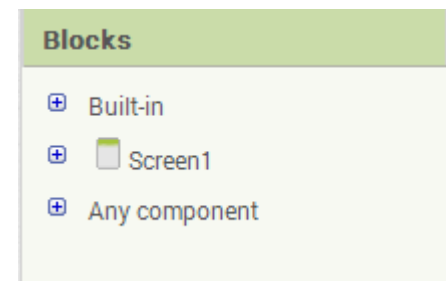
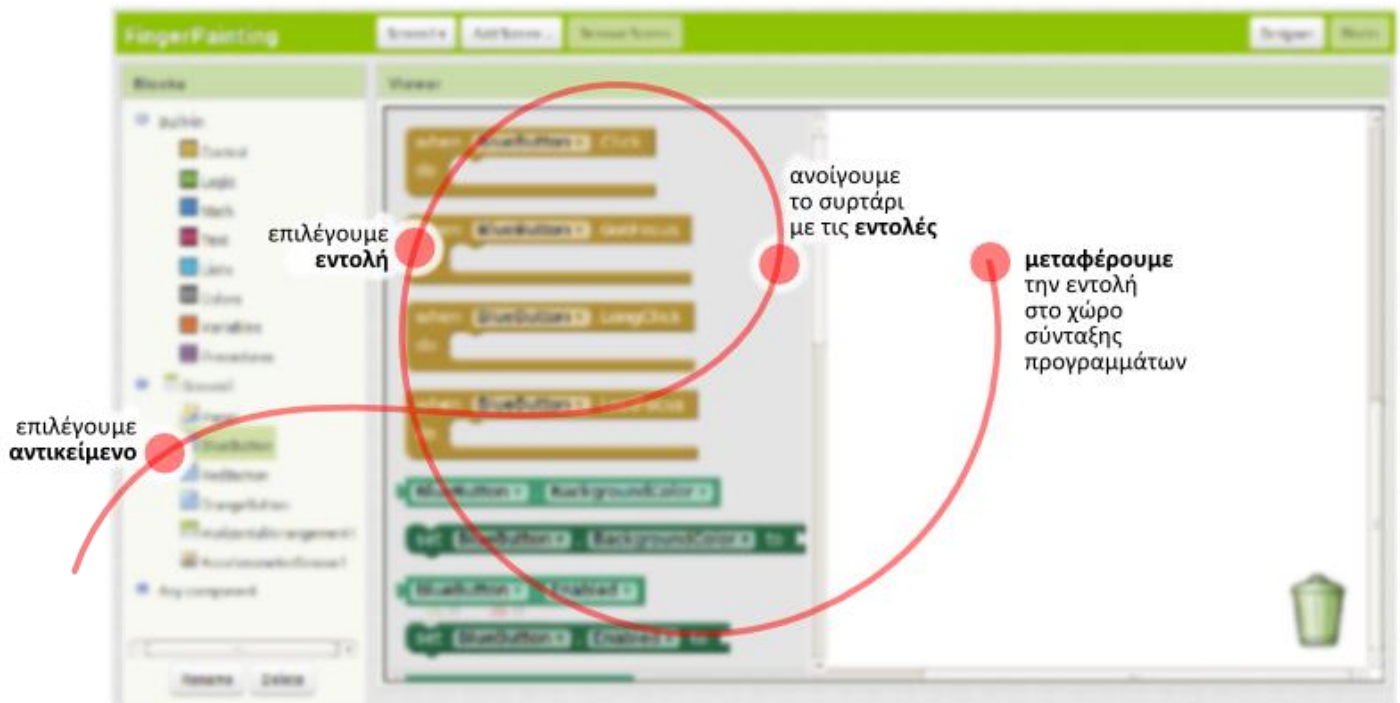
Βήμα 6: Συγγραφή προγράμματος – Γεγονός

10. Μέχρι στιγμής, στο παράθυρο σχεδίασης (**Designer**) έχουμε μεν σχεδιάσει ένα μέρος της «οθόνης» της εφαρμογής μας, αλλά δεν έχουμε ορίσει καθόλου την συμπεριφορά της. Θα μεταφερθούμε λοιπόν στο παράθυρο συγγραφής προγράμματος **Blocks** (το κουμπί μετάβασης βρίσκεται πάνω δεξιά) για να συσχετίσουμε ενέργειες με γεγονότα και, ουσιαστικά, να προγραμματίσουμε, προσθέτοντας τις κατάλληλες εντολές. Μεταβαίνουμε στο τμήμα εντολών της εφαρμογής, κάνοντας κλικ στο κουμπί **Blocks**.



11. Το ρίξιμο των ζαριών θα ενεργοποιείται όταν ο χρήστης αγγίζει το κουμπί «**Ρίξε ζάρια**». Θα πρέπει να συνδέσουμε το γεγονός αυτό (το άγγιγμα του κουμπιού) με τις κατάλληλες ενέργειες. Πρέπει, λοιπόν, να τοποθετήσουμε στην περιοχή εντολών την κατάλληλη εντολή που θα ανιχνεύει το άγγιγμα του κουμπιού.

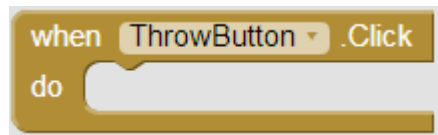
Όταν προγραμματίζουμε μια συγκεκριμένη συμπεριφορά για την εφαρμογή μας, πρακτικά συναρμολογούμε πλακίδια εντολών (και όχι μόνο). Για κάθε πλακίδιο που θέλουμε να προ-σθέσουμε ανατρέχουμε στην κατάλληλη ομάδα στα αριστερά της οθόνης, αναζητούμε και επιλέγουμε το πλακίδιο που χρειαζόμαστε και το σέρνουμε στον χώρο σύνταξης των προγραμμάτων. Η διαδικασία μοιάζει, λοιπόν, με εκείνη που ακολουθήσαμε για να προσθέσουμε αντικείμενα στην περιοχή σχεδίασης στο **Βήμα 3** και περιγράφεται ενδεικτικά στην παρακάτω εικόνα:



Στα αριστερά της οθόνης διακρίνουμε *τρεις ομάδες πλακιδίων*, blocks.

- **Built-in:** Ενσωματωμένα
- **Screen1:** Αντικείμενα που έχουμε προσθέσει στην εφαρμογή.
- **Any component.**

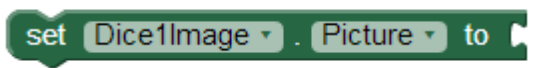
Στην περίπτωση μας, επιλέγουμε από την ομάδα Screen1 το αντικείμενο ThrowButton. Σέρνουμε με το ποντίκι την εντολή **when [throwButton].click ... do** στο χώρο σύνταξης του προγράμματος, δεξιά.



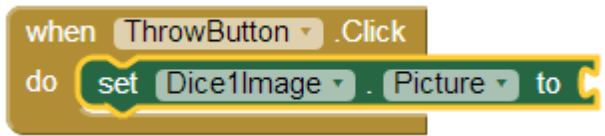
ThrowButton. Σέρνουμε με το ποντίκι την εντολή **when [throwButton].click ... do** στο χώρο σύνταξης του προγράμματος, δεξιά.

Βήμα 7: Τυχαίοι αριθμοί

12. Για να προσομοιώσουμε το ρίξιμο των ζαριών θα πρέπει να παράγουμε 2 τυχαίους αριθμούς, έναν για το κάθε ζάρι και να θέτουμε, ανάλογα με τον αριθμό που θα παραχθεί, και την κατάλληλη εικόνα ζαριού στα αντίστοιχα σημεία. Αρχικά για να αλλάξουμε την εικόνα που απεικονίζει το αντικείμενο **Dice1Image**, θα επιλέξουμε το συγκεκριμένο αντικείμενο για να βρούμε την εντολή **set Dice1Image.Picture to ...**



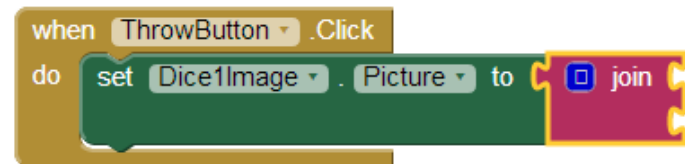
Κουμπώνουμε την εντολή κάτω από την προηγούμενη εντολή για να καταλήξουμε στο διπλανό σενάριο.



13. Στη συνέχεια θα πρέπει να προσθέσουμε στην εγκοπή της εντολής το όνομα του αρχείου εικόνας. Εδώ πρέπει να προσθέσουμε ένα κείμενο που θα περιέχει το όνομα του αρχείου. Το ένα τμήμα του ονόματος θα είναι ο αριθμός της εικόνας, τον οποίο θα παράγουμε με τυχαίο τρόπο και το δεύτερο τμήμα του κειμένου θα είναι η επέκτασή του **.png**. Επομένως θα πρέπει να συνδυάσουμε δύο τμήματα κειμένου.



Για το σκοπό αυτό θα χρησιμοποιήσουμε, από την ομάδα **Text**, της **Built-in** την εντολή **join**. Την εντολή αυτή θα πρέπει να την κουμπώσουμε με την εντολή αλλαγής εικόνας καταλήγοντας στο επόμενο σενάριο.



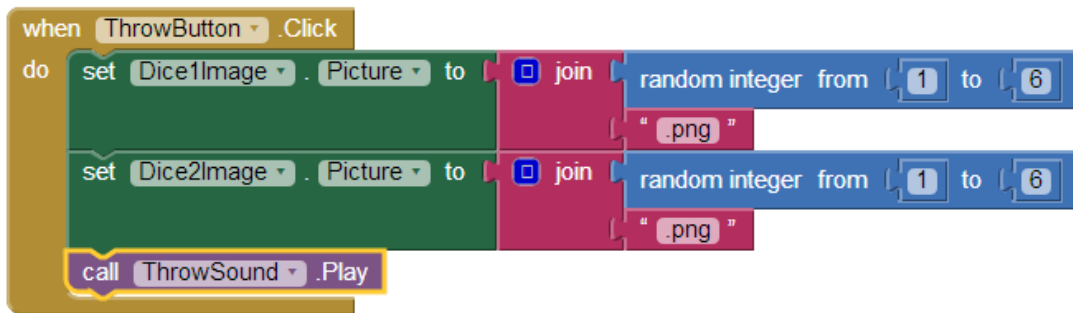
14. Στην πρώτη εγκοπή της εντολής θα βάλουμε το όνομα του αρχείου εικόνας και στη δεύτερη εγκοπή την κατάληξή του. Το όνομα του αρχείου της εικόνας είναι ένας αριθμός από το 1 μέχρι το 6. Στο σημείο αυτό θα πρέπει να παράγουμε έναν τυχαίο αριθμό σε αυτό το διάστημα που θα τον χρησιμοποιήσουμε σαν όνομα της εικόνας. Για το σκοπό αυτό θα ανατρέξουμε στην ομάδα **Math** της **Built-in** όπου εντοπίζουμε την εντολή **random integer from 1 to 100**. Θα προσαρμόσουμε τους αριθμούς που παράγονται στο διάστημα **1 – 6**, όσοι είναι και οι αριθμοί των ζαριών μας, αλλάζοντας το δεύτερο όριο της εντολής από 100 σε 6.
15. Στη 2η εγκοπή της εντολής **join** θα προσθέσουμε το κείμενο **.png** που είναι η κατάληξη του ονόματος αρχείου. Για το σκοπό αυτό, θα χρειαστούμε, από την ομάδα **Text**, το πλακίδιο κειμένου “ “, στο οποίο θα προσθέσουμε το **.png**. Ακολουθεί η τελική μορφή του κώδικα.

Βήμα 8: Διπλασιασμός εντολών

16. Είναι προφανές ότι θέλουμε να εκτελεστούν ακριβώς οι ίδιες ενέργειες για το αντικείμενο **Dice2Image**. Αντί να ξανακάνουμε τη διαδικασία που περιγράφηκε προηγουμένως μπορούμε απλά να διπλασιάσουμε την εντολή **set Dice1Image.Picture to** και να την τροποποιήσουμε κατάλληλα για να λειτουργεί με την εικόνα του 2ου ζαριού. Κάνουμε δεξί κλικ πάνω στην εντολή που θέλουμε να διπλασιάσουμε και επιλέγουμε την εντολή **Duplicate**, διπλασιασμός. Η μόνη αλλαγή που απαιτείται είναι να αλλάξουμε το όνομα του αντικείμενου από **Dice1Image** σε **Dice2Image**. Στη συνέχεια θα πρέπει να ενώσουμε τις εντολές όπως φαίνεται στο στιγμιότυπο.

Βήμα 9: Αναπαραγωγή ήχου

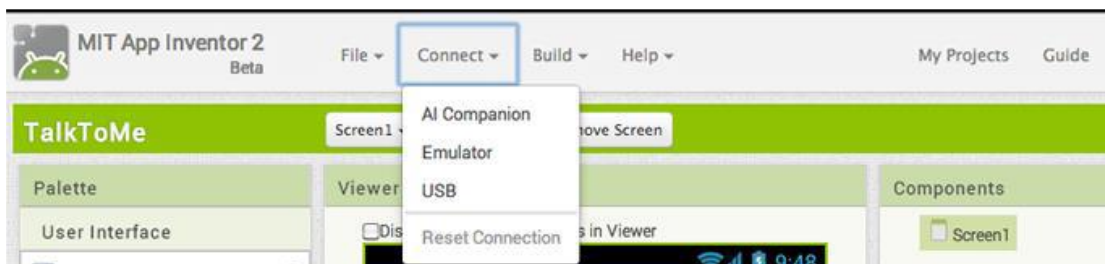
17. Η αναπαραγωγή του ήχου των ζαριών θα ενεργοποιείται και αυτή με το γεγονός του αγγίγματος του κουμπιού **Πίξε ζάρια**. Για να ξεκινήσει η αναπαραγωγή του ήχου θα χρησιμοποιήσουμε την εντολή **call ... Play** που θα βρούμε στο αντικείμενο **ThrowSound** της ομάδας **Screen1** την οποία και θα κωμώσουμε κάτω από τις υπόλοιπες. Ακολουθεί ολοκληρωμένο το πρόγραμμά μας.



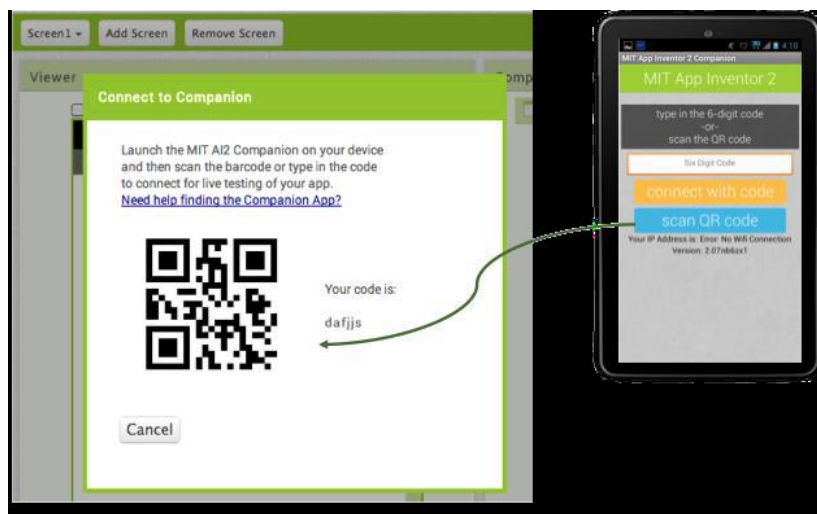
Η εφαρμογή μας είναι έτοιμη και μπορούμε να την δοκιμάσουμε στην android συσκευή μας.

Βήμα 10: Έλεγχος της εφαρμογής μας

18. Όταν θέλετε να ελέγξετε την εφαρμογή σας, επιλέγετε από το άνω μενού του App Inventor την εντολή **Connect - AI Companion**.



Εμφανίζεται ένα παράθυρο με έναν QR κώδικα. Ανοίγετε τώρα την εφαρμογή MIT AI2 Companion στη συσκευή σας και πατώντας το κουμπί **Scan QR Code** σαρώνετε τον κώδικα και η σύνδεση ολοκληρώνεται σε ελάχιστα δευτερόλεπτα. Εναλλακτικά, ή όταν κάτι δεν πάει καλά με την ανάγνωση του QR κώδικα, φυσικά μπορείτε να πληκτρολογήσετε τον εξαψήφιο κωδικό.



Μπορείτε πλέον να δοκιμάσετε την εφαρμογή σας σε πραγματικό χρόνο, ακόμα και να προσθέσετε ή να αφαιρέσετε αντικείμενα στο παράθυρο σχεδίασης, να προσθέσετε ή να τροποποιήσετε εντολές στο παράθυρο εντολών και να δείτε απευθείας τι συνέπειες έχουν οι ενέργειές σας.

19. Για να ξαναδοκιμάζετε όμως τις τροποποιήσεις σας στη συσκευή σας θα πρέπει κάθε φορά να επαναφέρετε τη σύνδεση επιλέγοντας **Connect – Reset Connection** πριν ξανασυνδεθείτε.

Βήμα 11: Προσθήκη αντικειμένου Accelerometer

Μία βελτίωση στο πρώτο μας App είναι η παρακάτω που εκμεταλλεύεται τους επιπλέον αισθητήρες της συσκευής σε σχέση με ένα παραδοσιακό υπολογιστικό σύστημα.

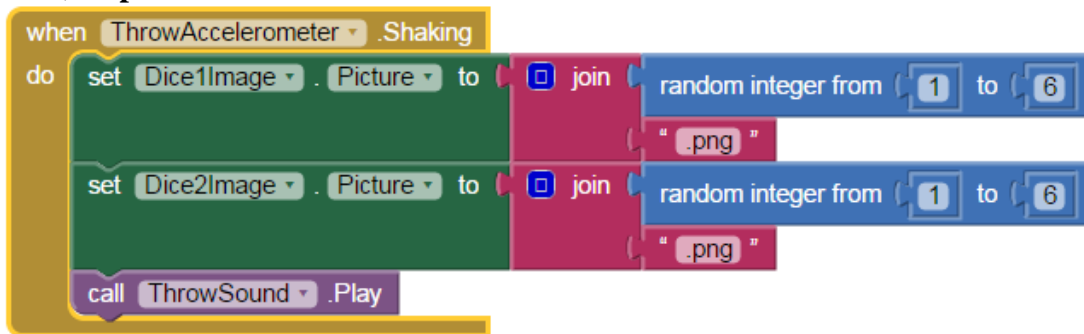
20. Προς το παρόν, το ρίξιμο των ζαριών ενεργοποιείται με το πάτημα ενός κουμπιού. Εναλλακτικά μπορείτε να χρησιμοποιήσετε τον αισθητήρα **Accelerometer**, επιταχυνσιόμετρο, της συσκευής σας, με τη βοήθεια του οποίου η συσκευή παίρνει πληροφορίες για το αν αυτή επιταχύνεται και σε ποιον άξονα καθώς και για το αν αυτή μετακινείται πέρα δώθε.

```
when AccelerometerSensor1 .Shaking
do
```

Επιστρέφουμε στο τμήμα σχεδίασης της εφαρμογής, **Designer**, και τοποθετούμε στην περιοχή σχεδίασης το αντικείμενο **AccelerometerSensor** από την ομάδα **Sensors**. Δώστε το όνομα *ThrowAccelerometer*.

21. Στη συνέχεια μεταβαίνουμε στο τμήμα εντολών, **Blocks**, και επιλέγουμε το κατάλληλο γεγονός ανίχνευσης του κουνήματος της συσκευής, που βρίσκεται κάτω από το αντικείμενο *ThrowAccelerometer*, για να το προσθέσουμε στις εντολές της εφαρμογής.

22. Οι εντολές που θα εκτελούνται με την κίνηση της συσκευής θα είναι ακριβώς οι ίδιες με αυτές του αγγίγματος του κουμπιού. Τις προσθέτουμε, αξιοποιώντας τη λειτουργία του διπλασιασμού εντολών, **Duplicate**.



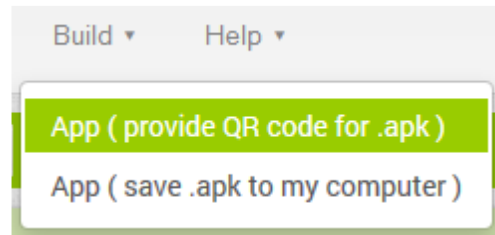
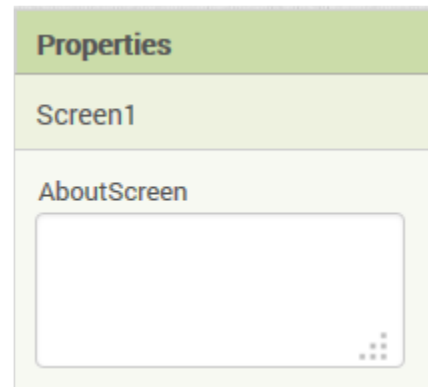
Ολοκληρώσατε την πρώτη σας εφαρμογή! Δοκιμάστε την στη συσκευή σας!



Βήμα 12: Πακετάρισμα και διαμοιρασμός (προαιρετικό)

Με τον προαναφερόμενο τρόπο μπορούμε μεν να δοκιμάσουμε την εφαρμογή που αναπτύσσουμε εύκολα και άμεσα, αλλά αυτό δεν αρκεί, αν θέλουμε η εφαρμογή να εγκατασταθεί μόνιμα στη συσκευή μας ή να την μοιραστήμε με άλλους. Απαιτείται λοιπόν να την «πακετάρουμε», δημιουργώντας το σχετικό αρχείο εγκατάστασής της (.apk).

23. Προτού δημιουργήσουμε το .apk αρχείο, καλό είναι να ορίσουμε μια εικόνα ως βασικό εικονίδιο της εφαρμογής, το οποίο θα εμφανίζεται, μετά την εγκατάστασή της στη συσκευή μας. Η πιο απλή λύση είναι να επιλέξουμε (εφόσον υπάρχει) μια χαρακτηριστική εικόνα από αυτές που χρησιμοποιούνται ήδη στην εφαρμογή μας και να την θέσουμε ως **Icon** της *Screen1* από το περιβάλλον σχεδίασης **Designer**.
24. Επιπρόσθετα, μπορούμε (πάλι προαιρετικά) να προσθέσουμε ένα μικρό κείμενο στην ιδιότητα **AboutScreen** της *Screen1*, το οποίο θα περιγράφει εν συντομία την εφαρμογή ή θα αναφέρει λίγα λόγια για το δημιουργό της.
25. Προκειμένου να δημιουργήσουμε το **.apk** αρχείο, μεταβαίνουμε στο μενού **Build** της βασικής γραμμής εργαλείων, όπως φαίνεται και στην εικόνα, όπου έχουμε δυο επιλογές:
- α. να εμφανιστεί ένας QR κωδικός, ώστε σαρώνοντάς τον να κατεβάσουμε το αρχείο εγκατάστασης απευθείας στη συσκευή μας (ο QR κωδικός λειτουργεί μόνο για το δημιουργό της εφαρμογής),
 - β. να αποθηκεύσουμε το αρχείο εγκατάστασης στον υπολογιστή μας για μετέπειτα διαμοιρασμό και εγκατάσταση με όποιον τρόπο προτιμάμε.
26. Αν επιθυμούμε να διαθέσουμε την εφαρμογή στο Google Play, απαιτείται η δημιουργία Publisher Account για τη μεταφόρτωση του .apk αρχείου. Για περισσότερες πληροφορίες: <http://appinventor.mit.edu/explore/ai2/google-play.html>.
27. Τέλος, αν θέλουμε να μοιραστούμε τον κώδικα της εφαρμογής μας, μαζί με το γραφικό περιβάλλον και τα συνοδευτικά αρχεία, μπορούμε να εξάγουμε το πηγαίο αρχείο, .aia, και να το διαθέσουμε σε άλλους, ώστε να το εισάγουν ως έργο, project, στο δικό τους λογαριασμό στο App Inventor. Αρκεί να μεταβούμε στη λίστα με τα έργα μας, My Projects, έπειτα να επιλέξουμε το έργο (ή τα έργα) που θέλουμε να εξάγουμε και στη συνέχεια, από το μενού Project, να επιλέξουμε την εντολή **Export selected project (.aia) to my computer**.



Δραστηριότητα

Χρησιμοποιώντας τα αρχεία εικόνας *1.png*, *2.png* και ήχου *CoinFlip.mp3* που θα βρείτε στον υποφάκελο **CoinFlip**, δημιουργήστε μία εφαρμογή προσομοίωσης του στριψίματος ενός νομίσματος. Αρχικά το νόμισμα θα έχει την μορφή της κορώνας (head) ή των γραμμάτων (tails) και κάθε φορά που θα πατάμε ένα κουμπί που θα δημιουργήσουμε ή θα κουνάμε το κινητό θα επιλέγεται με τυχαίο τρόπο η νέα εικόνα αλλά και θα ακούγεται ο ήχος που εισαγάγαμε.