

**UEL Number:** 1635291,

**Date:** 20/5/18

**Subject:** Ομαδική Εργασία

**Lesson:** Web Application Development

**Professor:** Χ. Φουτσιτζής



Περιεχόμενα

[Εισαγωγή 3](#_Toc514672658)

[Application Proposal: Παιχνίδι Ερωτήσεων Quiz 4](#_Toc514672659)

[Ανάλυση εντολών Activity Quiz : 6](#_Toc514672660)

[Αναλυτικότερα για το κώδικα του Activity Quiz : 8](#_Toc514672661)

[Ανάλυση εντολών της Quiz Activity : 9](#_Toc514672662)

[Αναλυτικότερα για το κώδικα του Quiz Activity : 11](#_Toc514672663)

[Ανάλυση κώδικα Question Library : 12](#_Toc514672664)

[Ανάλυση κώδικα Start.java : 14](#_Toc514672665)

[Ανάλυση κώδικα activity\_Start : 16](#_Toc514672666)

[Αναλυτικότερα για το κώδικα του Activity\_Start : 17](#_Toc514672667)

[Ανάλυση κώδικα activity\_About : 18](#_Toc514672668)

[Γραφικά και συνολική εμφάνιση 19](#_Toc514672669)

[Προβλήματα 20](#_Toc514672670)

[Βελτιώσεις 20](#_Toc514672671)

[ΒΙΒΛΙΟΓΡΑΦΙΑ : 21](#_Toc514672672)

# Εισαγωγή

Η εργασία αυτή αφορά την δημιουργία μίας εφαρμογής με την χρήση του εργαλείου της Google, Android Studio. Η εφαρμογή που θα δημιουργηθεί είναι ένα κουίζ ερωτήσεων γνώσης που θα αφορά ένα ευρύ κοινό νέων ατόμων μεταξύ 18 και 25 χρονών στους οποίους αρέσουν τα παιχνίδια γνώσεων. Για τα γραφικά της εφαρμογής θα χρησιμοποιηθεί το Adobe Photoshop 6.

# Application Proposal: Παιχνίδι Ερωτήσεων Quiz

Η εφαρμογή θα είναι ένα παιχνίδι ερωτήσεων σε μορφή quiz με τρεις επιλογές σε μια ερώτηση γενικών γνώσεων όπου η σωστή απάντηση θα είναι μόνο μια. Για να προχωρήσει παρακάτω ο παίχτης θα πρέπει πρώτα να βρει την σωστή απάντηση. Όταν επιλεχθεί μία λάθος απάντηση εμφανίζεται στο κάτω μέρος της οθόνης ένα κείμενο “Wrong Answer”. Όταν ο παίχτης επιλέξει την σωστή απάντηση, τότε προχωράει αυτόματα στην επόμενη ερώτηση. Για κάθε σωστή απάντηση θα υπάρχει ένας μετρητής όπου θα προσθέτει το score που έχει κάνει ο παίχτης. Η εφαρμογή θα αφορά χρήστες νέους σε ηλικία μεταξύ 18 και 25 χρονών με ενδιαφέρον να μάθουν, όπως φοιτητές οι οποίοι θέλουν να περάσουν ευχάριστα και παιδαγωγικά την ώρα τους και να ενημερωθούν με καινούργιες πληροφορίες.

Πιο συγκεκριμένα θα μπορούσε να αφορά ένα μαθητή ή φοιτητή όπως κάποιον που του αρέσει να παίζει παιχνίδια γνώσεων με τους φίλους του αλλά και μόνος του. Οπότε, θα έβρισκε σαν μια ευχάριστη απασχόληση για τον ελεύθερο του χρόνο ένα τέτοιο παιχνίδι το οποίο θα μπορούσε να παίξει οπουδήποτε αφού θα το είχε στο κινητό του.

Ένα τέτοιο παράδειγμα θα μπορούσε να είναι ο Κώστας.

Η.Γ: 22/04/1998

Φύλο : Άντρας

Πόλη : Αθήνα

Ηλικία : 20

Απασχόληση : Φοιτητής Οικονομικών Σπουδών

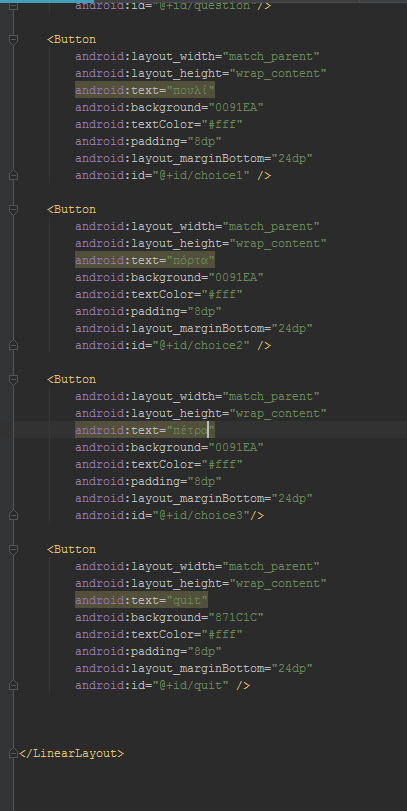


Ο Κώστας έχει εμμονή με τις γενικές γνώσεις. Είναι αρχηγός της ομάδας μπάσκετ της σχολής του και μέλος μιας ομάδας με φίλους του όπου διοργανώνουν επιτραπέζια παιχνίδια γνώσεων. Προσπαθεί σκληρά στη σχολή του και μαζί με τις προπονήσεις δε του μένει πολύ χρόνος. Εκτός σχολής, ο Κώστας βγαίνει με τους φίλους του και συνήθως συζητάνε για καινούργια παιχνίδια γνώσεων που έχουν βρει, επιτραπέζια ή εφαρμογές στο κινητό. Έχει μία mid-range συσκευή Samsung J5 του 2017 με λογισμικό Android 6.0 Marshmallow. Εκτός από το να μιλάει με τους φίλους του, μέσω εφαρμογών όπως το Messenger του Facebook και να παίζει σύντομα και απλά παιχνίδια κατά τη διάρκεια της διαδρομής που κάνει από το σπίτι που διαμένει μέχρι την σχολή δεν το χρησιμοποιεί για περαιτέρω λόγους.

Οι γονείς του Κώστα τον υποστηρίζουν πολύ και έχουν δώσει ένα τεράστιο χρονικό διάστημα για να τον βοηθήσουν να επιτύχει τους στόχους του. Ο Κώστας ζει σε μια πολυκατοικία στα νότια προάστια της πόλης αλλά η σχολή του είναι στο κέντρο. Οπότε κατά τη διαδρομή προσπαθεί πάντα να βρίσκει παιχνίδια στο κινητό του για να περνάει ευχάριστα η ώρα του. Η κοινωνική ζωή του Κώστα βασίζεται στον αθλητισμό και στους φίλους του.

# Ανάλυση εντολών Activity Quiz :





**Android**: id Αναγνωριστικό πόρων. Ένα μοναδικό όνομα πόρων για το στοιχείο, το οποίο μπορεί να χρησιμοποιηθεί για να αποκτήσει μια αναφορά στην ViewGroup από την εφαρμογή

**Android: layout\_width:** Διάσταση ή λέξη-κλειδί. Είναι το πλάτος ως τιμή διάστασης ή μια λέξη-κλειδί.

**Αndroid:layout\_height:** Διάσταση ή λέξη-κλειδί. Είναι το ύψος ως τιμή διάστασης ή μια λέξη-κλειδί.

**Android: padding Bottom:** Ορίζει τo γέμισμα στο κάτω άκρο της οθόνης.

**Android: padding Left:** Ορίζει τo γέμισμα στο αριστερό άκρο της οθόνης.

**Android: padding Right:** Ορίζει τo γέμισμα στο δεξιό άκρο της οθόνης.

**Android: padding Top:** Ορίζει τo γέμισμα στο πάνω άκρο της οθόνης.

**Android: text:** Παρέχει classes που χρησιμοποιούνται για την απόδοση ή την παρακολούθηση κειμένου και κειμένου σε οθόνη.

**Android: layout:** Ένα layout ορίζει τη δομή για μια διεπαφή χρήστη στην εφαρμογή, όπως σε μια δραστηριότητα.

**Android: background:**  Ένα σχέδιο που μπορεί να χρησιμοποιηθεί ως φόντο

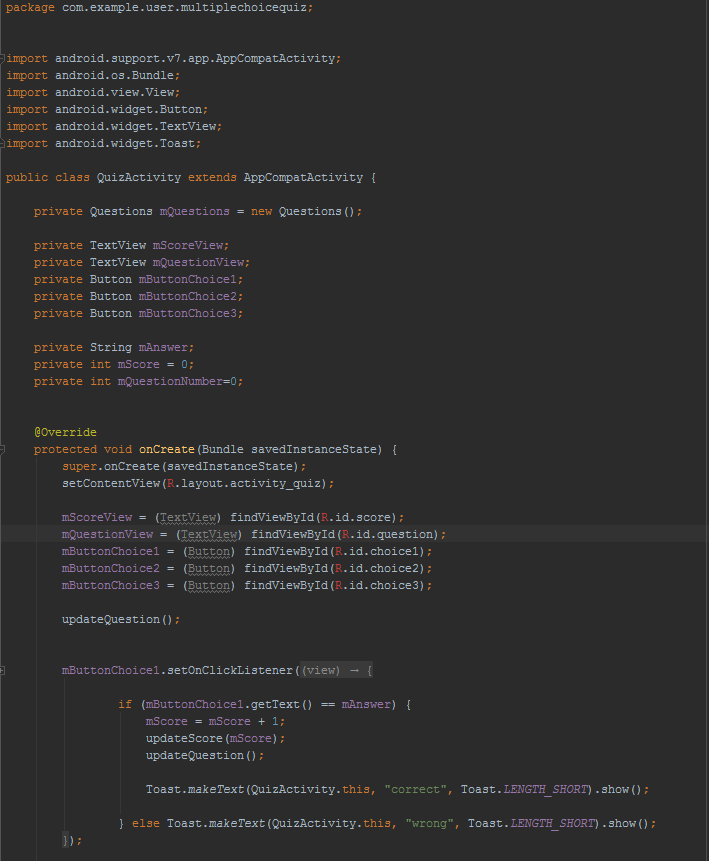
# Αναλυτικότερα για το κώδικα του Activity Quiz :

Δημιουργούμε ένα private object. Έπειτα, δημιουργούμε μεταβλητές για να κρατήσουν την βαθμολογία του χρήστη, για τα κουμπιά, τις ερωτήσεις και τις απαντήσεις και τα ορίζουμε όλα ως private. Μετά δημιουργούμε μεταβλητές για την απάντηση, το score και τον αριθμό της ερώτησης. Επίσης ορίζουμε το 0 σαν αρχική τιμή για το score και τον αριθμό της ερώτησης. Οπότε, συνδέουμε τις μεταβλητές με τα views και με την xml. Μετά για να είναι σωστή η απάντηση θα πρέπει το κείμενο από το κουμπί που πατήθηκε να είναι ίδιο με τη σωστή απάντηση έτσι ώστε το score να ανέβει συν ένα πόντο. Αν είναι σωστή η απάντηση το score θα μεγαλώσει κατά έναν βαθμό, θα εμφανίσει στον χρήση το μήνυμα Correct Answer (με χρήση Toast) και θα προχωρήσει στην επόμενη ερώτηση. Αν η απάντηση είναι λάθος θα εμφανίσει στον χρήστη ένα το μήνυμα Wrong Answer (Toast) και δεν θα προχωρήσει σε επόμενη ερώτηση μέχρι ο χρήστης να απαντήσει σωστά.

Το relative layout θα κρατήσει το score και το μετρητή των βαθμών. Θέτουμε το πρώτο text view όπου μέσα θα περιέχει το κείμενο με το score και το τοποθετούμε το πάνω και αριστερό μέρος της οθόνης. Στο επόμενο text view θέτουμε τη βαθμολογία του score να ξεκινάει από το 0.

Ξεκινάμε καινούργιο relative layout. Μέσα σε αυτό βάζουμε καινούργιο text view στο οποίο ορίζουμε μια ερώτηση. Έπειτα δημιουργούμε τέσσερα κουμπιά με διαφορετικό id από τα οποία τα τρία θα είναι οι απαντήσεις και το τέταρτο η επιλογή quit σε περίπτωση που ο χρήστης θέλει να αποχωρίσει από το παιχνίδι.

# Ανάλυση εντολών της Quiz Activity :



**Public class:** δημιουργεί μια δημόσια class

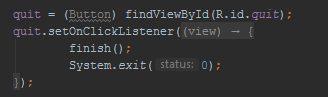
**Private TextView mScore, mQuestionView:** δίνει όνομα στα TextView από το xml αρχείο ώστε στην συνέχεια να τα εμπλουτίσει με χρήσιμα για την εφαρμογή χαρακτιριστικά.

**Private Button mButtonChoice1, 2, 3:** δίνει όνομα στα Buttons από το xml αρχείο ώστε στην συνέχεια να τα εμπλουτίσει με χρήσιμα για την εφαρμογή χαρακτιριστικά συνδέει τα Buttons1, 2, 3 από το xml αρχείο με τα choices από το QuestionLibrary αντίστοιχα.

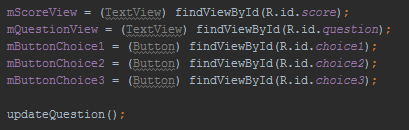
**Private Int mScore:** δηλώνει μία int μεταβλητή mScore ώστε να προστεθούν στη συνέχεια χαρακτιριστικά ως προς την χρησιμότητα της.

**Private int Question Number:** δηλώνει μία int μεταβλητή Question Number ώστε να προστεθούν στη συνέχεια χαρακτηριστικά ως προς την χρησιμότητα της.

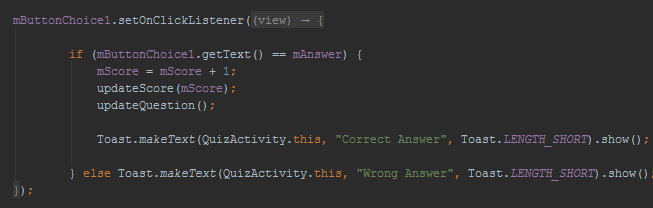
**Private String mAnswer:** δηλώνει μία String μεταβλητή mAnswer ώστε να προστεθούν στη συνέχεια χαρακτιριστικά ως προς την χρησιμότητα της.



Στον παραπάνω κώδικα δίνουμε χρήση στο button Quit ώστε όταν ο χρήστης το πατάει να τερματίζετε η εφαρμογή (finish();) και να κλείνει η εφαρμογή (System.exit(0);)).



Στον παραπάνω κώδικα συνδέονται τα TextView και Buttons από το Activity Quiz με τα αντίστοιχα στο Questions δίνοντας έτσι λειτουργία σε αυτά.



Στον παραπάνω κώδικα δημιουργείτε ένα if χάρη στο οποίο αν η απάντηση που θα δώσει ο χρήστης από το Button 1 ένα το οποίο είναι το choice 1 είναι σωστή, τότε στην μεταβλητή score θα προστεθεί ο αριθμός 1 και θα προχωρήσει στην επόμενη απάντηση. Το ίδιο ισχύει και για τα Button 2 και 3.

**Update Question:** κάθε φορά που απαντιέται μια ερώτηση θα προχωράει στην επόμενη.

**Toast.make Text Correct:** Αν η απάντηση είναι σωστή θα βγει το μήνυμα ότι είναι σωστή

**Toast.make Text Correct:** Αν η απάντηση είναι λάθος θα βγει το μήνυμα ότι είναι λάθος

# Αναλυτικότερα για το κώδικα του Quiz Activity :

Δημιουργεί μια δημόσια class με όνομα Quiz Activity. Μέσω του Private TextView mScore, mQuestionView, mButtonchoice, mButttonchoice2 και mButtonchoice3συνδέει τα ονόματα στα TextView από το xml και δίνει όνομα στα Buttons από το xml αρχείο και συνδέει τα Buttons1, 2, 3 με τα choices από το Question Library.

Με το Private String mAnswer: δηλώνει μία μεταβλητή για να κρατάει την απάντηση. Έπειτα μέσω του Private Int mScore: δηλώνει μία int μεταβλητή mScore για να κρατάει το score και να ξεκινάει από τη τιμή 0 . Επίσης με το Private int Question Number δηλώνει επίσης μία int μεταβλητή Question Number για να μετράει τη θέση των ερωτήσεων και να ξεκινάει από τη πρώτη ερώτηση.

Μετά μέσω του update Question κάθε φορά που θα απαντιέται μια ερώτηση θα προχωράει πάντα στην επόμενη και ανάλογα αν είναι σωστή ή λάθος θα εμφανίζεται το αντίστοιχο μήνυμα.

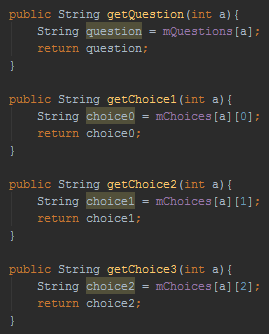
# Ανάλυση κώδικα Question Library :

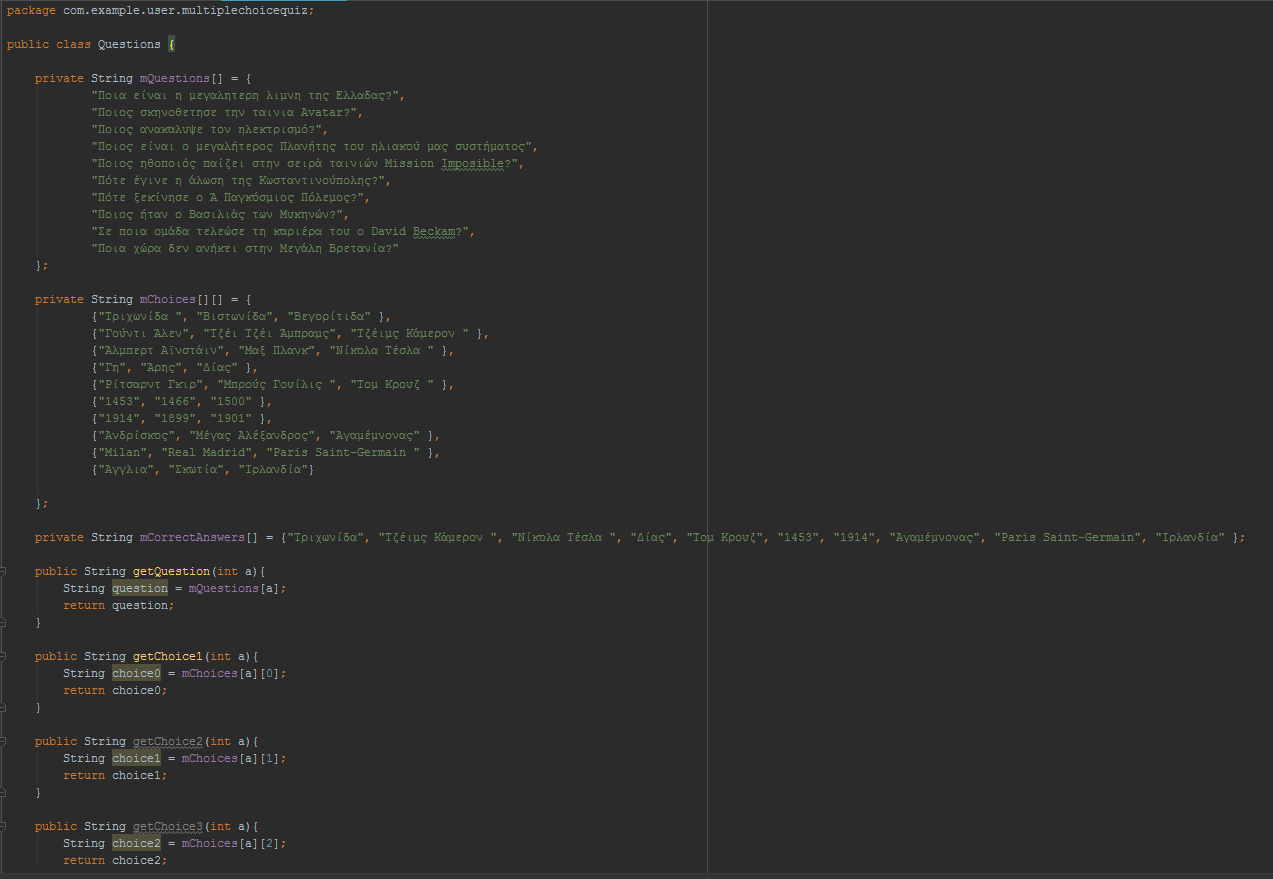






Δημιουργούμε τρείς μεταβλητές μελών μέσα στην δημόσια class QuestionLibrary. Κάθε μεταβλητή θα είναι ένα array. To πρώτο θα είναι ένα κανονικό array, το δεύτερο θα είναι μια διστιχία και το τρίτο πάλι κανονικό. Το πρώτο array θα περιέχει τις ερωτήσεις, το δεύτερο τις επιλογές – απαντήσεις και το τρίτο θα περιέχει τις σωστές απαντήσεις. Στο επόμενο βήμα δημιουργήθηκαν μέθοδοι για να επιστρέφουν τις ερωτήσεις και τη σωστή απάντηση.





Η πρώτη μέθοδος getQuestion θα επιστρέψει την ερώτηση από το mQuestions array.

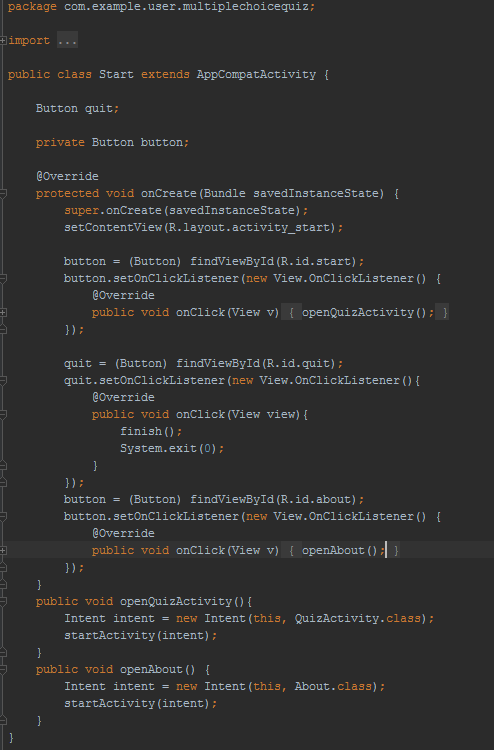
H μέθοδος getChoice1 θα επιστρέψει το πρώτο στοιχείο σε ένα δισδιάστατο array,

Η μέθοδος getChoice2 θα επιστρέψει το δεύτερο στοιχείο σε ένα array και η

μέθοδος getChoice3 θα επιστρέψει το τρίτο στοιχείο σε ένα δισδιάστατο array.

Η τελευταία μέθοδος είναι αυτή που θα επιστρέφει την σωστή επιλογή.

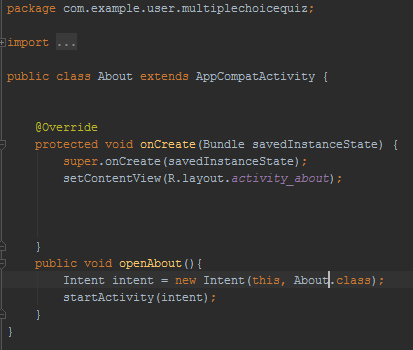
# Ανάλυση κώδικα Start.java :



Δημιουργούμε μια δημόσια κλάση Start και ένα κουμπί Quit

Ανάλυση κώδικα About.java :

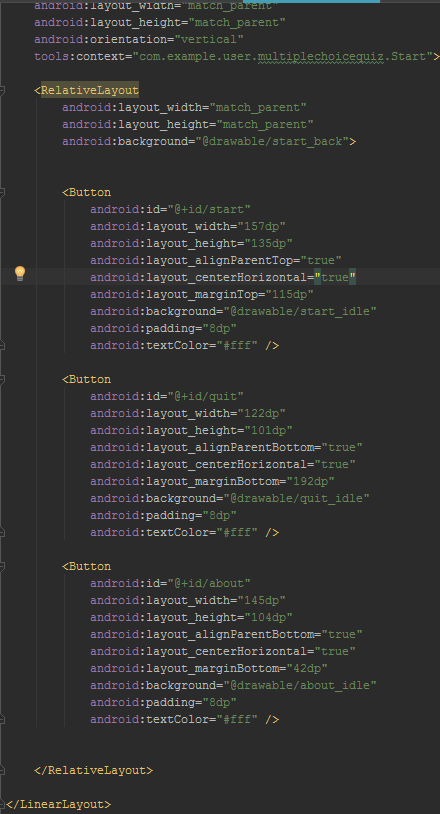
.



Δημιουργούμε μια δημόσια κλάση About

Όταν θα πατιέται το κουμπί About θα ανοίγει η αντίστοιχη σελίδα

# Ανάλυση κώδικα activity\_Start :



**Android: id** Αναγνωριστικό πόρων. Ένα μοναδικό όνομα πόρων για το στοιχείο, το οποίο μπορεί να χρησιμοποιηθεί για να αποκτήσει μια αναφορά στην ViewGroup από την εφαρμογή

**Android: layout\_width** Διάσταση ή λέξη-κλειδί. Είναι το πλάτος ως τιμή διάστασης ή μια λέξη-κλειδί.

**Αndroid:layout\_height** Διάσταση ή λέξη-κλειδί. Είναι το ύψος ως τιμή διάστασης ή μια λέξη-κλειδί.

**Android: padding** Ορίζει τo γέμισμα της οθόνης.

**Android: text** Παρέχει classes που χρησιμοποιούνται για την απόδοση ή την παρακολούθηση κειμένου και κειμένου σε οθόνη.

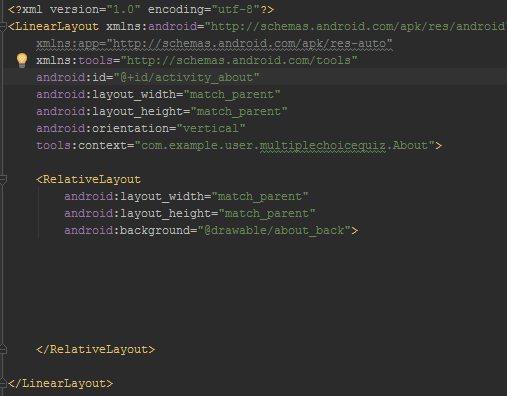
**Android: layout** Ένα layout ορίζει τη δομή για μια διεπαφή χρήστη στην εφαρμογή, όπως σε μια δραστηριότητα.

**Android: background** Ένα σχέδιο που μπορεί να χρησιμοποιηθεί ως φόντο

# Αναλυτικότερα για το κώδικα του Activity\_Start :

Δημιουργούμε τρία διαφορετικά κουμπιά ένα για να ξεκινάει το παιχνίδι (Start), ένα για να βγαίνει από το παιχνίδι ο χρήστης (Quit) και ένα με σχετικές πληροφορίες (About). Είναι στην ουσία η αρχική μας οθόνη. Με το που ανοίξει ο χρήστης την εφαρμογή είναι το πρώτο παράθυρο που βλέπει.

# Ανάλυση κώδικα activity\_About :



**Android: id** Αναγνωριστικό πόρων. Ένα μοναδικό όνομα πόρων για το στοιχείο, το οποίο μπορεί να χρησιμοποιηθεί για να αποκτήσει μια αναφορά στην ViewGroup από την εφαρμογή

**Android: layout\_width** Διάσταση ή λέξη-κλειδί. Είναι το πλάτος ως τιμή διάστασης ή μια λέξη-κλειδί.

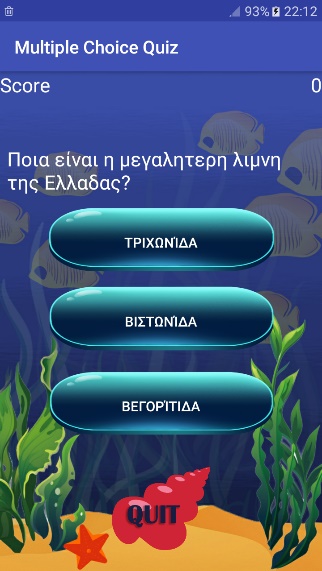
**Αndroid:layout\_height** Διάσταση ή λέξη-κλειδί. Είναι το ύψος ως τιμή διάστασης ή μια λέξη-κλειδί.

**Android: background** Ένα σχέδιο που μπορεί να χρησιμοποιηθεί ως φόντο

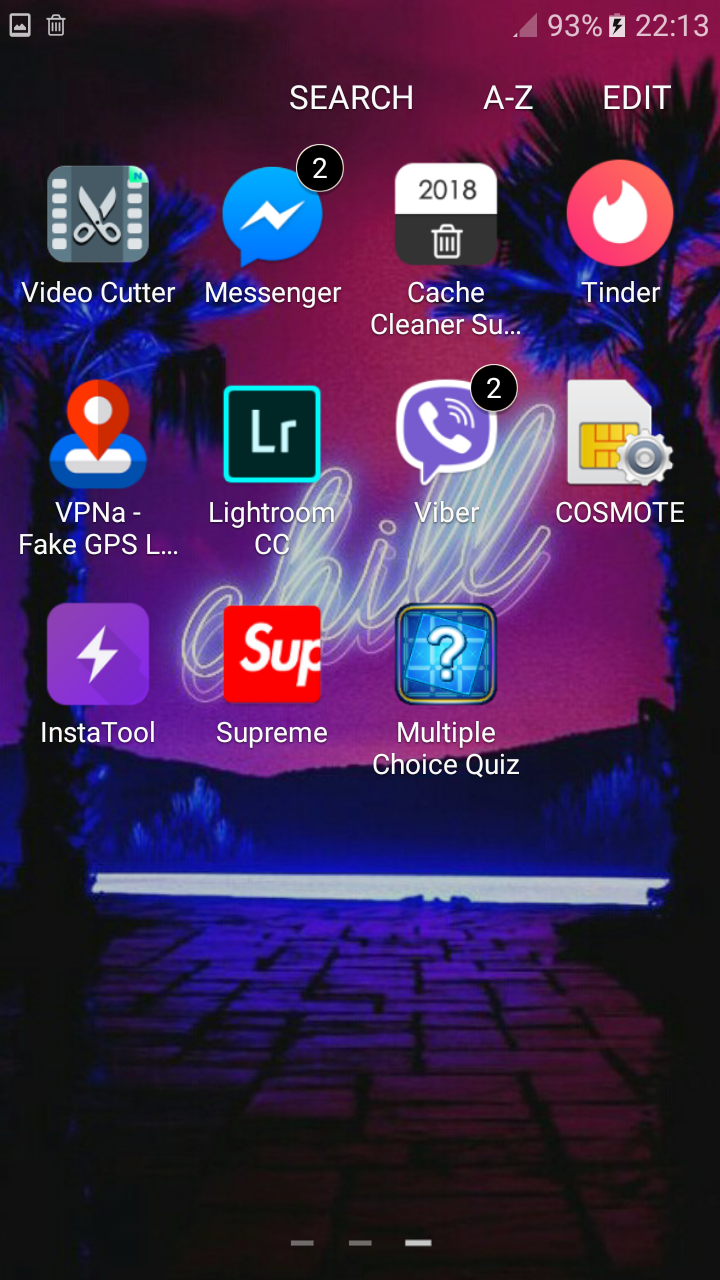
# Γραφικά και συνολική εμφάνιση

Αφότου δημιουργήθηκε ο κώδικας της εφαρμογής και έγινε έλεγχος για το αν λειτουργεί σωστά, το επόμενο βήμα ήταν ο σχεδιασμός της εμφάνισης. Αρχικά, έγινε σχεδίαση στο χέρι για το θέμα και το πώς θα είναι κάθε στοιχείο (κουμπιά, background κτλ). Έπειτα έγινε σχεδιασμός στο Photoshop ενός Blueprint του αρχικού παραθύρου. Με βάση το blueprint σχεδιάστηκαν ξεχωριστά το background της εφαρμογής και τα κουμπιά.

Στην πρώτη εικόνα φαίνεται η αρχική σελίδα, στην δεύτερη αφότου έχει πατηθεί το START και στην Τρίτη το ABOUT. Όταν πατηθεί το QUIT της αρχικής σελίδας τερματίζεται η εφαρμογή και όταν πατηθεί το QUIT του παιχνιδιού μας γυρίζει ξανά στην αρχική, έτσι ώστε ο παίχτης μπορεί να ξεκινήσει καινούρια παρτίδα ή να τερματίσει την εφαρμογή.

Τέλος, δημιουργήθηκε ένα εικονίδιο για την εφαρμογή.



# Προβλήματα

Προβλήματα που συναντήθηκαν ήταν κυρίως στο κομμάτι της εμφάνισης. Έπρεπε να δημιουργηθούν και τοποθετηθούν πολύ σωστά τα γραφικά ώστε να μην φαίνονται παραμορφωμένα λόγο της ανάλυσης του κινητού το οποίο χρησιμοποιήθηκε για τον έλεγχο της εφαρμογής. Στο κομμάτι της τοποθέτησης εικόνας για κουμπί “Image Button” δεν καταφέραμε να το δημιουργήσουμε, όποτε χρησιμοποιήσαμε ένα workaround. Δηλώσαμε την εικόνα σαν background του button και την μορφοποιήσαμε ανάλογα στο design κομμάτι του Android Studio. Το αποτέλεσμα είναι ακριβώς αυτό που θέλαμε και είχαμε σχεδιάσει το οποίο θα μπορούσε να είχε γίνει με το “Image Button”. Η λύση μας αποδείχθηκε πολύ αποτελεσματική.

# Βελτιώσεις

Αργότερα θα θέλαμε να προσθέσουμε στην εφαρμογή μας την δυνατότητα στον χρήστη να συνδέεται με το Google+ ή το Facebook λογαριασμό του. Έτσι ώστε εμείς να έχουμε μια βάση δεδομένων όπου θα βλέπουμε το ποιοι παίζουν το παιχνίδι μας, σε τι ηλικίες βρίσκονται κτλ. Αυτό θα μας βοηθούσε στην βελτίωση της εμπειρίας της εφαρμογής.

# ΒΙΒΛΙΟΓΡΑΦΙΑ :

1. (Andrew Hoog. 2011 Android Forensics)

2. (Wallace Jackson. 2014 Learn Android App Development)

3. (M. Harwani.2013 Android Programming Unleashed)

4. (Zigurd R. Mednieks, Laird Dornin 2015 Programming Android)