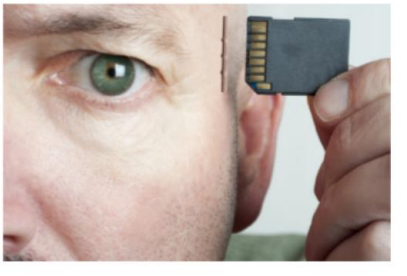
Machine Translated by Google

תכנות זיכרון האפליקציה שלך

איור .1-16



בדיוק כמו שאנשים צריכים לזכור דברים, כך גם אפליקציות. פרק זה בוחן כיצד ניתן לתכנת אפליקציה לזכור מידע.

כשמישהו אומר לך את מספר הטלפון של פיצה, המוח שלך מאחסן אותו בחריץ זיכרון. אם מישהו קורא כמה מספרים כדי שתוכל להוסיף, אתה מאחסן את המספרים ותוצאות הביניים בחריצי זיכרון. במקרים כאלה, אינך מודע לחלוטין לאופן שבו המוח שלך אוגר מידע או זוכר אותו.

לאפליקציה יש גם זיכרון, אבל הפעולות הפנימיות שלה הרבה פחות מסתוריות מאלה של המוח שלך. בפרק זה תלמדו כיצד להגדיר זיכרון של אפליקציה, כיצד לאחסן בו מידע וכיצד לאחזר מידע זה במועד מאוחר יותר.

בשם חריצי זיכרון

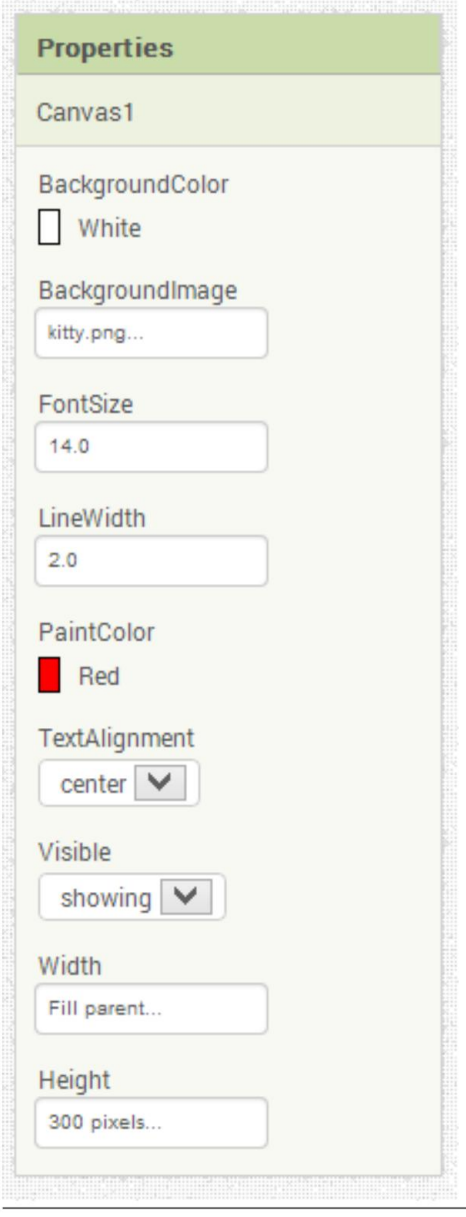
זיכרון של אפליקציה מורכב מקבוצה של חריצי זיכרון בעלי שם. חלק מחריצי הזיכרון הללו נוצרים כאשר אתה גורר רכיב לתוך האפליקציה שלך; חריצים אלו נקראים מאפיינים. אתה יכול גם להגדיר חריצי זיכרון בעלי שם שאינם משויכים לרכיב מסוים; אלה נקראים משתנים. בעוד שמאפיינים משויכים בדרך כלל למה שנראה באפליקציה, ניתן לחשוב על משתנים כזיכרון ה"שריטה" הנסתר של האפליקציה.

נכסים

המשתמש באפליקציה יכול לראות רכיבים גלויים כגון לחצנים, תיבות טקסט ותוויות. עם זאת, באופן פנימי, כל רכיב מוגדר לחלוטין על ידי קבוצה של מאפיינים. הערכים המאוחסנים בחריצי הזיכרון של כל מאפיין קובעים כיצד הרכיב מופיע.

אתה מגדיר את ערכי המאפיינים ישירות .Component Designer-בלדוגמה, איור 16-1מציג את הלוח לשינוי המאפיינים של רכיב . Canvas

Machine Translated by Google



איור 16-2. אתה יכול להגדיר מאפייני רכיב באתה מגדיר את הערכים ההתחלתיים של המאפיינים (הם לא מציגים את הערכים הנוכחיים בזמן שאפליקציה פועלת) ;Designer-

לרכיב Canvas יש מאפיינים רבים מסוגים שונים. למשל, ה
BackgroundColor- ו PaintColor הם חריצי זיכרון שמכילים צבע. תמונת הרקע מכילה שם פלמי (kitty.png). המאפיין Visible מכיל ערך בוליאני

Machine Translated by Google

ערך true)או ,falseתלוי אם התיבה מסומנת). חריצי הרוחב והגובה מכילים מספר או ייעוד מיוחד (למשל, "מילוי הורה").

כאשר אתה מגדיר מאפיין ,Component Designer-באתה מציין את האותיות הראשוניות ערך הנכס -ערכו כאשר האפליקציה מתחילה. ניתן גם לשנות ערכי נכסים כשהאפליקציה פועלת, עם בלוקים. עם זאת, הערכים המוצגים ,Component Designer-בכגון אלו באיור ,1-16אינם משתנים; אלה תמיד מציגים רק את הערכים ההתחלתיים. זה יכול להיות מבלבל כשאתה בודק אפליקציה -הערך הנוכחי של מאפייני האפליקציה אינו גלוי.

הגדרת משתנים

כמו מאפיינים, משתנים נקראים חריצי זיכרון, אך הם אינם משויכים לרכיב מסוים. אתה מגדיר משתנה כאשר האפליקציה שלך צריכה לזכור משהו שאינו מאוחסן במאפיין רכיב. לדוגמה, ייתכן שאפליקציית משחק תצטרך לזכור לאיזו רמה המשתמש הגיע. אם מספר הרמה עומד להופיע ברכיב , Labelייתכן שלא תזדקק למשתנה, כי אתה יכול פשוט לאחסן את הרמה במאפיין Textשל רכיב . Label

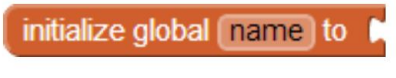
אבל אם מספר הרמה אינו משהו שהמשתמש יראה, תגדיר משתנה שבו לאחסן אותו.

חידון הנשיאים (פרק 8)הוא דוגמה נוספת לאפליקציה שצריכה משתנה.

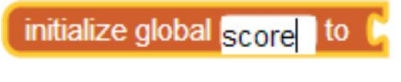
באותה אפליקציה, רק שאלה אחת של החידון צריכה להופיע בכל פעם בממשק המשתמש, ובכל זאת לחידון יש שאלות רבות (שרובן נשמרות מוסתרות מהמשתמש בכל עת). לפיכך, עליך להגדיר משתנה כדי לאחסן את רשימת השאלות.

בעוד שמאפיינים נוצרים באופן אוטומטי בעת גרירת רכיב לתוך המעצב, משתנים מוגדרים במפורש בעורך הבלוקים על ידי גרירת בלוק גלובלי לאתחל . אתה יכול לתת שם למשתנה על ידי לחיצה על הטקסט "שם" בתוך הבלוק, ותוכל לציין ערך ראשוני עבורו על ידי גרירת מספר, טקסט, צבע, או יצירת בלוק רשימה וחיבורו. להלן השלבים שאתה עושה. בצע כדי ליצור משתנה בשם ציון עם ערך התחלתי של :0

1. בבלוקים מובנים, פתח את מגירת המשתנים וגרור החוצה את הבלוק הגלובלי של האתחול .



2. שנה את שם המשתנה על ידי לחיצה על הטקסט "שם" והקלדה "ציון".



3. ממגירת המתמטיקה, גרור החוצה בלוק מספרים וחבר אותו לחלק הפתוח של ההגדרה המשתנה כדי להגדיר את הערך ההתחלתי.socket

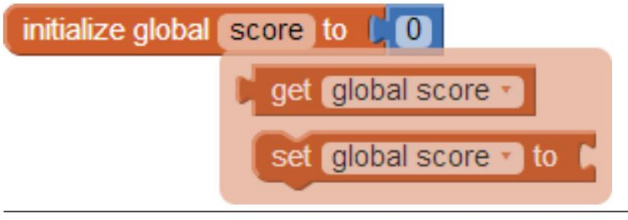
Machine Translated by Google

כאשר אתה מגדיר משתנה, אתה מורה לאפליקציה להגדיר חריץ זיכרון בשם לאחסון ערך. חריצי זיכרון אלה, כמו במאפיינים, אינם גלויים למשתמש כאשר האפליקציה פועלת.

בלוק המספרים שאתה מחבר מציין את הערך שיש למקם בחריץ כאשר האפליקציה מתחילה. מלבד אתחול עם מספרים או טקסט, אתה יכול גם לאתחל את המשתנה באמצעות יצירת רשימה או יצירת בלוק רשימה ריק . זה מודיע לאפליקציה שהמשתנה יאחסן רשימה של חריצי זיכרון במקום ערך בודד. למידע נוסף על רשימות, ראה פרק 19.

הגדרה וקבלת משתנה

כאשר אתה מגדיר משתנה, App Inventorיוצר עבורו שני בלוקים: הגדר וקבל . אתה יכול לגשת לבלוקים אלה על ידי ריחוף מעל שם המשתנה בבלוק האתחול, כפי שמוצג באיור 16-2.



איור 16-3. בלוק האתחול מכיל בלוקים של set and get עבור אותו משתנה

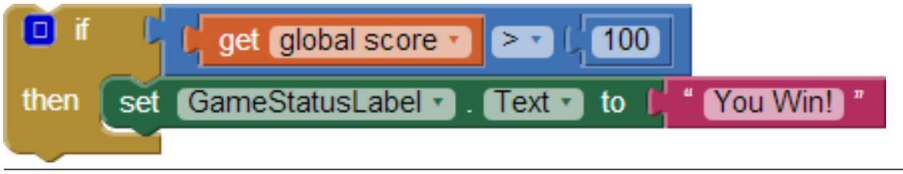
ההגדרה גלובלית לחסימה מאפשרת לך לשנות את הערך המאוחסן במשתנה. לדוגמה, בלוק המספרים באיור 16-3 מציב את הערך 5 בציון המשתנה. המונח "גלובלי" בציון הגלובלי שנקבע לחסימה מתייחס לעובדה שניתן להשתמש במשתנה בכל מטפלי האירועים והנהלים של התוכנית. עם הגרסה החדשה ביותר של App Inventor, אתה יכול גם להגדיר משתנים שהם "מקומיים" להליך מסוים או למטפל באירועים - כלומר, ניתן להשתמש במשתנים מקומיים רק על ידי הפרוצדורה או האירוע שאליהם הם משויכים (עוד על זה קצת מאוחר יותר בפרק).

איור 16-4. הצבת מספר 5 בציון המשתנה

אתה משתמש בבלוק שכותרתו קבל ציון גלובלי כדי לאחזר את הערך של משתנה. לדוגמה, אם אתה רוצה לבדוק אם הערך בתוך חריץ הזיכרון היה גדול מ

Machine Translated by Google

,100תחבר את הבלוק לקבל ציון גלובלי למבחן אם , כפי שמוצג באיור .16-4



איור .16-5שימוש בגוש הניקוד הגלובלי כדי לקבל את הערך המאוחסן במשתנה

הגדרת משתנה לביטוי

כפי שראית, אתה יכול להכניס ערכים פשוטים כמו 5למשתנה, אבל לעתים קרובות תגדיר את המשתנה לביטוי מורכב יותר (ביטוי הוא המונח של מדעי המחשב לנוסחה). לדוגמה, כאשר המשתמש לוחץ על Nextכדי להגיע לשאלה הבאה באפליקציית חידון, תצטרך להגדיר את המשתנה currentQuestionלאחד יותר מהערך הנוכחי שלו. כאשר מישהו מאבד עשר נקודות באפליקציית משחק, אתה צריך לשנות את משתנה הניקוד ל-01 פחות מהערך הנוכחי שלו. במשחק כמו MoleMash(פרק ,(3אתה משנה את המיקום האופקי (x)של השומה למיקום אקראי בתוך בד.

אתה תבנה ביטויים כאלה עם קבוצה של בלוקים שמתחברים לסט גלובלי לחסום.

הגדלת משתנה

אולי הביטוי הנפוץ ביותר הוא הגדלה של משתנה, או קביעת משתנה על סמך הערך הנוכחי שלו. לדוגמה, במשחק, כאשר שחקן קולע נקודה, ניתן להגדיל את הציון המשתנה ב-5. איור 16-5מציג את הבלוקים ליישום התנהגות זו.

איור .16-6הגדלת הציון המשתנה ב-5

אם אתה יכול להבין סוגים אלה של בלוקים, אתה בדרך להיות א
מְתַכנֵת. אתה קורא את הבלוקים האלה כ"קבע את הציון fve-ליותר ממה שהוא כבר", וזו דרך נוספת לומר להגדיל את המשתנה שלך. הדרך שבה זה עובד היא שהבלוקים מתפרשים מבפנים החוצה, לא משמאל לימין. לפיכך, החסימות הפנימיות ביותר -הציון הגלובלי לקבל והגוש מספר - 5מוערכים תחילה. לאחר מכן, הבלוק +מתבצע והתוצאה "מוגדרת" לציון המשתנה.

Machine Translated by Google

נניח שהיה 10בחריץ הזיכרון לציון לפני בלוקים אלה; ה
האפליקציה תבצע את השלבים הבאים:

.1אחזר את ה-01 מחריץ הזיכרון של הציון (הערך את בלוק .(get

.2הוסף לזה 5כדי לקבל .15

.3הכנס את התוצאה, ,15לתוך חריץ הזיכרון של הציון (בביצוע הסט).

בניית ביטויים מורכבים

במגירת Math, App Inventorמספק מגוון רחב של פונקציות מתמטיות הדומות לאלו שתמצא בגיליון אלקטרוני או במחשבון. ישנם אופרטורים אריתמטיים (למשל, ,(/,*,–,+ בלוקים להפקת ערכים אקראיים ואופרטורים כגון sqrt, cosinusוסינוס.

אתה יכול להשתמש בלוקים אלה כדי לבנות ביטוי מורכב ואז לחבר אותם בתור הביטוי בצד ימין של קבוצה גלובלית לחסימה. לדוגמה, כדי להעביר ספרייט תמונה לעמודה אקראית בתוך גבולות בד, תגדיר ביטוי המורכב מבלוק כפל ,(*)בלוק חיסור ,(–)מאפיין Canvas1.Width , מאפיין ImageSprite1.תכונת רוחב , ובלוק שבר אקראי , כפי שמוצג באיור 16-6.

איור .16-7אתה יכול להשתמש בלוקים במתמטיקה כדי לבנות ביטויים מורכבים כמו זה

כמו בדוגמה המצטברת בסעיף הקודם, הבלוקים מתפרשים על ידי האפליקציה בצורה מבפנים החוצה. נניח של- Canvasיש רוחב של 300ול- ImageSpriteיש רוחב של ,50 האפליקציה תבצע את השלבים הבאים:

.1אחזר את ה-003 וה-05 מחריצי הזיכרון עבור Canvas1.Widthו- ,ImageSprite.Width בהתאמה.

.2הורידו: 300 – 50 = 250.

.3קרא לפונקציית השבר האקראי כדי לקבל מספר בין 0ל-1 (נניח .(.5

.4הכפל: 250 * .5 = 125.

.5הכנס את ה-521 לתוך חריץ הזיכרון עבור המאפיין . ImageSprite1.X

Machine Translated by Google

הצגת משתנים

כאשר אתה משנה מאפיין רכיב, כמו בדוגמה הקודמת, ממשק המשתמש מושפע ישירות. זה לא נכון לגבי משתנים; לשינוי משתנה אין השפעה ישירה על מראה האפליקציה. אם רק תגדיל ציון משתנה אבל לא משנה את ממשק המשתמש בדרך אחרת, המשתמש לעולם לא ידע שיש שינוי. זה כמו עץ הפתגם שנפל ביער: אם אף אחד לא היה שם לשמוע את זה, האם זה באמת קרה?

לפעמים, אתה לא רוצה להפגין מיד שינוי בממשק המשתמש כאשר משתנה משתנה. לדוגמה, במשחק אתה עשוי לעקוב אחר סטטיסטיקות (למשל, זריקות שהוחמצו) שיופיעו רק כשהמשחק יסתיים.
זהו אחד היתרונות של אחסון נתונים במשתנה בניגוד למאפיין רכיב: אתה יכול להציג רק את הנתונים שאתה רוצה כאשר אתה רוצה להציג אותם.
אתה יכול גם להפריד את החלק החישובי של האפליקציה שלך ממשק המשתמש, מה שיקל על שינוי ממשק משתמש זה מאוחר יותר.
לדוגמה, עם משחק, אתה יכול לאחסן את הניקוד ישירות בתווית או במשתנה. אם אתה מאחסן אותו בתווית, תגדיל את מאפיין הטקסט של התווית כאשר נצברו נקודות, והמשתמש יראה את השינוי ישירות. אם אחסנת את הניקוד במשתנה והגדלת את המשתנה כאשר נצברו נקודות, תצטרך לכלול בלוקים כדי להעביר גם את הערך של המשתנה לתווית.

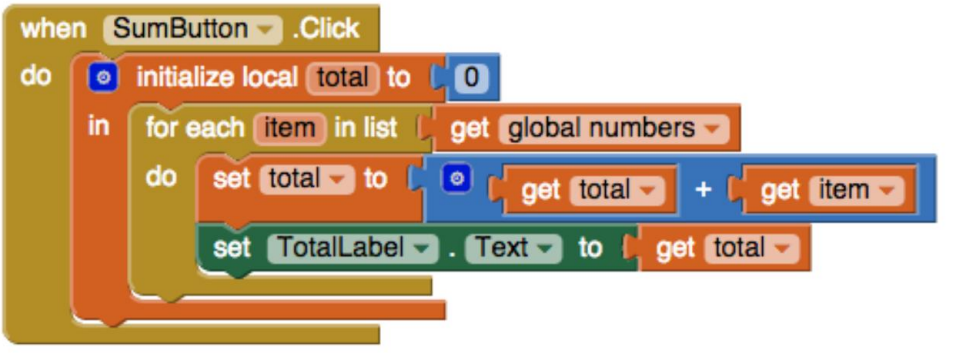
עם זאת, אם החלטתם לשנות את האפליקציה כדי להציג את הציון בצורה שונה, אולי עם סליידר, יהיה קל יותר לשנות את הפתרון המשתנה. לא תצטרך לשנות את כל המקומות שמשנים את הניקוד; תצטרך רק לשנות את הבלוקים שמציגים את הניקוד.

משתנים מקומיים

המשתנים שתוארו בפרק זה עד כה הם משתנים גלובליים ואתה מגדיר אותם עם אתחול גלובלי לחסימה. ה"גלובלי" מתייחס לעובדה שניתן להשתמש במשתנה בכל מטפלי האירועים והנהלים. אומרים למשתנים כאלה יש היקף גלובלי.

עם הגרסה העדכנית ביותר של App Inventor, כעת תוכל גם להגדיר משתנים מקומיים , כלומר משתנים שהשימוש בהם (ההיקף) מוגבל למטפל או פרוצדורה בודדים (ראה איור 16-7.).

Machine Translated by Google



איור 16-8. המשתנה "סך הכל" הוא מקומי; ניתן להשתמש בו רק באירוע SumButton.Click

אם המשתנה נחוץ רק במקום אחד, כדאי להגדיר אותו כמקומי, כמו
המשתנה "סך הכל" נמצא באיור 16-7. על ידי כך, אתה מגביל את התלות באפליקציה שלך
ומבטיח שלא תשנה בטעות משתנה. תחשוב על משתנה מקומי כמו הזיכרון הפרטי במוח
שלך - אתה בהחלט לא רוצה שלמוחים אחרים תהיה גישה אליו!

סיכום

כאשר אפליקציה מופעלת, היא מתחילה לבצע את פעולותיה ולהגיב לאירועים המתרחשים.
כאשר מגיבים לאירועים, האפליקציה צריכה לפעמים לזכור דברים. עבור משחק, זה עשוי להיות
הניקוד של כל שחקן או הכיוון שאליו חפץ נע.

האפליקציה שלך זוכרת דברים בתוך מאפייני רכיב, אבל כאשר אתה צריך חריצי זיכרון
נוספים שאינם משויכים לרכיב, אתה יכול להגדיר משתנים.
אתה יכול לאחסן ערכים במשתנה ולאחזר את הערך הנוכחי, בדיוק כמו שאתה עושה עם
מאפיינים.
בדומה לערכי מאפיינים, ערכי משתנים אינם גלויים למשתמש הקצה. אם אתה רוצה
שמשתמש הקצה יראה את המידע המאוחסן במשתנה, אתה מוסיף בלוקים שמציגים את
המידע הזה בתווית או ברכיב ממשק משתמש אחר.