**ΕΡΓΑΣΙΑ Β ΤΕΤΡΑΜΗΝΟΥ Γ’ ΓΥΜΝΑΣΙΟΥ**

**ΠΕΡΙΓΡΑΦΗ ΕΦΑΡΜΟΓΗΣ**: Θα κατασκευάσουμε μια εφαρμογή που θα δίνει τη δυνατότητα να παρουσιάζουμε ένα σύνολο από φωτογραφίες. Ο χρήστης της εφαρμογής θα έχει τη δυνατότητα να μετακινηθεί από τη μια φωτογραφία στην επόμενη με 3 διαφορετικούς τρόπους. (α) Αγγίζοντας ένα κουμπί, (β) σύροντας το δάκτυλό του πάνω στην οθόνη και (γ) αυτόματα κάθε 3 δευτερόλεπτα.

**ΒΗΜΑΤΑ ΥΛΟΠΟΙΗΣΗΣ**

ΒΗΜΑ 1: Ξεκινώντας, δημιουργούμε ένα καινούργιο project με όνομα **PhotoGallery**.

Ορίζουμε τις ιδιότητες της Screen1:

**AlignHorizontal**: Center

**Screen Orientation**: Portrait

**Scrollable**: No

**Title**: Photo Gallery

ΒΗΜΑ 2: Φορτώνουμε τα αρχεία που θα χρειαστούμε από το φάκελο **PhotoGallery**.

ΒΗΜΑ 3: Θα χρειαστούμε το αντικείμενο του καμβά, πάνω στο οποίο θα εμφανίζονται οι φωτογραφίες του άλμπουμ. Επιπλέον θα προσθέσουμε τα κουμπιά για μετακίνηση σε προηγούμενη και επόμενη φωτογραφία, το ρολόι για την αυτόματη εναλλαγή και ένα αντικείμενο Player για την αναπαραγωγή της μουσικής. Τέλος σε μια ετικέτα θα εμφανίζουμε τον αριθμό της εικόνας που απεικονίζεται εκείνη τη στιγμή στην οθόνη.

|  |  |  |
| --- | --- | --- |
| **ΑΝΤΙΚΕΙΜΕΝΟ** | **ΟΝΟΜΑ ΑΝΤΙΚΕΙΜΕΝΟΥ** | **ΙΔΙΟΤΗΤΕΣ** |
| Layout > HorizontalArrangement | ButtonsArea | AlignHorizontal: Center  BackgroundColor: επιλογή σας  Width: Fill Parent |
| **Μέσα στο HorizzontalArrangment** | | |
| User Interface> Button | PreviousButton | Image : previous.png  Text : ΚΕΝΟ |
| User Interface > Label | PicNumberLabel | FontSize : 25  Text : ΚΕΝΟ  TextAlignment: center  Width : 100 pixels |
| User Interface Button | NextButton | Image : next.png  Text : ΚΕΝΟ |
| **Κάτω από το HorizzontalArrangment** | | |
| Drawing and Animation > Canvas | GalleryCanvas | BackgroundImage : 1.jpg |
| Sensors> Clock | TimerClock | TimerInterval : 3000 (δηλαδή 3 sec) |
| Media > Player | MusicPlayer | Source : Averil White - Death of the Hero.mp3 |

ΒΗΜΑ 4: Για να επιτύχουμε την εναλλαγή των εικόνων θα χρειαστεί να δημιουργήσουμε μια μεταβλητή, η οποία θα παίρνει διαδοχικά τις τιμές από 1 μέχρι 6 και ξανά από την αρχή. Η τρέχουσα τιμή της μεταβλητής θα μας δείχνει και τη φωτογραφία που πρέπει να απεικονίσουμε πάνω στον καμβά (για το λόγο αυτό και τα αρχεία των εικόνων έχουν ονομαστεί κατάλληλα με αριθμούς από το 1 μέχρι το 6).

Για να ορίσουμε μεταβλητή στην προβολή **Blocks** στα αριστερά επιλέγουμε **VARIABLES>INITIALIZE GLOBAL “name” to**

Ας ονομάσουμε τη μεταβλητή μας **imageIndex** με αρχική τιμή 1. 

Όταν ο χρήστης κάνει κλικ στο κουμπί NEXT, η εφαρμογή θα αυξάνει κατά 1 την τιμή της μεταβλητής κάνοντας έλεγχο, όταν ξεπεράσει την τιμή 6, να της δίνει ξανά την τιμή 1. Με τον τρόπο αυτό οι εικόνες θα προβάλλονται κυκλικά. Στη συνέχεια η εφαρμογή θα ανανεώνει την εικόνα φόντου του αντικειμένου **GalleryCanvas** και το κείμενο της ετικέτας με τον αριθμό της εικόνας που προβάλλεται. Συγκεκριμένα θα γίνονται οι ακόλουθες εντολές. ΌΤΑΝ ΚΑΝΕΙΣ ΚΛΙΚ ΣΤΟ ΚΟΥΜΠΙ NEXT

Αύξησε την τιμή της μεταβλητής κατά 1.

Θέσε ως εικόνα φόντου του καμβά το αρχείο με όνομα ίδιο με την τιμή της μεταβλητής και κατάληξη .jpg

Θέσε το κείμενο της ετικέτας ίσο με τον αριθμό της εικόνας που προβάλλεται.

Αν αυτή ξεπεράσει το 6, δώσε της την τιμή 1, όρισε ως εικόνα φόντου του καμβά την 1.png και όρισε ως κείμενο ετικέτας ίσο με τον αριθμό της εικόνας.

Παρατηρήστε τώρα, ότι πρέπει να τοποθετήσουμε τις παραπάνω εντολές κάτω από τα 3 διαφορετικά γεγονότα, δηλαδή το άγγιγμα του κουμπιού, την πυροδότηση του ρολογιού και το σύρσιμο του δάκτυλου στην οθόνη. Κάτι τέτοιο δε φαίνεται και πολύ λογικό, αφού θα περιγράφουμε ακριβώς τις ίδιες ενέργειες και στις 3 περιπτώσεις. Με τον τρόπο αυτό φορτώνουμε την εφαρμογή μας με τον ίδιο κώδικα ξανά και ξανά, κάνοντας την δυσανάγνωστη και λιγότερο κατανοητή σε κάποιον που τη διαβάζει.

Για να επιλύσουμε το παραπάνω πρόβλημα, μπορούμε να τοποθετήσουμε τις εντολές για την εναλλαγή των εικόνων μέσα σε μια **διαδικασία**. Η διαδικασία είναι ένα υποπρόγραμμα, δηλαδή ένα μικρό και αυτόνομο τμήμα της εφαρμογής στο οποίο τοποθετούμε μια σειρά εντολών. Στην ουσία χρησιμοποιώντας διαδικασίες δημιουργούμε νέες εντολές που μπορούμε στη συνέχεια να καλούμε (χρησιμοποιούμε) από οποιοδήποτε σημείο του προγράμματος.

Για να δημιουργήσουμε μια νέα διαδικασία μεταβαίνουμε στην ομάδα Procedures και επιλέγουμε την εντολή

**to [procedure] do**

Θα μετονομάσουμε τη διαδικασία μας από procedure σε **changeImageNext**, και θα τοποθετήσουμε τις εντολές για την εναλλαγή των εικόνων μέσα σε αυτή. (Είναι οι εντολές που καθορίζονται από το πιο πάνω πλαίσιο) οι εντολές που επαναλαμβάνονται , θα εφαρμοστούν 1 φορά.

ΒΗΜΑ 5: Για να εκτελεστούν οι εντολές που βρίσκονται μέσα σε μια διαδικασία θα πρέπει να την καλέσουμε. Η εντολή για την κλήση μιας διαδικασίας που αναπτύσσουμε βρίσκεται και αυτή στην ομάδα Procedures και είναι **call** **changeImageNext**.

Θα καλέσουμε τη διαδικασία και στα 3 γεγονότα:

* **When Timer.Clock.Timer do**
* **When NextButton.Click do**
* **When PreviousButton.Click do**

ΒΗΜΑ 6: Κάντε τις απαραίτητες ενέργειες ώστε κατά την εκκίνηση της εφαρμογής να ξεκινάει και η αναπαραγωγή της μουσικής (αρχικοποίηση οθόνης). Ρυθμίστε κατάλληλα το αντικείμενο Player, ώστε όταν η μουσική τελειώνει να ξεκινάει ξανά από την αρχή (ιδιότητα loop στην προβολή Designer).

BHMA 7: Κάντε τις απαραίτητες ενέργειες, ώστε ο χρήστης της εφαρμογής να μπορεί να μετακινείται στην προηγούμενη εικόνα με δύο τρόπους.

* Είτε αγγίζοντας το κουμπί Προηγούμενο,
* είτε σέρνοντας το δάκτυλό του προς τα αριστερά.

Υλοποιήστε μια διαδικασία με όνομα **changeImagePrevious** που θα εκτελεί τις απαραίτητες ενέργειες, όπως και πριν.

Για να ανιχνεύσετε τη φορά προς την οποία μετακινείται το δάκτυλο του χρήστη πάνω στον καμβά θα χρειαστείτε την ιδιότητα **heading** (κατεύθυνση), την τιμή της οποίας σας παρέχει το γεγονός **when [GalleryCanvas] Flung.**

* Για να ανιχνεύσετε την κίνηση του δάκτυλου προς τα δεξιά ελέγξτε αν η ιδιότητα **heading** έχει τιμές από -30 μέχρι 30. Αν ελεγχθεί η ιδιότητα heading και είναι στο διάστημα [-30 ως 30] τότε θα θεωρείται ότι έχει επιλεγεί το κουμπί ΕΠΟΜΕΝΟ
* Για να ανιχνεύσετε την κίνηση του δάκτυλου προς τα αριστερά ελέγξτε αν η ιδιότητα **heading** έχει τιμές από -200 μέχρι -160. Αν ελεγχθεί η ιδιότητα heading και είναι στο διάστημα [-200 ως -160] τότε θα θεωρείται ότι έχει επιλεγεί το κουμπί ΠΡΟΗΓΟΥΜΕΝΟ

Βήμα 8: Συγχωνεύοντας τις δύο διαδικασίες σε μία

Αν παρατηρήσουμε προσεκτικά τις δύο διαδικασίες που υλοποιήσαμε για την μετακίνηση στην επόμενη και στην προηγούμενη εικόνα, θα δούμε ότι εκτελούν στην ουσία σχεδόν τις ίδιες εντολές. Αυτό που τις διαφοροποιεί είναι η αύξηση ή η μείωση του αντίστοιχου αριθμού εικόνας κατά ένα. Επίσης, διαφέρουν στον έλεγχο για το αν ο αριθμός ξεπέρασε το 6 ή έγινε μικρότερος του 1, αλλά και αυτό εξαρτάται από το αν ο αριθμός αυξάνεται ή μειώνεται.

*Μπορούμε να πετύχουμε τους σκοπούς μας με μια διαδικασία αντί για δυο;*

Έχουμε τη δυνατότητα να στέλνουμε τιμές σε μια διαδικασία τη στιγμή που την καλούμε. Οι τιμές αυτές ονομάζονται παράμετροι της διαδικασίας. Στην ουσία οι παράμετροι είναι μεταβλητές που μεταφέρουν τιμές από την εφαρμογή προς τη διαδικασία. Για να προσθέσουμε μια ή περισσότερες παραμέτρους σε μια διαδικασία κάνουμε κλικ στο μπλε τετράγωνο που βρίσκεται στα αριστερά του ονόματος της διαδικασίας. Ας επιλέξουμε τη διαδικασία με όνομα **changeImageNext**. Για να προσθέσουμε παραμέτρους, από το παράθυρο που εμφανίζεται, σέρνουμε το πλακίδιο **input** μέσα στο πλακίδιο **inputs**. Βάζουμε τόσα πλακίδια, όσες και οι παράμετροι που θέλουμε να προσθέσουμε στη διαδικασία. Στη θέση του x βάζουμε το όνομα που θέλουμε να έχει κάθε παράμετρος. Στην εφαρμογή μας θα την ονομάσουμε **number**.

Επίσης θα μετονομάσουμε τη διαδικασία changeImageNext σε **changeImage**. Η πρώτη αλλαγή που θα κάνουμε στις εντολές της διαδικασίας είναι να προσθέτουμε την τιμή της παραμέτρου number αντί να προσθέτουμε το 1, χρησιμοποιώντας την παράμετρο. Επιπλέον θα προσθέσουμε τους 2 ελέγχους με 2 εντολές if.

Βήμα 9: Η διαδικασία που υλοποιήσαμε πρέπει κάθε φορά που καλείται να συνοδεύεται από έναν αριθμό, που είναι η τιμή που θέλουμε να δώσουμε στην παράμετρο **number**. Επομένως, στα σημεία που θέλουμε να μετακινούμαστε στην επόμενη εικόνα, θα δίνουμε σαν τιμή το +1, ενώ στα σημεία που θέλουμε να γίνεται μετακίνηση στην προηγούμενη εικόνα, θα δίνουμε σαν τιμή το -1.

**Σημείωση**: Μη ξεχάσετε να απενεργοποιείται το ρολόι (TimerClock) όταν τρέχουν οι διαδικασίες **NextButton.Click** και **PreviousButton.Click**  και μετά τα ενεργοποιούνται πάλι.