Machine Translated by Google

סיור מפת פריז

איור 6-1.

בפרק זה, תבנה אפליקציית מדריכי טיולים לטיול לפריז. יצירת אפליקציית מפות שפועלת במלואה עשויה להיראות מאוד מסובכת, אבל App Inventorמספק שני רכיבים ברמה גבוהה כדי לעזור: ,ActivityStarterהמאפשר לך להפעיל אפליקציה נוספת מהאפליקציה שלך, כולל מפות ,Google-ו ,WebViewerשמציג כל דף אינטרנט שאתה רוצה בתוך לוח משנה של האפליקציה שלך. תחקור את שני המרכיבים הללו ותבנה שתי גרסאות שונות של מדריך טיולים.

מה תלמד

פרק זה מציג את הרכיבים והמושגים הבאים של ממציא האפליקציות:

•הרכיב Activity Starterלהפעלת אפליקציות אנדרואיד אחרות מהמכשיר שלך
אפליקציה.

•רכיב WebViewerלהצגת דפי אינטרנט בתוך האפליקציה שלך.

•כיצד להשתמש במשתני רשימה לאחסון מידע עבור האפליקציה שלך.

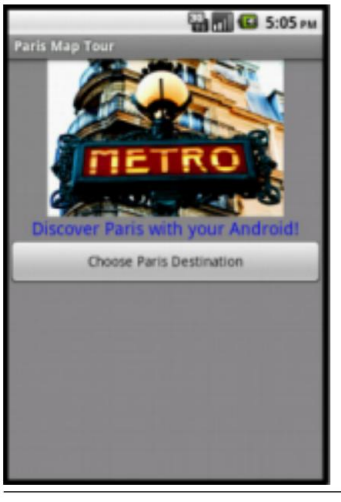
Machine Translated by Google

•רכיב ListPickerלתת למשתמש את היכולת לבחור מתוך רשימה של מיקומים.

•כיצד לבנות כתובת URLבאופן דינמי כדי להציג מפות שונות.

עיצוב הרכיבים

צור פרויקט חדש ב-ppA Inventorוקרא לו ."ParisMapTour"ממשק המשתמש עבור לאפליקציה יש רכיב Imageעם תמונה של פריז, רכיב Labelעם קצת טקסט, רכיב ListPickerשמגיע עם כפתור משויך, ובזה גרסה ראשונה, רכיב ActivityStarter(לא גלוי). אתה יכול לעצב את רכיבים באמצעות תמונת המצב באיור .6-1



איור .6-2אפליקציית Paris Map Tourפועלת באמולטור

תזדקק לרכיבים המפורטים בטבלה 6-1כדי לבנות אפליקציה זו. גרור כל אחד רכיב מהלוח לתוך ה-reweiV ושם אותו כפי שצוין.

טבלה .6-1רכיבים לסיור המפה בפריז

קומפוננטה	מַטָרָה			
[illegible]	[illegible]			
[illegible]	[illegible]			
[illegible]	[illegible]			
ActivityStarter1	ActivityStarterקישוריות הפעל את אפליקציית המפות כאשר נבחר יעד.			

Machine Translated by Google

הגדרת המאפיינים של ActivityStarter

ActivityStarterהוא רכיב שבאמצעותו אתה יכול להפעיל כל אפליקציית אנדרואיד, כולל מפות Googleאו אפליקציה אחרת משלך. תחילה תבנה את ParisMapTourכך שאפליקציית המפות תופעל כדי להציג מפות מסוימות בהתבסס על בחירת המשתמש. לאחר מכן המשתמש יכול להקיש על כפתור החזרה כדי לחזור לאפליקציה שלך ולבחור יעד אחר.

ActivityStarterהוא רכיב ברמה נמוכה יחסית בכך שתצטרך להגדיר חלק נכסים עם מידע שיהיה מוכר למתכנת Java Android SDK, אך זר לחלוטין ל-99.99% האחרים מהעולם. עבור אפליקציה זו, הזן את המאפיינים כפי שצוינו בטבלה 6-2, והיזהר - הם תלויי רישיות, כלומר חשוב אם אות היא אות גדולה או קטנה.

טבלה 6-2. מאפייני ActivityStarterלהפעלת מפות Google

ערך	תכונה
android.intent.action.VIEW	פעולה
com.google.android.apps.maps	ActivityPackage
com.google.android.maps.MapsActivity	ActivityClass

בעורך הבלוקים, תגדיר עוד מאפיין אחד, DataUri, המאפשר לך לספק כתובת URL להפעלת מפה ספציפית במפות Google. יש להגדיר מאפיין זה בעורך הבלוקים במקום Component Designer-במכיוון שהוא צריך להיות דינמי: הוא ישתנה על סמך האם המשתמש יבחר לבקר במגדל אייפל, בלובר או בקתדרלת נוטרדאם.

נגיע לעורך בלוקים בעוד רגע, אבל יש עוד כמה פרטים לטפל לפני שתוכל לעבור לתכנות ההתנהגות עבור הרכיבים שלך:

1. הורד את הקובץ metro.jpg כדי לטעון לפרויקט שלך. לאחר מכן, הגדר אותו כמאפיין Picture של Image1.

2. רכיב ListPicker מגיע עם כפתור; כאשר המשתמש לוחץ עליו, האפשרויות מופיעות. הגדר את הטקסט של כפתור זה על ידי שינוי מאפיין הטקסט של ListPicker1 ל"בחר יעד בפריז".

הוספת התנהגויות לרכיבים

בעורך הבלוקים, תצטרך להגדיר רשימה של יעדים ושתי התנהגויות:

Machine Translated by Google

•כאשר האפליקציה מתחילה, האפליקציה טוענת את היעדים לתוך ListPicker
רכיב כך שהמשתמש יוכל לבחור אחד.

•כאשר המשתמש בוחר יעד מתוך ,ListPickerאפליקציית המפות
מופעל ומציג מפה של יעד זה. בגרסה הראשונה של האפליקציה,
פשוט תפתח את מפות ותנחה אותו להריץ חיפוש אחר היעד הנבחר.

יצירת רשימת יעדים

פתח את עורך בלוקים וצור משתנה עם רשימת יעדי פריז לפי
באמצעות הבלוקים המפורטים בטבלה .6-3

טבלה .6-3בלוקים ליצירת משתנה יעדים

אתחול משתנים גלובליים ("יעדים") צור רשימה של היעדים.		
פשוט רשימת פריטים לרשימה.		
פריט הראשון.		טקסט ("סיור אייפל")
פריט השני.		טקסט ("מוזיאון הלובר")
פריט השלישי.		טקסט ("קתדרלת נוטרדאם")

כאשר אתה גורר את בלוק יצירת רשימה לתוך האפליקציה שלך, יהיו לו רק שניים זמינים
שקעים. אתה יכול להוסיף עוד אחד על ידי לחיצה על הסמל הכחול כהה אותו והוספת שלישי
פריט.

לאחר שעשית את זה, פשוט צור את בלוקי הטקסט עבור כל יעד ומקום
אותם בשלושת השקעים של צור רשימה, כפי שמוצג באיור .6-2

איור .6-3רשימה של שלושה פריטים

מתן אפשרות למשתמש לבחור יעד

הרשימה שזה עתה הגדרת אינה מופיעה בממשק המשתמש -אין משתנים. אתה תעשה
השתמש ברכיב ListPickerכדי להציג את רשימת הפריטים לבחירת המשתמש.
אתה טוען מראש את האפשרויות לתוך ListPickerעל ידי הגדרת המאפיינים Elementsלרשימה.
עבור אפליקציה זו, אתה רוצה להגדיר את המאפיין Elementsעבור ListPickerל-
רשימת היעדים שיצרת זה עתה. כי זה צריך להיות מוגדר רק פעם אחת, אתה תעשה זאת

Machine Translated by Google

הגדר התנהגות זו באירוע . Screen1.Initializeאתה תצטרך את הבלוקים שהם
המפורטים בטבלה .6-4

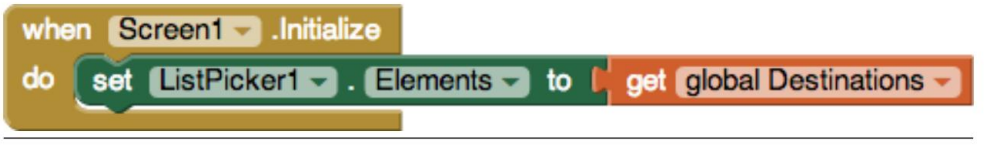
Machine Translated by Google

טבלה 6-4. חסימות להפעלת ListPicker כאשר האפליקציה מתחילה

מַטָרָה	מִגרָה	סוג בלוק
אירוע זה מופעל כאשר האפליקציה מתחילה.	מסך .1אתחול	
הגדר מאפיין זה לרשימה שברצונך שתופיע.	הגדר את	1rekciPtsiL-לListPicker1.Elements
רשימת היעדים.	גרור החוצה מבלוק אתחול משתנה	להשיג יעדים גלובליים

איך הבלוקים עובדים

מסך .1אתחול מופעל כאשר האפליקציה מתחילה. איור 6-3ממחיש שהמטפל באירועים מגדיר את המאפיין Elementsשל ListPickerכך ששלושת היעדים יופיעו.



איור 6-4. אתחל את ListPickerעם שלוש האפשרויות כאשר האפליקציה מופעלת

בדוק את האפליקציות שלך לחץ על התחבר והגדר בדיקה חיה עם המכשיר או האמולטור שלך. לאחר מכן, לחץ על הכפתור שכותרתו "בחר יעד בפריז". בוחר הרשימה אמור להופיע עם שלושת הפריטים. בשלב זה, שום דבר לא אמור לקרות כשאתה בוחר פריט.

פתיחת מפות עם כתובת אתר לחיפוש

לאחר מכן, תתכנתו את האפליקציה כך שכאשר המשתמש בוחר באחד מהיעדים, ActivityStarterמפעיל את מפות גוגל ומחפש את המיקום הנבחר.
ראשית, שקול את כתובת האתר http://maps.google.com?q=Parisכאשר אתה מקליד כתובת אתר זו בשורת הכתובת של דפדפן, היא מציגה מפה של פריז. ה "?"משותף לכתובות URLרבות; זה מסמל שמגיע פרמטר. פרמטר הוא המידע שהאתר צריך כדי לעבד את הבקשה. במקרה זה, שם הפרמטר הוא "q", קיצור של "שאילתה", והערך שלו הוא "פריז". הוא מורה למפות גוגל איזו מפה להציג.
באפליקציה זו, תבנה כתובת URLבאופן דינמי, ותוסיף את ערך הפרמטר על סמך המיקום שהמשתמש בוחר. כך תוכלו להציג מפות שונות בהתאם לבחירותיו של המשתמש.

Machine Translated by Google

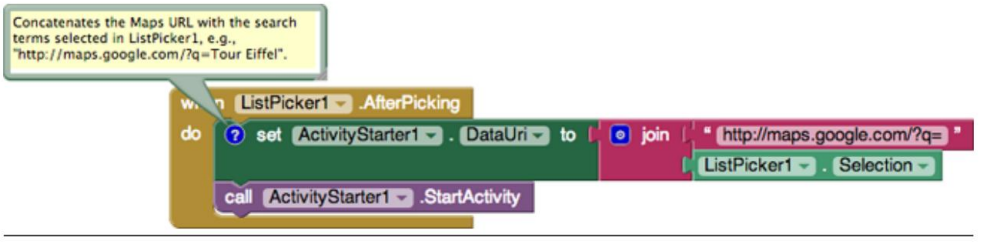
כאשר המשתמש בוחר פריט מתוך רכיב , ListPickerה-
אירוע ListPicker.AfterPickingמופעל. במטפל האירועים עבור ,AfterPicking
אתה צריך להגדיר את DataUriשל רכיב ActivityStarterכך שהוא יודע
איזו מפה לפתוח, ולאחר מכן עליך להפעיל את מפות Googleבאמצעות
.ActivityStarter.StartActivityהבלוקים עבור פונקציונליות זו מפורטים ב
טבלה .6-5

טבלה .6-5חסימות להפעלת מפות Googleעם Activity Starter-ה

מַטָרָה	מִגֵרָה	סוג בלוק
אירוע זה מופעל כאשר המשתמש בוחר מתוך ListPicker.	ListPicker1	ListPicker1.AfterPicking
ה-irUataD מורה למפות באיזו מפה להיפתח לְהַשִׁיק.	הגדר את ActivityStarter1-ל ActivityStarter1.DataUri	
בנה את ה-irUataD משתי[illegible] טקסט.		
החלק הראשון של DataUri [illegible] ("http://maps.google.com/?q=		
[illegible] ListPicker1.Selection.		
ActivityStarter1.StartActivity ActivityStarter1		ActivityStarter1הפעל את מפות.

איך הבלוקים עובדים

כאשר המשתמש בוחר מתוך ,ListPickerהפריט הנבחר מאוחסן ב
ListPicker.Selectionואירוע AfterPickingמופעל. כפי שנראה ב
איור ,6-4המאפיין DataUriמוגדר לאובייקט טקסט המשלב "http://
"maps.google.com/?qעם הפריט הנבחר. לכן, אם המשתמש בחר בפריט הראשון, "סיור
,"Eifel"ה- DataUriיוגדר "http://maps.google.com/?q= Tour Eifel."-ל



איור .6-5הגדרת ה-IRUataD להפעלת המפה שנבחרה

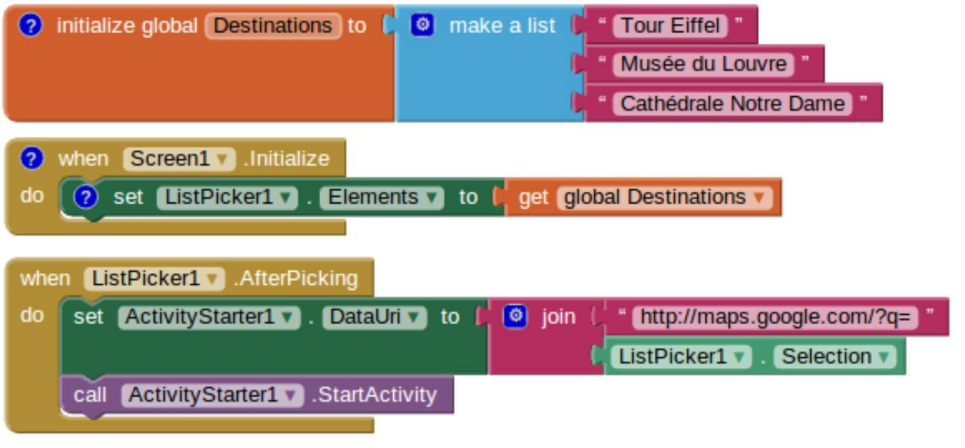
כי אתה כבר מגדיר את המאפיינים האחרים של ActivityStarterכך שהוא
יודע לפתוח את מפות, בלוק ActivityStarter1.StartActivityמפעיל את המפות
האפליקציה ומפעילה את החיפוש שנקבע על ידי DataUri.

Machine Translated by Google

בדוק את האפליקציה שלך הפעל מחדש את האפליקציה ולחץ שוב על הלחצן "בחר יעד בפריז". כאשר אתה בוחר באחד מהיעדים, האם מופיעה מפה של אותו יעד? האם תוכל לחזור לאפליקציה שלך עם כפתור החזרה של המכשיר?

האפליקציה השלמה: סיור במפה עם מתחיל פעילות

איור 6-5מציג את תצורת הבלוק הסופי עבור גרסה 1של סיור המפה של פריז.



איור .6-6אפליקציית סיור המפה המלאה (גרסה 1)

סיור וירטואלי עם הצופה באינטרנט

ActivityStarterהוא רכיב חשוב מכיוון שהוא מספק גישה לכל אפליקציה אחרת במכשיר. אבל, יש דרך אחרת לבנות מדריך טיולים שמשתמש ברכיב אחר, במקום זאת; ה- WebViewer .WebViewerהוא חלונית שאתה מציב ישירות בתוך האפליקציה שלך שמתנהג כמו דפדפן. אתה יכול לפתוח כל דף אינטרנט, כולל מפת גוגל, במציג, ותוכל לשנות באופן תכנותי את הדף שיופיע. בניגוד ,ActivityStarter-להמשתמש שלך אף פעם לא עוזב את האפליקציה שלך, כך שאתה לא צריך לסמוך על כך שהוא ילחץ על כפתור החזרה כדי לחזור.

בגרסה השנייה הזו של האפליקציה, תשתמש ב- WebViewerוגם תבלבל את האפליקציה כך שהיא פותחת כמה תצוגות מוגברת ותצוגות רחוב של המונומנטים של פריז.
אתה תגדיר רשימה שנייה ותשתמש בסכימה מסובכת יותר כדי להחליט איזו מפה להציג. כדי להתחיל, תחילה תחקור את מפות Googleכדי לקבל את כתובות האתרים של כמה מפות ספציפיות. אתה עדיין תשתמש באותם ציוני דרך פריזאיים עבור היעדים, אך כאשר המשתמש בוחר אחד, תשתמש באינדקס (המיקום ברשימה) לבחירתו כדי לבחור ולפתוח מפה ספציפית עם זום או תצוגת רחוב. .

Machine Translated by Google

לפני שתמשיך, אולי תרצה לשמור את הפרויקט שלך (באמצעות שמירה בשם) כדי שיהיה לך עותק של סיור המפה של ActivityStarter שיצרת עד כה. בדרך זו, אם אתה עושה משהו שגורם לבעיות באפליקציה שלך, אתה תמיד יכול לחזור לגרסה העובדת הזו ולנסות שוב.

הוסף את ה-WebViewer

במעצב, מחק את הרכיב ActivityStarter . לאחר מכן, מהמגירה של ממשק המשתמש, גרור פנימה רכיב WebViewer והצב אותו מתחת לרכיבים האחרים.
בטל את הסימון של המאפיין Screen1.Scrollable כדי שה- WebViewer יציג דפים בצורה נכונה.

מציאת כתובת האתר למפות ספציפיות

השלב הבא הוא פתיחת מפות Google במחשב שלך כדי למצוא את המפות הספציפיות שברצונך להפעיל עבור כל יעד:

1. במחשב שלך, דפדף אל http://maps.google.com.
2. חפשו ציון דרך (למשל, מגדל אייפל).
3. התקרב לרמה הרצויה.
4. בחר את סוג התצוגה הרצויה (למשל, Street View).
5. תפוס את כתובת האתר. בגרסה הקלאסית של מפות, אתה לוחץ על כפתור הקישור ליד בפינה השמאלית העליונה של חלון המפות והעתק את כתובת האתר של המפה.
בגרסה החדשה יותר של מפות גוגל אתה יכול פשוט לתפוס את כתובת האתר מסרגל הכתובות.

השתמש בתוכנית זו כדי ליצור כמה מפות מגניבות של המונומנטים של פריז ולחלץ את כתובות אתרים. טבלה 6-6 מספקת כמה דוגמאות אם אתה מעדיף להשתמש בהן (כתובות האתר קוצרו עם השירות bit.ly).

טבלה 6-6. כתובות אתרים של סיורים וירטואליים עבור מפות Google

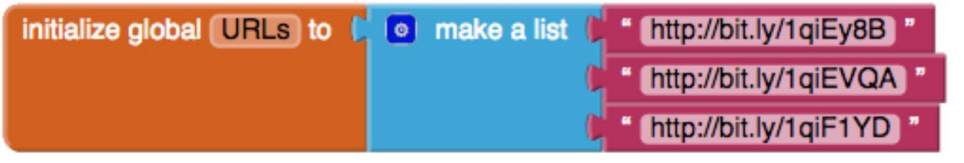
ציון דרך	כתובת האתר של מפות
סיור באייפל	bit.ly/1qiEVQA
מוזיאון הלובר	http://1qiEy8B
קתדרלת נוטרדאם (תצוגת רחוב) http://bit.ly/1qiF1YD	http://bit.ly/

כדי להציג כל אחת מהמפות הללו בדפדפן, הדבק את כתובות האתרים מטבלה 6-6 בשורת הכתובות.

Machine Translated by Google

הגדרת רשימת כתובות האתרים

תזדקק לרשימה בשם כתובות אתרים, המכילה כתובת URL עבור כל אחד מהיעדים. לִיצוֹר
רשימה זו כפי שמוצג באיור 6-6 כך שהפריטים מתאימים לפריטים ב-
רשימת היעדים (כלומר, כתובת האתר הראשונה צריכה להתאים ליעד הראשון, האייפל
מִגדָל).



איור 6-7. העתק והדבק את כתובות האתרים לתוך בלוקי הטקסט של רשימת כתובות האתרים

שינוי בורר הרשימה.התנהגות בחירה לאחר

בגרסה הראשונה של אפליקציה זו, התנהגות ListPicker.AfterPicking מגדירה את DataUri
לשילוב של "http://maps.google.com/?q=" והיעד של המשתמש
בחר מתוך הרשימה (למשל, "טור אייפל"). בגרסה השנייה הזו, ה- AfterPicking
ההתנהגות חייבת להיות מתוחכמת יותר, מכיוון שהמשתמש בוחר מתוך רשימה אחת
(יעדים), אבל האפליקציה בוחרת מרשימת כתובות האתרים עבור כתובת האתר. ספציפית,
כאשר המשתמש בוחר פריט מתוך ListPicker, אתה צריך לדעת את האינדקס של
הבחירה כדי שתוכל להשתמש בה כדי לבחור את כתובת האתר הנכונה מהרשימה. נסביר יותר
על מה זה אינדקס בעוד רגע, אבל זה עוזר להגדיר את הבלוקים תחילה לטובים יותר
להמחיש את הרעיון. יש לא מעט בלוקים הנדרשים לפונקציונליות הזו, כולם
המפורטים בטבלה 6-7.

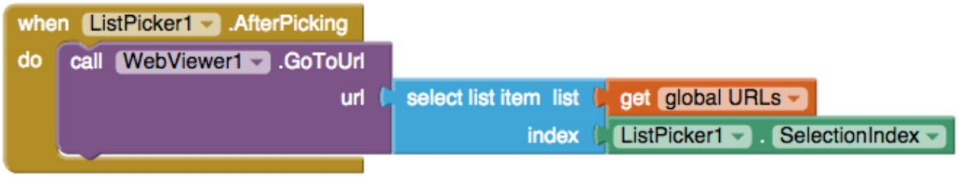
טבלה 6-7. חסימות לבחירת פריט רשימה על סמך בחירת המשתמש

מַטָרָה	סוג בלוק	שם בלוק
אירוע זה מופעל כאשר המשתמש בוחר פריט.	ListPicker1	ListPicker1.AfterPicking
האינדקס (מיקום) של הפריט הנבחר.	ListPicker1	ListPicker1.SelectionIndex
בחר פריט מרשימת כתובות האתרים.	בחר פריט רשימה	
רשימת כתובות האתרים.	גרור אותו מהמשתנה אִתחוּל	לקבל כתובות URL גלובליות
טען את כתובת האתר במציג כדי להציג את המפה.	WebViewer	WebViewer.GoToURL

Machine Translated by Google

איך הבלוקים עובדים

כאשר המשתמש בוחר פריט מתוך ListPicker, אירוע AfterPicking מופעל, כפי שמוצג באיור 6-7. הפריט הנבחר - למשל, "טור אייפל" - נמצא ב- ListPicker.Selection. השתמשת בנכס זה בגרסה הראשונה של אפליקציה זו. עם זאת, ל-rekciPtsiL יש גם מאפיין SelectionIndex, המתאים למיקום היעד הנבחר ברשימה. אז אם נבחר "סיור אייפל", אינדקס הבחירה יהיה 1; אם "מוזיאון הלובר" ייבחר, זה יהיה 2; ואם "קתדרלת נוטרדאם דה פריז" תיבחר, היא תהיה 3.



איור 6-8. פתח את כתובת האתר שנבחרה ב-reweiVbeW

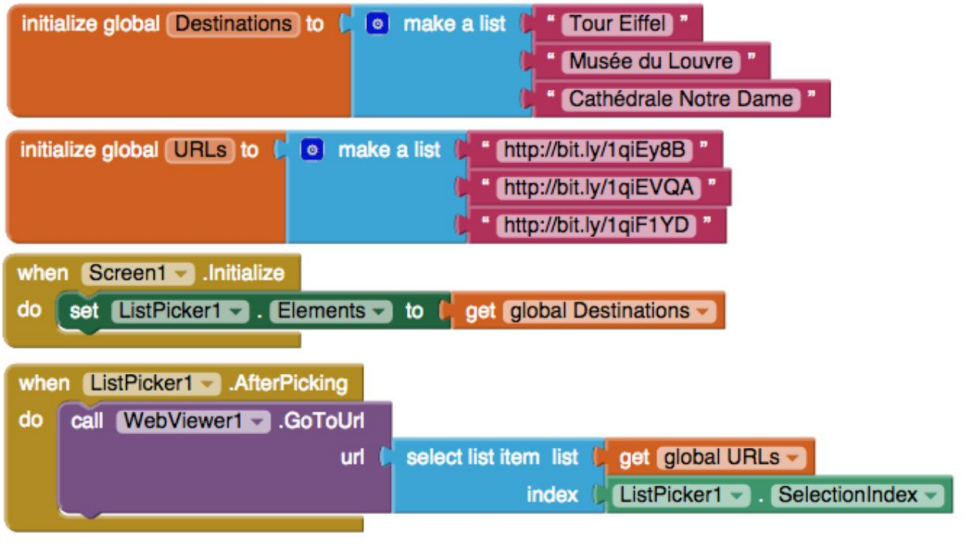
אתה משתמש ב- ListPicker.SelectionIndex כדי לבחור פריט מרשימת כתובות האתרים . זה עובד מכיוון שהפריטים בשתי הרשימות, יעדים וכתובות URL, מסונכרנים: היעד הראשון מתאים לכתובת ה-LRU הראשונה, השני לשני והשלישי לשלישי. לכן, למרות שהמשתמש בוחר פריט מרשימה אחת, אתה יכול להשתמש בבחירה שלו (טוב, האינדקס לבחירתו) כדי לבחור את כתובת האתר הנכונה להצגה.

בדוק את האפליקציה שלך במכשיר, לחץ על הכפתור שכותרתו "בחר יעד בפריז". הרשימה אמורה להופיע עם שלושת הפריטים. בחר אחד מהפריטים וראה איזו מפה מופיעה.

האפליקציה השלמה: סיור במפה (מציג אינטרנט)

איור 6-8 מציג את תצורת הבלוק הסופי עבור גרסה שנייה זו של סיור המפה של פריז.

Machine Translated by Google



איור 6-9. אפליקציית סיור המפה המלאה (גרסת WebViewer)

וריאציות

הנה כמה וריאציות מוצעות לנסות:

- צור סיור וירטואלי במקום העבודה או בית הספר שלך, או לחופשה הבאה שלך יַעַד.
- חקור את ActivityStarter והשתמש בו כדי לשלוח אימייל או להפעיל אפליקציה כגון YouTube (ראה http://bit.ly/1qiFx8Z לעזרה).
- קשה: צור אפליקציית סיור וירטואלי הניתנת להתאמה אישית המאפשרת למשתמש ליצור מדריך למיקום לפי בחירתה על ידי הזנת השם של כל יעד יחד עם כתובת האתר של המפה המתאימה. תצטרך לאחסן את הנתונים במסד נתונים של TinyWebDB וליצור אפליקציית סיור וירטואלי שעובדת עם הנתונים שהוזנו. לדוגמא כיצד ליצור מסד נתונים של TinyWebDB, ראה MakeQuiz/TakeQuiz אפליקציה.

סיכום

הנה כמה מהרעיונות שכיסינו בפרק זה:

- אתה יכול להשתמש במשתני רשימה כדי להחזיק נתונים כגון יעדי מפה וכתובות URL.
- רכיב ListPicker מאפשר למשתמש לבחור מתוך רשימה של פריטים. ה מאפיין האלמנטים של ListPicker מחזיק את הרשימה, המאפיין Selection מחזיק את

Machine Translated by Google

הפריט שנבחר, SelectionIndexמחזיק את המיקום של הפריט שנבחר, ואירוע AfterPickingמופעל כאשר המשתמש בוחר פריט מהרשימה.

•רכיב ActivityStarterמאפשר לאפליקציה שלך להפעיל אפליקציות אחרות. פרק זה הדגים את השימוש שלו עם אפליקציית Google Maps, אבל אתה יכול להפעיל גם דפדפן או כל אפליקציית אנדרואיד אחרת, אפילו עוד אחת שיצרת בעצמך.

•אתה יכול להשתמש ב- ListPicker.SelectionIndexכדי לקבל את המיקום של פריט שא המשתמש בוחר מתוך רשימה. לאחר מכן תוכל להשתמש באינדקס הזה כדי לבחור מידע מרשימה אחרת (שהפריטים שלה מסונכרנים עם הרשימה הראשונה). למידע נוסף על משתני רשימה ורכיב , ListPickerראה פרק 19.

Machine Translated by Google

Machine Translated by Google

אנדרואיד, איפה המכונית שלי?

חניתם הכי קרוב לאצטדיון שאפשר, אבל כשהקונצרט מסתיים, אין לכם מושג איפה המכונית שלכם. החברים שלך חסרי מושג באותה מידה. למרבה המזל, לא איבדת את טלפון האנדרואיד שלך, שלעולם לא שוכח כלום, ואתה זוכר שיש לך את האפליקציה החמה החדשה, אנדרואיד, איפה המכונית שלי? עם האפליקציה הזו, אתה לוחץ על כפתור כשאתה מחנה את המכונית שלך, והאנדרואיד משתמש בחיישן המיקום שלו כדי לתעד את קואורדינטות ה-SPG של המכונית והכתובת. מאוחר יותר, כאשר אתה פותח מחדש את האפליקציה, היא נותנת לך הנחיות מהמקום בו אתה נמצא כעת למיקום השמור -הבעיה נפתרה!

מה תלמד

אפליקציה זו מכסה את המושגים הבאים:

•קביעת המיקום של מכשיר האנדרואיד באמצעות חיישן המיקום
רְכִיב.

•הקלטת נתונים מתמשכת במסד נתונים ישירות על המכשיר באמצעות .TinyDB

•שימוש ברכיב WebViewerכדי לפתוח את מפות Googleמהאפליקציה שלך ולהראות
כיוונים ממקום אחד לאחר.

Machine Translated by Google

מתחילים

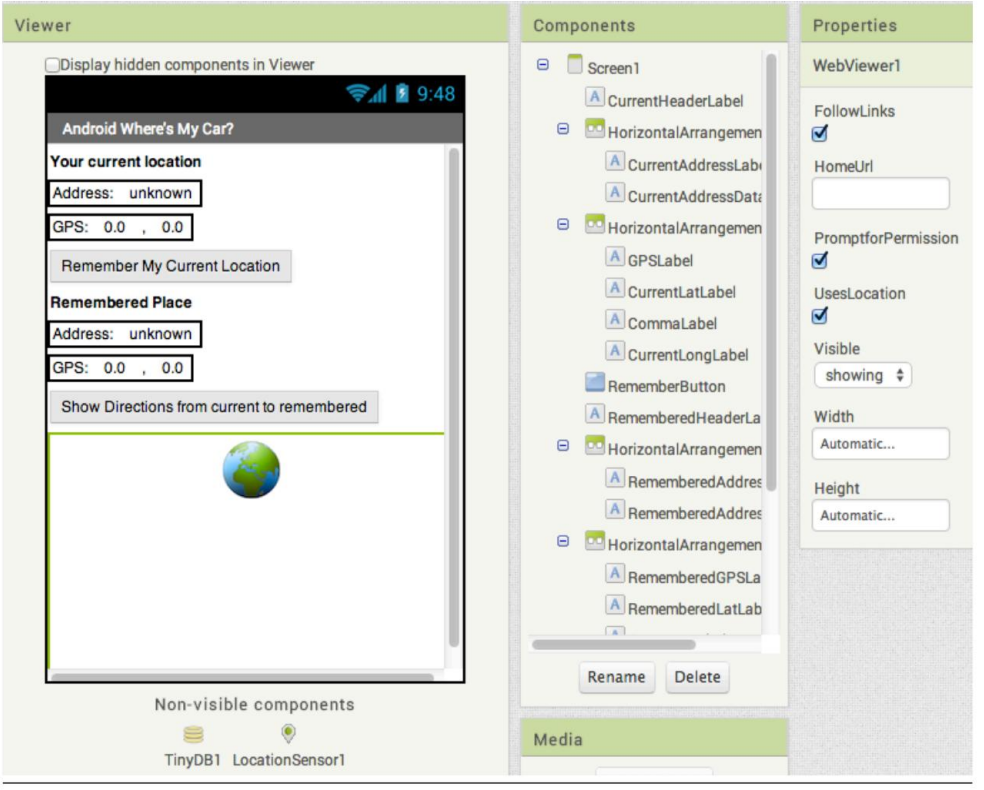
התחבר לאתר App Inventorוהתחל פרויקט חדש. מכיוון שלשמות פרויקטים לא יכולים להיות רווחים, תן לזה "AndroidWhere". הגדר את כותרת המסך ל"אנדרואיד, איפה המכונית שלי?" חבר את המכשיר או האמולטור שלך לבדיקה חיה.

עיצוב הרכיבים

ממשק המשתמש לאנדרואיד, Where's My Car?מורכב מתוויות כדי להציג את המיקומים הנוכחיים והזכורים שלך, וכפתורים כדי להקליט מיקום ולהראות הנחיות אליו. תזדקק לכמה תוויות שרק מציגות טקסט סטטי; לדוגמה, GPSLabelיספק את הטקסט "GPS:" המופיע בממשק המשתמש. תוויות אחרות, כגון CurrentLatLabel,יציגו נתונים מחיישן המיקום. עבור תוויות אלה, תספק ערך ברירת מחדל, (0,0)שישתנה ככל שה-SPG יקבל מידע על מיקום.

תזדקק גם לשלושה רכיבים שאינם גלויים: חיישן מיקום להשגת ה
מיקום נוכחי, TinyDBלאחסון מיקומים מתמשך ו- WebViewerלהצגת הנחיות מפות Googleבין המיקומים הנוכחיים והמאוחסנים.
אתה יכול לבנות את הרכיבים מתמונת המצב של מעצב הרכיבים באיור 7-1.

Machine Translated by Google



איור 7-1. האנדרואיד, איפה המכונית שלי? אפליקציה Component Designer-ב

תזדקק לרכיבים בטבלה 7-1.

טבלה 7-1. כל הרכיבים לאפליקציה

מַטָרָה	איך תקרא לזה	לוח הַצְבָעִים קְבוּצָה	סוג רכיב
הצג את הכותרת "שלך מיקום נוכחי".	ממשק משתמש CurrentHeaderLabel		תווית
סדר את פרטי הכתובת.	פריסה אנכית1 אופקי		
הצג את הטקסט "כתובת:".	ממשק משתמש CurrentAddressLabel		
הצגת נתונים דינמיים: ה כתובת נוכחית.	ממשק משתמש CurrentAddressDataLabel		
סדר את פרטי ה-GPS.	פריסה אנכית2 אופקי		
הצג את הטקסט ".GPS:"	ממשק משתמש GPSLabel		
הצגת נתונים דינמיים: ה קו הרוחב הנוכחי.	ממשק משתמש CurrentLatLabel		
הצג ".,"	ממשק משתמש CommaLabel		

Machine Translated by Google

מַטָרָה	איך תקרא לזה	לוּחַ הַצְבָעִים קְבוּעָה	סוג רכיב
הצגת נתונים דינמיים: ה קו אורך נוכחי.	ממשק משתמש CurrentLongLabel		תווית
לחץ כדי להקליט את הנוכחי מקום.	ממשק משתמש RememberButton		לַחְצָן
לסדר נזכר פרטי כתובת.	ממשק משתמש סידור אופקי2		תווית
הצג את הטקסט "מקום זכור".	ממשק משתמש RememberedAddressLabel		תווית
הצגת נתונים דינמיים: ה כתובת זכורה.	ממשק משתמש RememberedAddressDataLabel		תווית
הצג את הטקסט "GPS".	ממשק משתמש RememberedGPSLabel		תווית
הצגת נתונים דינמיים: ה זכר את קו הרוחב.	ממשק משתמש RememberedLatLabel		תווית
הצג ",".	ממשק משתמש Comma2Label		תווית
הצגת נתונים דינמיים: ה זכר קו אורך.	ממשק משתמש RememberedLongLabel		תווית
לחץ כדי להציג את המפה.	כפתור הוראות ממשק משתמש		לַחְצָן
חוש מידע GPS.	חיישן מיקום 1		חיישן מיקום
אחסן את הזכור מיקום בהתמדה.	TinyDB1		TinyDB
הצג הנחיות.	ממשק משתמש WebViewer1		WebViewer

הגדר את המאפיינים של הרכיבים בצורה הבאה:

- הגדר את המאפיין Textעבור התוויות עם טקסט קבוע כמפורט בטבלה 7-1.
- הגדר את מאפיין הטקסט של התוויות עבור נתוני GPSדינמיים ל-"0.0".
- הגדר את המאפיין Textשל התוויות עבור כתובות דינמיות ל"לא ידוע".
- בטל את הסימון של המאפיין Enabledשל ה- RememberButtonושל DirectionsButton.
- בטל את הסימון של המאפיין Screen.Scrollableכך שה- WebViewerיתאים ל- מֶסֶך.

הוספת התנהגויות לרכיבים

תזדקק להתנהגויות הבאות עבור אפליקציה זו:

Machine Translated by Google

•כאשר חיישן המיקום מקבל קריאה, הכנס את נתוני המיקום הנוכחיים לתוך
התוויות המתאימות של ממשק המשתמש. זה יאפשר למשתמש לדעת את
החיישן קרא מיקום והוא מוכן לזכור אותו.

•כאשר המשתמש לוחץ על ה- ,RememberButtonהעתק את נתוני המיקום הנוכחיים אל
התוויות עבור המיקום הזכור. תצטרך גם לאחסן את
זכור נתוני מיקום כך שהם יהיו שם אם המשתמש יסגר ו
משיקה מחדש את האפליקציה.

•כאשר המשתמש לוחץ על כפתור ההוראות, הפעל את מפות Googleב-
WebViewerכך שיציג הנחיות למיקום הזכור.

•כאשר האפליקציה מופעלת מחדש, טען את המיקום הזכור ממסד הנתונים
לתוך האפליקציה.

הצגת המיקום הנוכחי

האירוע LocationSensor.LocationChangedמתרחש לא רק כשהמכשיר
שינויים במיקום, אך גם כאשר החיישן מקבל קריאה ראשונה. לפעמים זה קודם
הקריאה תימשך כמה שניות, ולפעמים לא תקבל קריאה כלל אם
קווי הראייה ללווייני GPSחסומים (ובהתאם להגדרות המכשיר). ל
מידע נוסף על GPSו- ,LocationSensorראה פרק 23.
כאשר אתה מקבל קריאת מיקום, האפליקציה צריכה למקם את הנתונים ב-
תוויות מתאימות. טבלה 7-2מפרטת את כל הבלוקים שתצטרך לעשות זאת.

טבלה 7-2.חסימות לקבלת קריאת מיקום והצגתה בממשק המשתמש של האפליקציה

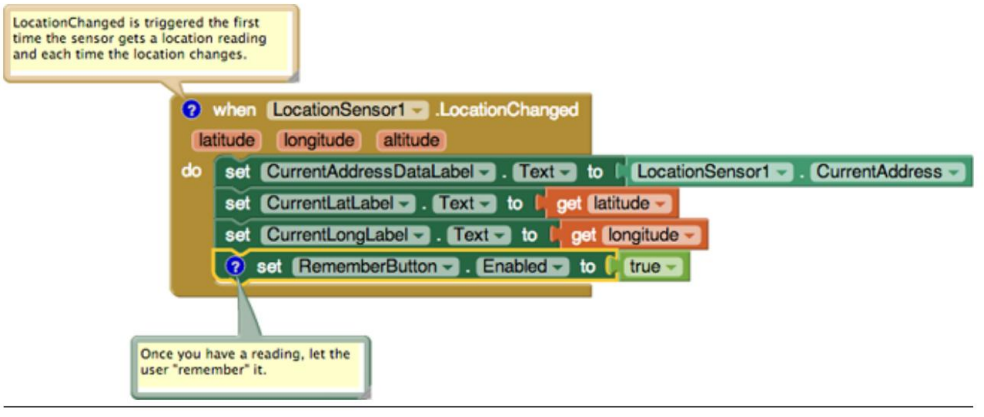
מַטָרָה	מְגֻרָה	סוג בלוק
זה המטפל באירועים שמופעל כאשר הטלפון מקבל GPSחדש קריאה.	LocationSensor1	LocationSensor1.LocationChanged
הצב את הנתונים החדשים בתווית של כתובת נוכחית.	CurrentAddressDataLabel	הגדר CurrentAddressDataLabel.Text ל
נכס זה נותן לך כתובת רחוב.	LocationSensor1	LocationSensor1.CurrentAddress
מקם את קו הרוחב לתוך המתאים תווית.	CurrentLatLabel	הגדר את CurrentLatLabel.Text ל
חבר לסט .CurrentLatLabel.Text to	גרור החוצה אירוע השתנה מיקום	לקבל קו רוחב
מניחים את קו האורך לתוך המתאים תווית.	CurrentLongLabel	הגדר CurrentLongLabel.Text ל
חבר לתוך הגדר .to CurrentLongLabel.Text	גרור החוצה אירוע השתנה מיקום	ערך קו אורך

Machine Translated by Google

מַטָרָה	מְגִרָה	סוג בלוק
זכור את הקריאה עבור הנוכחי מקום.	RememberButton	הגדר את RememberButton.Enabledל
חבר לסט RememberButton.Enabled ל.	הִגָיוֹן	נָכוֹן

איך הבלוקים עובדים

איור 2-7ממחיש שקו הרוחב והאורך הם פרמטרים של
אירוע השתנה מיקום . אתה יכול לתפוס הפניות לפרמטרים של אירועים באמצעות עכבר
עליהם. CurrentAddressאינו ארגומנט; אלא, זה נכס של ה
LocationSensor, אז אתה תופס אותו מהמגירה של . LocationSensorחיישן המיקום
עושה עבורך עבודה נוספת על ידי התקשרות למפות Googleכדי לקבל כתובת רחוב
מתאים למיקום ה-SPG.
מטפל באירועים זה מאפשר גם את ה- .RememberButtonאתחלנו אותו כנכה
(לא מסומן) Component Designer-בכי אין מה למשתמש לעשות
זכור עד שהחיישן יקבל קריאה, אז עכשיו נתכנת את ההתנהגות הזו.



איור .2-7שימוש LocationSensor-בכדי לקרוא את המיקום הנוכחי

בדוק את האפליקציה שלך אתה כנראה רוצה להסתובב כדי לבדוק זאת
אפליקציה. אז תצטרך לבנות את האפליקציה ולהתקין אותה אצלך
טלפון על ידי בחירה App -> Build-ב(ספק קוד QRעבור .kpa).
כאשר אתה מפעיל את האפליקציה, אתה אמור לראות כמה נתוני GPSמופיעים
והמאופשרים. אם לא תצליח לקרוא, RememberButton
בדוק את הגדרות האנדרואיד שלך עבור מיקום ואבטחה ונסה
יוצאים החוצה. למידע נוסף, ראה פרק .23

Machine Translated by Google

הקלטת המיקום הנוכחי

כאשר המשתמש לוחץ על ה- ,RememberButtonנתוני המיקום העדכניים ביותר צריכים להיות להציב בתוויות להצגת הנתונים הזכורים. טבלה 7-3מראה לך איזה בלוקים שתזדקק לפונקציונליות הזו.

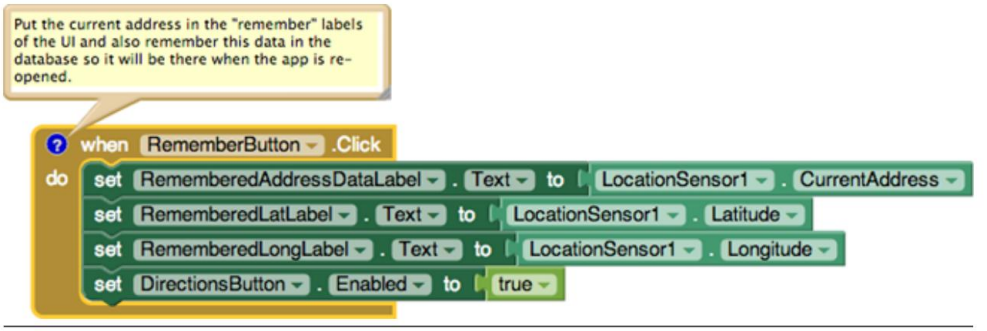
טבלה .7-3בלוקים להקלטה והצגת המיקום הנוכחי

מַטָרָה	מְגֵרָה	סוג בלוק
מופעל כאשר המשתמש לוחץ "זכור."	RememberButton	RememberButton.Click
הכנס את נתוני הכתובת של החיישן לתוך התווית של הכתובת הזכורה.	RememberedAddressDataLabel	הגדר RememberedAddressDataLabel.Text ל
נכס זה נותן לך רחוב כתובת.	LocationSensor1	LocationSensor1.CurrentAddress
מקם את קו הרוחב הנחוש לתוך תווית "זכור".	RememberedLatLabel	הגדר RememberedLatLabel.Text ל
חבר לסט RememberedLatLabel.Text ל.	חיישן מיקום 1	LocationSensor.Latitude
הנח את קו האורך הנחוש לתוך תווית "זכור".	הגדר את RememberedLongLabel	RememberedLongLabel.Text ל
חבר לסט .to RememberedLongLabel.Text	חיישן מיקום 1	חיישן מיקום. קו אורך
מפה את המקום הזכור.	הגדר את DirectionsButton	DirectionsButton.Enabled ל-
חבר ל-tes DirectionsButton.Enabled ל.	הִגָיוֹן	נָכוֹן

איך הבלוקים עובדים

כאשר המשתמש לוחץ על ה- ,RememberButtonהקריאות הנוכחיות עבור חיישן המיקום מוכנסים לתוך התוויות "נזכרות", כפי שמוצג באיור 7-3.

Machine Translated by Google



איור 7-3. הצבת מידע המיקום הנוכחי בתוויות "נזכרות".

תבחין גם שכפתור הכיוון מופעל. זה יכול להיות מסובך,
כי אם המשתמש ילחץ על כפתור ה- Directionsמיד, המיקום הזכור יהיה זהה למיקום הנוכחי,
כך שהמפה שתופיע לא תספק הרבה מבחינת כיוונים. אבל זה לא משהו שמישהו עשוי לעשות;
לאחר שהמשתמש יזוז (למשל, הולך לקונצרט), המיקום הנוכחי והמיקום הזכור יתבדו.

בדוק את האפליקציה שלך הורד את הגרסה החדשה של האפליקציה
RememberButton. לחץ על האם הנתונים מההגדרות

הצגת הנחיות למיקום הזכור

כאשר המשתמש לוחץ על כפתור המסלולים, אתה רוצה שהאפליקציה תפתח את מפות
Googleולאחר מכן תציג את ההנחיות מהמיקום הנוכחי של המשתמש למיקום הזכור (למשל,
היכן המכונית חונה).
רכיב WebViewerיכול להציג כל דף אינטרנט, כולל מפות .Google
אתה תתקשר ל- WebViewer.GoToURLכדי לפתוח את המפה, אבל אתה רוצה לפתוח כתובת אתר שתעשה זאת
הצג הנחיות מהמיקום הנוכחי למיקום הזכור.
אחת הדרכים להציג מסלול במפות היא באמצעות כתובת אתר בצורה הבאה: ?maps
http://maps.google.com/
saddr=37.82557,-122.47898&daddr=37.81079,-122.47710הקלד את כתובת האתר
בדפדפן -האם תוכל לדעת איזה נקודת ציון מפורסמת היא מפנה אותך
ברחבי?
עבור אפליקציה זו, עליך לבנות את כתובת האתר ולהגדיר את כתובת המקור שלה (saddr)ו
כתובת יעד (daddr)פרמטרים באופן דינמי (בבלוקים). צירפת טקסט בעבר בפרקים קודמים
באמצעות ;joinאנחנו נעשה את זה גם כאן, תוך חיבור לחשמל

Machine Translated by Google

נתוני ה-SPG עבור המיקומים הזכורים והנוכחיים. אתה תשים את כתובת האתר שאתה בונה לתוך חריץ הפרמטרים של WebViewer.GotToURL. טבלה 7-4מפרטת את כל הבלוקים שתזדקק להם לזה.

טבלה 7-4. בלוקים להקלטה והצגת המיקום הנוכחי

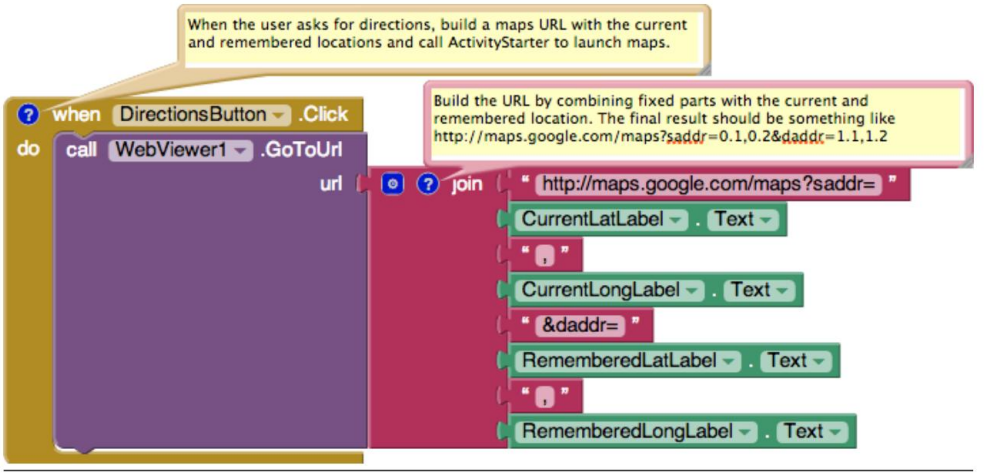
מַטָרָה	מְגֻרָה	סוג בלוק
מופעל כאשר המשתמש לוחץ על "הנחיות".	כפתור הוראות	כפתור הוראות. לחץ
הגדר את כתובת האתר של המפה שברצונך להביא למעלה.	WebViewer1	WebViewer1.GoToURL
בנה כתובת URL ממספר חלקים.	טֶקסט	לְהִצְטָרֵף
החלק המקובע של כתובת האתר, כתובת המקור.	טֶקסט	טקסט ("http://maps.google.com/maps? saddr=")
קו הרוחב הנוכחי.	CurrentLatLabel	CurrentLatLabel.Text
שים פסיק בין קו הרוחב ו ערכי קו אורך.	טֶקסט	טקסט (",")
קו האורך הנוכחי.	CurrentLongLabel	CurrentLongLabel.Text
הפרמטר השני של כתובת האתר, ה- כתובת יעד.	טֶקסט	טקסט ("&daddr=")
RememberedLatLabel	קו הרוחב הזכור.	RememberedLatLabel.Text
שים פסיק בין הערכים עבור קו הרוחב וקווי אורך.	טֶקסט	טקסט (",")
RememberedLongLabel	קו האורך הזכור.	RememberedLongLabel.Text

איך הבלוקים עובדים

כאשר המשתמש לוחץ על כפתור ה- Directions, המטפל באירועים בונה כתובת URL עבור מפה וקורא ל- WebViewer.GoToURL כדי לפתוח את המפה, כפי שמוצג באיור 7-4. הצטרף רגיל לבנות את כתובת האתר של המפה.

כתובת האתר המתקבלת מורכבת מהדומיין של מפות Google (http://maps.google.com/maps) יחד עם שני פרמטרים של כתובת URL, saddr ו- daddr, המציינים את המקור ואת מיקומי יעד עבור ההנחיות. עבור אפליקציה זו, ה- saddr מוגדר לקו הרוחב וקו האורך של המיקום הנוכחי, וה- daddr מוגדר לקו הרוחב ו קו האורך של המיקום המאוחסן עבור המכונית.

Machine Translated by Google



איור 7-4. בניית כתובת האתר לשימוש לפתיחת המפה ב-reweiVbeW

בדוק את האפליקציה שלך הורד את הגרסה החדשה של האפליקציה לטלפון שלך ובדוק שוב. כשנכנסת קריאה, לחץ על ואז צאו לטייל. כאשר אתה לוחץ על האם המפה מראה לך איך לחזור לרכבתך? לאחר התבוננות במפה, לחץ על כפתור החזרה, האם אתה חוזר לאפליקציה שלך?

RememberButton
כפתור הבית

אחסון המיקום הזכור בהתמדה

כעת יש לך אפליקציה הפועלת במלואה שזוכרת מיקום התחלה ומשרטטת מפה חזרה למיקום זה מכל מקום בו נמצא המיקום הנוכחי של המשתמש. עם זאת, אם המשתמש "זוכר" מיקום ואז סוגר את האפליקציה, הנתונים הזכורים לא יהיו זמינים כאשר האפליקציה תיפתח מחדש. מה שאתה באמת רוצה זה שהמשתמש יוכל לתעד את מיקום המכונית, לסגור את האפליקציה ולעבור לאירוע כלשהו, ואז להפעיל מחדש את האפליקציה מאוחר יותר כדי לקבל הנחיות למיקום המוקלט.

אם אתה כבר חושב על האפליקציה ללא הודעות טקסט בזמן נהיגה (פרק 4), אתה בדרך הנכונה. עליך לאחסן את הנתונים באופן קבוע במסד נתונים באמצעות TinyDB. תשתמש בסכימה דומה לזו שבה השתמשת באפליקציה זו:

1. כאשר המשתמש לוחץ על ה- RememberButton, אחסן את נתוני המיקום ב- מאגר מידע.

2. כאשר האפליקציה מופעלת, טען את נתוני המיקום ממסד הנתונים למשתנה או רכוש.

Machine Translated by Google

תתחיל בכך שתשנה את מטפל האירועים RememberButton.Clickכך שיישמר
הנתונים הזכורים. כדי לאחסן את קו הרוחב, קו האורך והכתובת, תזדקק לשלושה
שיחות ל- .TinyDB.StoreValueטבלה 5-7מפרטת את הבלוקים הנוספים שתזדקק להם.

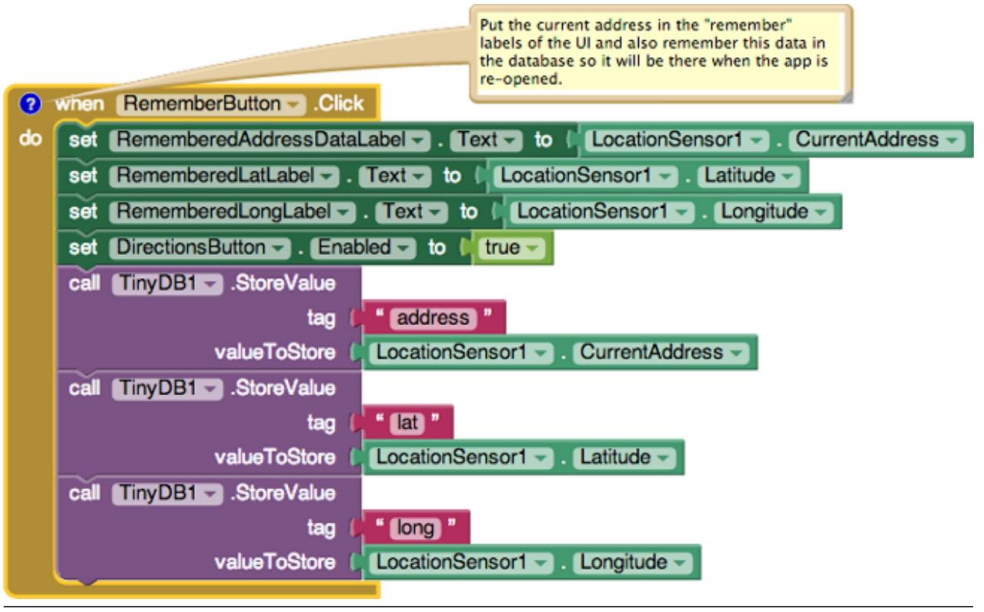
טבלה .5-7בלוקים להקלטה והצגת המיקום הנוכחי

מַטָרָה	מְגֶרָה	סוג בלוק
אחסן את הנתונים במסד הנתונים של המכשיר.	TinyDB	TinyDB1.StoreValue (3)
חבר את זה לשקע "תג" של TinyDB1.StoreValue.	טֶקסט	טקסט ("כתובת")
הכתובת לאחסון בהתמדה; חבר את זה ל שקע "ערך" של .TinyDB1.StoreValue	LocationSensor1	LocationSensor1.CurrentAddress
חבר את זה לשקע "תג" של השני TinyDB1.StoreValue.	טֶקסט	טקסט ("lat")
קו הרוחב לאחסון בהתמדה; חבר את זה ל שקע "ערך" של השני TinyDB1.StoreValue.	LocationSensor1	LocationSensor1.CurrentLatitude
חבר את זה לשקע "תג" של השלישי TinyDB1.StoreValue.	טֶקסט	טקסט ("ארוך")
קו האורך לאחסון בהתמדה; חבר את זה ל שקע "ערך" של השלישי TinyDB1.StoreValue.	LocationSensor1	LocationSensor1.Current Longitude

איך הבלוקים עובדים

כפי שמוצג באיור TinyDB1.StoreValue ,5-7מעתיק את נתוני המיקום מה-
מאפייני LocationSensorלתוך מסד הנתונים. כפי שאתה אולי זוכר No SMS-מ
בזמן נהיגה, לפונקציה StoreValueיש שני ארגומנטים, התג והערך. ה
תג מזהה את הנתונים שברצונך לאחסן, והערך הוא הנתונים האמיתיים שאתה
רוצה לשמור -במקרה זה, נתוני חיישן המיקום .

Machine Translated by Google



איור 7-5. אחסון נתוני המיקום הזכורים במסד נתונים

אחזור המיקום הזכור כאשר האפליקציה מופעלת

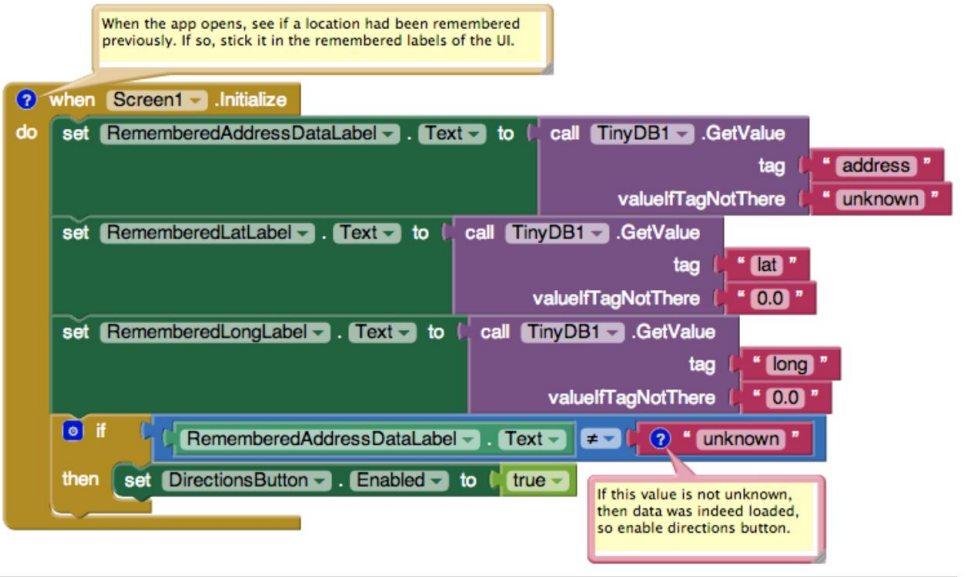
אתה מאחסן נתונים במסד נתונים כדי שתוכל לזכור אותם מאוחר יותר. באפליקציה זו, אם משתמש מאחסן מיקום ולאחר מכן סוגר את האפליקציה, ברצונך לאחזר את המידע הזה ממסד הנתונים ולהראות אותו כאשר המשתמש מפעיל מחדש את האפליקציה.

כפי שנדון בפרקים הקודמים, אירוע Screen.Initialize מופעל כאשר האפליקציה שלך מופעלת. אחזור נתונים ממסד נתונים הוא דבר נפוץ מאוד בהפעלה, וזה בדיוק מה שאנחנו רוצים לעשות עבור האפליקציה הזו.

תשתמש בפונקציית TinyDB.GetValue כדי לאחזר את נתוני ה-SPG המאוחסנים. מכיוון שאתה צריך לאחזר את הכתובת המאוחסנת, קו הרוחב וקו האורך, תצטרך שלוש שיחות ל- GetValue. כמו ללא הודעות טקסט בזמן נהיגה, תצטרך לבדוק אם אכן יש נתונים זמינים (אם זו הפעם הראשונה שאתה מפעיל את האפליקציה שלך, TinyDB.GetValue תחזיר טקסט ריק).

כאתגר, בדוק אם אתה יכול ליצור בלוקים אלה ולאחר מכן השווה את היצירה שלך לבלוקים המוצגים באיור 7-6.

Machine Translated by Google



איור .6-7כאשר האפליקציה מופעלת, טען במיקום הזכור ממסד הנתונים

איך הבלוקים עובדים

כדי להבין את החסימות הללו, דמיינו שני מקרי שימוש: משתמש שיפתח את האפליקציה בפעם הראשונה, והמשתמש יפתח אותה מאוחר יותר, לאחר שהקליט בעבר נתוני מיקום. בפעם הראשונה שהמשתמש פותח את האפליקציה, לא יהיו נתוני מיקום במסד הנתונים לטעינה.
בהשקות עוקבות, אם יש נתונים מאוחסנים, ברצונך לטעון את נתוני המיקום המאוחסנים קודם לכן ממסד הנתונים.

הבלוקים קוראים TinyDB1.GetValue-לשלוש פעמים, פעם אחת עבור כל אחד משדות הנתונים ששמרת בעבר: "כתובת", "lat"ו-"gnol". הפרמטר valueIfTagNotThereמוגדר לערך ברירת מחדל עבור כל אחד מהם, כך שאם עדיין אין נתונים במסד הנתונים, התוויות יוגדרו לערכי ברירת המחדל (זהה לאלו שהוגדרו במעצב).
בלוק ifמשמש כדי לקבוע אם יש להפעיל את כפתור ה- . Directionsזה צריך להיות אם הנתונים אכן אוחזרו ממסד הנתונים. הבדיקה שבה נעשה שימוש היא להשוות את RememberedAddressDataLabelלערך ברירת המחדל שלו, לא ידוע. אם זה לא ידוע, זה חייב להיות מוחלף בכתובת זכורה כלשהי.

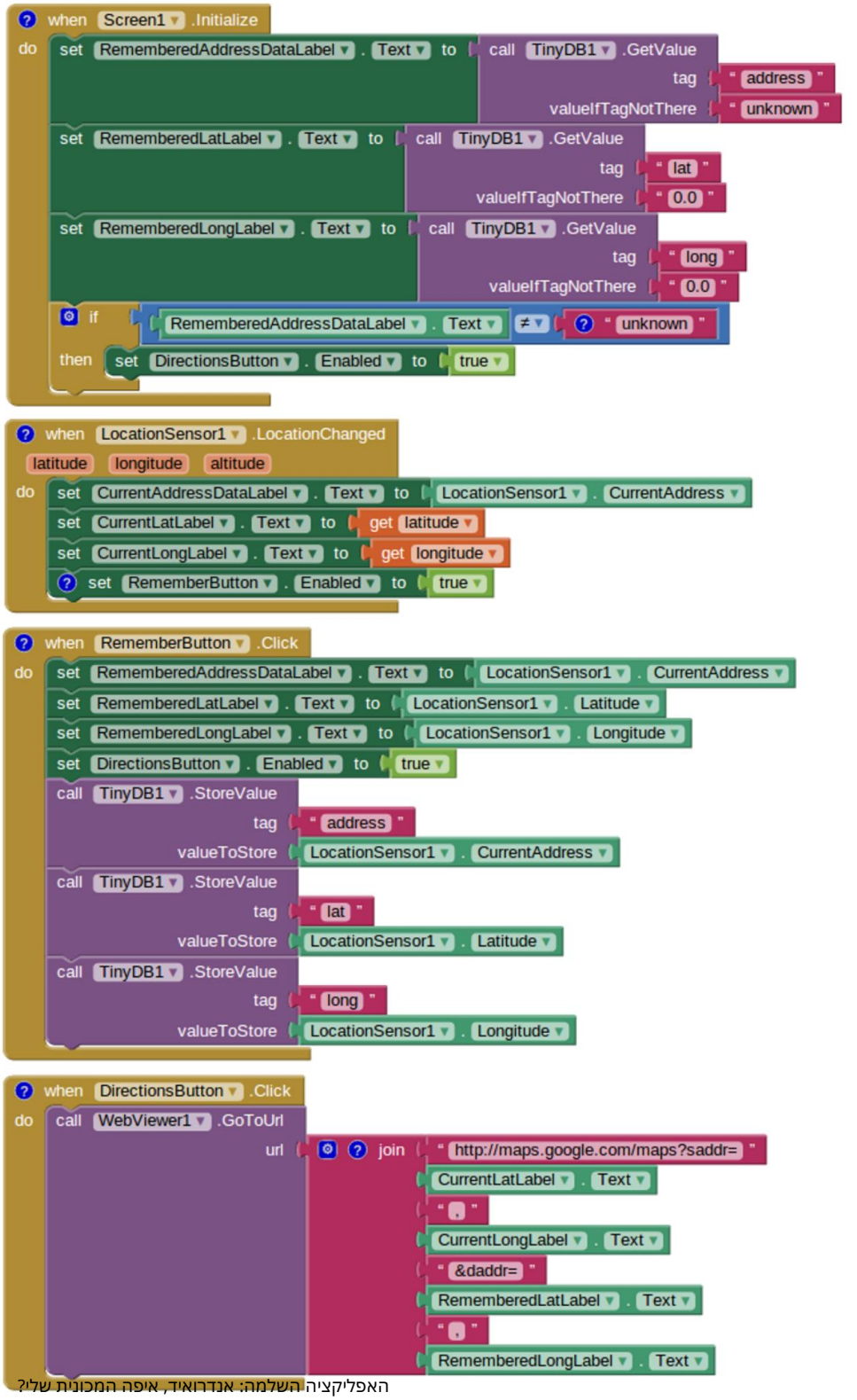
בדוק את האפליקציה שלך הורד את הגרסה החדשה של האפליקציה לטלפון שאפשרה לך לשמור את RememberButton

Machine Translated by Google

האפליקציה השלמה: אנדרואיד, איפה המכונית שלי?

איור 7-7מציג את הבלוקים הסופיים עבור אנדרואיד השלם, Where's My Car?אפליקציה.

Machine Translated by Google



Machine Translated by Google

וריאציות

להלן כמה וריאציות שתוכל להתנסות בהן:

•צור אנדרואיד, איפה כולם?, אפליקציה המאפשרת לקבוצה של אנשים לעקוב אחר מיקומו של זה. בין אם אתם מטיילים ביער או נפרדים בפארק, האפליקציה הזו יכולה לעזור לחסוך זמן ואולי אפילו חיים. הנתונים עבור אפליקציה זו משותפים, כך שתצטרך להשתמש במסד נתונים אינטרנטי וברכיב TinyWebDBבמקום .TinyDBראה פרק 22 למידע נוסף.

•צור אפליקציית Breadcrumbשעוקבת אחר מקום הימצאו שלך על ידי הקלטת כל שינוי במיקום ברשימה. עליך להקליט פירור לחם חדש רק אם המיקום השתנה בכמות מסוימת, או שחלף פרק זמן מסוים, מכיוון שאפילו תנועה קלה יכולה ליצור קריאת מיקום חדשה.
יהיה עליך לאחסן את המיקומים המוקלטים ברשימה. עיין בפרק 19לקבלת עזרה.

סיכום

הנה כמה מהרעיונות שכיסינו במדריך זה:

•הרכיב LocationSensorיכול לדווח על קו הרוחב, קו האורך וכתובת הרחוב הנוכחית של המשתמש. אירוע ה- LocationChangedשלו מופעל כאשר החיישן מקבל את הקריאה הראשונה שלו וכאשר הקריאה משתנה (המכשיר זז). למידע נוסף על חיישן המיקום, ראה פרק 23.

•רכיב ה- WebViewerמציג כל דף אינטרנט, כולל מפות .Googleאם ברצונך להציג הנחיות בין קואורדינטות ,GPSכתובת האתר תהיה בפורמט הבא, אך תחליף את הנתונים לדוגמה המוצגים כאן בקואורדינטות GPSבפועל: http://maps.google.com/maps/?saddr=0.1 ,0.1&daddr=0.2,0.2

•אתה משתמש ב- joinכדי לחבר (לשרשר) פריטי טקסט נפרדים לאובייקט טקסט בודד. אתה יכול להשתמש בו כדי לשרשר נתונים דינמיים עם טקסט סטטי. עם כתובת האתר של מפות, קואורדינטות ה-SPG הן הנתונים הדינמיים.

•באמצעות ,TinyDBאתה יכול לאחסן נתונים באופן קבוע במסד הנתונים של הטלפון. בעוד שהנתונים במשתנה או בנכס אובדים כאשר אפליקציה נסגרת, ניתן לטעון נתונים המאוחסנים במסד הנתונים בכל פעם שהאפליקציה נפתחת. למידע נוסף על TinyDB ומסדי נתונים, ראה פרק 22.

Machine Translated by Google

חידון נשיאים

איור 1-8

חידון הנשיאים הוא משחק טריוויה של מנהיגים לשעבר של ארצות הברית. למרות שהחידון הזה עוסק בנשיאים, אתה יכול להשתמש בו כתבנית לבניית חידונים או מדריכי לימוד בכל נושא.

בפרקים הקודמים הוצגת לך לכמה מושגי תכנות בסיסיים. עכשיו, אתה מוכן למשהו יותר מאתגר. תגלו שהפרק הזה דורש קפיצה מושגית מבחינת מיומנויות תכנות וחשיבה מופשטת. בפרט, תשתמש בשני משתני רשימה כדי לאחסן את הנתונים -במקרה זה, השאלות והתשובות של החידון -ותשתמש במשתנה אינדקס כדי לעקוב אחר היכן נמצא המשתמש בחידון. כשתסיים, תהיה חמוש בידע ליצור אפליקציות חידונים ואפליקציות רבות אחרות הדורשות עיבוד רשימה.

פרק זה מניח שאתה מכיר את היסודות של App Inventor: שימוש ב-
מעצב רכיבים לבניית ממשק המשתמש, ושימוש בעורך הבלוקים כדי לציין מטפלי אירועים ולתכנת את התנהגות הרכיבים. אם אינך מכיר את היסודות הללו, הקפד לעיין בפרקים הקודמים לפני שתמשיך.

אתה תעצב את החידון כך שהמשתמש ימשיך משאלה לשאלה לפי
לחיצה על כפתור הבא ומקבלת משוב על התשובות.

Machine Translated by Google

מה תלמד

אפליקציה זו, המוצגת באיור 8-1, מכסה:

- הגדרת משתני רשימה לאחסון השאלות והתשובות ברשימות.
- רצף באמצעות רשימה באמצעות אינדקס; בכל פעם שהמשתמש לוחץ על הבא, תציג את השאלה הבאה.
- שימוש בהתנהגויות מותנות (אם): ביצוע פעולות מסוימות רק בתנאים ספציפיים. אתה תשתמש בבלוק אם כדי לטפל בהתנהגות האפליקציה כאשר המשתמש מגיע לסוף החידון.

איור 8-2. חידון הנשיאים בפעולה

- החלפת תמונה כדי להציג תמונה שונה עבור כל שאלת חידון.

מתחילים

נווט לאתר App Inventor והתחל פרויקט חדש. תן לפרויקט את השם "PresidentsQuiz" והגדר את כותרת המסך ל"stnediserP Quiz". חבר את המכשיר או האמולטור שלך לבדיקה חיה.

תצטרך להוריד את התמונות הבאות עבור החידון מה-appinventor.org למחשב שלך:

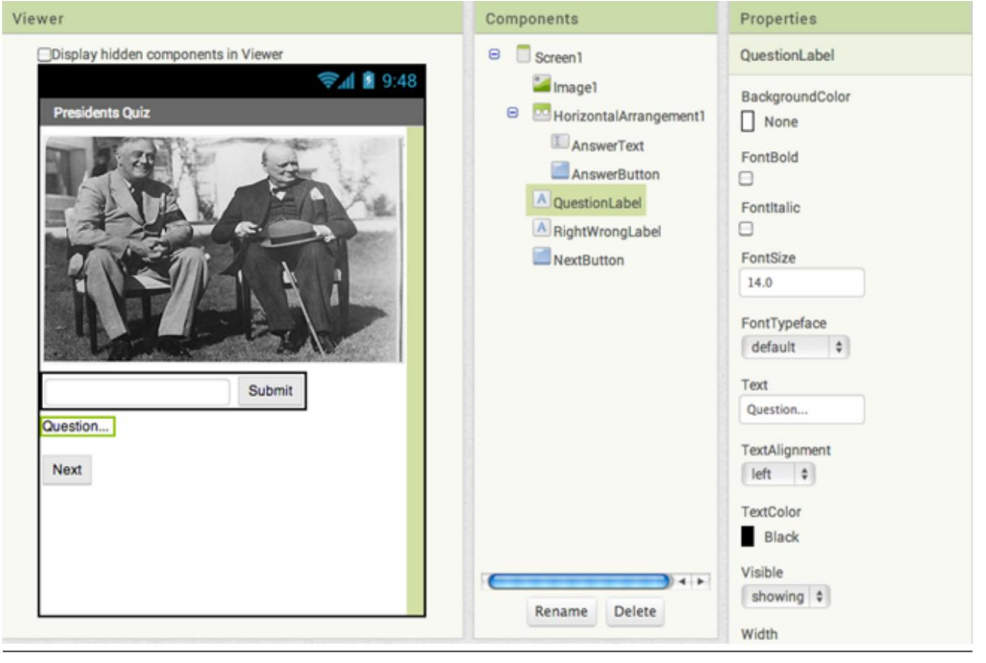
- http://appinventor.org/bookFiles/PresidentsQuiz/roosChurch.gif
- http://appinventor.org/bookFiles/PresidentsQuiz/nixon.gif
- http://appinventor.org/bookFiles/PresidentsQuiz/carterChina.gif
- http://appinventor.org/bookFiles/PresidentsQuiz/atomic.gif

אתה תטען את התמונות האלה לפרויקט שלך בסעיף הבא.

עיצוב הרכיבים

לאפליקציית Presidents Quiz יש ממשק פשוט להצגת השאלות ולאפשר למשתמש לענות. אתה יכול לבנות את הרכיבים מתמונת המצב של מעצב הרכיבים המוצגת באיור 8-2.

Machine Translated by Google



איור .8-3חידון הנשיאים במעצב

כדי ליצור ממשק זה, טען תחילה את התמונות שהורדת לפרויקט. ב
באזור המדיה, לחץ על הוסף ובחר אחד מהקבצים שהורדת (למשל, .(roosChurch.gif
עשה את אותו הדבר עבור שלוש התמונות האחרות. לאחר מכן, הוסף את הרכיבים המפורטים בטבלה .8-1

טבלה .8-1רכיבים לאפליקציית חידון הנשיאים

מַטָרָה	קבוצת צבעים איך תקרא לזה		סוג רכיב
תמונת ממשק משתמש 1			תמונה
			תווית
המשתמש יזין את תשובתו כאן.	ממשק משתמש AnswerText		תיבת טקסט
המשתמש לוחץ על זה כדי לשלוח תשובה.	ממשק משתמש AnswerButton		לַחְצָן
הצג "נכון!" או "לא נכון!"	ממשק משתמש RightWrongLabel		תווית
המשתמש לוחץ על זה כדי להמשיך השאלה הבאה.	ממשק משתמש NextButton		לַחְצָן

Machine Translated by Google

1. הגדר את Image1.Picture לקובץ התמונה הראשונה roosChurch.gif, התמונה שאמורה להופיע. הגדר את הרוחב שלו ל"מלא הורה" ואת הגובה שלו ל-002.

2. הגדר את QuestionLabel.Text ל-"שאלה..." (תזין את השאלה הראשונה ב- עורך בלוקים).

3. הגדר את AnswerText.Hint ל"הזן תשובה". הגדר את מאפיין הטקסט שלו לריק. מהלך \לזוז \לעבור זה לתוך Horizontal Arrangement1.

4. שנה את AnswerButton.Text ל-"timbuS" והעבר אותו אל סידור אופקי 1.

5. שנה את NextButton.Text ל-"Next".

6. שנה את RightWrongLabel.Text לריק .

הוספת התנהגויות לרכיבים

יהיה עליך לתכנת את ההתנהגויות הבאות:

- כאשר האפליקציה מתחילה, מופיעה השאלה הראשונה, כולל התואמת לה תמונה.
- כאשר המשתמש לוחץ על כפתור Next, מופיעה השאלה השנייה. כשזה לחצו שוב, השאלה השלישית מופיעה, וכן הלאה.
- כאשר המשתמש מגיע לשאלה האחרונה ולוחץ על הלחצן Next, הראשון השאלה צריכה להופיע שוב.
- כאשר המשתמש עונה על שאלה, האפליקציה תדווח האם היא נכונה.

תקודד את ההתנהגויות האלה אחת אחת, תוך כדי בדיקה.

הגדרת רשימות השאלות והתשובות

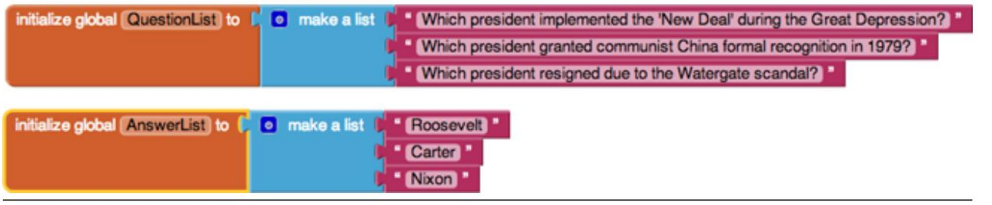
כדי להתחיל, הגדר שני משתני רשימה על סמך הפריטים בטבלה 8-2: QuestionList כדי להחזיק את רשימת השאלות, ו- AnswerList כדי להחזיק את רשימת התשובות המתאימות. איור 8-3 מציג את שתי הרשימות שתיצור בעורך הבלוקים.

טבלה 8-2. משתנים להחזקת רשימות שאלות ותשובות

שם המשתנה		מטרה
QuestionList		משתנים אחסן את רשימת השאלות (שנה את שמה ל- QuestionList).
AnswerList		משתנים אחסן את רשימת התשובות (שנה את שמה ל- AnswerList).
		אחסן את הפריטים של השאלות.

Machine Translated by Google

מטרת מגירה		סוג בלוק
אחסן את רשימת השאלות.		טקסט (שלושה מהם)
הפוך את הפריטים של רשימת התשובות.		עשה רשימה
אחסן את רשימת התשובות.		טקסט (שלושה מהם)



איור .8-4הרשימות לחידון

הגדרת משתנה האינדקס

האפליקציה צריכה לעקוב אחר השאלה הנוכחית כשהמשתמש לוחץ על הבא
כדי להמשיך בחידון. אתה תגדיר משתנה בשם currentQuestionIndexעבור
זה, והמשתנה ישמש כאינדקס הן לשאלות השאלות והן
רשימת תשובות. טבלה 8-3מפרטת את הבלוקים שתצטרך לעשות זאת, ואיור 8-4מראה מה
המשתנה הזה יראה.

טבלה .8-3יצירת האינדקס

מטרת מגירה		סוג בלוק
משתנים החזק את האינדקס (המיקום) של השאלה/תשובה הנוכחית.		אתחול גלובלי ל ("CurrentQuestionIndex")
הגדר את הערך ההתחלתי של currentQuestionIndexל-1 (הראשון שְׁאֵלָה).	מתמטיקה	מספר (1

איור .8-5התחלת משתנה האינדקס עם ערך של 1

הצגת השאלה הראשונה

כעת, לאחר שהגדרת את המשתנים הדרושים לך, תוכל לציין את האינטראקטיביות של האפליקציה
התנהגות. כמו בכל אפליקציה, חשוב לעבוד בהדרגה ולהגדיר אחת
התנהגות בכל פעם. כדי להתחיל, חשוב רק על השאלות, ובמיוחד,
הצגת השאלה הראשונה ברשימה בעת הפעלת האפליקציה. נחזור ו
להתמודד עם התמונות ותשובות קצת מאוחר יותר.

Machine Translated by Google

אתה רוצה שקוביות הקוד שלך יעבדו ללא קשר לשאלות הספציפיות המופעלות הרשימה. בדרך זו, אם תחליט לשנות את השאלות או ליצור חידון חדש על ידי העתקה ושינוי של אפליקציה זו, תצטרך לשנות רק את השאלות בפועל בהגדרות הרשימה, ולא תצטרך לשנות אף מטפל באירועים.

אז להתנהגות הראשונה הזו, אינך רוצה להתייחס ישירות לשאלה הראשונה, "איזה נשיא יישם את 'העסקה החדשה' במהלך השפל הגדול?" במקום זאת, אתה רוצה להתייחס, באופן מופשט, לשקע הראשון ב- QuestionList(ללא קשר לשאלה הספציפית שם). כך, הבלוקים עדיין יעבדו גם אם תשנה את השאלה בשקע הראשון הזה.

אתה בוחר פריטים מסוימים ברשימה עם בלוק פריט הבחירה . הבלוק מבקש ממך לציין את הרשימה ואינדקס (מיקום ברשימה). אם רשימה כוללת שלושה פריטים, אתה יכול להזין 1, 2, או 3בתור האינדקס.

עבור התנהגות ראשונה זו, כאשר האפליקציה מופעלת, אתה רוצה לבחור את הפריט הראשון ב- QuestionListולמקם אותו בתווית השאלות. נזכר מאנדרואיד, איפה המכונית שלי? אפליקציה, אם אתה רוצה שמשהו יקרה כשהאפליקציה שלך מופעלת, אתה מתכנת את ההתנהגות הזו במטפל באירועים . Screen1.Initializeאתה יכול להשתמש בלוקים המפורטים בטבלה 8-4.

Machine Translated by Google

טבלה .8-4חסימות לטעינת השאלה הראשונית כשהאפליקציה מתחילה

מַטָרָה	מִגרָה	סוג בלוק
1מסך		מסך .1אתחול
שים את השאלה הראשונה .QuestionLabel ב-	set QuestionLabel.Text ל-QuestionLabel	
בחר את השאלה הראשונה מרשימת השאלות.	בחר מספר ברשימה	רשימות
הרשימה של שאלות הבחינה.	קבל משתנה גלובלי	משתנים
בחר את השאלה הראשונה באמצעות אינדקס של 1.	בחר מספר ברשימה	מתמטיקה

איך הבלוקים עובדים

אירוע Screen1.Initializeמופעל כאשר האפליקציה מתחילה. כפי שנראה ב
איור ,8-5הפריט הראשון של המשתנה QuestionListנבחר וממוקם לתוך
QuestionLabel.Text.לכן, כאשר האפליקציה מופעלת, המשתמש יראה את השאלה הראשונה.

איור .8-6בחירת השאלה הראשונה

בדוק את האפליקציה שלך לחץ על התחבר והתחבר למכשיר שלך או ל-
אמולטור לבדיקה חיה. כאשר האפליקציה שלך נטענת, האם אתה רואה את
הפריט הראשון של ,QuestionList"איזה נשיא יישם את
'ניו דיל' בתקופת השפל הגדול?"

איטרציה דרך השאלות

כעת, תכנת את ההתנהגות של ה- .NextButtonכבר הגנת על
currentQuestionIndexכדי לזכור את השאלה הנוכחית. כאשר המשתמש לוחץ על
,NextButtonהאפליקציה צריכה להגדיל את ה- currentQuestionIndex(כלומר, לשנות אותו
מ-1 ל-2 או מ-2 ל-3, וכן הלאה). לאחר מכן תשתמש בערך המתקבל של
currentQuestionIndexכדי לבחור את השאלה החדשה להצגה. כאתגר, ראה אם אתה
יכול לבנות את הבלוקים האלה בעצמך. כשתסיים, השווה את התוצאות שלך
נגד איור .8-6

Machine Translated by Google

איור .7-8עוברים לשאלה הבאה

איך הבלוקים עובדים

שורת הבלוקים הראשונה מגדילה את המשתנה currentQuestionIndex. אם currentQuestionIndex ל- יש 1, הוא משתנה ל-2. אם יש לו 2, הוא משתנה ל-3, וכן הלאה. לאחר שינוי המשתנה currentQuestionIndex, האפליקציה משתמשת בו כדי לבחור את השאלה החדשה להצגה. כאשר המשתמש לוחץ על NextButton בפעם הראשונה, בלוקי ההגדלה ישנו את currentQuestionIndex מ-1 ל-2, כך שהאפליקציה תבחר בפריט השני מתוך QuestionList, "איזה נשיא העניק לסין הקומוניסטית הכרה רשמית ב-9791?" בפעם השנייה שלוחצים על currentQuestionIndex NextButton, יוגדר מ-2 ל-3, והאפליקציה תבחר בשאלה השלישית ברשימה, "איזה נשיא התפטר עקב שערוריית ווטרגייט?"

תכונה חיקוי NextButton.Click דומה להשלות בין הבלוקים של Screen.Initialize אלה ב- מטפל באירועים. בתוך ה מסך. אתחול בלוקים, האפליקציה השתמשה בפריט רשימה עם מספר קונקרטי (1)כדי לבחור את פריט הרשימה. בלוקים אלה, אתה בוחר את פריט הרשימה באמצעות משתנה בתור האינדקס. האפליקציה לא בוחרת את הפריט הראשון ברשימה, או את השני או השלישי; הוא בוחר בפריט currentQuestionIndexth, וכך ייבחר פריט אחר בכל פעם שהמשתמש לוחץ על NextButton. זהו שימוש נפוץ מאוד לאינדקס -הגדלת הערך שלו כדי למצוא ולהציג או לעבד פריטים ברשימה.

Machine Translated by Google

בדוק את האפליקציה שלך בדוק את התנהגות ה- NextButtonכדי לראות אם
האפליקציה פועלת כהלכה. לחץ על הלחצן הבא בטלפון.
האם הטלפון מציג את השאלה השנייה, "איזה נשיא
העניקה לסין הקומוניסטית הכרה רשמית ב-9791?" זה
צריך, והשאלה השלישית אמורה להופיע כשאתה לוחץ
שוב כפתור .Nextאבל אם תלחץ שוב, אתה אמור לראות
שגיאה: "ניסיון להשיג פריט 4מתוך רשימה באורך ".3האפליקציה
יש באג! אתה יודע מה הבעיה? נסה לעצב את זה
לצאת לפני שממשיכים הלאה.

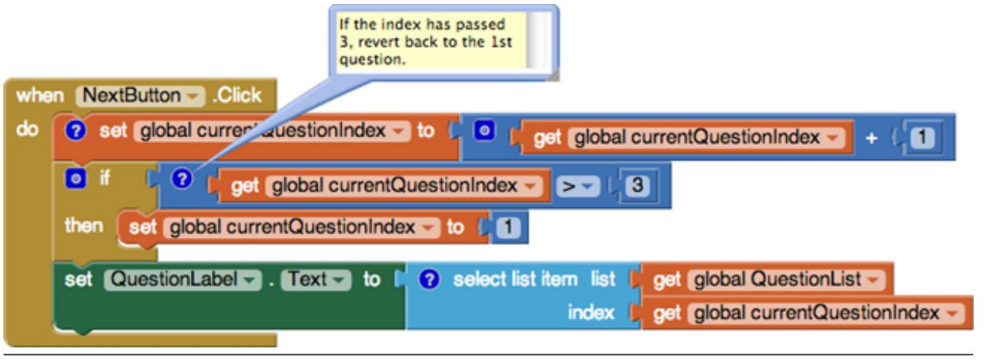
הבעיה עם הקוד עד כה היא שהוא פשוט עולה לקוד הבא
שאלה בכל פעם ללא כל חשש לסוף החידון. מתי
currentQuestionIndexכבר 3והמשתמש לוחץ על ה-nottuBtxeN, האפליקציה משתנה
currentQuestionIndexמ-3 עד .4לאחר מכן הוא קורא לפריט בחר רשימה כדי לקבל את
currentQuestionIndexthפריט -במקרה זה, הפריט הרביעי. כי יש רק
שלושה פריטים במשתנה ,QuestionListמכשיר האנדרואיד לא יודע מה לעשות
לַעֲשׂוֹת. כתוצאה מכך, הוא מציג שגיאה ומאלץ את האפליקציה להפסיק. איך אנחנו יכולים לתת לאפליקציה
יודעים שזה הגיע לסוף החידון?
האפליקציה צריכה לשאול שאלה כאשר לוחצים על , NextButtonולהפעיל
בלוקים שונים בהתאם לתשובה. כי אתה יודע שהאפליקציה שלך מכילה שלושה
שאלות, אחת הדרכים לשאול את השאלה תהיה, "האם המשתנה
currentQuestionIndexגדול מ-3?" אם התשובה היא כן, עליך להגדיר
currentQuestionIndexחזרה ל-1 ולקחת את המשתמש חזרה לשאלה הראשונה. ה
בלוקים שתזדקק לכך מופיעים בטבלה .8-5

טבלה .8-5בלוקים לבדיקת ערך האינדקס לסוף הרשימה

סוג בלוק	מְגֵרָה	מַטָרָה
אם	לִשְׁלוֹט	גלה אם המשתמש נמצא בשאלה האחרונה.
>=	מתמטיקה	בדוק אם currentQuestionIndexגדול מ-3.
קבל currentQuestionIndex העולמי	גרור מ אתחול בלוק	שים את זה בצד שמאל של .=
מספר 3	מתמטיקה	שים את זה בצד ימין של =כי 3הוא המספר של פריטים ברשימה.
currentQuestionIndexהעולמי ל	גרור את אתחול בלוק	הגדר ל-1 כדי לחזור לשאלה הראשונה.
מספר 1	מתמטיקה	הגדר את האינדקס ל-1.

הבלוקים צריכים להופיע כמתואר באיור .8-7

Machine Translated by Google



איור 8-8. בודקים אם האינדקס עבר את השאלה האחרונה

איך הבלוקים עובדים

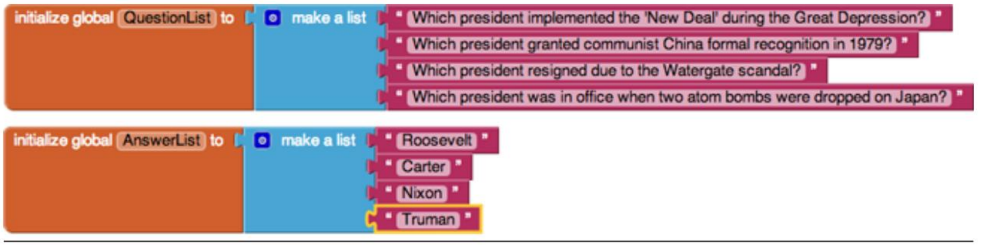
כאשר המשתמש לוחץ על הלחצן Next, האפליקציה מגדילה את האינדקס כפי שעשתה קודם לכן. אבל אז, כפי שמוצג באיור 8-8, הוא בודק אם currentQuestionIndex גדול מ-3, שהוא מספר השאלות. אם הוא גדול מ-3, currentQuestionIndex מוחזר ל-1, והשאלה הראשונה מוצגת. אם הוא 3 או פחות, החסימות בתוך בלוק if לא מבוצעות, והשאלה הנוכחית מוצגת כרגיל.

בדוק את האפליקציה שלך הרם את הטלפון ולחץ על הלחצן הבא. השאלה השנייה, "איזה נשיא העניק לסין הקומוניסטית הכרה רשמית ב-9791?" אמור להופיע בטלפון, כמו קודם. כשתלחץ שוב, השאלה השלישית אמורה להופיע בטלפון. עכשיו, לגבי ההתנהגות שאתה באמת בודק: אם תלחץ שוב, אתה אמור לראות את השאלה הראשונה ("איזה נשיא יישם את 'העסקה החדשה' במהלך השפל הגדול?") מופיעה בטלפון.

הופך את החידון לקל לשינוי

אם הבלוקים שלך עבור NextButton עובדים, טפחו לעצמכם על השכם; אתה בדרך להיות מתכנת! אבל מה אם הוספת שאלה (ותשובה) חדשה לחידון? האם הבלוקים שלך עדיין יעבדו? כדי לחקור זאת, תחילה הוסף שאלה רביעית לרשימת השאלות ותשובה רביעית לרשימת התשובות, כפי שמוצג באיור 8-8.

Machine Translated by Google



איור .8-9הוספת פריט לשתי הרשימות

בדוק את האפליקציה שלך לחץ על NextButtonמספר פעמים. תשים לב שהשאלה הרביעית לעולם לא מופיעה, לא משנה כמה פעמים תלחץ על הבא. אתה יודע מה הבעיה? לפני שתמשיך לקרוא, בדוק אם אתה יכול לתקן את הבלוקים כך שהשאלה הרביעית תופיע.

הבעיה כאן היא שהבדיקה כדי לקבוע אם המשתמש נמצא בשאלה האחרונה היא ספציפית מדי; הוא שואל אם המשתנה currentQuestionIndexגדול מ-3. אתה יכול פשוט לשנות את המספר 3ל-4, והאפליקציה תפעל שוב כהלכה. עם זאת, הבעיה עם הפתרון הזה היא שבכל פעם שאתה משנה את השאלות ורשימות התשובות, עליך לזכור גם לבצע את השינוי הזה במבחן אם.
תלות כזו בתוכנת מחשב מובילה לרוב לבאגים, במיוחד כאפליקציה גדל במורכבות. אסטרטגיה הרבה יותר טובה היא לעצב את הבלוקים כך שהם יעבדו לא משנה כמה שאלות יש. כלליות כזו מקלה על אם אתה, כמתכנת, רוצה להתאים אישית את החידון שלך לנושא אחר. זה גם חיוני אם הרשימה שאיתה אתה עובד משתנה באופן דינמי - לדוגמה, חשבו על אפליקציית חידון המאפשרת למשתמש להוסיף שאלות חדשות (אתה תבנה זאת בפרק .(10כדי שתוכנית תהיה כללית יותר, היא לא יכולה להתייחס למספרים קונקרטיים כמו ,3כי זה עובד רק עבור חידונים של שלוש שאלות.

לכן, במקום לשאול אם הערך של currentQuestionIndexגדול מהמספר הספציפי ,3 שאל אם הוא גדול ממספר הפריטים ב- .QuestionListאם האפליקציה שואלת שאלה כללית יותר זו, היא תעבוד גם כאשר תוסיף או תסיר פריטים מרשימת השאלות. בואו נשנה את המטפל באירועים NextButton.Clickכדי להחליף את הבדיקה הקודמת שהתייחסה ישירות ל-3. תזדקק לבלוקים המפורטים בטבלה .8-6

טבלה .8-6חסימות כדי לבדוק את אורך הרשימה

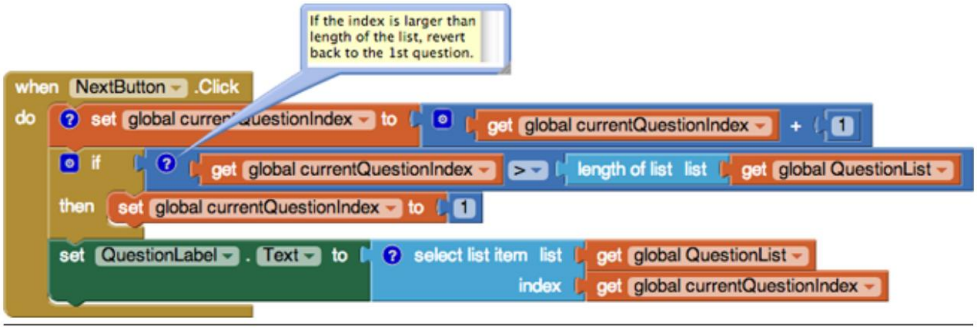
מַטָרָה	מְגֵרָה	סוג בלוק
שאל כמה פריטים יש ברשימת השאלות.	רשימות	אורך הרשימה

Machine Translated by Google

מַטָרָה	מִגרָה	סוג בלוק
הכניסו את זה לשקע "רשימה" של אורך הרשימה.	גרור פנימה מבלוק האתחול	קבל רשימת שאלות גלובלית

איך הבלוקים עובדים

מבחן ה- ifמשווה כעת את ה- currentQuestionIndexלאורך של השאלות, כפי שמוצג באיור .8-11לכן, אם currentQuestionIndexהוא ,5והאורך של QuestionListהוא ,4ה- currentQuestionIndexיוחזר ל-1. שימו לב, מכיוון שהבלוקים כבר לא מתייחסים ל-3 או למספר ספציפי כלשהו, ההתנהגות תעבוד לא משנה כמה פריטים נמצאים ברשימה.



איור .8-10בדיקה אם האינדקס גדול מאורך הרשימה (במקום 3)

בדוק את האפליקציה שלך כאשר אתה לוחץ על הלחצן ,Nextהאם האפליקציה עוברת כעת בין ארבע השאלות, ועוברת לראשונה אחרי הרביעית?

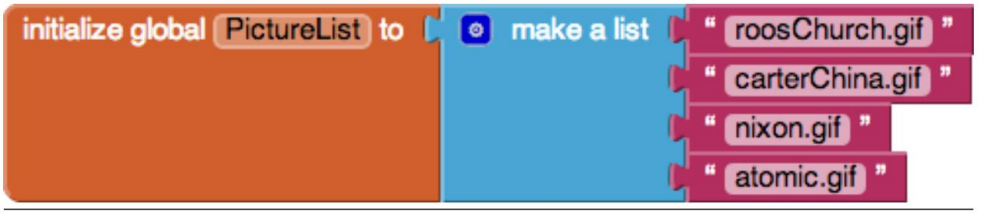
החלפת התמונה עבור כל שאלה

כעת, לאחר שתכנתת את כל ההתנהגויות למעבר בין השאלות (והפכת את הקוד שלך לחכם וגמיש יותר על ידי הפיכתו יותר מופשט), המשימה הבאה שלך היא לגרום לתמונות לעבוד כמו שצריך. כרגע, האפליקציה מציגה את אותה תמונה לא משנה איזו שאלה נשאלת. אתה יכול לשנות זאת כך שתופיע תמונה הקשורה לכל שאלה כאשר המשתמש לוחץ על הבא.

קודם לכן, הוספת ארבע תמונות כמדיה עבור הפרויקט. כעת, תיצור רשימה שלישית, ,PictureListעם שמות התמונה כפריטים שלה. תוכל גם לשנות את מטפל האירועים NextButton.Clickכדי להחליף את התמונה בכל פעם, בדיוק כפי שאתה מחליף את טקסט השאלה. (אם אתה כבר חושב להשתמש ב- currentQuestionIndexכאן, אתה בדרך הנכונה!) ראשית, צור PictureListואתחול אותה עם שמות התמונות. ודא שהשמות זהים לחלוטין לשמות

Machine Translated by Google

flenamesשטענת למקטע המדיה של הפרויקט. איור 8-10מראה כיצד
בלוקים עבור PictureListצריכים להיראות.



איור PictureList .8-11עם שמות תמונות כפריטים

לאחר מכן, שנה את המטפל באירועים NextButton.Clickכך שישנה את התמונה
שמופיע עבור כל שאלה. המאפיין Image.Pictureמשמש לשינוי
תמונה המוצגת. כדי לשנות את ,NextButton.Clickתצטרך את הבלוקים הרשומים ב
טבלה .8-7

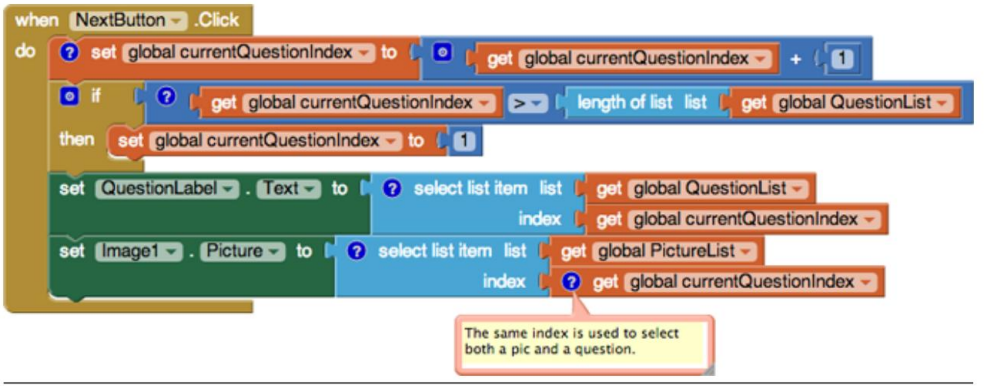
טבלה .8-7בלוקים להוספת התמונה המלווה את השאלה

מַטָרָה	מִגרָה	סוג בלוק
הגדר את זה כדי לשנות את התמונה.	תמונה 1	הגדר תמונה1.תמונה ל
בחר את התמונה המתאימה לתמונה הנוכחית שְׁאֵלָה.	רשימות	בחר פריט רשימה
בחר שם שם מרשימה זו.	גרור החוצה בלוק אתחול	קבל PictureListגלובלי
בחר בפריט .currentQuestionIndex-ה	גרור החוצה בלוק אתחול	קבל currentQuestionIndex העולמי

איך הבלוקים עובדים

ה- currentQuestionIndexמשמש כאינדקס הן עבור השאלות והן עבור
.PictureListכל עוד הגדרת את הרשימות שלך כראוי כך שהשאלה הראשונה
מתאים לתמונה הראשונה, השנייה לשניה, וכן הלאה, האינדקס היחיד
יכול לשרת את שתי הרשימות, כפי שמוצג באיור .8-11לדוגמה, התמונה הראשונה,
,roosChurch.gifהיא תמונה של הנשיא פרנקלין דלאנו רוזוולט (יושב עם בריטים
ראש הממשלה וינסטון צ'רצ'יל), ו"רוזוולט" היא התשובה לשאלה הראשונה.

Machine Translated by Google



איור 8-12. בחירת התמונה הנוכחיתQuestionIndex בכל פעם

בדוק את האפליקציה שלך לחץ על הבא כמה פעמים. עושה תמונה אחרת להופיע בכל פעם שאתה לוחץ על הלחצן הבא?

בדיקת תשובות המשתמש

עד כה, יצרת אפליקציה שפשוט עוברת בין שאלות ותשובות (בשילוב עם תמונה של התשובה). זו דוגמה מצוינת לאפליקציות שמשתמשות ברשימות, אבל כדי להיות אפליקציית חידון אמיתית, היא צריכה לתת למשתמשים משוב אם התשובות שלהם כן נכון. בואו נוסיף את הבלוקים כדי לעשות זאת. הממשק שלנו מוגדר כך שהמשתמש הקלד תשובה ב- AnswerTextולאחר מכן לחץ על AnswerButton. האפליקציה חייבת השווה את הערך של המשתמש עם התשובה לשאלה הנוכחית, באמצעות אם אחרת לחסום כדי לבדוק. לאחר מכן יש לשנות את RightWrongLabelכדי לדווח אם ה התשובה נכונה. יש לא מעט בלוקים הדרושים כדי לתכנת את ההתנהגות הזו, כולם המפורטים בטבלה 8-8.

טבלה 8-8. בלוקים לציון האם תשובה נכונה

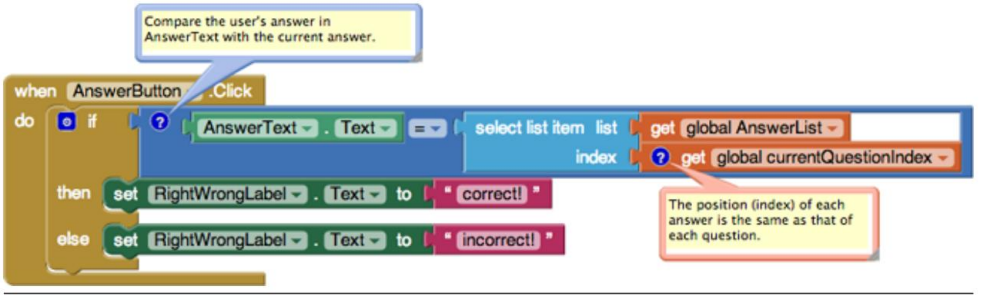
מַטָרָה	מְגִרָה	סוג בלוק
מופעל כאשר המשתמש לוחץ על כפתור תשובה.	כפתור תשובה	AnswerButton.Click
אם התשובה נכונה, עשה דבר אחד; אחרת, לעשות אחרת.	לְשַׁלוֹט	אחרת
שאל אם התשובה נכונה.	מתמטיקה	=
מכיל את תשובת המשתמש.	תשובה טקסט	AnswerText.Text
בחר את התשובה הנוכחית מרשימת התשובות.	רשימות	בחר פריט רשימה
הרשימה לבחירה.	גרור החוצה בלוק אתחול	קבל רשימת תשובות גלובלית

Machine Translated by Google

מַטָרָה	מִגרָה	סוג בלוק
מספר השאלה (והתשובה) הנוכחית.	גרור החוצה בלוק אתחול	קבל currentQuestionIndex העולמי
דווח על התשובה כאן.	הגדר RightWrongLabel-ל RightWrongLabel.Text	
הצג זאת אם התשובה נכונה.	טקסט ("נכון!")	
דווח על התשובה כאן.	הגדר RightWrongLabel-ל RightWrongLabel.Text	
הצג זאת אם התשובה שגויה.	טקסט ("לא נכון!")	

איך הבלוקים עובדים

איור 12-8מדגים כיצד מבחן אם אחרת שואל האם התשובה למשתמש
מספק (AnswerText.Text)שווה לפריט currentQuestionIndexבמערך
רשימת תשובות. אם currentQuestionIndexהוא ,1האפליקציה תשווה את תשובת המשתמש עם
הפריט הראשון ברשימת התשובות, "רוזוולט". אם currentQuestionIndexהוא ,2האפליקציה תהיה
השווה את תשובת המשתמש לתשובה השנייה ברשימה, "קרטר" וכן הלאה. אם
תוצאת הבדיקה חיובית, הסניף לאחר מכן מבוצע והRightWrongLabel מוגדר
לתקן!" אם הבדיקה היא שקר, הסניף elseמבוצע והRightWrongLabel כן
מוגדר ל"לא נכון!"



איור .13-8בודק את התשובה

בדוק את האפליקציה שלך נסה לענות על אחת השאלות. זה אמור
דווח אם ענית על השאלה בדיוק כפי שצוין
ברשימת התשובות. בדוק עם נכון ולא נכון
תשובה. סביר להניח שתבחין בכך כדי שתשובה תסומן כ
נכון, זה חייב להיות התאמה מדויקת וספציפית מקרה למה
שהזנת ברשימת התשובות. הקפד לבדוק גם את הדברים
לעבוד על שאלות עוקבות.

האפליקציה אמורה לעבוד, אבל אולי תשים לב שכאשר תלחץ על הלחצן הבא,
הנכון!" או "לא נכון!" הטקסט והתשובה הקודמת עדיין שם, כפי שמוצג ב

Machine Translated by Google

איור 8-13, למרות שאתה מסתכל על השאלה הבאה. זה די תמים, אבל משתמשי האפליקציה שלך בהחלט ישימו לב לבעיות כאלה בממשק משתמש. כדי לבטל את התווית RightWrongLabelואת ה- AnswerText, תכניס את הבלוקים הרשומים בטבלה 8-9 בתוך המטפל באירועים . NextButton.Click

איור 8-14. החידון בפעולה כאשר התשובה הקודמת מופיעה כאשר היא לא אמורה

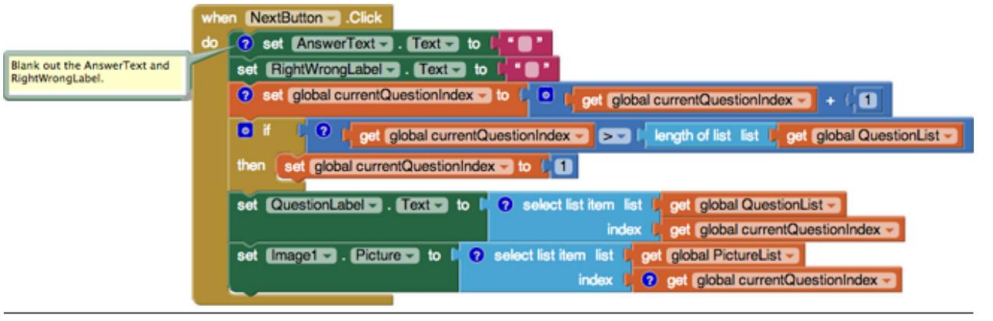
Machine Translated by Google

טבלה .8-9חסימות כדי לנקות את התווית RightWrongLabel

מִגרָה	סַעִיף
RightWrongLabel.Text	הגדר RightWrongLabel-ל- זו התווית שיש לבטל.
	כאשר המשתמש לוחץ על NextButton, נקה את הקודם המשוב של התשובה.
	תשובת המשתמש לשאלה הקודמת.
טֶקסט	כאשר המשתמש לוחץ על הלחצן Next, נקה את ה- תשובה קודמת.

איך הבלוקים עובדים

כפי שמוצג באיור 8-14, על ידי לחיצה על כפתור Next, המשתמש עובר לבא שאלה, כך ששתי השורות העליונות של המטפל באירועים מנקות את RightWrongLabel ו טקסט התשובה .



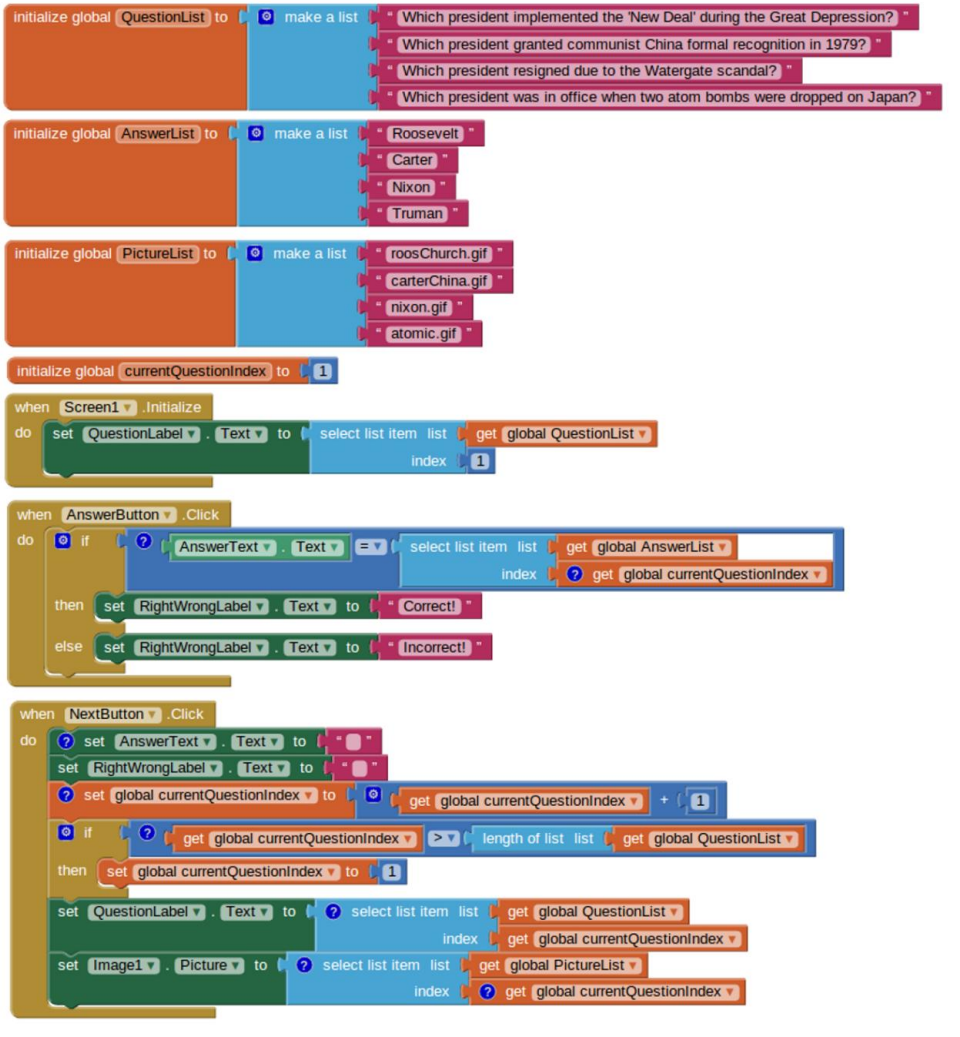
איור .8-15 PictureList עם שמות תמונות כפריטים

בדוק את האפליקציה שלך ענה על שאלה ולחץ על שלח ולאחר מכן לחץ כפתור הבא. האם תשובתך הקודמת והמשוב שלה לְהֵעָלֵם?

האפליקציה השלמה: חידון הנשיאים

איור 8-15 מציג את תצורת הבלוק הסופי עבור חידון הנשיאים.

Machine Translated by Google



איור .8-16בלוקים לחידון הנשיאים

וריאציות

כאשר אתה מקבל את החידון הזה לעבוד, אולי תרצה לחקור כמה וריאציות, כגון הבאות:

•במקום להציג רק תמונות עבור כל שאלה, נסה להפעיל קליפ קול או סרטון קצר. עם סאונד, אתה יכול להפוך את החידון שלך לאפליקציית .Name That Tune

•החידון מאוד נוקשה מבחינת מה שהוא מקבל כתשובה תקפה. ישנן מספר דרכים לשפר זאת, תוך ניצול היתרון של בלוקים לעיבוד טקסט במגירת הטקסט. אחת הדרכים היא להשתמש בלוק upcaseבמגירת הטקסט כדי

Machine Translated by Google

המר את תשובת המשתמש ואת התשובה האמיתית לכל אותיות רישיות לפני ההשוואה.
אחר הוא להשתמש בבלוק text.containsכדי לראות אם התשובה של המשתמש היא
הכלול בתשובה בפועל. אפשרות נוספת היא לספק מספר תשובות לכל שאלה ולבדוק על ידי איטרציה (לפנים) דרכן כדי לראות אם יש התאמה.

•דרך נוספת לשפר את בדיקת התשובות היא לשנות את החידון כך שיהיה רב ברירה. תזדקק לרשימה נוספת כדי להחזיק את אפשרויות התשובות לכל שאלה. התשובות האפשריות יהיו רשימה של רשימות, כאשר כל תת-רשימה תכיל את אפשרויות התשובות לשאלה מסוימת. השתמש ברכיב ListPickerכדי לתת למשתמש את היכולת לבחור תשובה. תוכל לקרוא עוד על רשימות בפרק 19.

סיכום

הנה כמה מהרעיונות שכיסינו במדריך זה:

•לרוב האפליקציות המתוחכמות למדי יש נתונים (לרוב מאוחסנים ברשימות) והתנהגות -שלהם מטפלי אירועים.

•השתמש בבלוק ifelseכדי לבדוק תנאים. למידע נוסף על
תנאים, ראה פרק 18.

•החסימות במטפלי אירועים צריכים להתייחס רק באופן מופשט לפריטי רשימה ולגודל רשימה כך שהאפליקציה תעבוד גם אם הנתונים ברשימה ישתנו.

•משתני אינדקס עוקבים אחר המיקום הנוכחי של פריט בתוך רשימה. כאשר אתה מגדיל אותם, השתמש ב- if blockכדי לטפל בהתנהגות האפליקציה כאשר המשתמש מגיע לסוף הרשימה.

Machine Translated by Google

Machine Translated by Google

קסילופון

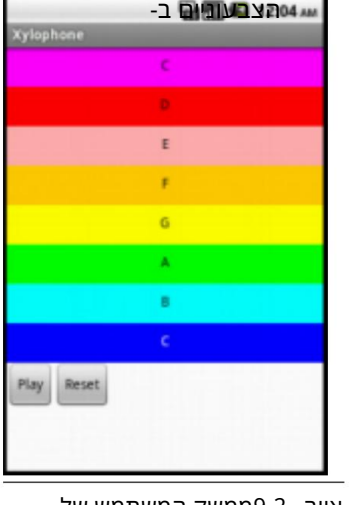
קשה להאמין ששימוש בטכנולוגיה כדי להקליט ולהשמיע מוזיקה רק משנת ,1878כאשר אדיסון רשם פטנט על הפטיפון. הגענו כל כך רחוק מאז -עם סינתיסייזרים מוזיקה, תקליטורים, דגימה ורמיקס, טלפונים שמשמיעים מוזיקה ואפילו ג'מינג למרחקים ארוכים דרך האינטרנט. בפרק זה, תוכלו לקחת חלק במסורת זו על ידי בניית אפליקציית Xylophone שמקליטה ומשמיעה מוזיקה.

איור .9-1

מה אתה תבנה

עם האפליקציה המוצגת באיור 9-1(יוצרה במקור על ידי ליז לוני מצוות ה-ppA ,(Inventor אתה יכול:

•נגן שמונה תווים שונים על ידי נגיעה בלחצנים הצבעוניים ב-



מַסָך.

•לחץ על לחצן הפעלה כדי להשמיע מחדש את ההערות שאתה שיחק קודם לכן.

•לחץ על כפתור איפוס כדי שהאפליקציה תנקה את כל ההערות ששיחקת קודם לכן כדי שתוכל להזין א שיר חדש.

מה תלמד

הדרכה זו מכסה את המושגים הבאים:

•שימוש ברכיב סאונד יחיד לנגינה קבצי אודיו שונים.

איור .9-2ממשק המשתמש של אפליקציית Xylophone

•שימוש ברכיב השעון כדי למדוד ולאכוף עיכובים בין פעולות.

Machine Translated by Google

•החלטה מתי ליצור נוהל.

•יצירת נוהל שקורא לעצמו.

•שימוש מתקדם ברשימות, כולל הוספת פריטים, גישה אליהם וניקוי
רשימה.

מתחילים

התחבר לאתר App Inventorוהתחל פרויקט חדש. תן לזה "קסילופון",
וגם הגדר את כותרת המסך ל"קסילופון". חבר את האפליקציה שלך למכשיר שלך או
אמולטור.

עיצוב הרכיבים

לאפליקציה הזו יש 13רכיבים שונים (מתוכם 8מהווים את המקלדת), שהם
המפורטים בטבלה .9-1בגלל שיש כל כך הרבה, זה יהיה די משעמם ליצור את כולם
מהם לפני שמתחילים לכתוב את התוכנית שלנו, אז נחלק את האפליקציה לתוכנה
חלקים פונקציונליים ולבנות אותם ברצף על ידי מעבר הלוך ושוב בין ה
מעצב ועורך הבלוקים, כפי שעשינו עם אפליקציית Ladybug Chaseבפרק 5.

טבלה .9-1כל הרכיבים לאפליקציית הקסילופון

מַטָרָה	קבוצת צבעים איך תקרא לזה	סוג רכיב
נגן מקש Cנמוך.	כפתור ממשק משתמש 1	לַחְצָן
נגן מקש D.	כפתור ממשק משתמש2	לַחְצָן
נגן מקש E.	כפתור ממשק משתמש 3	לַחְצָן
נגן מקש F.	כפתור ממשק משתמש 4	לַחְצָן
נגן מקש G.	כפתור ממשק משתמש 5	לַחְצָן
נגן מפתח.	כפתור ממשק משתמש 6	לַחְצָן
נגן מקש B.	כפתור ממשק משתמש 7	לַחְצָן
נגן מפתח Cגבוה.	כפתור ממשק משתמש8	לַחְצָן
נגן את התווים.	כְּלֵי תִקְשׁוֹרֶת צליל 1	נשמע
הפעל את השיר.	ממשק משתמש כפתור הפעלה	לַחְצָן
אפס את זיכרון השיר.	כפתור איפוס ממשק משתמש	לַחְצָן
מקם את לחצני ההפעלה והאיפוס אחד ליד השני.	סידור אופקי 1 פריסת סידור אופקי	
עקוב אחר עיכובים בין הערות.	שעון ממשק משתמש 1	שָׁעוֹן

Machine Translated by Google

יצירת המקלדת

ממשק המשתמש שלנו יכלול מקלדת בעלת שמונה צלילים לסולם מז'ור פנטטוני (שבעה צלילים) החל מ-C Lowועד .C High אנו ניצור את המקלדת המוזיקלית הזו בחלק זה.

יצירת כפתורי ההערה הראשונה

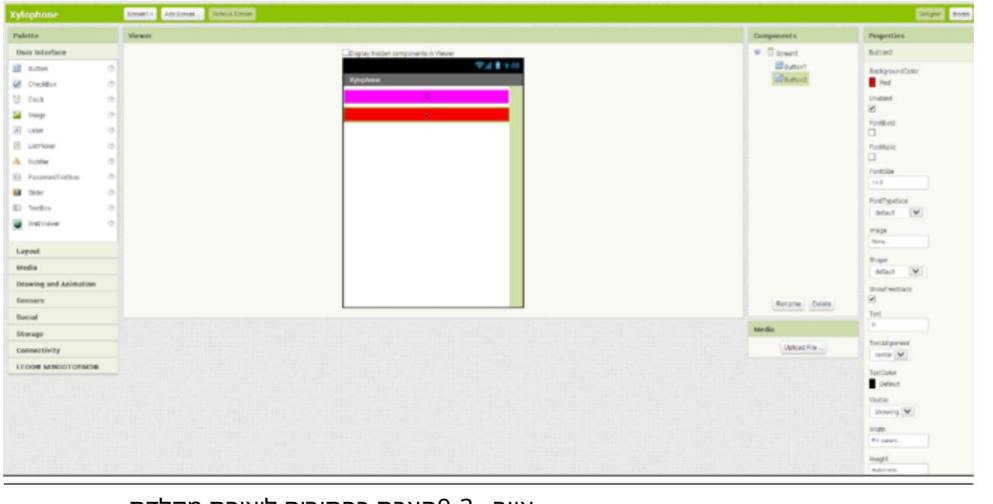
התחל ביצירת שני מקשי הקסילופון הראשונים, אותם ניישם ככפתורים.

1. מהקטגוריה ממשק משתמש, גרור לחצן אל המסך. עזוב את זה שם ככפתור1. אנחנו רוצים שזה יהיה פס מגנטה ארוך, כמו זה בקסילופון, אז הגדר את המאפיינים שלו באופן הבא:

- שנה את המאפיין BackgroundColor למגנטה.
- שנה את המאפיין טקסט ל- "C".
- הגדר את המאפיין Width ל-"lliF" parent" כך שהוא משתרע לאורך כל הדרך המסך.
- הגדר את המאפיין Height ל-04 פיקסלים.

2. חזור על כפתור שני, בשם Button2, והצב אותו מתחת Button1. - להשתמש ב אותם ערכי מאפיין Width - ו Height, אך הגדר את המאפיין BackgroundColor שלו לאדום ואת תכונת הטקסט שלו ל-"D".

(מאוחר יותר, נחזור על שלב 2 עבור שישה לחצני הערה נוספים.)
התצוגה Component Designer - בצריכה להיראות בערך כמו איור 9-2.



איור 9-3. הצבת כפתורים ליצירת מקלדת

Machine Translated by Google

התצוגה בטלפון שלך צריכה להיראות דומה, אם כי לא תהיה כזו
חלל ריק בין שני הכפתורים הצבעוניים.

הוספת רכיב הסאונד

לא יכול להיות לנו קסילופון בלי צלילים, אז גרור פנימה רכיב סאונד , השאר את שמו כ-
.Sound1שנה את המאפיין MinimumInterval מערך ברירת המחדל שלו של 500אלפיות
השנייה ל-0. זה מאפשר לנו להשמיע את הסאונד בתדירות גבוהה ככל שנרצה, במקום להמתין
חצי שנייה (500אלפיות שניות) בין השמעה. אל תגדיר את מאפיין ה- Sourceשלו , אותו נגדיר
בעורך הבלוקים.
הורד את הצלילים: http://appinventor.org/bookFiles/Xylophone/1.wav- ו-
2.wav. http://appinventor.org/bookFiles/Xylophone/שלא כמו בפרקים הקודמים, שבהם היה
בסדר לשנות שמות של אנשי תקשורת, חשוב להשתמש בשמות המדויקים האלה מסיבות
שיתבררו בקרוב. אתה יכול להעלות את ששת הצלילים הנותרים כשתכוונו מאוחר יותר.

חיבור הצלילים ללחצנים

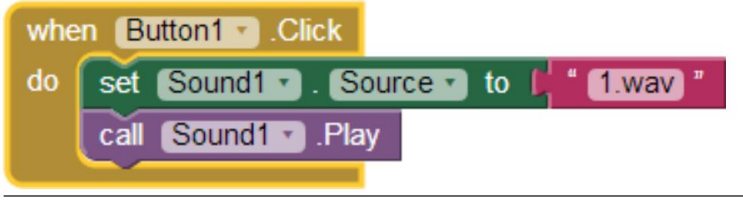
ההתנהגות שעלינו לתכנת היא הפעלת סאונד בלחיצה על הכפתור המתאים. באופן ספציפי,
אם לוחצים על , Button1נרצה לשחק ב-1.vaw; אם לוחצים על כפתור2, נרצה לשחק
ב-2.vaw; וכולי. אנו יכולים להגדיר זאת בעורך הבלוקים, כפי שמוצג באיור 9-3, על ידי ביצוע
הפעולות הבאות:

1. מהמגירה של Button1, גרור החוצה את בלוק Button1.Click .

2. מהמגירה של Sound1, גרור החוצה את בלוק הסט Sound1.Source , והצב אותו ב-
לחצן 1. לחץ על חסום.

3. הקלד "טקסט" כדי ליצור בלוק טקסט. (זה מהיר יותר מאשר מעבר ללשונית Built-In ולאחר
מכן למגירת הטקסט, אם כי גם זה יעבוד.) הגדר את ערך הטקסט שלו ל-"1.vaw" והצב אותו
בבלוק Sound1.Source .

4. הוסף בלוק Sound1.Play .



איור 9-4. השמעת צליל בעת לחיצה על כפתור

נוכל לעשות את אותו הדבר עבור Button2, כפי שמוצג באיור 9-4 (רק משנים את ערך
הטקסט), אבל הקוד יחזור על עצמו נורא.

Machine Translated by Google

איור 5-9.הוספת צלילים נוספים

קוד חוזר הוא סימן טוב שעליך ליצור פרוצדורה, שכבר ביצעת במשחק MoleMashבפרק 3ובמשחק Ladybug Chaseבפרק 5.באופן ספציפי, ניצור פרוצדורה שלוקחת מספר כפרמטר , מגדיר את המקור של Sound1לפל המתאים ומשמיע את הצליל. זוהי דוגמה נוספת - refactoring-לשיפור היישום של תוכנית מבלי לשנות את ההתנהגות שלה, מושג שהוצג במדריך .MoleMashאנו יכולים להשתמש בגוש ההצטרפות של מגירת הטקסט כדי לשלב את המספר (למשל, 1)ואת הטקסט ".wav"כדי ליצור את ה-emanelf המתאים (למשל, "1.wav").להלן השלבים ליצירת ההליך שאנו צריכים:

1.תחת הכרטיסייה מובנה, עבור אל מגירת ההליכים וגרור החוצה את ה- to נוהל לחסום. (אלא אם צוין אחרת, עליך לבחור את הגרסה עם "עשה", לא "תוצאה".)

2.הוסף את הפרמטר על-ידי לחיצה על הסמל הכחול הקטן ב- ,do block to procedureגרירה מעל קלט, ושינוי השם שלו מ-"x" ל-"number". אולי תרצה לעיין באיור 5-6מפרק 5 .

3.לחץ על שם ההליך, שהוא כברירת מחדל "פרוצדורה" והגדר אותו ל "PlayNote".

4.גרור את בלוק Sound1.Sourceמכפתור1 . לחץ לתוך PlayNoteמימין למילה "עשה". העבר את בלוק Sound1.Playגם ל-etoNyalP.

5.גרור את גוש 1.wavלפח האשפה.

6.ממגירת הטקסט, גרור את גוש ההצטרפות לתוך השקע של . Sound1.Source

7.הקלד "מספר" והעבר אותו לשקע העליון של בלוק החיבור (אם לא כבר שם).

8.ממגירת הטקסט, גרור את גוש הטקסט לתוך השקע השני של החיבור לַחסוֹם.

9.שנה את ערך הטקסט ל-".vaw". (זכור לא להקליד את המרכאות.)

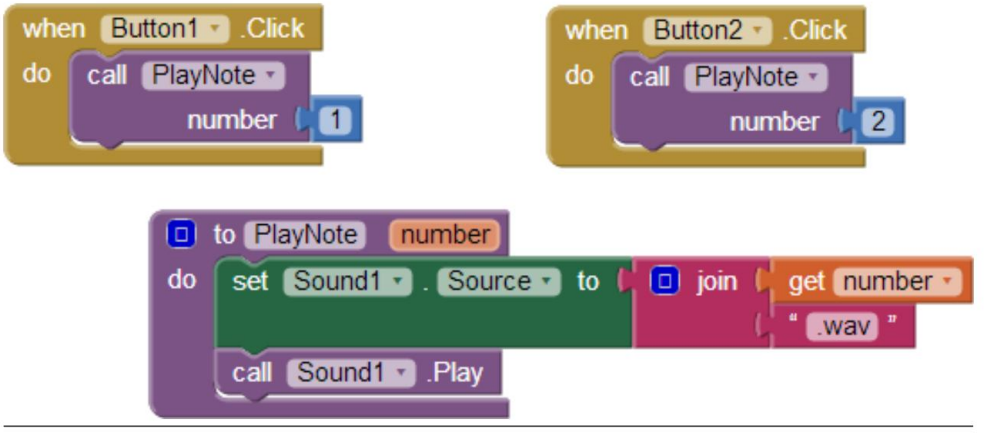
10.מהמגירה של נהלים, גרור החוצה בלוק PlayNoteשיחה והצב אותו ב- גוף ריק של .Button1.Click

11.הקלד "1"והכנס אותו לשקע "מספר".

Machine Translated by Google

כעת, כאשר לוחצים על כפתור 1, ההליך PlayNote ייקרא, כאשר פרמטר המספר שלו יהיה בעל הערך 1. הוא צריך להגדיר את Sound1.Source ל-"vaw.1" ולהשמיע את הצליל.

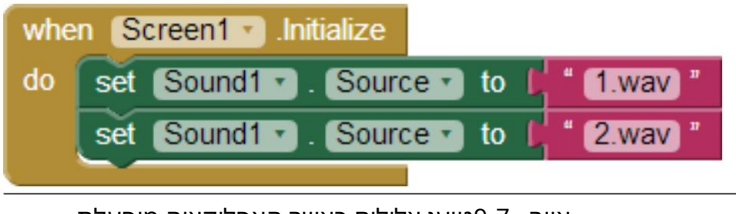
צור כפתור דומה 2. בלוק קליק עם קריאה ל- PlayNote עם פרמטר 2.
(אתה יכול להעתיק את הבלוק הקיים של הקריאה PlayNote ולהעביר אותו לגוף של Button2.Click, הקפד לשנות את הפרמטר.) התוכנית שלך צריכה להיראות כמו איור 9-5.



איור 9-6. יצירת נוהל לנגינה של תו

הדרכה לאנדרואיד לטעון את הצלילים

אם ניסיתם את השיחות הקודמות ל- PlayNote, ייתכן שהתאכזבתם מכך שלא שמעתם את הצליל שציפיתם לו או על ידי שגיאה או עיכוב בלתי צפוי.
הסיבה לכך היא שאנדרואיד צריך לטעון צלילים בזמן ריצה, מה שכרוך בפיגור מסוים לפני שניתן להפעיל אותם. בעיה זו לא עלתה קודם לכן מכיוון ששמות שממוקמים במאפיין המקור של רכיב סאונד ב-rengiseD נטענים אוטומטית כאשר התוכנית מתחילה. מכיוון שאנו לא מגדירים את Sound1.Source עד לאחר הפעלת התוכנית, תהליך האתחול הזה לא מתרחש. עלינו לטעון במפורש את הצלילים כאשר התוכנית מופעלת, כפי שמוצג באיור 9-6.



איור 9-7. טוען צלילים כאשר האפליקציה מופעלת

Machine Translated by Google

בדוק את האפליקציה שלך גע בלחצנים ובדוק אם התווים מתנגנים ללא דיחוי. (אם אינך שומע דבר, ודא שעוצמת הקול של המדיה בטלפון שלך אינה מוגדרת להשתיק.)

יישום ההערות הנותרות

כעת, כששני הכפתורים וההערות הראשונים מיושמים ופועלים, הוסף את ששת התווים הנותרים על ידי חזרה למעצב והורדת קטעי הסאונד:

- http://appinventor.org/bookFiles/Xylophone/3.wav
- http://appinventor.org/bookFiles/Xylophone/4.wav
- http://appinventor.org/bookFiles/Xylophone/5.wav

http://appinventor.org/bookFiles/Xylophone/7.wav

• http://appinventor.org/bookFiles/Xylophone/6.wav •

• http://appinventor.org/bookFiles/Xylophone/8.wav

לאחר מכן, צור שישה כפתורים חדשים, בצע את אותם שלבים כפי שעשית עבור הקודם שניים אך מגדירים את מאפייני הטקסט וצבע הרקע שלהם באופן הבא:

•כפתור ,"E") 3ורוד)

- •כפתור ,"F") 4כתום)

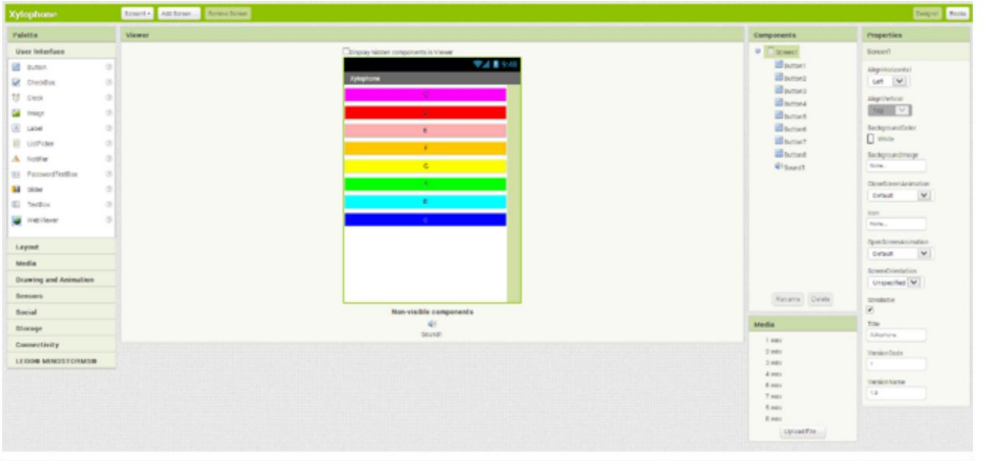
כפתור ,"G") 5צהוב)

•כפתור ,"A") 6ירוק)

•כפתור ,"B") 7ציאן) •כפתור ,"C") 8כחול)

ייתכן שתרצה גם לשנות את המאפיין TextColor של Button8 ללבן, כפי שמוצג באיור 9-7, כדי שיהיה קריא יותר.

Machine Translated by Google



איור .9-8הכנסת הלחצנים והצלילים הנותרים Component Designer-ב

בחזרה בעורך הבלוקים, צור קוביות קליק עבור כל אחד מהלחצנים החדשים עם שיחות מתאימות ל- .PlayNoteבאופן דומה, הוסף כל צליל חדש ל- Screen.Initialize, כפי שמוצג באיור .9-8

איור .9-9תכנות אירועי לחיצה על הכפתור כך שיתאימו לכל מקשי המקלדת

בדוק את האפליקציה שלך כעת אמורים להיות לך כל הכפתורים, וכל אחד מהם ינגן צליל אחר כשתלחץ עליו.

Machine Translated by Google

הקלטה והשמעת הערות

השמעת תווים על ידי לחיצה על כפתורים היא מהנה, אבל היכולת להקליט ולהשמיע שירים היא אפילו טובה יותר. כדי ליישם השמעה, נצטרך לשמור תיעוד של תווים מושמעים. בנוסף לזכור את המגרשים (צלילים) שנוגנו, עלינו לרשום גם את משך הזמן בין תווים, אחרת לא נוכל להבחין בין שני תווים שנוגנו ברצף מהיר לבין שניים מנוגנים בדממה של 10שניות ביניהם.

האפליקציה שלנו תשמור על שתי רשימות, שלכל אחת מהן תהיה רשומה אחת עבור כל תו שהושמע:

•הערות, שיכילו את שמות קבצי הקול לפי סדר השמעתם.

•זמנים, אשר יתעדו את נקודות הזמן שבהן התווים הושמעו.

הערה לפני שתמשיך, אולי תרצה לעיין ברשימות, שמכוסות בחידון הנשיאים בפרק 8ובפרק 19.

אנחנו יכולים לקבל את מידע התזמון מרכיב , Clockשבו נשתמש גם לתזמן נכון את התווים להשמעה.

הוספת הרכיבים

ב-rengiseD, תצטרך להוסיף רכיב שעון וכפתורי הפעלה ואיפוס, אותם תשים בסידור אופקי:

1.ממגירת החיישנים, גרור פנימה רכיב שעון . זה יופיע ב-
סעיף "רכיבים שאינם נראים לעין". בטל את הסימון של המאפיין TimerEnabledשלו כי אנחנו לא רוצים שהטיימר שלו יכבה עד שנאמר לו לעשות זאת במהלך ההשמעה.

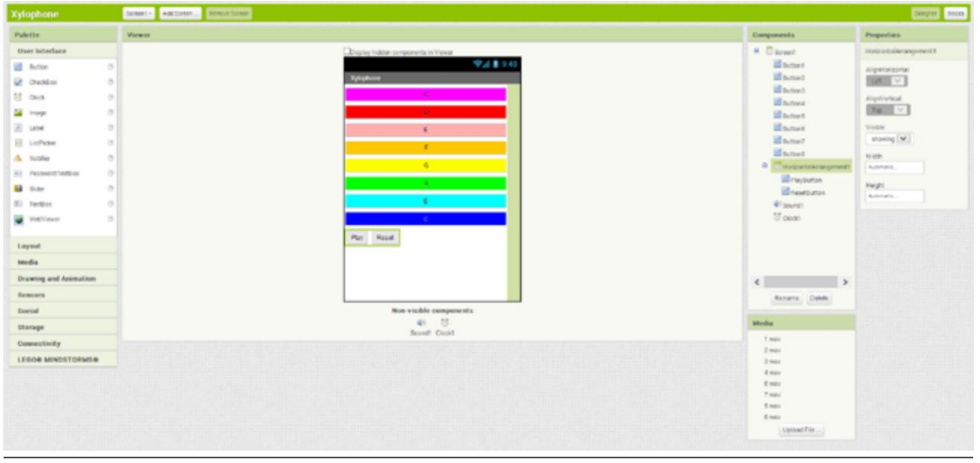
2.עבור אל מגירת ה-tuoyaL וגרור רכיב Horizontal Arrangementמתחת ללחצן הקיים. הגדר את מאפיין הרוחב שלו ל"מלא הורה".

3.מהמגירה של ממשק המשתמש, גרור פנימה כפתור. שנה את השם ל- "PlayButton"ו
הגדר את מאפיין הטקסט שלו ."Play"-ל

4.גרור פנימה כפתור נוסף, והצב אותו מימין ל- PlayButton. שנה את שם הלחצן החדש "ResetButton"והגדר את מאפיין הטקסט שלו ל"איפוס ."

תצוגת המעצב צריכה להיראות כמו איור 9-9.

Machine Translated by Google



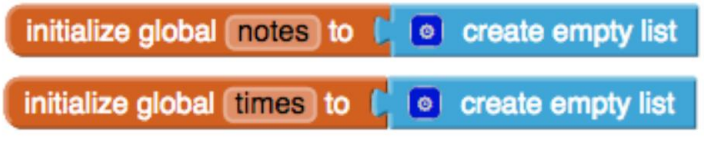
איור 9-10. הוספת רכיבים להקלטה והשמעת צלילים

הערות וזמנים להקלטה

כעת עלינו להוסיף את ההתנהגות הנכונה בעורך הבלוקים. נצטרך לשמור רשימות של הערות וזמנים ולהוסיף לרשימות בכל פעם שהמשתמש לוחץ על כפתור.

1. צור משתנה חדש על ידי מעבר למגירת המשתנים וגרירת אתחול גלובלי לחסימה ממגירת ההגדרה.
2. שנה את שם המשתנה ל"הערות".
3. פתח את מגירת הרשימות וגרור יצירת בלוק רשימה ריק החוצה, והצב אותו ב-שקע של אתחול גלובל לחסום.

זה מגדיר משתנה חדש בשם "הערות" להיות רשימה ריקה. חזור על השלבים עבור משתנה אחר, שעליך לקרוא לו "זמנים". בלוקים חדשים אלה צריכים להיראות כמו אלה באיור 9-10.



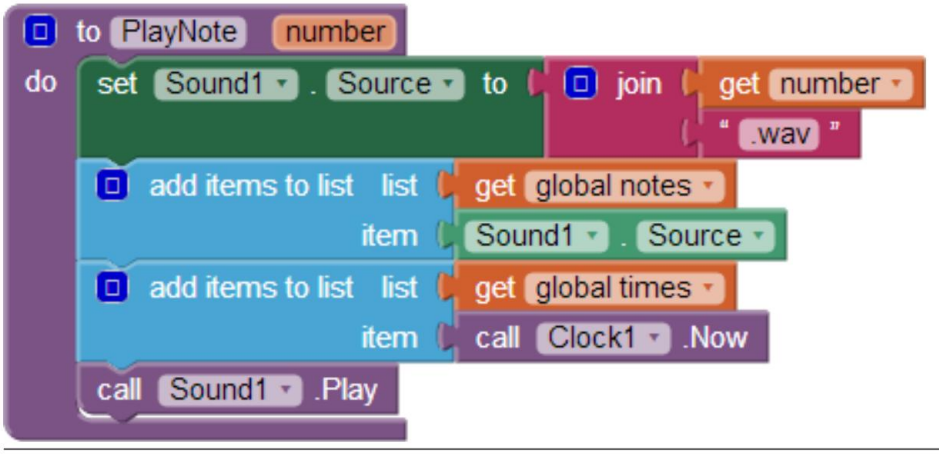
איור 9-11. אתחול שני משתנים כדי לאחסן את ההערות ואת מידע התזמון

איך הבלוקים עובדים

בכל פעם שמתנגן תו, אנחנו צריכים לשמור גם את שם הצליל (להערות הרשימה) וגם את הרגע בזמן שבו הוא הושמע (לזמני הרשימה). כדי להקליט

Machine Translated by Google

הרגע בזמן, נשתמש בבלוק , Clock1.Nowהמחזיר את הרגע הנוכחי בזמן (למשל, 12במרץ 2011, 8:33:14בבוקר), לאלפית השנייה הקרובה. ערכים אלה, שהתקבלו דרך הבלוקים Sound1.Sourceו- , Clock1.Nowיש להוסיף לרשימות הערות וזמנים , בהתאמה, כפי שמוצג באיור 9-11.



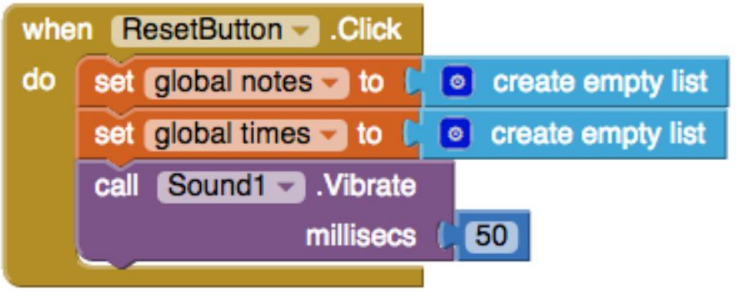
איור 9-12. הוספת הצלילים המושמעים לרשימה

Machine Translated by Google

לדוגמה, אם תשחק ב"שורה, שורה, תחתור בסירה שלך" [EDCCC], בסופו של דבר יהיו לרשימות שלך חמישה ערכים, שעשויים להופיע באופן הבא:

- הערות: 1.wav, 1.wav, 1.wav, 2.wav, 3.wav
- זמנים [תאריכים הושמטו]: 12:00:01, 12:00:02, 12:00:03, 12:00:03.5, 12:00:04

כאשר המשתמש לוחץ על כפתור האיפוס, אנו רוצים ששתי הרשימות יחזרו למצב המקורי והריק שלהן. מכיוון שהמשתמש לא יראה שום שינוי, נחמד להוסיף בלוק Sound1.Vibrate קטן כדי לציין שהלחיצה על המפתח נרשמה. איור 9-12מציג את הבלוקים להתנהגות זו.



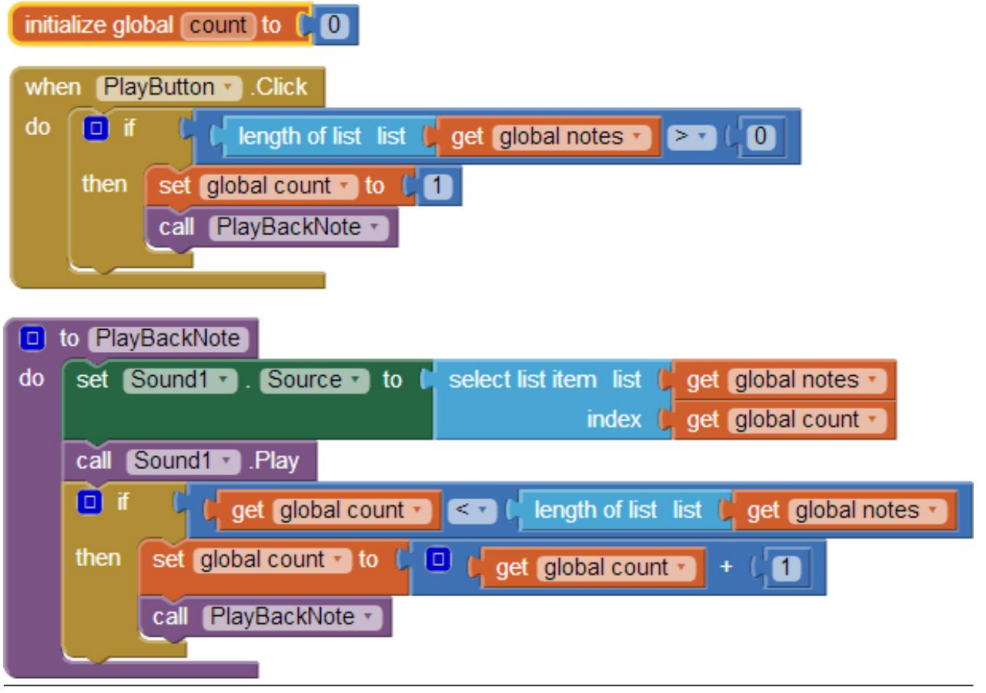
איור 9-13. מתן משוב כאשר המשתמש מאפס את האפליקציה

השמעת הערות

בתור ניסוי מחשבתי, בואו נסתכל תחילה כיצד ליישם השמעת תווים מבלי לדאוג לתזמון. נוכל (אך לא נעשה) זאת על ידי יצירת בלוקים אלה כפי שמוצג באיור 9-13:

- ספירת משתנים כדי לעקוב אחר איזה פתק אנחנו נמצאים.
- נוהל חדש, PlayBackNote, שמנגן את התו הזה ועובר אל הבא.
- קוד להפעלה בעת לחיצה על PlayButton שמגדיר את הספירה ל-1 וקורא ל- PlayBackNoteאלא אם כן אין הערות שמורות.

Machine Translated by Google



איור 9-14. השמעת התווים המוקלטים

איך הבלוקים עובדים

זו עשויה להיות הפעם הראשונה שאתה רואה נוהל מתקשר לעצמו. למרות שבמבט ראשון זה אולי נראה מזויף, זהו למעשה מושג חשוב ורב עוצמה במדעי המחשב הנקרא רקורסיה.

כדי לקבל מושג טוב יותר על אופן פעולת הרקורסיה, הבה נעבור דרך מה קורה אם משתמש מנגן/מקליט שלושה תווים (1.wav, 3.wav ו-vaw.6) ולאחר מכן לוחץ על כפתור ההפעלה. ראשית, PlayButton.Click מתחיל לפעול. מכיוון שאורך הערות הרשימה הוא 3, שהוא גדול מ-0, הספירה מוגדרת ל-1, ו- PlayBackNote נקרא:

1. בפעם הראשונה שנקרא PlayBackNote , ספירה = 1:

Sound1.Source מוגדר לפריט הראשון בתווים, שהוא 1.wav.

Sound1.Play נקרא, מנגן את התו הזה.

מכיוון שהספירה (1) קטנה מאורך התווים, (3) הספירה גדלה ל-2, ו- PlayBackNote נקרא שוב.

2. בפעם השנייה שנקרא PlayBackNote , count = 2:

Sound1.Source מוגדר לפריט השני בתווים, שהוא 3.wav.

Sound1.Play נקרא, מנגן את התו הזה.

Machine Translated by Google

מכיוון שהספירה (2)קטנה מאורך התווים ,(3)הספירה גדלה ל-3, ו- PlayBackNoteנקרא שוב.

.3בפעם השלישית שנקרא , PlayBackNoteספירה :3 =

Sound1.Source מוגדר לפריט השלישי בתווים, שהוא 6.wav.

Sound1.Play נקרא, מנגן את התו הזה.

כי הספירה (3)אינה קטנה מאורך התווים ,(3)שום דבר אחר קורה, וההשמעה הושלמה.

הערה למרות הרקורסיה היא רבת עוצמה, היא גם יכולה להיות מסוכנת. בתור ניסוי מחשבתי, שאל את עצמך מה היה קורה אם המתכנת שכח להכניס את הבלוקים ב- PlayBackNoteשהגדילו את הספירה.

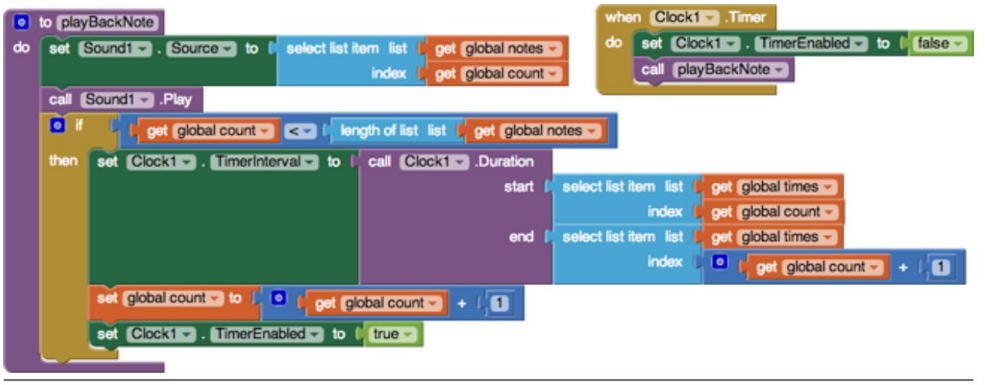
למרות שהרקורסיה נכונה, יש בעיה שונה עם הקודמת
דוגמה: כמעט לא עובר זמן בין שיחה אחת ל- Sound1.Playלמשנהו, כך שכל צליל מופרע על ידי התו הבא, מלבד האחרון. אף הערה (חוץ מהאחרונה) אינה רשאית להסתיים לפני שהמקור של Sound1משתנה ו- Sound1.Playנקרא שוב. כדי להשיג את ההתנהגות הנכונה, עלינו ליישם השהייה בין שיחות ל- .PlayBackNote

השמעת הערות עם עיכובים מתאימים

ניישם את ההשהיה על ידי הגדרת הטיימר על השעון לפרק הזמן בין ההערה הנוכחית להערה הבאה. לדוגמה, אם התו הבא מנוגן 3,000אלפיות שניות 3)שניות) לאחר התו הנוכחי, נגדיר את Clock1.TimerIntervalל-3,000, ולאחר מכן יש לקרוא ל-PlayBackNote שוב .

בצע את השינויים המוצגים באיור 14-9בגוף ה- ifבלוק ב- PlayBackNoteוצור ו- lllבמטפל האירועים , Clock1.Timerהמפרט מה צריך לקרות כשהטיימר כבה.

Machine Translated by Google



איור 9-15. הוספת השהיה בין ההערות

איך הבלוקים עובדים

נניח את התכנים הבאים עבור שתי הרשימות:

- הערות: 1.wav, 3.wav, 6.wav
- שעות: 12:00:00, 12:00:01, 12:00:04

כפי שמראה איור 9-14, PlayButton.Click מגדיר את הספירה ל-1 וקורא ל- PlayBackNote.

1. בפעם הראשונה שנקרא PlayBackNote , ספירה = 1:

Sound1.Source מוגדר לפריט הראשון בתווים, שהוא "1.wav".

Sound1.Play נקרא, מנגן את התו הזה.

מכיוון שהספירה (1) קטנה מאורך התווים Clock1.TimerInterval (3), מוגדר לכמות הזמן בין הפריט הראשון (12:00:00) לשני בזמן 1 (12:00:01) שְׁנִיָה. הספירה מוגברת ל-2. שעון1. טיימר מופעל ומתחיל בספירה לאחור.

שום דבר אחר לא קורה במשך שנייה אחת, אז שעון1. טיימר פועל, השבתה זמנית של הטיימר וקריאת PlayBackNote.

2. בפעם השנייה שנקרא PlayBackNote , count = 2:

Sound1.Source מוגדר לפריט השני בתווים, שהוא "3.wav".

Sound1.Play נקרא, מנגן את התו הזה.

מכיוון שהספירה (2) קטנה מאורך התווים Clock1.TimerInterval (3), מוגדר לכמות הזמן בין הפריט השני (12:00:01) לשלישי בזמנים 3 (12:00:04) שניות. הספירה מוגברת ל-3 . שעון1. טיימר מופעל ומתחיל בספירה לאחור.

Machine Translated by Google

שום דבר אחר לא קורה במשך 3שניות, כאשר השעון .1טיימר פועל,
השבתה זמנית של הטיימר וקריאת PlayBackNote.

.3בפעם השלישית שנקראPlayBackNote , ספירה :3 =

Sound1.Source מוגדר לפריט השלישי בתווים, שהוא ."6.wav"

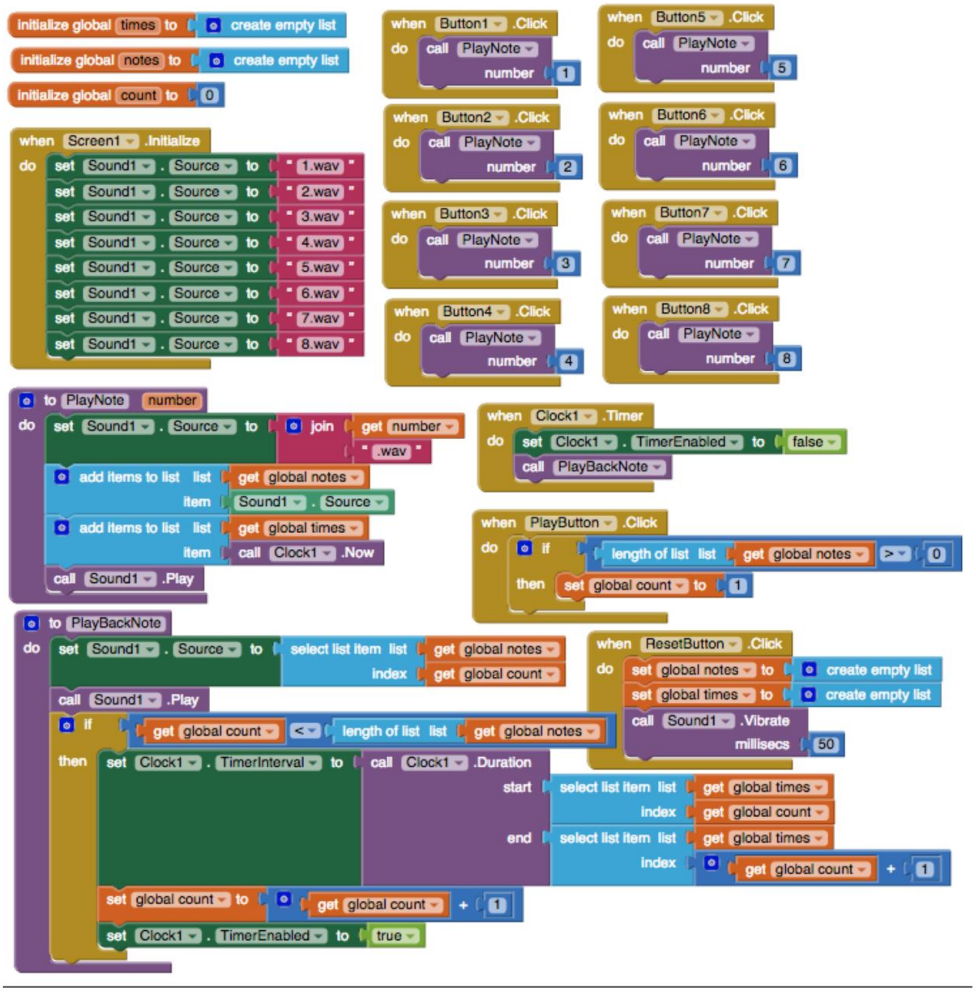
Sound1.Play נקרא, מנגן את התו הזה.

מכיוון שהספירה (3)אינה קטנה מאורך התווים ,(3)שום דבר אחר לא קורה. ההשמעה הושלמה.

האפליקציה השלמה: קסילופון

איור 9-15מציג את תצורת הבלוק הסופי עבור אפליקציית Xylophone.

Machine Translated by Google



איור .16-9הבלוקים לקסילופון

וריאציות

הנה כמה תרחישים חלופיים שכדאי לחקור:

•נכון לעכשיו, אין שום דבר שימנע ממשתמש ללחוץ על ResetButtonבמהלך ההשמעה, מה שיגרום לתוכנית לקרוס. (אתה יכול להבין למה?) שנה את כפתור ההפעלה. לחץ על כך שישבית את כפתור איפוס. כדי להפעיל אותו מחדש כשהשיר הושלם, שנה את בלוק if- וב nottuByalP .לחץ לבלוק , if else והפעל מחדש את ResetButtonבחלק . else

•באופן דומה, המשתמש יכול כעת ללחוץ על לחצן ההפעלה בזמן ששיר כבר מופיע משחק. (האם אתה יכול להבין מה יקרה?) הפוך את זה PlayButton.Click-ל

Machine Translated by Google

משבית את PlayButtonומשנה את הטקסט שלו "Playing..."-לאתה יכול להפעיל אותו מחדש ולאפס את הטקסט בבלוק , ifelseכמתואר בתבליט הקודם.

•הוסף כפתור עם שם של שיר, כגון "Für Elise". אם המשתמש לוחץ עליו, אכלס את רשימות ההערות והזמנים בערכים המתאימים, הגדר את הספירה ל-1 והתקשר ל- PlayBackNote. כדי להגדיר את הזמנים המתאימים, תמצא את הבלוק Clock1.MakeInstantFromMillisשימושי.

•אם המשתמש לוחץ על תו, הולך ועושה משהו אחר, ואז חוזר שעות לאחר מכן ולוחץ על תו נוסף, התווים יהיו חלק מאותו שיר, וזה כנראה לא מה שהמשתמש התכוון אליו. שפר את התוכנית על ידי (1הפסקת ההקלטה לאחר מרווח זמן סביר כלשהו, כגון דקה; או, (2 הגבלה על משך הזמן המשמש עבור Clock1.TimerIntervalעל ידי שימוש בבלוק המקסימלי ממגירת המתמטיקה.

•ציין חזותית איזה תו מתנגן על ידי שינוי מראה הלחצן -לדוגמה, על ידי שינוי הטקסט, צבע הרקע או צבע הקדמי שלו.

סיכום

הנה כמה מהרעיונות שכיסינו במדריך זה:

•ניתן לנגן קבצי שמע שונים מרכיב סאונד בודד על ידי שינוי מאפיין המקור שלו. זה איפשר לנו לקבל רכיב סאונד אחד במקום שמונה. רק הקפד לטעון את הצלילים בעת האתחול כדי למנוע עיכובים (איור .(9-6

•רשימות יכולות לספק לתוכנית זיכרון, עם תיעוד של פעולות המשתמש המאוחסנות ברשימה ומאוחר יותר אוחזרו ועובדו מחדש. השתמשנו בפונקציונליות זו כדי להקליט ולהשמיע שיר.

•ניתן להשתמש ברכיב השעון כדי לקבוע את השעה הנוכחית. מְחַסֵּר שני ערכי זמן נותנים לנו את משך הזמן בין שני אירועים.

•אתה יכול להגדיר את המאפיין TimerIntervalעבור Clockבתוך התוכנית, כגון איך אנחנו מגדירים אותו למשך הזמן בין התחלות של שני תווים.

•לא רק שאפשר אלא לפעמים רצוי שהליך יתקשר לעצמו. זוהי טכניקה רבת עוצמה הנקראת רקורסיה. בעת כתיבת פרוצדורה רקורסיבית, ודא שיש מקרה בסיס שבו ההליך מסתיים, במקום לקרוא לעצמו, אחרת התוכנית תעבור בלולאה אינסופית.