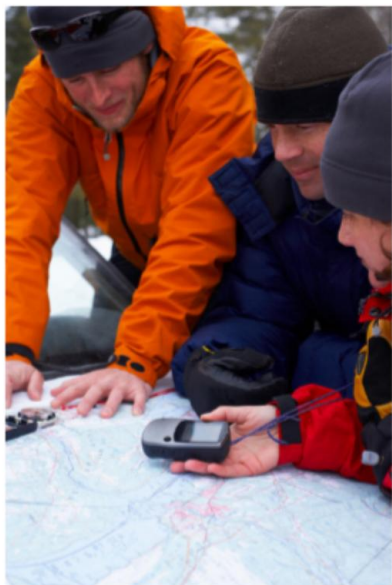


קריאה ותגובה לחיישנים

איור 23-1



כוון את הטלפון שלך לשמיים, Google Sky Map-ו יגיד לך באילו כוכבים אתה מסתכל. הטה את הטלפון שלך, ותוכל לשלוט במשחק שאתה משחק. קח את הטלפון שלך לריצה היומית שלך, ואפליקציה מתעדת את המסלול שלך. כל האפליקציות הללו אפשריות מכיוון שלמכשירים הניידים שאנו נושאים יש חיישני היי-טק לזיהוי המיקום, הכיוון וההאצה שלנו.

בפרק זה, תחקור את רכיבי OrientationSensor, LocationSensor, AccelerometerSensor-ו-App Inventor LocationSensor, תלמדו על מערכת המיקום הגלובלית (GPS); מדדי התמצאות כגון גובה, גלגול ואזימוט; וקצת מתמטיקה לעיבוד קריאות מד תאוצה.

יצירת מודעות למיקום

אפליקציות

עד לפופולאריזציה של הטלפון החכם, המחשוב היה בנעילת שולחן העבודה. כן, מחשבים ניידים הם ניידים, אבל לא באותו מובן כמו המכשירים הזעירים שאנו סוחבים כעת בכיסים שלנו. המחשוב עזב את המעבדה ואת המשרד, וכעת הוא מתרחש בעולם, מעבר למגבלות של ארבעה קירות.

אפקט אחד משמעותי של נשיאת המחשוב שלנו איתנו הוא חדש, מעניין מאוד פיסת נתונים עבור כל אפליקציה: מיקום נוכחי. לדעת היכן אנשים נמצאים בזמן שהם מסתובבים בעולם יש השלכות מרחיקות לכת ופוטנציאל לעזור לנו מאוד בחיינו. יש לזה גם פוטנציאל לפלוש לפרטיות שלנו ולהוות פגיעה באנושות.

האנדרואיד, איפה המכונת שלי? אפליקציה (פרק 7) היא דוגמה לאפליקציה מודעת למיקום המספקת הטבה אישית. זה מאפשר לך לזכור מיקום קודם כדי שתוכל לחזור אליו במועד מאוחר יותר. האפליקציה הזו פרטית, כלומר פרטי המיקום שלך מאוחסנים רק במסד הנתונים של המכשיר שלך.

קבוצות יכולות גם להשתמש בחיישן מיקום. לדוגמה, קבוצת מטיילים עשויה לרצות עקבו אחר מקום הימצאו של זה במדבר, או שקבוצה של שותפים עסקיים עשויים לרצות למצוא אחד את השני בכנס גדול (או בר). יש אנשים שמשתמשים באפליקציות "צ'ק-אין" כאלה מדי יום.

סוג אחר של אפליקציה מודעות למיקום משתמש בכלי מציאות רבודה. אפליקציות אלה משתמשות המיקום שלך והכיוון של הטלפון כדי לספק מידע שכבת-על שמגביר את ההגדרה הטבעית. אז אולי תכונן טלפון לבניין ותראה את המחיר שלו בשוק הנדל"ן, או שאתה יכול ללכת ליד צמח אקזוטי בגן בוטני ואפליקציה יכולה לספר לך על המינים שלו.

מערכת המיקום הגלובלית

כדי ליצור אפליקציה מודעת למיקום, תחילה עליך להבין כיצד פועלת מערכת המיקום הגלובלית (GPS). נתוני GPS נוצרים באמצעות סדרה של לוויינים גיאוסטנדרניים המתחזקים על ידי ממשלת ארצות הברית. כל עוד יש לך קו ראייה ללא הפרעה לשלושה לוויינים לפחות במערכת, הטלפון שלך יכול לקבל קריאה. קריאת GPS מורכבת מקו הרוחב, קו האורך והגובה שלך. קו הרוחב הוא מרחק צפון או דרום ביחס לקו המשווה, כאשר ערכי צפון חיוביים ודרום שליליים. הטווח הוא 90-עד 90. איור 23-1 מציג מפת גוגל של נקודה ליד קיטו, אקוודור. קו הרוחב המוצג במפה הוא 0.01- בקושי דרומית לקו המשווה!



איור 23-2. אקוודור, נמצאת על קו המשווה

קו אורך הוא כמה רחוק אתה ממזרח או מערב מהמרידיאן הראשי; לקואורדינטות מזרח יש ערכים חיוביים וקואורדינטות מערב הן שליליות. פריים מרידיאן, המשתרע מצפון לדרום, ממוקם בגריניץ', עיירה ליד לונדון המהווה את ביתו של מצפה הכוכבים המלכותי, הארגון שבמקור הקים את פריים מרידיאן כבסיס למדידה לאסטרונומים ולנוטים כאחד. המפה באיור 23-2 מציגה את גריניץ' וקו האורך שלה 0.0.



איור 23-3. מצפה הכוכבים המלכותי בגריניץ' יורה קרן אור לאורך מרידיאן פריים

ערכי קו האורך נעים בין 180-ל-081. איור 23-3 מציג נקודה ברוסיה, מאוד קרוב לאלסקה, שיש לה קו אורך 180.0 אפשר לומר שמיקום כזה נמצא באמצע העולם מגריניץ' (קו אורך 0.0).

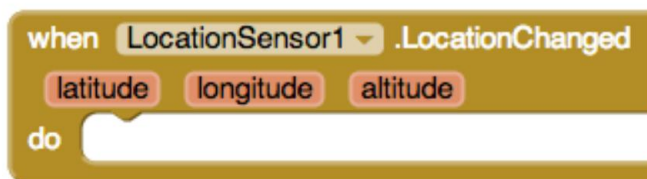


איור 23-4. לנקודה ליד הגבול בין רוסיה לאלסקה יש קו אורך 180

חישה מיקום עם ממצאי אפליקציה

App Inventor מספק את רכיב ה- `LocationSensor` לגישה למידע GPS. לרכיב יש מאפיינים עבור קו רוחב, קו אורך וגובה. זה גם מתקשר עם מפות Google, שתוכל לקבל קריאה עבור כתובת הרחוב הנוכחית שלך.

`LocationSensor.LocationChanged` בתמונה באיור 23-4, הוא המטפל באירועי מפתח עבור `LocationSensor`.

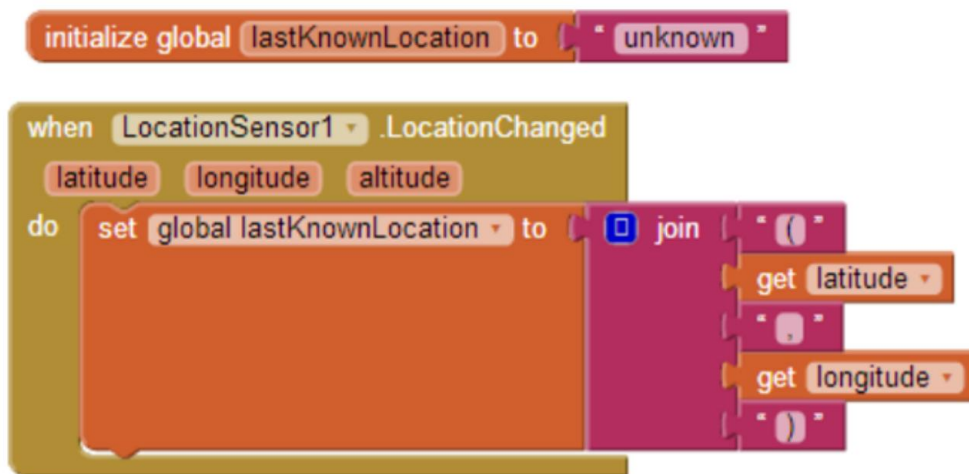


איור 23-5. המטפל באירועים `LocationSensor1.LocationChanged`

אירוע זה מופעל בפעם הראשונה שהחיישן יוצר קריאה ובכל פעם אחר כך שהטלפון מזיז מספיק כדי שמידע חדש ייקרא. לעתים קרובות יש עיכוב של לא מעט שניות לפני הקריאה הראשונה של אפליקציה, ולפעמים המכשיר לא יכול לקבל קריאה כלל. לדוגמה, אם אתה בתוך הבית ולא מחובר, WiFi-לייטכן שהמכשיר לא יקבל קריאה. לטלפון שלך יש גם הגדרות שבאמצעותן תוכל להפעיל קריאת GPS כדי לחסוך בחיי סוללה; זוהי סיבה אפשרית נוספת לכך שהרכיב לא יכול לקבל קריאה. מסיבות אלו, אל תניח שה- `LocationSensor`

לנכסים יש הגדרה חוקית עד לאירוע `LocationSensor.LocationChanged` מתרחשת.

אחת הדרכים להתמודד עם הלא ידועים בחישה מיקום היא ליצור משתנה `lastKnownLocation` אתו, "unknown"-לולאחר מכן בקש מהמטפל באירועים `LocationSensor.LocationChanged` לשנות את הערך של אותו משתנה, כפי שמוצג באיור 23-5.



איור 23-6. הערך של המשתנה `lastKnownLocation` משתנה בכל פעם שהמיקום משתנה

על ידי תכנות המטפל באירועים `LocationSensor.LocationChanged` באופן זה, אתה תמיד יכול להציג את המיקום הנוכחי או להקליט אותו במסד נתונים, כאשר "לא ידוע" מופיע עד הקריאה הראשונה. אסטרטגיה זו משמשת בתוכנית ללא הודעות טקסט בזמן נהיגה! אפליקציה (פרק 4) האפליקציה הזו מגיבה אוטומטית להודעות SMS וכוללת "לא ידוע" או את הקריאה האחרונה שנלקחה בתגובה.

אתה יכול גם לשאול במפורש אם לחיישן יש קריאה באמצעות `LocationSensor.HasLongitudeLatitude` מאיור 23-6 בלוק



איור 23-7. בדיקה אם לחיישן יש קריאה באמצעות בלוק `HasLongitudeLatitude`

בדיקת גבולות

שימוש נפוץ אחד באירוע `LocationChanged` הוא לבדוק אם המכשיר נמצא בתוך גבול, או אזור מוגדר. לדוגמה, קחו בחשבון את הקוד באיור 23-7, המרעיד את הטלפון בכל פעם שקריאה חדשה מראה שאדם התרחק מ-1.0 מעלות קו אורך מהפריים מרידיאן.



איור 23-8. קריאה אינה קרובה לפריים מרידיאן, הטלפון רוטט

לבדיקת גבולות כזו יש יישומים רבים; למשל, אזהרת שחרורים על תנאי אם הם מתקרבים למרחק שנקבע בחוק מביתם, או התראה להורים או למורים אם ילד עוזב את אזור מגרש המשחקים. אם תרצה לראות דוגמה קצת יותר מורכבת, ראה את הדיון בפרק 18 על בלוקים מותנים.

ספקי מידע על מיקום: GPS, WIFI ומזהה סלולרי

מכשיר אנדרואיד יכול לקבוע את המיקום שלו במספר דרכים. השיטה המדויקת ביותר - בטווח של מטרים ספורים - היא באמצעות לווייני ה-GPS. עם זאת, לא תקבל קריאה אם אתה בפנים או שיש גורדי שחקים או מכשולים אחרים סביבך; אתה צריך נתיב ברור לפחות לשלושה לוויינים במערכת.

אם GPS אינו זמין או שהמשתמש השבית אותו, המכשיר יכול לקבל את מיקומו דרך רשת אלחוטית. אתה חייב להיות ליד נתב WiFi, וקריאת המיקום שתקבל הוא קו הרוחב/קו האורך של אותה תחנת WiFi. דרך שלישית שבה מכשיר יכול לקבוע מיקום היא באמצעות מזהה תא. מזהה סלולרי מספק מיקום הטלפון בהתבסס על עוצמת האותות ממגדלי הטלפון הסלולרי הסמוכים. זה בדרך כלל לא מאוד מדויק אלא אם כן יש לך מגדלים סלולריים רבים בקרבתך. עם זאת, הוא אכן משתמש בכמות הסוללה הנמוכה ביותר בהשוואה לקישוריות GPS או WiFi.

שימוש בחיישן הכיוון

אתה יכול להשתמש ב-`OrientationSensor` עבור אפליקציות דמויות משחק בהן המשתמש שולט בפעולה על ידי הטיית המכשיר. זה יכול לשמש גם כמצפן כדי לגלות לאיזה כיוון (צפון/דרום, מזרח/מערב) הטלפון מכוון.

ל- OrientationSensor יש חמישה מאפיינים, שכולם אינם מוכרים לרוב האנשים מלבד מהנדסי אווירונאוטיקה:

גלגול (שמאל-ימין)

הגלגול הוא 0 מעלות כשהמכשיר מיושר, עולה ל-09 מעלות כשהמכשיר מוטה לכיוון הצד השמאלי, ויורד ל-09 מעלות כשהמכשיר מוטה לכיוון הצד הימני.

גובה גובה (מעלה-גב)

גובה הצליל הוא 0 מעלות כשהמכשיר מיושר, עולה ל-09 מעלות כשהמכשיר מוטה כך שהחלק העליון שלו מצביע כלפי מטה, ועולה עוד יותר ל-081 מעלות כשהמכשיר מתהפך. באופן דומה, כשהמכשיר מוטה כך שהחלק התחתון שלו מצביע כלפי מטה, Pitch-היורד ל-09 מעלות ולאחר מכן יורד ל-081 מעלות כאשר הוא מופנה עד הסוף. על.

אזימוט (מצפן)

אזימוט הוא 0 מעלות כאשר החלק העליון של המכשיר מצביע לצפון, 90 מעלות כאשר הוא מצביע מזרחה, 180 מעלות כאשר הוא מצביע לדרום, ו-072 מעלות כאשר הוא מצביע מערבה.

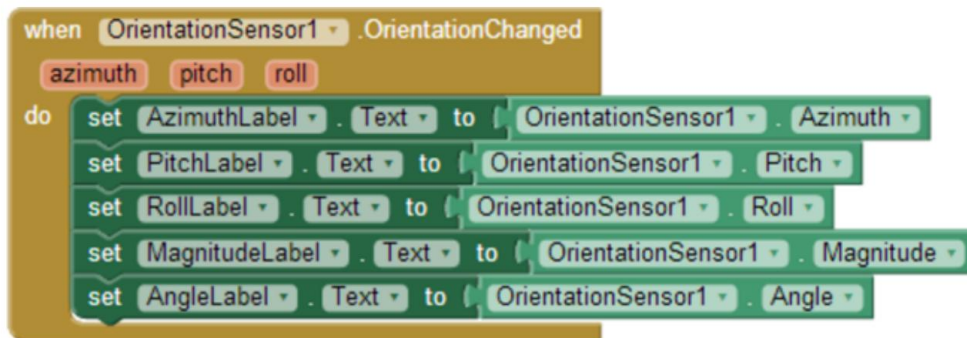
גודל (מהירות של כדור מתגלגל)

גודל מחזיר מספר בין 10-1 המציין עד כמה המכשיר מוטה. ערכו מציין את הכוח שמפעיל כדור שמתגלגל על פני המכשיר.

זווית (זווית של כדור מתגלגל)

זווית מחזירה את הכיוון בו מרצפת המכשיר. כלומר, הוא מציין את כיוון הכוח שיופעל על ידי כדור שמתגלגל על פני המכשיר.

OrientationSensor מספק את האירוע , OrientationChanged המופעל בכל פעם שהכיוון משתנה. כדי לחקור את המאפיינים הללו עוד יותר, בואו נכתוב אפליקציה שממחישה כיצד המאפיינים משתנים כשהמשתמש מטה את המכשיר. פשוט הוסף חמש תוויות כותרות וחמש תוויות אחרות כדי להציג את הערכים הנוכחיים של המאפיינים ברשימה הקודמת. לאחר מכן, הוסף את הבלוקים המוצגים באיור 8-23.



איור 23-9. חסימות להצגת נתוני OrientationSensor

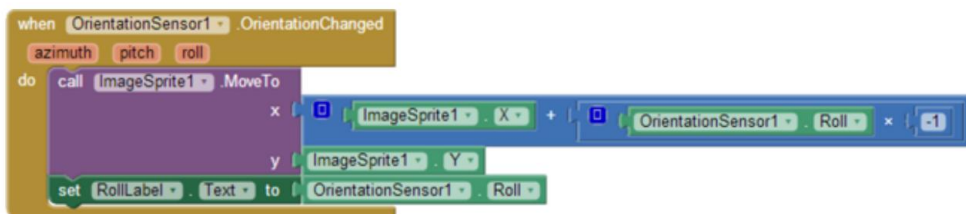
שימוש בפרמטר הגלגול כדי להזיז אובייקט

הפעם, בואו ננסה להזיז תמונה שמאלה או ימינה על המסך בהתבסס על הטיית המשתמש, כפי שאתם עשויים לעשות במשחק יריות או נהיגה. גרור החוצה קנבס והגדר את הרוחב "מלא הורה" ואת הגובה ל-002 פיקסלים. לאחר מכן, הוסף ImageSprite או Ball בתוך Canvas, והוסף תחתיו תווית בשם RollLabel כדי להציג ערך מאפיין, כפי שמוצג באיור 23-9.



איור 23-10. משחק משתמש לחקירה כיצד ניתן להשתמש בגלגול כדי להזיז תמונה

המאפיין Roll של OrientationSensor יציין אם הטלפון מוטה שמאלה או ימינה - אם תחזיק את הטלפון זקוף ותטה אותו מעט שמאלה, תקבל קריאה חיובית עבור הגלילה; אם תטה אותו מעט ימינה, תקבל קריאה שלילית. לכן, אתה יכול לאפשר למשתמש להזיז אובייקט עם מטפל באירועים כמו זה שמוצג באיור 23-10.



איור 23-11. תגובה לשינויים במאפיין Roll עם האירוע OrientationChanged

הבלוקים מכפילים את הגלילה ב-1, מכיוון שהטיה שמאלה נותנת גלגול חיובי וצריכה להזיז את האובייקט שמאלה (ובכך להקטין את קואורדינטת ה-x). לסקירה של אופן הפעולה של מערכת הקואורדינטות באפליקציות מונפשות, ראה פרק 17.

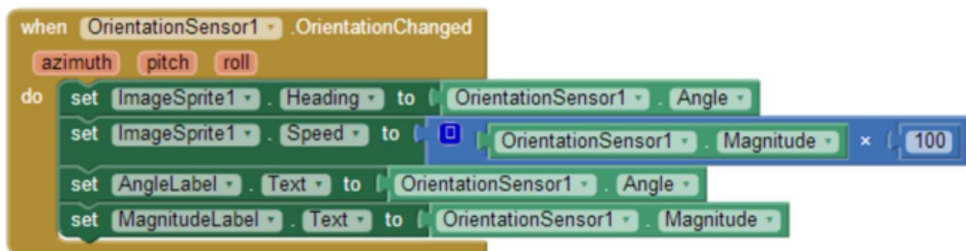
שימו לב שהאפליקציה הזו פועלת רק כשהמכשיר במצב פורטרט (זקוף), לא במצב נוף. כפי שהוא, אם תטה את הטלפון יותר מדי, המסך ישתנה למצב נוף והתמונה תישאר מרוחקת בצד שמאל של המסך. הסיבה היא שאם המכשיר נמצא על הצד, הוא מוטה שמאלה וכך תמיד יקבל קריאה חיובית לגלגול. קריאת גלגול חיובית, כפי שמוצגת בבלוקים באיור 23-10, תקטין את קואורדינטת ה-x.

שימו לב ש-ppA Inventor מספק את המאפיין Screen.ScreenOrientation, שבו אתה יכול להשתמש כדי לנעול את הכיוון אם אתה לא רוצה שהוא יעבור בין המצבים.

תנועה לכל כיוון על ידי שימוש בכותרת ובגודל

הדוגמה בסעיף הקודם מזיזה את התמונה שמאלה או ימינה. אם אתה רוצה לאפשר תנועה בכל כיוון, אתה יכול להשתמש במאפייני הזווית והגודל של ה-OrientationSensor. המאפיינים המשמשים להזזת פרת משה רבנו במשחק המתואר בפרק 5.

באיור 23-11, ניתן לראות את הבלוקים של אפליקציית בדיקה שבה המשתמש מטה את המכשיר כדי להזיז דמות לכל כיוון (צריך שתי תוויות וספרייט תמונה לדוגמה זו).



איור 23-12. הזזת דמות באמצעות זווית וגודל

נסה את זה. המאפיין, Magnitude ערך בין 1-70, מציין כיצד במידה רבה המכשיר מוטא. באפליקציית בדיקה זו, התמונה זו מהר יותר ככל שערך הגודל עולה.

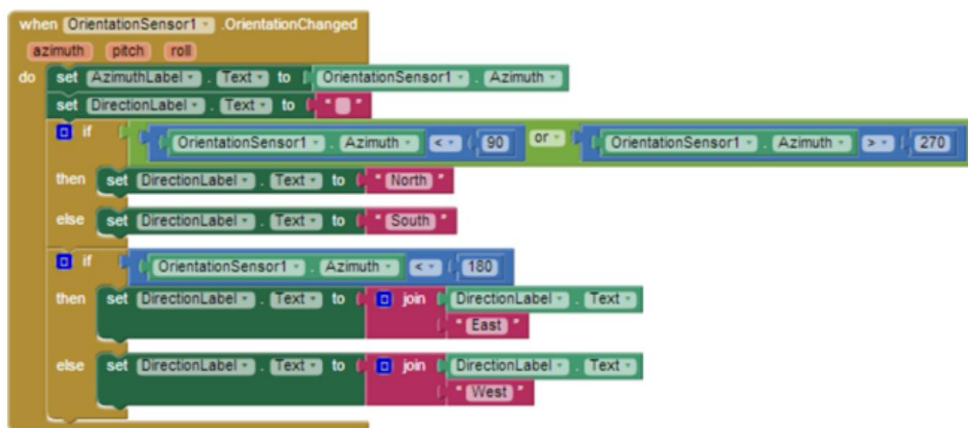
שימוש בטלפון כמצפן

אפליקציות ואפליקציות מצפן כמו Google Sky Map צריכות לדעת את כיוון הטלפון בעולם, מזרח/מערב וצפון/דרום. מפת השמיים משתמשת במידע כדי לכסות מידע על קבוצות הכוכבים שאליהן הטלפון מצביע.

קריאת אזימוט שימושית עבור סוג זה של אוריינטציה. אזימוט הוא תמיד בין 0-360 מעלות, כאשר 0 הוא צפון; 90 מזרח; 180 דרום; ו-270 מערב. לפיכך, קריאה של 45 פירושה שהטלפון מצביע לצפון מזרח, 135 אומר דרום מזרח, 225 אומר דרום מערב ו-315 אומר צפון מערב.

הבלוקים באיור 12-23 מיועדים למצפן פשוט שמציג בטקסט אשר הכיוון שאליו הטלפון מצביע (למשל, צפון מערב).

כפי שאולי שמתם לב, הבלוקים מציגים רק אחת מארבע אפשרויות: צפון מערב, צפון מזרח, דרום מערב ודרום מזרח. כאתגר, בדוק אם אתה יכול לשנות אותו כך שיראה רק כיוון בודד (צפון, דרום, מזרח או מערב) אם הקריאה מציינת שאתה מצביע בטווח של כמה מעלות ממנו.



איור 13-23. תכנות מצפן פשוט

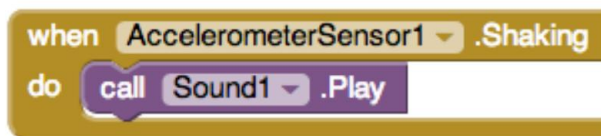
שימוש במד התאוצה

תאוצה היא קצב השינוי של המהירות לאורך זמן. אם אתה לוחץ את הרגל על דוושת הדלק של המכונית שלך, המכונית מאיץ -מהירותה עולה בקצב מסוים. מד תאוצה כמו זה שבמכשיר האנדרואיד שלך מודד תאוצה, אבל מסגרת ההתייחסות שלו היא לא המכשיר במנוחה, אלא המכשיר בנפילה חופשית: אם אתה

זרוק את הטלפון, הוא ירשום קריאת תאוצה של 0.0 במילים פשוטות, הקריאות לוקחות בחשבון את כוח המשיכה.
אם אתה רוצה לדעת יותר על הפיזיקה של העניין, תצטרך להתייעץ הספרים שלך הקשורים לאיינשטיין. אבל בחלק זה, נחקור את מד התאוצה מספיק כדי להתחיל. אפילו נבחן אפליקציה שיכולה לעזור להציל חיים!

תגובה לרעד של המכשיר

אם השלמת את אפליקציית Hello Purr בפרק 1, כבר השתמשת בחיישן האצה. באפליקציה הזו השתמשת באירוע Accelerometer.Shaking כדי לגרום לקיטי מiao כשהטלפון נער, כפי שמוצג באיור 13-23.



איור 14-23 השמעת צליל כאשר הטלפון רועד

שימוש בקריאות של חיישן התאוצה

כמו שאר החיישנים, למד התאוצה יש אירוע למועד שינוי הקריאות, AccelerometerSensor.AccelerationChanged. לאירוע הזה יש שלושה ארגומנטים התואמים לתאוצה בתלת מימד:

xAccel

חיובי כאשר המכשיר מוטה ימינה (כלומר, הצד השמאלי שלו מורם), ושלילי כאשר המכשיר מוטה שמאלה (הצד הימני שלו מורם).

yAccel

חיובי כאשר החלק התחתון של המכשיר מורם, ושלילי כאשר החלק העליון שלו מורם.

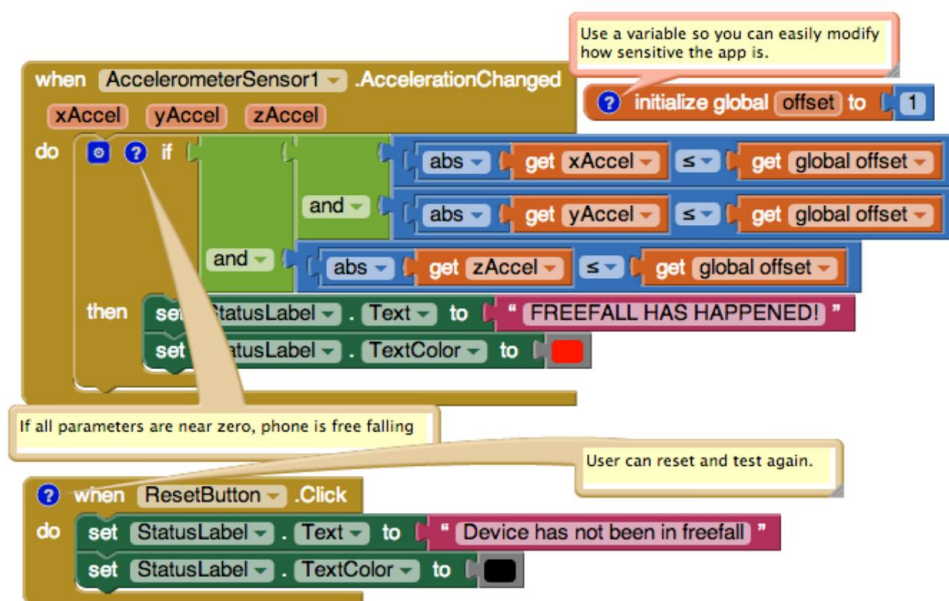
zAccel

חיובי כאשר תצוגת המכשיר פונה כלפי מעלה, ושלילי כאשר התצוגה פונה כלפי מטה.

זיהוי נפילה חופשית

אנו יודעים שאם כל קריאות התאוצה קרובות ל-0, המכשיר נופל חופשי לקרקע. עם זאת בחשבון, אנו יכולים לזהות אירוע נפילה חופשית על ידי בדיקת הקריאות באירוע AccelerometerSensor.AccelerationChanged. יכול להשתמש בלוקים כאלה, עם הרבה בדיקות, כדי לזהות מתי קשיש נפל ולשלוח אוטומטית הודעת SMS בתגובה.

איור 14-23 מציג את החסימות של אפליקציה שפשוט מדווחת שיש לנפילה חופשית התרחש (ומאפשר למשתמש ללחוץ על כפתור איפוס כדי לבדוק שוב).¹



איור 15-23 דיווח מתי התרחשה נפילה חופשית

בכל פעם שהחיישן מקבל קריאה, הבלוקים בודקים את מידות x , y , z ראה אם הם קרובים ל-0 (אם הערך המוחלט שלהם קטן מ-1). אם שלושתם קרובים ל-0, האפליקציה משנה תווית סטטוס כדי לציין שהטלפון נמצא בנפילה חופשית. כאשר המשתמש מקיש על כפתור האיפוס, תווית הסטטוס מאופסת למצבה המקורי ("המכשיר לא היה בנפילה חופשית").

סיכום

חיישנים מעוררים עניין רב באפליקציות לנייד מכיוון שהם מאפשרים למשתמשים שלך ליצור אינטראקציה אמיתית עם הסביבה שלהם. על ידי שימוש במחשוב נייד, אתה פותח עולם שלם של הזדמנויות בחוויות משתמש ופיתוח אפליקציות. עם זאת, תצטרך לחשוב היטב כיצד, היכן ומתי אתה משתמש בחיישנים באפליקציות שלך. לאנשים רבים יש חששות בפרטיות, וייתכן שהם לא ישתמשו באפליקציה שלך אם הם מודאגים לגבי מה שאתה עושה עם נתוני החיישנים שלהם. עוד,

¹ אתה יכול ללחוץ לחיצה ימנית על בלוק ולבחור "קלטות מוטבעות" כדי לשנות את האופן שבו בלוקים מופיעים. זה נעשה עבור הבלוקים בדוגמה זו כדי להקטין את הרוחב של המטפל באירועים.

עם כל האפשרויות במשחקים, רשתות חברתיות, נסיעות ועוד, האפשרויות להטמעות חיוביות
הן כמעט אינסופיות.

