

# فرم افزار های توسعه موبایل

دانشگاه فنی و حرفه‌ای امام خمینی (ره) قاین



# بازم

- ارتباط (برای بیان سوالات از ساعت ۸ الی ۱۴/۳۰ و ۱۶ الی ۲۲ میتوانید در شبکه های اجتماعی ایتا، واتس آپ و یا از طریق پیامک با شماره زیر در تماس باشید)
- شماره تماس در شبکه های اجتماعی: ۰۹۱۱۳۱۹۰۴۶۶
- ایمیل: **Hrezapour.ir@Gmail.com**
- سایت: **www.HRezapour.ir**

• حضور در جلسات ۳ نمره

• انجام تمرینات ۳ نمره

• پروژه میان ترم ۷ نمره

• پروژه پایان ترم ۷ نمره



# فهرست

- SDK،NDK ،SDK
- محیطهای توسعه و ابزارهای هیبرید که نیازی به یادگیری زبان برنامه نویسی ندارند
- معرفی و بررسی ابزارهای توسعه نرم افزار موبایل App Inventor
- نحوه تعاملات با صفحه لمسی و آشنایی با رویدادها
- چپنشهها و ابزار Canvas
- انیمیشن
- تایمر و شمارنده
- کار با سنسورها
- ابزار بررسی صدا و دوربین
- GSP امکان اتصال به وبسرویسه



# ابزارهای همه منظوره

- این ابزارها شامل SDK یا همان Software Development Kits میباشند که شما را قادر به توسعه اپلیکیشن برای پلتفرم خاصی مانند اندروید، آی او اس یا ویندوز میکند. علاوه بر این IDE یا همان Integrated Development Environments نیز که روند کار را تسهیل میبخشد، در این دسته بندی قرار میگیرند.

- SDKهای نیتیو پلتفرم

- یک SDK معمولاً شامل سورس های اصلی کامپایل شده، پشتیبانی از لایبرری های نرم افزاری و سایر کامپوننت ها مانند مستندات، اپلیکیشن های نمونه و غیره میباشد. این ابزار برای سهولت بخشیدن به توسعه موبایل، طراحی و ساخته شده است. SDKهای پلتفرم را میتوان به عنوان شکل قابل استفاده ای از سورس کد تعریف کرد.



# SDK و NDK اندروید

- میتوانید Android SDK را از طریق این وبسایت دانلود کنید. تنها لازم است پکیج ابزارهای کامند لاین SDK را که با سیستم عامل شما هم خوانی دارند دانلود کرده و از SDK Manager موجود در آن برای دانلود سایر پکیج ها استفاده کنید. سپس میتوانید از این SDK برای توسعه اپلیکیشن با استفاده یا بدون استفاده از IDE بهره بگیرید.

- بر خلاف SDK که توسعه بر مبنای جاوا را تسهیل میبخشد، Android Native Development Kit یا NDK توسعه دهندگان را قادر به ساخت اپلیکیشن به زبان های سی و سی پلاس پلاس میکند که در صورت استفاده از لایبرری های خارجی که به زبان سی و سی پلاس پلاس نوشته شده اند، به کمک شما خواهد آمد. علاوه بر این میتوانید از عملکرد سریع و حداکثر بهره گیری از قابلیت های سخت افزارهای گوشی نیز اطمینان حاصل کنید. بنابراین اگر عملکرد و کارایی برای شما اهمیت ویژه ای دارد، بهتر است NDK را مورد استفاده قرار دهید.



# SDK آی او اس

- مشابه SDK اندروید، میتوانید بدون پرداخت هیچگونه هزینه ای SDK آی او اس را دانلود کنید، کافی است یک حساب کاربری در Apple Developer Center داشته باشید، البته برای اجرای اپلیکیشن هایی که با استفاده از این ابزار توسعه داده شده اند، باید در Developer Program پولی آی او اس ثبت نام کنید. با این حال باز هم میتوانید اپلیکیشن خود را روی امولاتور اجرا و تست کنید.
- وقتی آمادگی لازم برای عرضه نهایی اپلیکیشن را پیدا کردید، میتوانید در Developer Program عضو شده و اپلیکیشن را برای اپ استور منتشر سازید.

## • SDK ویندوز ۱۰

- مایکروسافت فرآیند توسعه اپلیکیشن برای دستگاه های مختلف مانند PC، موبایل، Xbox One و HoloLens را یکپارچه سازی کرده و برای این کار باید Universal Windows Platform یا به اختصار UWP را مورد استفاده قرار دهید. میتوانید Windows 10 SDK را از Microsoft Windows Dev Center دانلود کرده و برای توسعه اپلیکیشن های موبایل با استفاده از سایر ابزارها مانند امولاتور (برای تست اپلیکیشن) مورد استفاده قرار دهید.
- در صورت استفاده از UWP، این SDK شما را قادر به وارد کردن اپلیکیشن های اندروید، آی او اس و ویندوز فعلی در آخرین پلتفرم ویندوز ۱۰ میکند. این قابلیت بسیار خوب و کاربردی است و توسعه دهندگان را قادر به گسترش مخاطبین مینماید.



# IDE های کامل

• IDE ها در تسهیل روند کاری و بیشترین بهره گیری از SDK به شما یاری می‌رسانند. توسعه دهندگان با کمک این ابزارها قادر به صرفه جویی در وقت خود هستند و میتوانند از قابلیت هایی مانند تکمیل کد، دیباگ تعاملی و بیلد خودکار بهره مند شوند.

- اندروید استودیو
- اندروید استودیو برای ساخت اپلیکیشن های اندروید و توسط کمپانی گوگل عرضه شده است. قابلیت تکمیل کد، شبیه سازی سریع و واقعی گوشی و تسریع سیستم های بیلد از امکانات قدرتمند آن محسوب میشوند. میتوانید یک کد واحد را در چندین دستگاه مانند پوشیدنی ها، گوشی ها و تلویزیون های هوشمند استفاده کنید.
- آخرین SDK اندروید نیز همراه اندروید استودیو عرضه میشود و توسعه دهندگان میتوانند از همان روز اول کدنویسی اپلیکیشن خود را آغاز کنند. میتوانید نسخه های قدیمی تر SDK را با استفاده از SDK Manager موجود در IDE دانلود کنید. این SDK شامل اپلیکیشن های نمونه، قالب های کد و ویزاردهای پروژه نیز میباشد که انجام عملیات را برای توسعه دهندگان ساده تر میکند.



# Xcode

- باید با Xcode آشنا باشید که شامل ios SDK تعدادی از سایر ابزارهای کاربردی است. مشابه اندروید استودیو این ابزار نیز یک محیط کارآمد برای ساخت اپلیکیشن برای مک، آیفون، آیپد و اپل واچ را فراهم می آورد. قابلیت های Xcode شامل ویرایشگر سورس کد قدرتمند، interface builder پیش ساخته، امولاتور گوشی پیشرفته و مستندات آپدیت شده میباشد.
- البته انتخاب های دیگری پیش روی شما قرار ندارند و اگر میخواهید اپلیکیشن خود را در اپ استور کمپانی اپل عرضه کنید، باید از Xcode استفاده نمایید. Xcode تمامی عملیات مربوط به امضای اپلیکیشن و مدیریت developer-key و اعتبارسنجی اپلیکیشن را به عهده میگیرد، از این رو توسعه دهندگان میتوانند بر روی کار اصلی خود تمرکز بیشتری داشته باشند.





# ویژوال استودیو

- ویژوال استودیو IDE مورد استفاده برای توسعه دهندگان ویندوز محسوب میشود. این ابزار توسط مایکروسافت عرضه شده و میتوانید از پشتیبانی های کمپانی بهره مند شوید. با استفاده از ویژوال استودیو میتوانید از کد واحدی برای اپلیکیشن های دسکتاپ، موبایل و سایر دستگاه ها استفاده کنید.
- مشابه Xcode و اندروید استودیو، Windows 10 SDK نیز همراه با IDE ارائه میشود و ویژوال استودیو قابلیت هایی مانند Peek To Definition و GoTo پیشرفته و غیره را در اختیار توسعه دهندگان قرار میدهد.
- بسته به امکانات و بودجه، میتوانید از سه نسخه Community، Professional و Enterprise یکی را انتخاب کنید. ویژوال استودیو در دو نسخه ویندوز و مک در دسترس است و میتوانید از آن برای توسعه اپلیکیشن موبایل بعدی خود استفاده نمایید.
- ویژوال استودیو یک IDE بسیار خوب برای کدنویسی اپلیکیشن های کراس پلتفرم در جاوااسکریپت مثلا React Native به شمار میرود. Visual Studio Code گزینه بسیار خوبی برای توسعه React Native همراه با اکستنشن React Native tools محسوب میشود. با استفاده از آن میتوانید کامندهای React Native را از طریق GUI به اجرا درآورده و تکمیل کد هوشمند را برای React Native API مشاهده کنید و حتی اپلیکیشن های خود را در ویرایشگر دیباگ کنید.



# اپلیکیشن هیبریدی چیست ؟

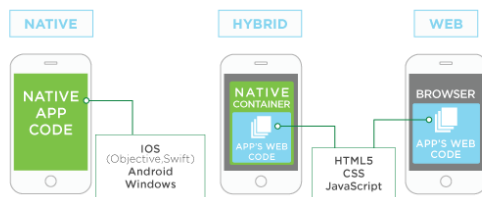
- یک اپ هیبریدی اساساً یک وب هست اما دارای یک قالب خفیف از اپ Native هست که بهش اجازه میده تا از ویژگی های بخصوصی از پلفرم Native و سخت افزار موبایل استفاده کنه. (برای مثال دوربین گوشی، تقویم، قابلیت push و ...) در صورتی که اپ های وب این دسترسی ها رو ندارند. اپ های هیبریدی مثل اپ های داخل وب از تکنولوژی های توسعه ی Front-end و زبان هایی مثل جاوا اسکریپت، HTML5 و CSS ساخته شدند و بهش قابلیت cross-platform داده شده.

## MOBILE APPS AT A GLANCE NATIVE VS. HYBRID VS. WEB APPLICATIONS

upwork

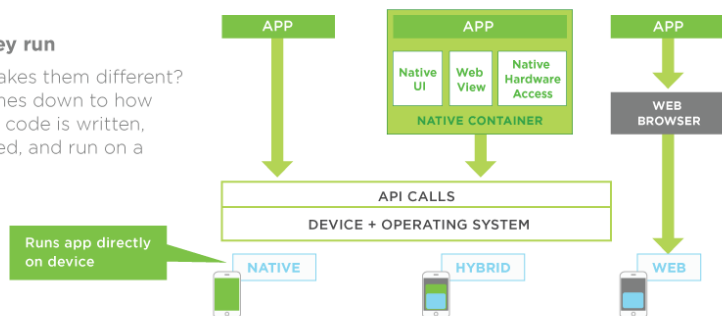
### How they're structured

When it comes to building your mobile app, you have three options: native, web, or hybrid.



### How they run

What makes them different? It all comes down to how an app's code is written, structured, and run on a device.



# مزایا و معایب اپ های هیبریدی

- شما فقط نیاز دارید که یک ورژن از اپلیکیشن خودتون بسازید. هیچ نیازی نیست که اپ رو در نسخه های مختلف طراحی کنید و توسعه بدید برای اینکه بخواد در یک پلتفرم اجرا بشه اما هنوز هم اپلیکیشنی رو دریافت می کنید که برای کاربران حس بومی بودن (Native داره). ساخت و نگهداری ورژن های مختلف از اپ Native شما نیاز به کار و تعداد توسعه دهندگان بیشتر داره. برای مثال یک کمپانی مثل فیسبوک که یک اپ بومی برای اندروید و iOS داره، صدها طراح و توسعه دهنده استخدام برای تیم اندروید و iOS استخدام می کنه. همواره آپدیت های جدید سیستم عامل ها رو در نظر میگیره و پشتیبانی و نگهداری اپ ها رو توسعه میده.
- همچنین یک اپ هیبریدی به توسعه دهندگان اجازه میده از مهارت هایی که دارند استفاده کنند – تکنولوژی های معمول مثل HTML5، CSS3 و جاوا اسکریپت. به همین دلیل این اپ ها برای توسعه دهندگان Front-end نیازی به یادگیری مجدد یک زبان جدید مثل Swift یا استفاده از ابزارهایی مثل کیت توسعه نرم افزار (SDK یا IDE ها که برای ساخت اپ های Native استفاده میشوند، ندارند.
- بنابراین اگر شما یک توسعه دهنده Front-end هستید میتونید اپ های بسیار زیبا و کارآمد برای موبایل بسازید طوری که حتی با اپ های Native رقابت کنه.
- با استفاده از این اپ ها شما برخی از مزایا رو قربانی cross-platform بودن و راحتی توسعه می کنید. اپ های Native برای یک پلتفرم خاص بهینه شدند و اجرای بهینه و خوبی در اون مورد دارند مخصوصا وقتی که دیتای سنگین یا گرافیک بالایی نیاز باشه. همچنین بدیل اینکه اپ های Native بصورت مستقیم نصب میشوند اون ها بهترین دسترسی رو به سخت افزار دارند. همه ی اینها مزایایی برای اپ های Native هستند اما فراموش نکنید اگر اپ های هیبریدی به خوبی طراحی شوند به سختی می توان تفاوت بین این دو را تشخیص داد.
- نکته : برای انتخاب نوع اپ حتما پیچیدگی اون رو در نظر بگیرید. معمولا برای اپ های ساده تر از اپ هیبریدی استفاده میشه و اپ هایی مثل بازی ها رو با Native می نویسند.



# ابزارهای محبوب برای ساخت اپ های هیبریدی

- از آنجایی که اپ های Native برای iOS و اندروید ایده آل هستند و برای پلتفرم خاص بهینه شدند، تکنولوژی های اپ های هیبریدی در حال گسترش اند و این باعث میشه یک گزینه قابل اعتماد و مناسب در هزینه و زمان برای توسعه دهندگان اپ موبایل باشه.
- فریمورک های اپ های هیبریدی موبایل کار رو راحت تر کردند، اون ها شامل کتابخانه هایی از کد، API ها و ویژگی های دیگری هستند که به سرعت و راحتی کدنویسی شما کمک می کنند.
- فریمورک هایی بر پایه HTML, JavaScript و CSS وجود دارند که درحقیقت کد رو به Native تبدیل می کنند.
- برخی از ابزارهای محبوب بر پایه وب برای اپ های هیبریدی
- (۱) PhoneGap یا Apache Cordova: یک فریمورک بر پایه وب و پلترم توسعه که به همراه API های جاوا اسکریپت آمده و به اپ های هیبریدی شما اجازه میده که در سطح دستگاه به API های سخت افزاری مثل دوربین، ویبره، تاچ اسکرین و ... دسترسی داشته باشید.



# ابزارهای محبوب برای ساخت اپ های هیبریدی

- Ionic Framework یک فریمورک رایگان و متن باز دیگه که برپایه وب هست و مزایای کامپوننت های بوت استرپ ۳ و AngularJS رو داره. این فریمورک دقیقا برای توسعه ی اپ های هیبریدی طراحی شده، از HTML, CSS و جاوا اسکریپت استفاده میکنه، تعداد زیادی ابزار و پلاگین داره و بسیار ساده و سریع برای یادگیری هست.



# ابزارهای محبوب برای ساخت اپ های هیبریدی

- Sencha Touch ۲: یک پلتفرم قدرتمند و سطح بالاست که از HTML5 و JavaScript و فریمورک ExtJS برای ساخت اپ های هیبریدی که بسیار شبیه Native هستند استفاده می کند. از مزایای دیگه ی اون میشه به واکنش گرا بودن، UI گرا بودن و مدل MVC اون نام برد.



# Sencha



# معرفی و کار با App Inventor

شروع کار عملی



**MIT**  
**APP INVENTOR**





# App Inventor

- یک گوشی هوشمند اندرویدی چه کارهایی می تواند انجام دهد؟
- در شکل زیر شما می توانید تعدادی از ویژگی های تلفن هوشمند که می توانید از طریق App Inventor روی آن ها کنترل داشته باشید، را ببینید. چیزی که وجود دارد، این است که برنامه نویسی یک دستگاه تک و منفرد که تمام این ویژگی ها را داشته باشد، قبلاً برای برنامه نویسان مبتدی امری غیرممکن بوده است.





# نصب App Inventor

- راه اندازی App Inventor
- App Inventor توسط گوگل توسعه داده شده و اکنون تحت مالکیت و توسعه داده شده توسط انستیتو فناوری ماساچوست (MIT) می باشد و می توان از طریق این وب سایت به آن دسترسی داشت: <Http://Appinventor.Mit.Edu/Explore>.
- شما برای استفاده از App Inventor به سه چیز نیاز دارید:
- یک کامپیوتر، با سیستم عامل ویندوز، مکینتاش یا لینوکس.
- یک اتصال اینترنت و مرورگر، مانند Chrome، Firefox یا Safari. پشتیبانی از Internet Explorer برای آینده طرح ریزی شده است.
- یک حساب گوگل، که رایگان بوده و در <https://accounts.google.com/signup> در دسترس می باشد.



- آیتم چهارمی که به شما توصیه می کنیم، داشتن یک گوشی یا تبلت اندرویدی می باشد، اگرچه در صورتی که یکی از آنها را ندارید، می توانید از یک امولاتور روی صفحه (شبیه ساز) استفاده کنید.

# اتصال به App Inventor

- گزینه ۱: استفاده از Wi-Fi با App Inventor همراه با نرم افزار تلفن
- اگر به Wi-Fi دسترسی دارید، خبر خوب این است که بدون اینکه مجبور به دانلود کردن هیچ برنامه ای برای کامپیوتر خود باشید، App Inventor را می توانید اجرا کنید.
- شما نیاز به نصب یک اپلیکیشن با نام Aicompanion روی گوشی خود دارید، که از وب سایت App Inventor و Google Play Store قابل دسترسی می باشد. و نیاز است که هم گوشی و هم کامپیوتر شما قادر به دسترسی به شبکه وای فای یکسانی باشند.
- قبل از اینکه شروع کنید، بهتر است نسخه های قبل App Inventor که روی سیستم خود دارید، را به طور کامل حذف کنید. سپس مراحل زیر را به منظور پیکربندی تلفن خود برای کار کردن با App Inventor انجام دهید:
- ۱- روی گوشی یا تبلت خود، به تنظیمات رفته، Applications و سپس Unknown Sources را انتخاب کنید. این گزینه به شما امکان استفاده از یک گوشی یا تبلت برای تست اپلیکیشن هایی که از Google Play Store دانلود نکرده اید، را می دهد.
- ۲- در صفحه Applications، گزینه Development را انتخاب کرده و هر دو گزینه USB Debugging و Stay Awake را انتخاب کنید. این گزینه مفید است تا هنگامی که به کامپیوتر خود متصل می شوید، روشن بماند، طوری که بدون اینکه مجبور به ضربه زدن به صفحه به طور دائم باشید، بتوانید برنامه نویسی کنید.



# اتصال به App Inventor

- ۳- اپلیکیشن Aicompanion را روی گوشی خود نصب کنید. لینک زیر به شما یک کد QR می‌دهد که می‌توانید برای دانلود اپلیکیشن رایگان، آن را اسکن کنید:

<Http://Mng.Bz/3ev2>.

- گزینه ۲: استفاده از امولاتور روی صفحه نمایش
- اگر نمی‌خواهید روی یک گوشی یا تبلت تست کنید، می‌توانید یک تلفن مجازی که امولاتور (Emulator، شبیه ساز) خوانده می‌شود، را روی صفحه نمایش کامپیوتر خود اجرا کنید. برای این کار، شما نیاز به نصب نرم افزار نصب App Inventor دارید.
- بعد از اینکه این نرم افزار نصب شد، هرزمان که بخواهید یک اپلیکیشن را تست کنید، باید برنامه Aistarter که به صورت خودکار به عنوان یک بخش از برنامه نصب، نصب خواهد شد) را اجرا کنید. در ادامه در مورد امولاتور بیشتر صحبت خواهیم کرد.



# اتصال به App Inventor

- قبل از اینکه شروع کنید، در صورت وجود داشتن نسخه‌های قبلی App Inventor روی سیستم خود، آن را به طور کامل حذف کنید. سپس مراحل زیر را انجام دهید:
- ۱- نرم افزار نصب App Inventor را نصب کنید. لینک زیر را دنبال کرده و به قسمت نصب برنامه برپا کردن App Inventor در صفحه بروید: [Http://Mng.Bz/Tk58](http://Mng.Bz/Tk58) سپس مراحل زیر را برای سیستم عامل خود دنبال کنید.
- 2- Aistarter را اجرا کنید. شما به این قسمت از نرم افزار برای ارتباط برقرار کردن بین کامپیوتر و امولاتور نیاز دارید. زمانی که نرم افزار نصب App Inventor را در مرحله ۱ نصب کردید، Aistarter نیز به صورت خودکار نصب شد. کاربران ویندوز و لینوکس نیاز دارند هرزمان که خواستند از App Inventor استفاده کنند، به طور دستی آن را یا از منوی Start یا منوی Programs اجرا کنند. کاربران مک آن را به صورت خودکار اجرا شده می بینند.



# اتصال به App Inventor

- گزینه ۳: اتصال از طریق کابل USB
- شما می توانید تلفن خود را از طریق یک کابل USB به طور مستقیم به کامپیوتر خود متصل کنید. این یک مزیت است که گوشی شما در حال کار کردن شارژ نیز شود. شما نیاز به نصب مشابه گزینه ۲ دارید، به اضافه اینکه نیاز به نصب راه انداز (Driver مناسب USB برای گوشی خود در کامپیوتری که استفاده می کنید، دارید.
- قبل از اینکه شروع کنید، بهتر است نسخه های قبلی App Inventor را از روی سیستم خود به طور کامل حذف کنید. سپس مراحل زیر را انجام دهید:
- ۱- برنامه نصب App Inventor را نصب کنید. لینک زیر را دنبال کرده کرده و به قسمت نصب برنامه برپا کردن App Inventor در صفحه بروید: [Http://Mng.Bz/3ev2](http://Mng.Bz/3ev2) سپس مراحل زیر را برای سیستم عامل خود دنبال کنید.
- ۲- اگر از یک کامپیوتر شخصی دارای ویندوز استفاده می کنید، شاید نیاز به نصب راه انداز (Driver برای گوشی یا تبلت اندرویدی خود داشته باشید. برنامه راه انداز، یک برنامه کوچک است که به کامپیوتر امکان "گفتگو" با وسایل سازگار مانند ماوس، صفحه کلید و گوشی های هوشمند را می دهد. برنامه نصب App Inventor شامل راه اندازهایی برای گوشی های رایج اندرویدی می باشد، اما اگر گوشی شما در لیست واقع در [Http://Mng.Bz/Yxxg](http://Mng.Bz/Yxxg) ظاهر نشد، نیاز به نصب یک راه انداز به صورت دستی خواهید داشت.



# اتصال به App Inventor

- ۳- گوشی خود را برای کار کردن با App Inventor به صورت زیر پیکربندی کنید:
  - a. روی گوشی یا تبلت خود، به قسمت تنظیمات و Applications رفته و Unknown Sources را انتخاب کنید. این گزینه به شما امکان استفاده از یک گوشی یا تبلت برای تست اپلیکیشن هایی که از Google Play Store دانلود نکرده اید، را می دهد.
  - b. در صفحه Applications، گزینه Development را انتخاب کرده و هر دو گزینه USB Debugging و Stay Awake را انتخاب کنید. این برای صفحه نمایش مفید است که هنگامی که به کامپیوتر خود متصل می کنید، روشن بماند، طوری که بتوانید بدون اینکه مجبور به ضربه زدن به صفحه به طور دائم باشید، برنامه نویسی کنید.
  - c. اپلیکیشن Aicompanion را روی گوشی خود نصب کنید. لینک زیر به شما یک کد QR می دهد که می توانید برای دانلود اپلیکیشن رایگان، آن را اسکن کنید:  
<Http://Mng.Bz/3ev2>.
- 4- Sistar را اجرا کنید. شما به این قسمت از نرم افزار برای ارتباط برقرار کردن بین کامپیوتر و تلفن از طریق USB نیاز دارید. زمانی که نرم افزار نصب App Inventor را در مرحله ۱ نصب کردید، Aistarter نیز به صورت خودکار نصب شد.



# طراحی اولین اپلیکیشن

## ۱- طراحی صفحه اپلیکیشن با استفاده از App Designer.

## ۲- دستور دادن به اپلیکیشن که چه کاری انجام دهد، از طریق برنامه نویسی Blocks Editor.

### ۳- تست برنامه خود با استفاده از گوشی یا امولاتور.

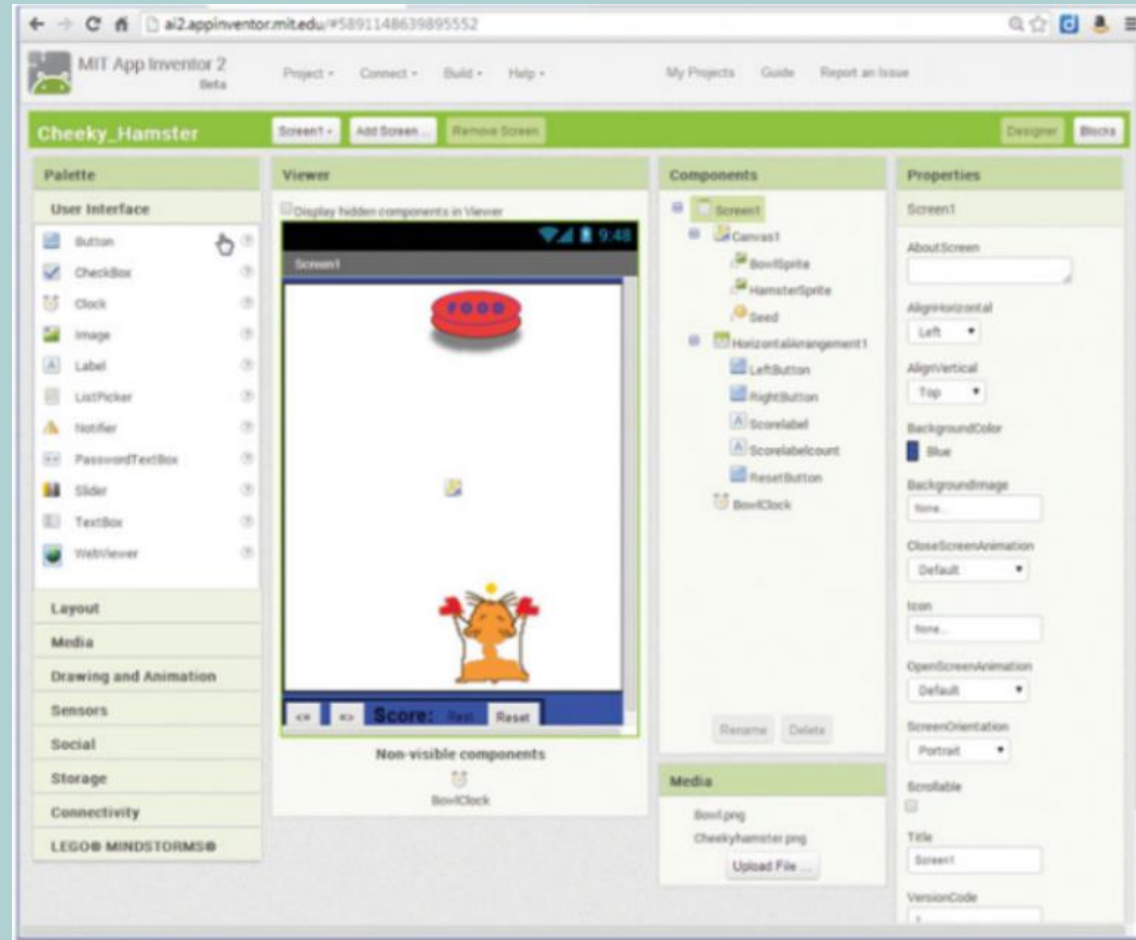
در زیر، این سه مرحله را می بینید.





# ۱. طراحی صفحه اپلیکیشن

در قسمت Designer برنامه App Inventor، یک پروژه جدید را شروع خواهید کرد. سپس صفحه اپلیکیشن را طراحی خواهید کرد (حتی اگر یک طرح ناقص باشد).





# Designer برنامه App Inventor به شما امکان می دهد:

یک پروژه جدید بسازید

● به آن کامپوننت (اجزاء) اضافه کنید

● نمای اپلیکیشن را طراحی کنید

شما از طریق مرورگر وب خود (مانند Chrome، Firefox یا Safari) کلیک کردن روی دکمه Designer، به Designer دسترسی خواهید داشت.

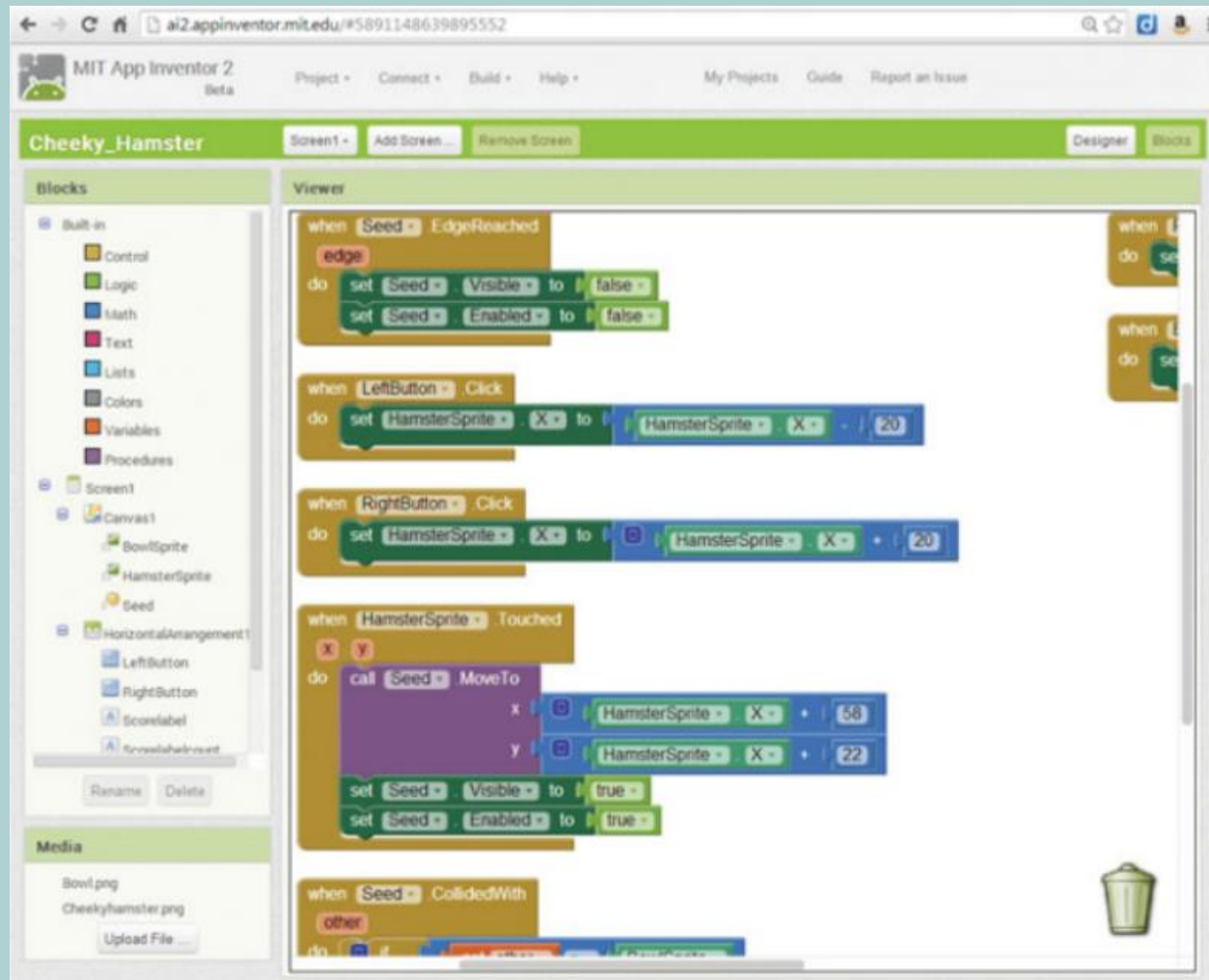
۲. دستور دادن به اپلیکیشن که چه کاری انجام دهد

سپس، شما Blocks Editor را برنامه نویسی خواهید کرد تا به اپلیکیشن بگویید چه کاری انجام دهد. Blocks Editor به شما امکان می دهد با استفاده از برنامه نویسی کردن بلوک ها، چگونگی کار کردن اپلیکیشن را کنترل کنید.



# Designer برنامه App Inventor به شما امکان می دهد:

شما با کلیک کردن روی دکمه Blocks از طریق مرورگر وب، اپلیکیشن‌های خود را می‌نویسید. در ضمن در هنگام برنامه نویسی می‌توانید بین Designer و Blocks جابه‌جا شوید.



## ۳. تست کردن برنامه

در آخر، برنامه را روی گوشی خود (نمایش داده شده در تصویر چپ) یا در امولاتور (تصویر سمت راست) تست خواهید کرد. شما می توانید با مشاهده کردن برنامه خود روی گوشی یا امولاتور، ببینید که به درستی کار می کند یا خیر.



## ۳. تست کردن برنامه

نکته: صفحه App Inventor دو دکمه در سمت راست دارد که به شما امکان می دهد بین Designer و Blocks Editor جابه جا شوید.

Blocks Editor یک واسط کشیدن و رهاکردنی ( Drag And Drop) می باشد، که به این معناست که شما می توانید بدون نیاز به یادگیری تعداد زیادی کلمه ناشناس، بی درنگ شروع به برنامه نویسی کنید.

صفحه Blocks Editor به دو بخش؛ یکی شامل تمام بلوک ها برای انتخاب کردن در سمت چپ و دیگری یک فضای خالی بزرگ در سمت راست تقسیم شده است.



# بلوک‌های Built-In

بلوک‌های Built-In

در جدول زیر، نگاهی اجمالی به هر کدام از ابزارهای دراور بلوک‌های Built-In خواهیم انداخت.

توضیح و مثال	بلوک‌هایی که در آن خواهید یافت	دراور بلوک‌ها
<p><b>Decision</b>: اگر روی دکمه happy کلیک کنید، تلفن یک نماد ☺ نمایش می‌دهد. اگر روی دکمه sad کلیک کنید، نماد ☹ نمایش داده می‌شود.</p> <p><b>Loop</b>: وقتی شما یک متن را برای گروهی از مخاطبین ارسال می‌کنید، تلفن یک حلقه را از اولین مخاطب تا آخرین آنها یا ارسال یک متن جداگانه در هر بار اجرا می‌کند. انجام دادن دستی آن خسته‌کننده خواهد بود!</p>	<p>بلوک‌های <b>Decision</b> که به برنامه می‌گویند که سپس و براساس تعدادی تست چه کاری انجام دهد</p> <p>بلوک‌های <b>Loops</b> که روش قدرتمندی از کارهای تکراری می‌باشند.</p>	Control
<p>مثال: تنظیم خصوصیت <b>Visible</b> یک دکمه با <b>False</b> دکمه را ناپدید می‌کند.</p> <p>تست‌های منطقی برای بلوک‌های تصمیم‌گیری پیچیده‌تر استفاده می‌شوند. برای مثال ممکن است بخواهید مطمئن شوید سن کاربر بیشتر از 8 و کمتر از 12 باشد.</p>	<p>مقادیر logic برابر است با <b>True</b> و <b>False</b></p> <p>تست منطقی: <b>NOT, OR, AND, =</b></p>	Logic
<p>برای مثال، در یک بازی شما می‌توانید از بلوک‌های <b>Math</b> برای اضافه کردن عدد سکه‌های جمع‌آوری شده و تبدیل آنها به یک امتیاز استفاده کنید. شما همچنین می‌توانید از بلوک‌های <b>Math</b> برای مشخص کردن اینکه این امتیاز از بالاترین امتیاز فعلی بیشتر شده یا نه، استفاده کنید.</p>	<p>بلوک‌هایی که اعداد را ساخته و مقایسه می‌کنند، به اضافه تمام نوعی که روی یک ماشین حساب می‌بینید</p>	Math
<p>هر قطعه متنی که می‌خواهید نمایش دهید، در یک بلوک <b>Text</b> ایجاد شده است. برای مثال، ممکن است نامی که یک کاربر نایب کرده را با یک پیام متنی ترکیب کنید، چیزی مانند "Hello Amy, how are you today?"</p>	<p>بلوک‌هایی که متون را ایجاد کرده، تغییر می‌دهد، اضافه می‌کند، متصل می‌کند، تقسیم کرده، جایگزین می‌کند و نیز مقایسه می‌کند.</p>	Text
<p>برای مثال ممکن است بخواهید به کاربر یک متو از انتخاب‌هایی که بتواند کلیک کند بدهید.</p>	<p>بلوک‌هایی که لیستی از اشیاء را ذخیره کرده، نمایش می‌دهد و با آنها کار می‌کند.</p>	Lists
<p>شما می‌توانید رنگ هر چیزی که روی صفحه می‌بینید، مانند متن، یک دکمه یا پس‌زمینه را تنظیم کرده یا تغییر دهید.</p>	<p>یک انتخاب از رنگ‌های استاندارد برای استفاده (شما همچنین می‌توانید رنگ‌های خود را با استفاده از نمودار رنگ ترکیب کنید، در <a href="http://mng.bz/qhY3">http://mng.bz/qhY3</a>)</p>	Colors
<p>یک متغیر، یک فضای ذخیره‌سازی نام‌دار برای یک قطعه کوچک از داده نظیر نام شما یا سن یا امتیاز بازی می‌باشد.</p>	<p>بلوک‌هایی برای ایجاد متغیرها</p>	Variables
<p>یک رویه، نوعی از برنامه کوچکی است که بلوک‌های دیگر می‌توانند شروع کنند. برای مثال، ممکن است یک رویه داشته باشید که هر زمان که بازی تمام شد، صفحه را ریست کند.</p>	<p>بلوک‌هایی برای ایجاد و اجرای رویه‌ها</p>	Procedures





# بلوک‌های Component-Specific

محتویات Blocks Editor در پاسخ به اجزایی که شما به Designer اضافه می کنید، تغییر می کند. بیایید یک رابط ساده با دو جزء بسازیم: یک دکمه با نام Pressbutton و یک صدا با نام Beepsound. Beepsound زمانی که به دراور (Screen1) که شامل بلوک‌های Component-Specific می باشد) در Blocks Editor نگاه کنید، درون آن دراورهای Beepsound و Pressbutton را خواهید دید. نکته: شما همیشه Screen1 را در Blocks Editor می بینید. این کامپوننت برای تمام اپلیکیشن‌ها وجود دارد، که یک صفحه خالی است و زمانی که اپلیکیشن را باز کنید، باز می شود.



## نکته آموزشی: انتزاع (Abstraction) چیست؟

انتزاع، یک روش ساده کردن برنامه نویسی می باشد. آنها به یک برنامه نویس امکان استفاده یا فراخوانی توابع سطح بالا را با استفاده از یک نام توصیفی مانند "Play Sound" می دهد. روش دیگر بیان آن این است که برنامه نویسان امکان برنامه نویسی با استفاده از زبانی که به "زبان طبیعی" آنها نزدیک تر است، را می دهد.

پشت بلوک ها، کدهای پنهان جاوا قرار دارد (جاوا یک زبان برنامه نویسی است) که شما می توانید از آنها چندین بار مجدداً استفاده کنید. اگر انتزاع بلوک ها نبود، هر زمانی که می خواستید یک صدا را پخش کنید، مجبور به نوشتن تعدادی کد جاوا مانند این بودید:



# نکته آموزشی: انتزاع (Abstraction) چیست؟

```
Import Java.Net.URL;
```

```
Import Javax.Swing.*;
```

```
Import Javax.Sound.Sampled.*;
```

```
Public Class Loopsound {
```

```
Public Static Void Main(String[] Args) Throws Exception {
```

```
URL Url = New URL(
```

```
    "Http://Pscore.Org/Media/Leftright.Wav");
```

```
Clip Clip = Audiosystem.Getclip();
```

```
// Getaudioinputstream() Also Accepts A File Or Inputstream
```

```
Audioinputstream Ais = Audiosystem.
```

```
Getaudioinputstream( Url );
```

```
Clip.Open(Ais);
```

```
Clip.Loop(Clip.LOOP_CONTINUOUSLY);
```

```
Swingutilities.Invokelater(New Runnable() {
```

```
Public Void Run() {
```

```
// A GUI Element To Prevent The Clip's Daemon Thread
```

```
// From Terminating At The End Of The Main()
```

```
Joptionpane.Showmessagedialog(Null, "Close To Exit!");
```

```
}
```

```
});
```





# محدودیت های امولاتور

امولاتور، یک برنامه است که روی کامپیوتر شخصی، مک، یا ماشین لینوکس شما اجرا شده و مانند گوشی شما عمل می کند. اما بعضی از ویژگی های گوشی های هوشمند روی امولاتور کار نمی کند. آنها شامل موارد زیر می باشند:

- فناوری صفحه لمسی (اگرچه شما می توانید از ماوس خود برای بعضی اعمال استفاده کنید)
- GPS
- پیام رسانی و مکالمه
- دوربین
- اسکن بارکد
- اتصال اینترنت
- پاسخ به چیزی که به صورت فیزیکی روی گوشی اتفاق افتاده است



# اولین اپلیکیشن شما: Hello World!

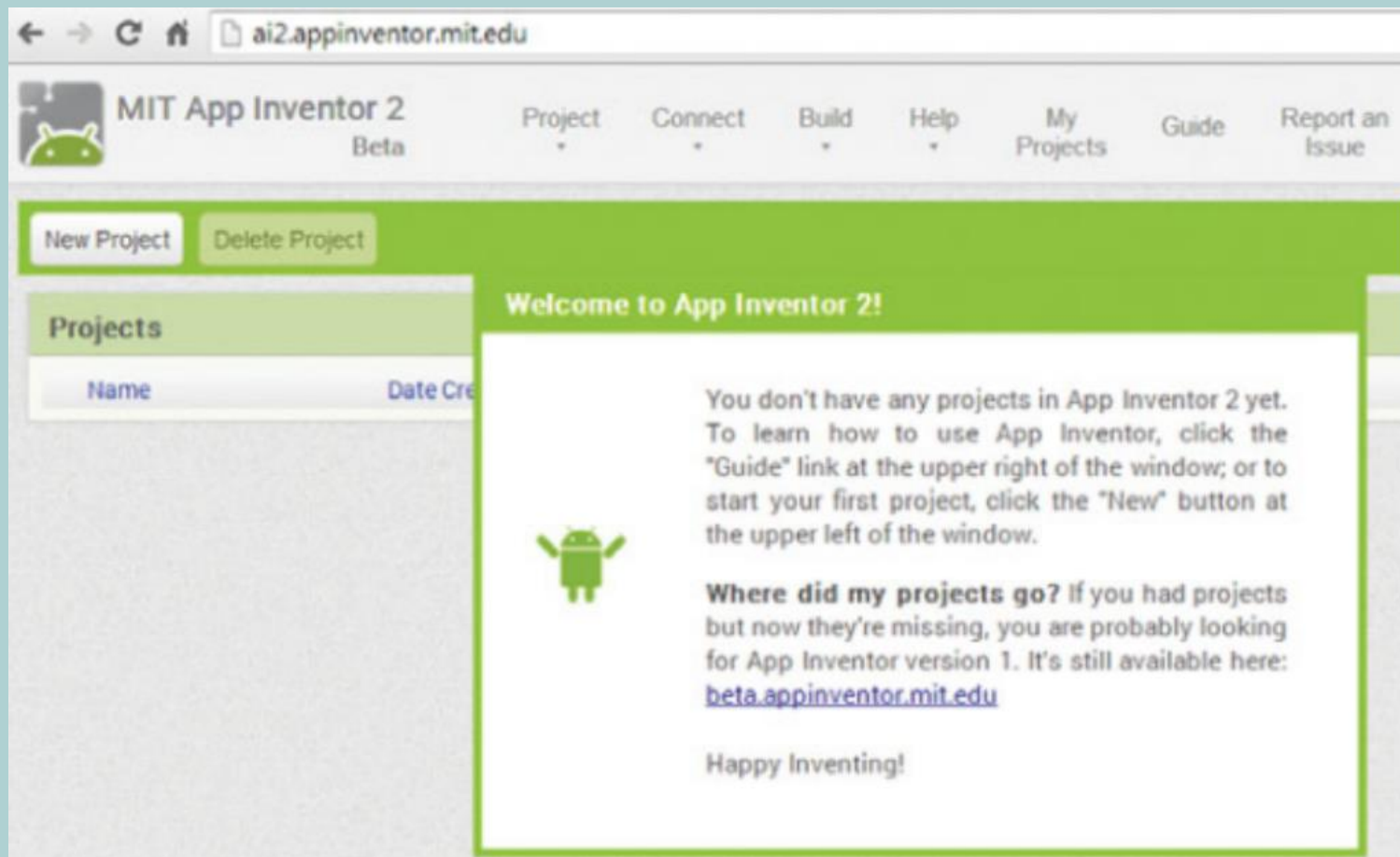
۱. باز کردن App Inventor

پس از اینکه App Inventor را روی کامپیوتر خود نصب کردید، بخش Create از وب سایت App Inventor ([Http://Appinventor.Mit.Edu](http://Appinventor.Mit.Edu)) را برای شروع انتخاب کنید. این کار، App Inventor Designer را باز می کند



# اولین اپلیکیشن شما: Hello World!

اگر تاکنون هیچ وقت از App Inventor استفاده نکرده اید، یک لیست خالی از پروژه‌ها و یک پیغام خوش آمدگویی مانند زیر را خواهید دید:



## ۲. شروع با یک پروژه جدید

نکته: در نام پروژه‌ها مجاز به قرار دادن هیچ فاصله ای نیستید، و قانون مشابه برای نام اجزاء ( Components) نیز برقرار است.

یک روش برای جدا کردن کلمات، بزرگ در نظر گرفتن حروف ابتدای هر کلمه است که برنامه نویسان به این روش، روش Camel Case (کوهان شتری) می گویند.



## ۲. شروع با یک پروژه جدید

پس از اینکه یک اپلیکیشن را ایجاد کرده یا باز کردید، صفحه طراحی (Design) باز می شود. این صفحه جایی است که شما اجزا را به اپلیکیشن خود اضافه کرده و ظاهر صفحه اپلیکیشن خود بر روی گوشی را طراحی می کنید. شما می توانید هر زمان که خواستید، با کلیک کردن روی لینک بالای صفحه به لیست اپلیکیشن های خود در My Projects برگردید.





### ۳. اضافه کردن یک جزء (Notifier) |خطاردهنده) به پروژه

۳. کار شما با Designer تمام شده. بنابراین با کلیک کردن روی دکمه Blocks. پنجره Blocks Editor را باز کنید.

شما می توانید از کامپوننت مفید Notifier برای تمام انواع پیغام ها و هشدارهای نمایش یافته استفاده کنید.

The screenshot shows the MIT App Inventor 2 web interface. The top bar includes the MIT App Inventor 2 logo and navigation links like Project, Connect, Build, and Help. Below the bar, there's a header for 'HelloWorld1' with buttons for 'Screen1', 'Add Screen', and 'Remove Screen'. The main workspace is divided into four panels: Palette, Views, Components, and Properties. The Palette panel on the left has a 'User Interface' section with various icons. The Views panel in the center shows a mobile screen preview. The Components panel on the right shows a list of components, with 'Notifier' selected. The Properties panel on the far right shows the settings for the selected 'Notifier' component, including 'BackgroundColor', 'Dark Gray', 'NotifierLength', 'Long', 'TextColor', and 'White'. Arrows point from the text instructions to specific parts of the interface: one from the 'Notifier' icon in the Palette to the 'Notifier' component in the Components panel, and another from the 'Notifier' component in the Components panel to the 'Notifier' component in the 'Non-visible components' section at the bottom of the Views panel.

۱. از پالت، کامپوننت Notifier را به Screen1 درگ کنید

۲. شما می توانید آیکن Notifier را ببینید که در بالا، در لیست کامپوننت ها و همچنین در پایین ظاهر شده است.

نکته: اگر نمی توانید Notifier را در پالت پیدا کنید، مطمئن شوید که در بخش User Interface به دنبال آن می گردید.



## ۴. نوشتن برنامه با استفاده از بلوک ها

معمولاً تا زمانی که کاربر کاری مانند کلیک کردن روی یک دکمه انجام ندهد، اپلیکیشن ها کاری انجام نمی دهند. این کار، یک رویداد خوانده می شود. بلوک های ویژه ای که یک رویداد را شناسائی کرده و سپس کاری انجام می دهند، اداره کننده های رویداد خوانده می شوند. شما می توانید یک اداره کننده رویداد را تشخیص دهید، زیرا دارای کلمه های When و Do در سمت چپ بلوک می باشند. پس از When، شما همیشه دو کلمه جدا شده با یک فاصله را می بینید: کلمه اول، نام کامپوننتی است که اداره کننده رویداد با آن کار می کند، و کلمه دوم، عملی است که اداره کننده رویداد در انتظار آن است (که همچنین یک Trigger یا محرک خوانده می شود). ممکن است پیچیده باشد، اما شرط می بندیم که شما می توانید بفهمید اداره کننده رویداد زیر به دنبال چه چیزی می باشد:

When Startbutton.Click

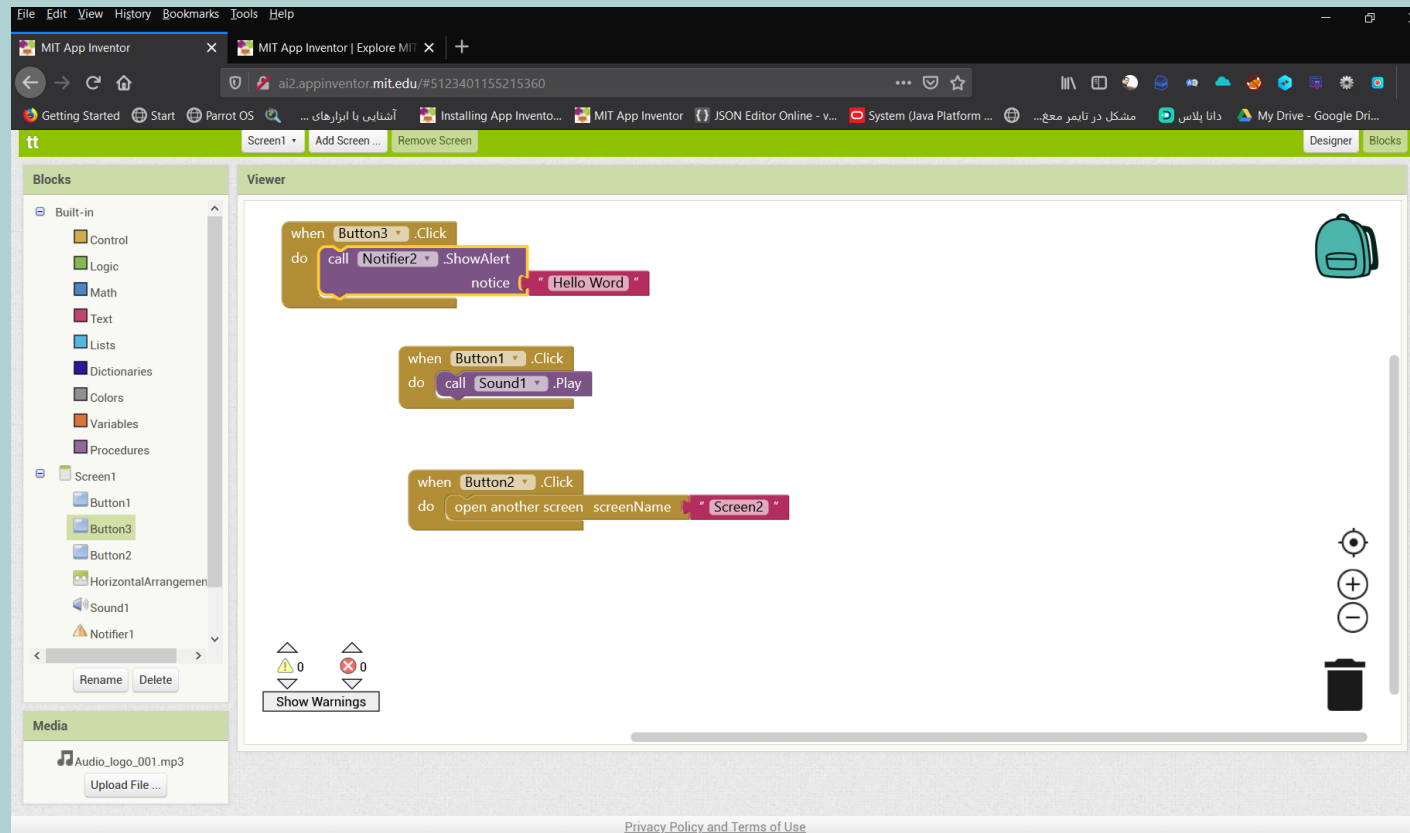
این اداره کننده رویداد در انتظار زمانی است که کاربر روی Startbutton کلیک کند، و زمانی که این کار را انجام داد، هرچه که در بخش Do بلوک قرار دارد، را اجرا می کند.

در این اپلیکیشن ساده، شما می خواهید به محض اینکه اپلیکیشن شروع شد، یک پیام ظاهر شود. از یک اداره کننده رویداد ویژه با نام Screen.Initialize برای انجام دادن این کار استفاده خواهید کرد. معنای آن این است که اپلیکیشن فوراً چیزی را انجام خواهد داد، حتی اگر کاربر یک دکمه را کلیک نکرده باشد.



## ۴. نوشتن برنامه با استفاده از بلوک ها

از اپلیکیشن می خواهید چه کاری انجام دهد؟ یک کادر هشدار ظاهر شود و پیغام "Hello World!" نشان دهد. تعداد زیادی از انواع مختلف (اکشن های ) Notifier (بلوک های بنفش) را خواهید دید. در این مورد، شما از یکی از آنها استفاده خواهید کرد که یک پیغام به کاربر نمایش می دهد و زمانی که کاربر روی دکمه OK کلیک کرد، ناپدید می شود. نام آن، Showmessagedialog بوده و یک پیام را به همراه یک دکمه نمایش می دهد (به دلیل اینکه بلوک های App Inventor دارای نام توصیفی می باشند، از نام Showmessagedialog نیز می توانید بفهمید چه کاری انجام می دهد).





## ۴. نوشتن برنامه با استفاده از بلوک ها

The screenshot shows the MIT App Inventor web interface. On the left is the 'Blocks' palette with categories like Control, Logic, Math, Text, Lists, Dictionaries, Colors, Variables, and Procedures. The 'Viewer' on the right shows a mobile app preview with a 'Press Button' button and a 'Notif' notification. The code blocks are as follows:

- when Screen2.Initialize**
  - do **call Notifier1.ShowDialog**
    - message: "سلام خوبی"
    - title: ""
    - buttonText: ""
- when Button1.Click**
  - do **open another screen** screenName: "Screen1"

At the bottom, there is a 'Show Warnings' button and a 'Media' section with an 'Audio\_logo\_001.mp3' file and an 'Upload File...' button.

The diagram illustrates the logic of the app. It shows the 'when Screen1.Initialize' block, which calls the 'Notifier1.ShowDialog' block. The 'Notifier1.ShowDialog' block has three inputs: 'message' (set to 'Hello World'), 'title' (set to 'I made this app!'), and 'buttonText' (set to 'OK'). The diagram also shows the 'when Screen2.Initialize' block, which calls the 'Notifier1.ShowDialog' block with the message 'سلام خوبی'.

۱. روی بلوک Text از بلوک های built-in کلیک کنید.

۲. سه بلوک Text را به شکاف های خالی درگ کنید.

شما می توانید یک رشته متنی را با کلیک کردن بین علائم نقل قول (کویشن مارک) تنظیم کنید.

سه رشته متنی را با آنچه در اینجا می بینید تغییر دهید.



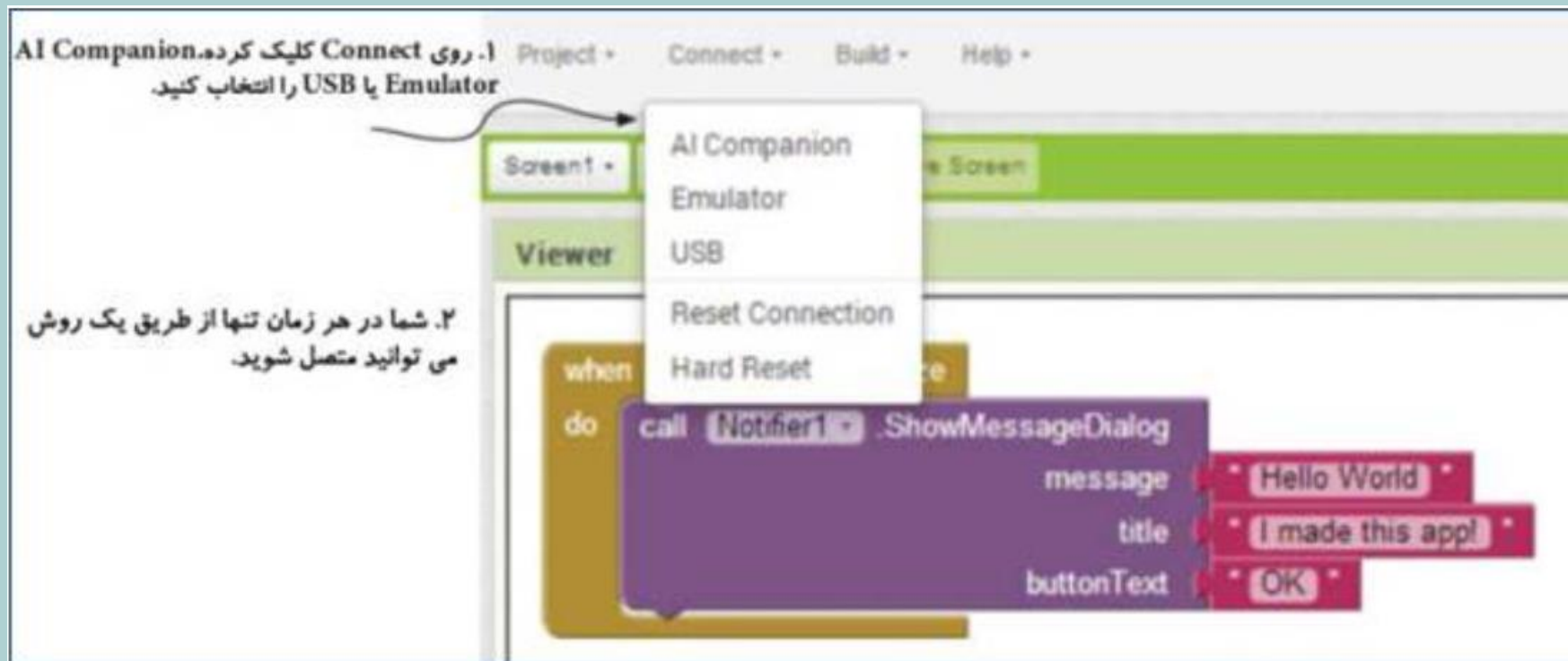
## ۵. تست کردن اپلیکیشن

داشتن اشتباه در هنگام برنامه نویسی زیاد پیش می آید، ممکن است بلوکی را به اشتباه انتخاب کرده باشید، یک مرحله را فراموش کرده یا حتی در اتصال بلوک‌ها به هم در حد چند میلی متر اشتباه کرده باشید. بنابراین مهم است که قبل از اینکه اپلیکیشن را به کاربران دیگر ارائه دهید، خود شما آن را تست کنید. شما می توانید اپلیکیشن را با استفاده از یک تلفن همراه هوشمند یا یک امولاتور روی صفحه تست کنید. App Inventor کامپوننت‌های Designer و بلوک‌های برنامه را به یک دسته از کدهای باینری که امولاتور یا گوشی می فهمند، ترجمه خواهد کرد. شما با استفاده از یک اتصال مستقیم بین Blocks Editor و گوشی یا امولاتور، اپلیکیشن را تست خواهید کرد. این روشی است که برای بیشتر اپلیکیشن‌های خود پس از ساخت آنها استفاده خواهید کرد، بعداً نگاهی می اندازیم به اینکه چگونه می توانید یک اپلیکیشن را پکیج (بسته بندی) کنید، طوری که حتی اگر گوشی شما به کامپیوتر متصل نبود، کار کند.

به یک تلفن هوشمند متصل شوید یا امولاتور را از منوی Connect در بالای صفحه خود اجرا کنید.



## ۵. تست کردن اپلیکیشن



## ۵. تست کردن اپلیکیشن

اگر یک تلفن دارید، می توانید به صورت وایرلس (Wi-Fi متصل شوید که ساده ترین گزینه است. به عنوان روش جایگزین می توانید آن را از طریق یک کابل USB وصل کنید. اگر تلفن ندارید، از امولاتور استفاده کنید. یکی از مراحل زیر را انجام دهید:

- برای یک اتصال بی سیم (وایرلس) به گوشی خود، AI Companion را انتخاب کنید. یک کد روی صفحه نمایش کامپیوتر شما ظاهر خواهد شد. شما می توانید با اجرای اپلیکیشن تلفنی AI Companion، این کد را وارد گوشی کرده یا توسط آن اسکن کنید.

- برای یک اتصال کابل USB، فرایند بستگی به نوعی از کامپیوتر که شما استفاده می کنید، دارد. شروع به متصل کردن گوشی خود به کامپیوتر با یک کابل USB کنید، و سپس دستورالعمل های لازم برای سیستم عامل خود (ویندوز، مک یا لینوکس) را در [Http://Mng.Bz/3ev2](http://Mng.Bz/3ev2) دنبال کنید.



## ۵. تست کردن اپلیکیشن



۱. AI Companion را از منوی Connect انتخاب کنید



۲. یک کد منحصر بفرد برای اپلیکیشن در هر دو فرم QR و متنی ظاهر خواهد شد.



۳. روی گوشی خود، اپلیکیشن MIT AI2 Companion را اجرا کنید.



۴. یا کد اپلیکیشن را تایپ کرده و روی دکمه Connect With Code کلیک کنید و یا روی دکمه Scan QR Code کلیک کرده و تلفن خود را روی کد QR قرار گرفته در صفحه نمایش کامپیوتر قرار دهید.





# استفاده از امولاتور نرم افزار

اگر برای تست کردن، تلفن ندارید، فرایند، به نوعی از کامپیوتر که شما استفاده می کنید، بستگی خواهد داشت. دستورالعمل های مناسب برای سیستم عامل خود (ویندوز، مک یا لینوکس) را در [Http://Mng.Bz/Tk58](http://Mng.Bz/Tk58) دنبال کنید. کمی طول می کشد تا امولاتور بارگذاری شود، بنابراین اجازه دهید کامل شود. آیکن اندروید را خواهید دید که در پایین صفحه نمایش ظاهر می شود. پس از اینکه امولاتور باز شد، می توانید از ماوس خود برای لغزاندن از میان قفل صفحه نمایش مجازی استفاده کنید.



# ذخیره سازی

## ذخیره سازی در فضای ابری و استفاده از Checkpoint ها (نقاط بررسی)

قبل از اینکه بخواهید اپلیکیشن را ذخیره کنید، مهم است بدانید چه چیزی را ذخیره می کنید. پروژه‌های App Inventor شامل طراحی‌های صفحه، چیدمان بلوک‌ها و موجودیت‌ها (گرافیک‌ها، صداها، ویدیوها و مانند آن که اپلیکیشن شما نیاز دارد) می باشد. پروژه‌های App Inventor تنها زمانی کار می کنند که در App Inventor اجرا شده روی کامپیوتر شما باز شود. برای کار کردن یک اپلیکیشن روی گوشی، بدون اتصال به App Inventor، شما نیاز به تبدیل آن به یک فایل پکیج کنید. پروژه‌های App Inventor به صورت خودکار هر چند ثانیه یک بار در زمانی که روی آنها کار می کنید، ذخیره می شوند.





# مشکل ذخیره سازی ابری و راه حل

- اگر در حال تمرین کردن بوده و تعداد زیادی تغییرات داشته باشید، و فراموش کرده باشید چه کاری انجام داده اید و سپس بخواهید به یک نسخه اولیه برگردید، اغلب با آخرین نسخه ای که ذخیره کرده اید، گیر خواهید افتاد.
  - اگر سرور دچار مشکل شود، هیچ کپی از پروژه‌های خود به صورت محلی در کامپیوتر خود نخواهید داشت.
- برای حل مشکل اول، App Inventor به شما امکان ذخیره پروژه‌ها و استفاده از Checkpoint ها را می‌دهد.

Save Project As یک کپی جدید از پروژه شما می‌سازد. شما به آن یک نام داده و سپس به کار کردن در پروژه جدید ادامه می‌دهید. پروژه قدیمی همیشه می‌تواند از منوی My Projects دوباره بارگذاری شود. اگر یک نسخه جدید از یک اپلیکیشن ایجاد می‌کنید، پس Save Project As بهترین گزینه است.

Save As - HelloWorld

New name:

Cancel OK

Project • Connect • Build • Help •

My Projects

Start new project ...

Import project (.aia) from my computer ...

Delete project

Save project

Save project as ...

Checkpoint ...

Export selected project (.aia) to my computer

App Inventor کار شما را به صورت خودکار ذخیره می‌کند. اگر می‌خواهید بیشتر مطمئن شوید، روی Save Project کلیک کنید.

checkpoint نمونه ای از پروژه شما را ذخیره می‌کند (در یک آرشیو از checkpoint ها ذخیره می‌شود). مزیت آن این است که اگر چیزی در مورد پروژه شما غلط از آب درآید (و App Inventor به صورت خودکار ذخیره کرده باشد)، شما همیشه می‌توانید به یک checkpoint قبلی برگردید. App Inventor به شما یک لیست از checkpoint ها را برای انتخاب کردن می‌دهد.

Checkpoint - HelloWorld

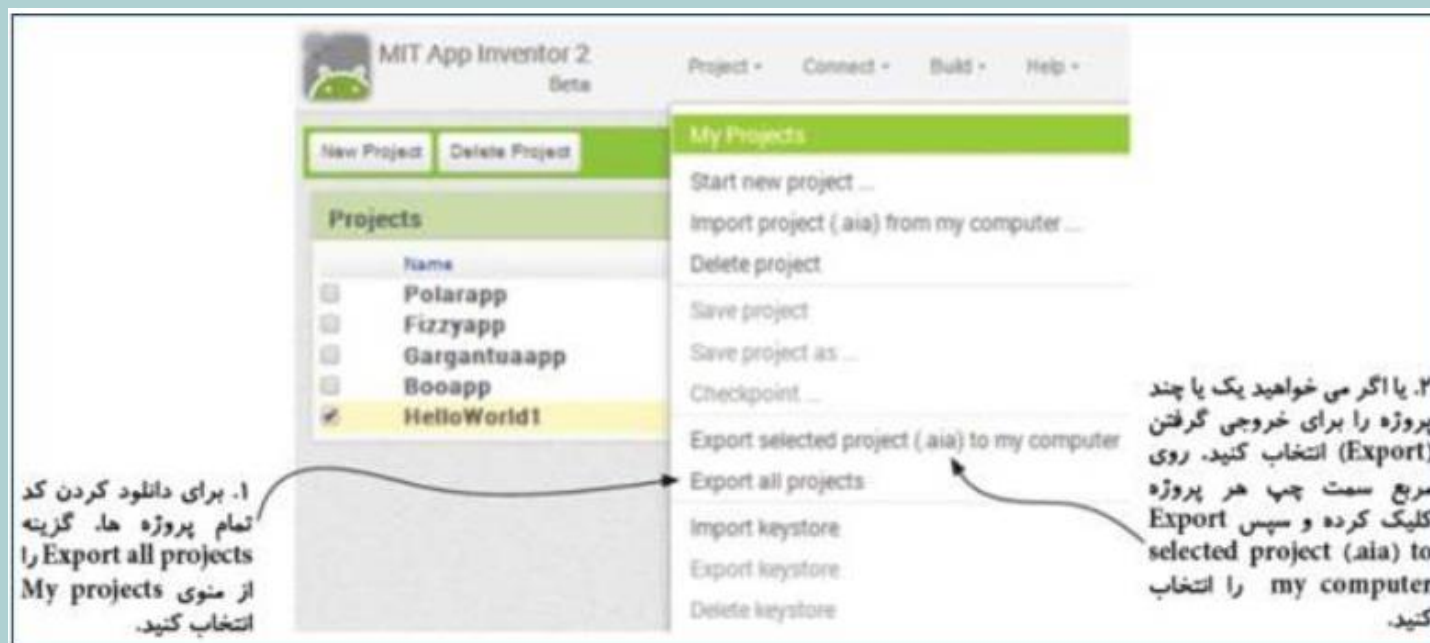
Checkpoint name:

Cancel OK



# ذخیره پروژه روی سیستم شخصی

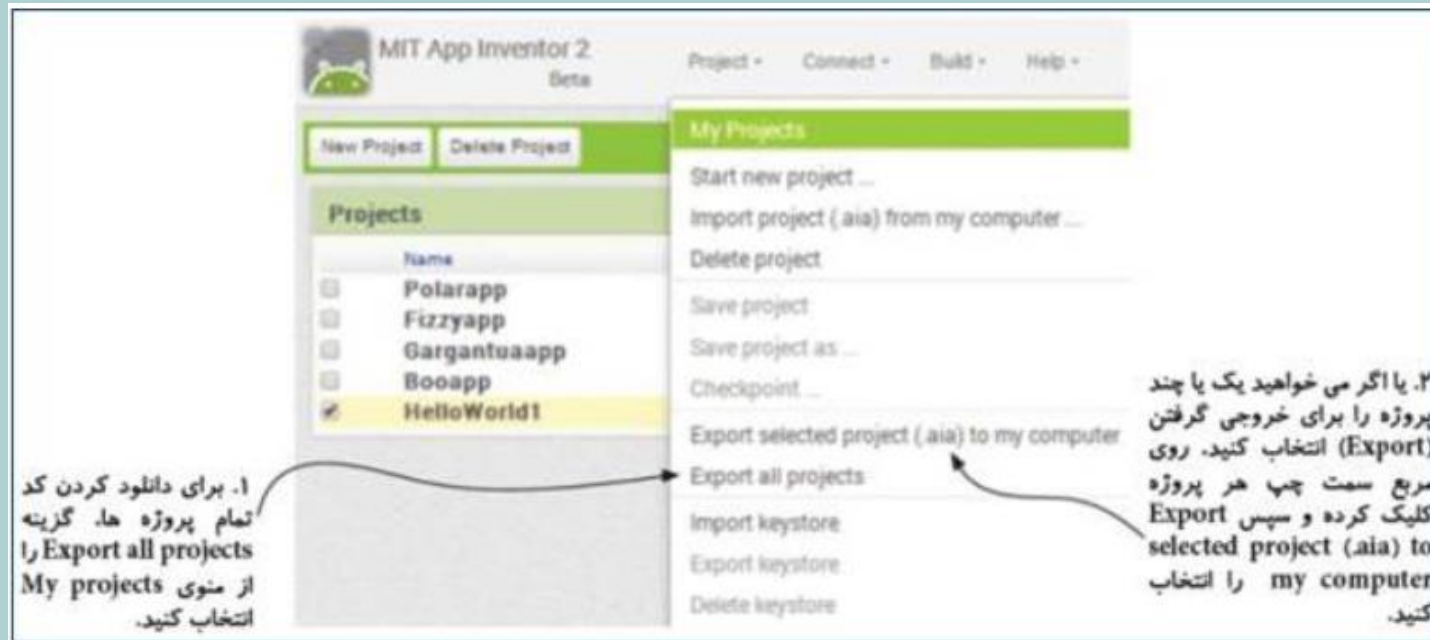
استفاده از Save As و Checkpoint، کپی‌های پروژه شما را در فضای ابری ذخیره می‌کند، اما اگر بخواهید یک کپی روی کامپیوتر خود به عنوان فایل پشتیبان (Backup) داشته باشید، چه کار می‌کنید؟



# ذخیره پروژه روی سیستم شخصی

استفاده از Save As و Checkpoint، کپی‌های پروژه شما را در فضای ابری ذخیره می‌کند، اما اگر بخواهید یک کپی روی کامپیوتر خود به عنوان فایل پشتیبان (Backup) داشته باشید، چه کار می‌کنید؟

فایل .Aia نتیجه در پوشه Downloads کامپیوتر شما ذخیره خواهد شد. از اینجا شما می‌توانید یک کپی محلی داشته و فایل پروژه را برای کار کردن روی آن توسط سایر App Inventor ها به اشتراک بگذارید، که آنها هم می‌توانند با (Import وارد کردن) فایل .Aia روی آن کار کنند.



# اپلیکیشن Hello World!، نسخه ۲

## کلیک دکمه: اپلیکیشن Hello World!، نسخه ۲

## اضافه کردن یک دکمه به برنامه Hello World! یک فرایند ساده چهار مرحله ای است:

## ۱ یک کپی جدید از اپلیکیشن ذخیره کنید.

## ۲ در App Inventor Designer یک دکمه به صفحه اضافه کنید.

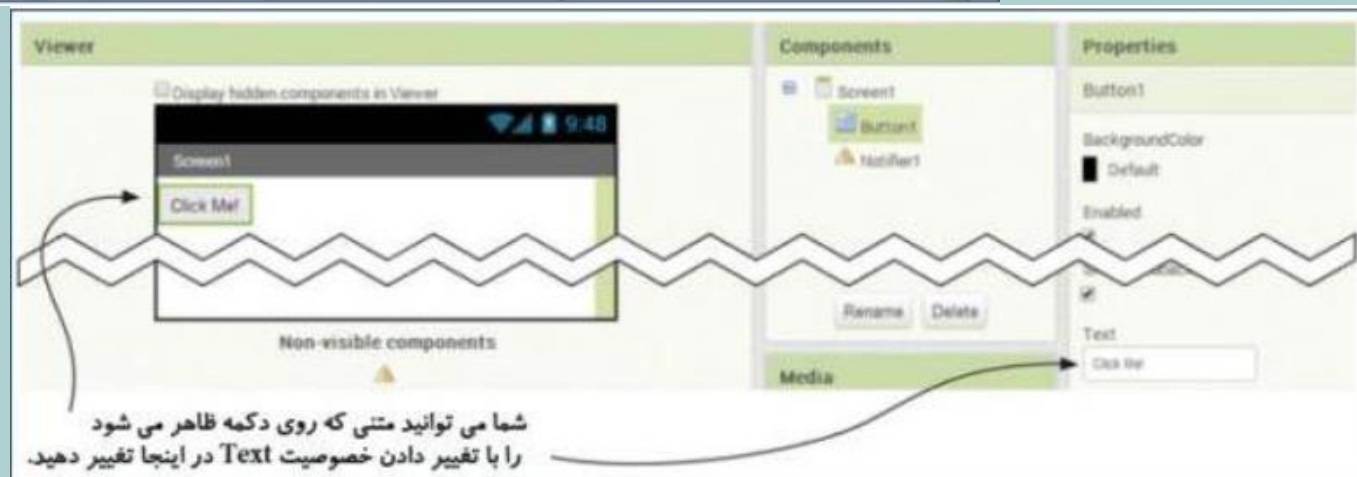
### ۳ محرک رویداد برای Notifier را تغییر دهید، طوری که به دکمه نگاه کند (به جای رویداد Screen1.Initialize)

## ۱۴ اپلیکیشن را تست کنید.



## ۲. اضافه کردن یک کامپوننت دکمه

لازم است به کاربر بگویید با دکمه چه کاری انجام دهد. متن دکمه را با Click Me! تغییر دهید.

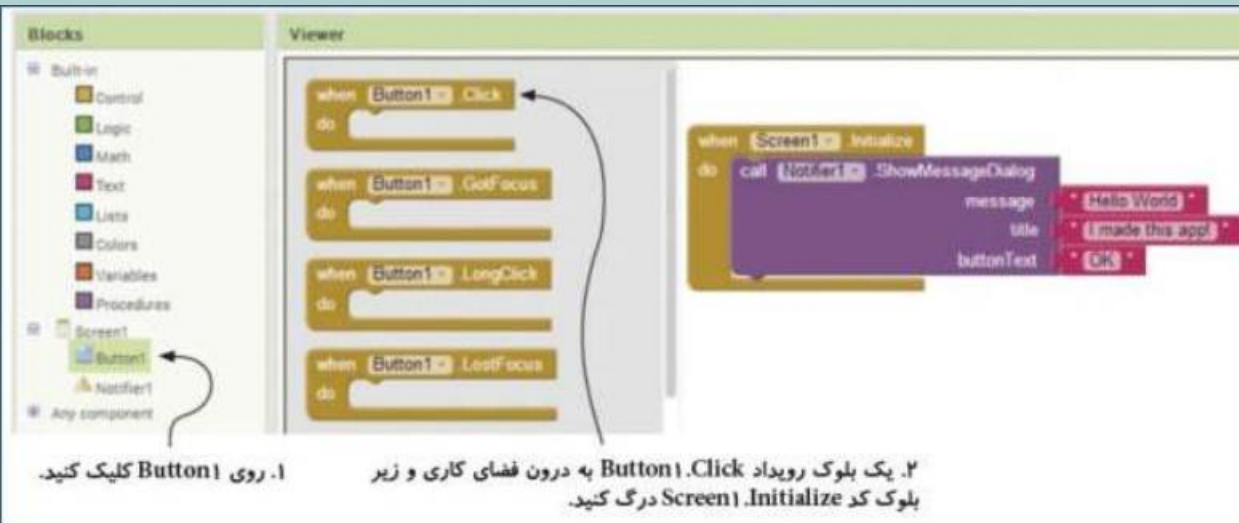




## ۳. برنامه نویسی بلوک ها

تاکنون بلوک هایی برای نمایش کادر هشدار نوشته اید. تنها چیزی که لازم است تغییر دهید، محرک رویداد است، به جای `Screen1.Initialize`، از یک رویداد `Click` تحریک شده با دکمه استفاده کنید. رویداد را به عنوان یک بلوک جدید اضافه کرده و سپس بلوک `Notifier` را از `Screen1.Initialize` به `Button1.Click` درگ کنید.

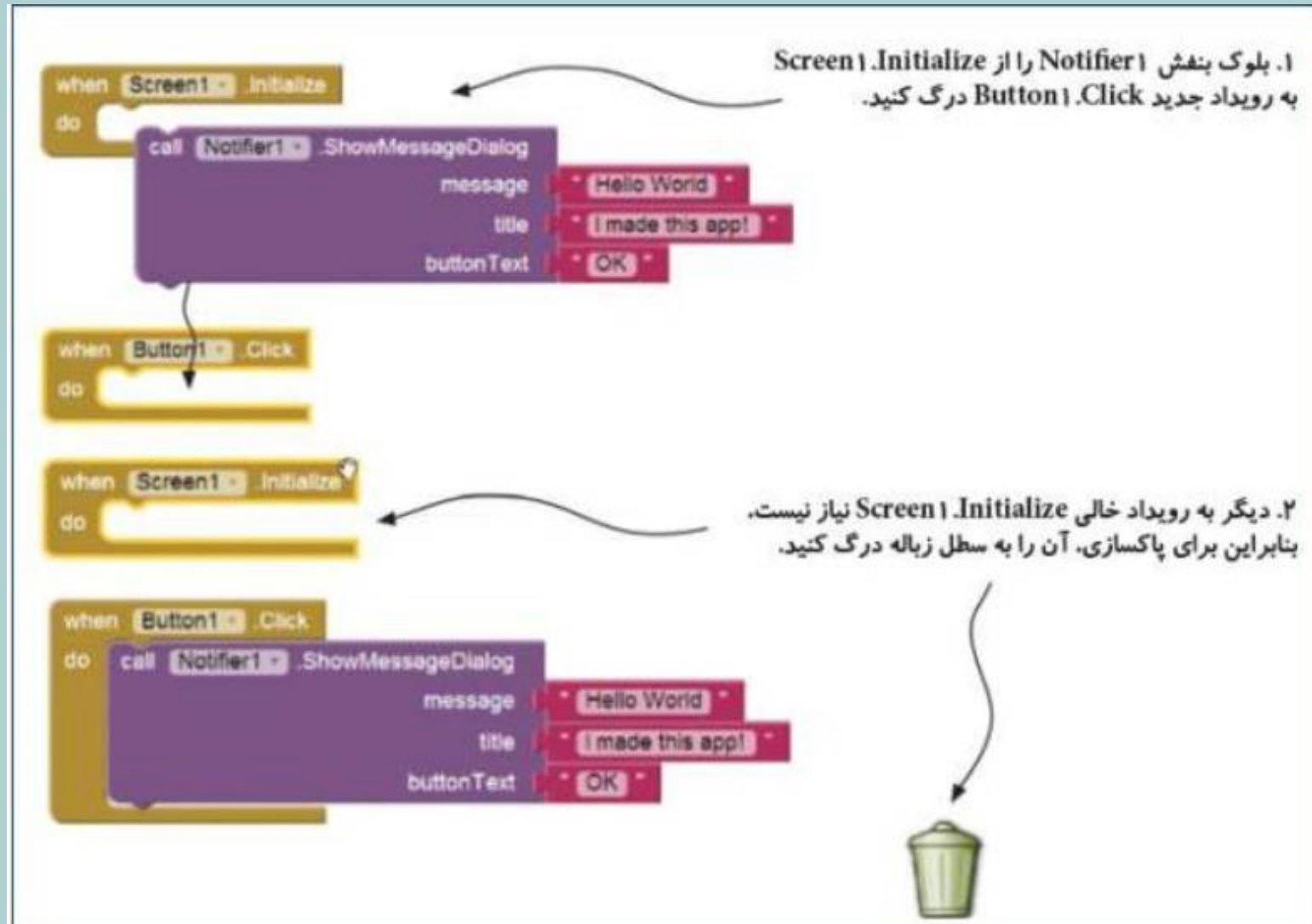
به `Blocks Editor` برگردید. یک رویداد `Button1.Click` درگ کنید.



اکنون کد بلوک `Notifier` را از `Screen1.Initialize` به `Button1.Click` درگ کنید.



### ۳. برنامه نویسی بلوک ها





## ۴. تست کردن اپلیکیشن

تمرین:

۱ یک دکمه و کادر هشدار جدید به اپلیکیشن خود اضافه کنید. کاری کنید که اگر کاربر روی دکمه دوم کلیک کرد، کادر هشدار، پیغام “Goodbye World” را نمایش دهد.

۲ سعی کنید خصوصیت رنگ Screen1 و Button1 را تغییر دهید. همچنین خصوصیت Font و Size مربوط به Button1 را تغییر دهید.



# تغییر دادن آیکن اپلیکیشن

آیکن App Inventor زیباست، اما شاید نخواهید تمام اپلیکیشن‌های شما شبیه هم باشند. شما می‌توانید قبل از پکیج کردن APK، تصویر موردنظر خود را به عنوان یک آیکن انتخاب کنید. بهتر است از یک آیکن با ۴۸ x ۴۸ پیکسل و در قالب PNG یا JPEG استفاده کنید. آن را به اپلیکیشن خود به صورت تصویر زیر اضافه کنید:



## فصل ۲: طراحی واسط کاربری (User Interface)

اپلیکیشن‌های فصل ۱ ساده بودند، اما به شما مطالب زیادی درباره اینکه چگونه بخش‌های مختلف App Inventor در کنار هم کار می‌کنند، آموزش دادند. در این فصل، شما اپلیکیشن‌ها را می‌سازید، طوری که سرگرم کننده به نظر آمده و با استفاده از گرافیک (تصاویر) و صدا کارهای جالب توجهی انجام دهند. اپلیکیشن ابتدایی که خواهید ساخت، Getting To Know Ewe نامیده می‌شود (Ewe: میش). گوسفند را ساخته و یک صدای بع (Baa) برای آن ایجاد می‌کنیم. سپس اپلیکیشن را توسعه می‌دهیم تا با لرزاندن گوشی، از ترس رم کرده و ناپدید شود، اپلیکیشنی که Ewe Escard Her! نامیده می‌شود.

مردم اپلیکیشن‌هایی را برای دانلود انتخاب می‌کنند که کار مفیدی انجام داده، جالب باشند و در صورت امکان به همان اندازه برای استفاده ساده باشند. "انجام دادن کاری مفید" چیزی است که این کتاب تماماً در رابطه با آن می‌باشد. این فصل روی دو جنبه آخر تمرکز می‌کند، ساختن یک اپلیکیشن که با یک واسط کاربری واقعاً ساده، جالب به نظر برسد.

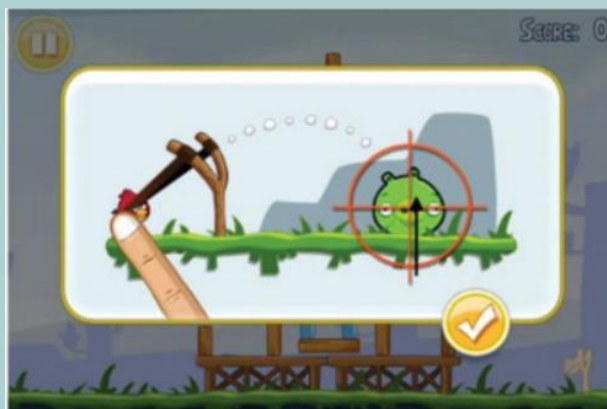


# واسط کاربری

یک واسط کاربری چیست؟

یکی از محبوب ترین بازی های اندروید، Angry Birds است. این بازی کاری مفید انجام می دهد: معماهای چالشی فراهم می کند (به اضافه اینکه به شما امکان ترکاندن می دهد). اگرچه این تنها دلیل برای محبوب بودن آن نیست، اما کارهای زیادی در رابطه با واسط کاربری آن انجام شده است.

اول، عالی به نظر می رسد: شخصیت ها، کار هنری، صداها و متن همه در کنار هم بدون نقص کار می کنند. دوم، بازی با آن به طرز باورنکردنی ساده است: کاربر تنها لازم است به بازی بگوید که پرنده بعدی اش در چه زاویه و سرعتی، باید پرتاب شود. بازی می تواند از کاربر بخواهد تا این مقادیر را در صفحه عددی تایپ کرده و Enter را فشار دهد. به جای آن، طراحان بازی ایده برجسته ای را طرح کرده اند که بازیکن پرنده را تحت یک زاویه از منجنیق دور کرده و سپس اجازه دهد پرتاب شود، دقیقاً مانند اینکه از یک منجنیق در زندگی واقعی استفاده می کند. این یکی از دلایلی است که مردم، Angry Birds را برای چندین ساعت بازی می کنند.

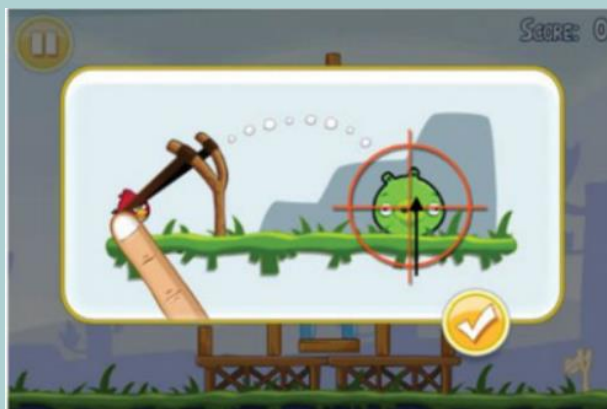


# واسط کاربری

آخرین چیزی که در مورد Angry Birds می توان گفت، بازخورد دادن نتیجه پرتاب پرنده ها به بازیکن است. بنابراین شما می بینید که پرنده پرواز کرده و پس از برخورد، صدای جلوه صوتی را شنیده و لرزیدن تلفن خود را حس خواهید کرد.

ما به ترکیب موارد بالا می گوییم: ( User Interface واسط کاربری)، یا به اختصار UI:

- چگونه دیده شدن اپلیکیشن - طراحی گرافیکی، تایپوگرافی (قلم ها) و رنگ
- چگونه مردم به اپلیکیشن بگویند چه چیزی می خواهند انجام شود
- چگونه اپلیکیشن اطلاعات را از طریق صفحه نمایش، بلندگوها یا بازخورد لمسی (وزوز کردن موتور کوچک گوشی) به کاربر برگشت دهد.





## نکته آموزشی: ورودی > پردازش > خروجی

## استفاده از Designer برای ساختن یک UI

## مراحل زیر را انجام دهید:

## ۱ ابتدا UI را در پنجره Designer طراحی کنید (حتی اگر تنها یک طرح خام باشد).

## ۲ Blocks Editor را برنامه نویسی کنید تا به اپلیکیشن بگویید چه کاری انجام دهد.

### ۳ برنامه را روی گوشی یا امولاتور خود تست کنید.



# نکته آموزشی: ورودی > پردازش > خروجی

شما برای رفت و برگشت بین این سه فعالیت، زمان زیادی را صرف خواهید کرد، همان طور که در تصویر زیر نشان داده شده است:





# نکته آموزشی: ورودی > پردازش > خروجی

نکته آموزشی: طراحی از بالا به پایین

در App Inventor، شما ابتدا UI را طراحی کرده و سپس یک برنامه کامپیوتری می نویسید. این یک مثال از دیدگاه بالا به پایین است.

یک روش جایگزین از انجام کارها در سایر برنامه ها، شروع کردن با پیچ و مهره های برنامه و بعداً نگران UI شدن می باشد، پایین به بالا.

هر دو دیدگاه مزایا و معایب خود را دارند.



## آشنایی با کامپوننت‌های ( Built-In تعبیه شده )

## نکته آموزشی: طراحی از بالا به پایین

در App Inventor، شما ابتدا UI را طراحی کرده و سپس یک برنامه کامپیوتری می نویسید. این یک مثال از دیدگاه بالا به پایین است.

یک روش جایگزین از انجام کارها در سایر برنامه ها، شروع کردن با پیچ و مهره های برنامه و بعداً نگران UI شدن می باشد، پایین به بالا.

## هر دو دیدگاه مزایا و معایب خود را دارند.

The screenshot displays the 'Components' tab in Android Studio, organized into several categories:

- App**: This section contains sub-categories:
  - Layout**: HorizontalArrangement, TableArrangement, VerticalArrangement.
  - Drawing and Animation**: Ball, Canvas, ImageSprite.
  - Connectivity**: ActivityStarter, BluetoothClient, BluetoothServer, Web.
  - Storage**: FusionTablesControl, TinyDB, TinyWebDB.
  - Media**: Camcorder, Camera, ImagePicker, Player, Sound, SoundRecorder, SpeechRecognizer, TextToSpeech, VideoPlayer.
  - Social**: ContactPicker, EmailPicker, PhoneCall, PhoneNumberPicker, Texting, Twitter.
  - Sensors**: AccelerometerSensor, BarcodeScanner, LocationSensor, NearField, OrientationSensor.
  - LEGO® MINDSTORMS®**: NxtColorSensor, NxtDirectCommands, NxtDrive, NxtLightSensor, NxtSoundSensor, NxtTouchSensor, NxtUltrasonicSensor.
- User Interface**: A separate section containing: Button, CheckBox, Clock, Image, Label, ListPicker, Notifier, PasswordTextBox, Slider, TextBox, and WebViewer.



# آشنایی با کامپوننت‌های (Built-In تعبیه شده)

به عنوان یک کاربر کامپیوتر و تلفن هوشمند، تعداد زیادی از کامپوننت‌های پالت App Inventor را تشخیص خواهید داد. عناصری مانند دکمه‌ها، برچسب‌ها و کادرهای متنی در صفحات وب، برنامه‌های رومیزی و اپلیکیشن‌های گوشی هوشمند رایج می‌باشند. به دلیل اینکه App Inventor این عناصر آماده را دارد، شما می‌توانید فرایند ساخت اپلیکیشن‌ها را سریع‌تر انجام دهید.



# آشنایی با کامپوننت‌های (Built-In تعبیه شده)

AMazeingPenguin Screen1 - Add Screen ...

Palette Viewer

User Interface

- Button
- CheckBox
- Clock
- Image
- Label
- ListPicker
- Notifier
- PasswordTextBox

☒ Display hidden components

**Button**

Button with the ability to detect clicks. Many aspects of its appearance can be changed, as well as whether it is clickable (Enabled), can be changed in the Designer or in the Blocks Editor.

[More information](#)

راهنمای کامپوننت  
شما می‌توانید اطلاعات بیشتر درباره هر کامپوننت را از پالت App Inventor پیدا کنید.

روی علامت سوال کنار هر عنصر پالت کلیک کنید تا یک توضیح کوتاه از آنچه انجام می‌دهد، نمایش داده شود.

نیاز به اطلاعات بیشتر دارید؟ روی لینک **More Information** کلیک کنید.



# اپلیکیشن Getting To Know Ewe

## هدف از این اپلیکیشن

این اپلیکیشن یک تصویر از گوسفند در یک زمین را نشان می دهد. هنگامی که کاربر با گوسفند بازی کند، بع بع می کند.

1

## درجه بازی:

## موجودیت های مورد نیاز:

تصاویر: تصویر پس زمینه چراگاه سرسبز و گوسفند. صداها: "بع"



# اپلیکیشن Getting To Know Ewe

## ۱. تنظیمات پروژه

با انتخاب Project > Start New Project، شروع به یک پروژه جدید App Inventor کنید. به آن نام Gettingtoknowewe داده و روی Ok کلیک کنید. باید با چهار ستون آشنا شده باشید: Palette، Viewer، Components و Properties.





# اپلیکیشن Getting To Know Ewe

شما می توانید ببینید که صفحه App Inventor تغییر یافته است. این تنها یک طرح اولیه از چیزی است که اپلیکیشن ما روی گوشی یا امولاتور به نظر خواهد رسید. شما می توانید تفاوت‌ها را با اتصال به گوشی یا اجرا کردن امولاتور بررسی کنید؛ همان طور که تغییرات را در Designer انجام می دهید، آن تغییرات فوراً روی گوشی ظاهر خواهد شد.

برای اکنون، شما تنها یک کامپوننت دارید: Screen1. به زودی کامپوننت‌های بیشتری اضافه خواهید کرد، اما ابتدا بیایید خصوصیات صفحه را تنظیم کنیم.

۱. خاصیت BackgroundColor صفحه را با None تنظیم کنید.

۲. یک تصویر از یک زمین سرسبز برای پشت سر گوسفند می خواهید. روی BackgroundImage کلیک کنید.

۳. در کادر ظاهر شده روی Upload File ... کلیک کنید.

۴. فایل GreenField.jpg را انتخاب کرده و سپس روی open کلیک کنید.

۵. روی ok کلیک کنید.

۶. برای Title، بنویسید: Getting To Know Ewe. این نام اپلیکیشن است که در بالای صفحه ظاهر می شود.



# نکته آموزشی: تنظیم کردن خصوصیات

تمام کامپوننت‌ها خصوصياتی دارند، چیزهایی که شما می‌توانید در مورد آن کامپوننت تغییر دهید. بعضی از کامپوننت‌ها تعداد زیادی خصوصیت دارند (مانند Screen، بقیه تنها تعداد کمی خصوصیت دارند (مانند Sound، و بعضی هیچ خصوصیتی ندارند (مانند Camera).  
Properties در قسمت سمت راست صفحه به شما امکان تنظیم کردن خصوصیات کامپوننت‌ها، هنگامی که اپلیکیشن شما برای اولین بار اجرا می‌شود، را می‌دهد.

بعداً برنامه‌هایی که شما خواهید نوشت، هم قادر به خواندن و هم تغییر خصوصیات خواهند بود. اینها خاصیت‌های (Geting دریافت کردنی) و (Setting تنظیم کردنی) خوانده می‌شوند. توجه خواهید کرد که بیشتر بلوک‌ها با کلمه Get یا Set شروع می‌شوند.



## ۲. اضافه کردن تصویر گوسفند

شما برای کامل کردن UI اپلیکیشن خود به دو کامپوننت نیاز دارید: یک تصویر گوسفند و یک صدای “Baa! با نگاه کردن به کامپوننت‌های نشان داده شده در زیر، به نظر شما کدام یک امکان درج گوسفند را می‌دهد؟

انتخاب کامپوننت

آیا Image را انتخاب کردید؟ اما با یک مشکل مواجه هستیم: image ها نمی‌توانند بگویند که لمس شده‌اند. در حقیقت، آنها اصلاً هیچ تشخیص ورودی ندارند، آنها تنها در صفحه قرار گرفته و زیبا به نظر می‌رسند.

Palette

User Interface

- Button
- CheckBox
- Clock
- Image
- Label
- ListPicker
- Notifier

شما گزینه‌های زیادی برای اشیایی که می‌توانند ورودی را تشخیص دهند، دارید، مانند زمانی که یک کاربر آنها را لمس کرده یا درگ می‌کند.

این بار، از Button استفاده کنید: ساده‌ترین گزینه.



## ۲. اضافه کردن تصویر گوسفند

افزودن و تغییر نام دکمه

۱. یک کامپوننت دکمه از پالت به زمین سرسبز خود درج کنید.

۲. نام دکمه را به SheepButton تغییر دهید.

۳. چه اتفاقی افتاد؟ نام این کامپوننت تغییر یافت. این کار هنگام استفاده از Blocks Editor مفید است، مخصوصاً اگر اپلیکیشنی با تعداد دکمه های زیاد می سازید. اما متن دکمه که کاربر می بیند، تغییر نیافته است.

۴. نام دکمه را به Button1 تغییر دهید (بدون هیچ فاصله) تغییر نام دهید.



شیه گوسفند درآوردن دکمه

۱. شما تصویری از یک گوسفند برای دکمه می خواهید. بنابراین روی خاصیت Image از SheepButton کلیک کنید.

۲. روی 'Upload File ...' کلیک کنید.

۳. فایل Sheep.png را انتخاب کنید.

۴. روی 'open' و سپس 'ok' کلیک کنید.

۵. روی 'Choose File' کلیک کنید.

۶. روی 'Open' کلیک کنید.



# نکته آموزشی: قالب PNG

ما از یک تصویر با قالب PNG و پس زمینه شفاف از یک گوسفند استفاده می کنیم. این بدان معناست که گوسفند یک کادر سفیدرنگ اطراف آن نخواهد داشت.





متن دکمه روی چشم‌های گوسفند قرار گرفته و نسبتاً بزرگ می باشد. بنابراین برای اصلاح این مشکلات، Text، Width و Height را تغییر می دهیم. روی خصوصیات مربوطه زوم کرده ایم تا به شما در یافتن آنها کمک کرده باشیم:





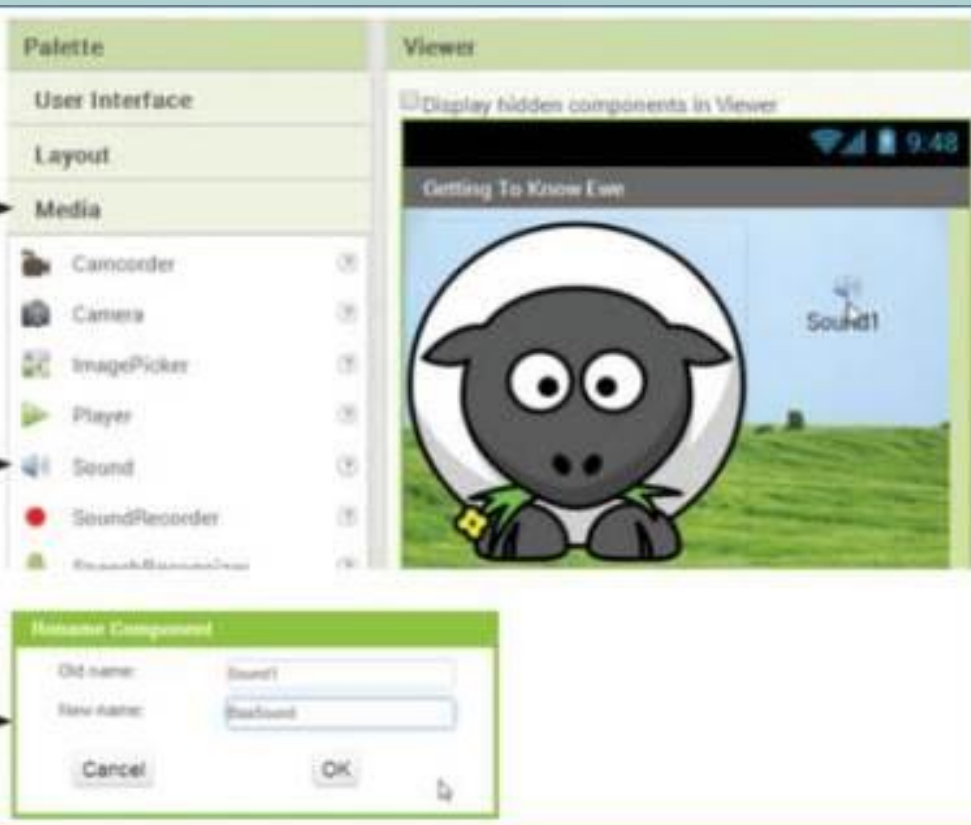
### ۳. اضافه کردن یک صدای "بع"

آخرین کامپوننتی که نیاز دارید، صدای "بع" می باشد.

۱. روی Media کلیک کنید.

۲. یک کامپوننت Sound را به صفحه درگ کنید. هنگامی که دکمه ماوس را رها می کنید، آیکن Sound1 ناپدید می شود. این آیکن دوباره در زیر صفحه Viewer ظاهر می شود.

۳. نام Sound1 را به Baasound تغییر دهید.



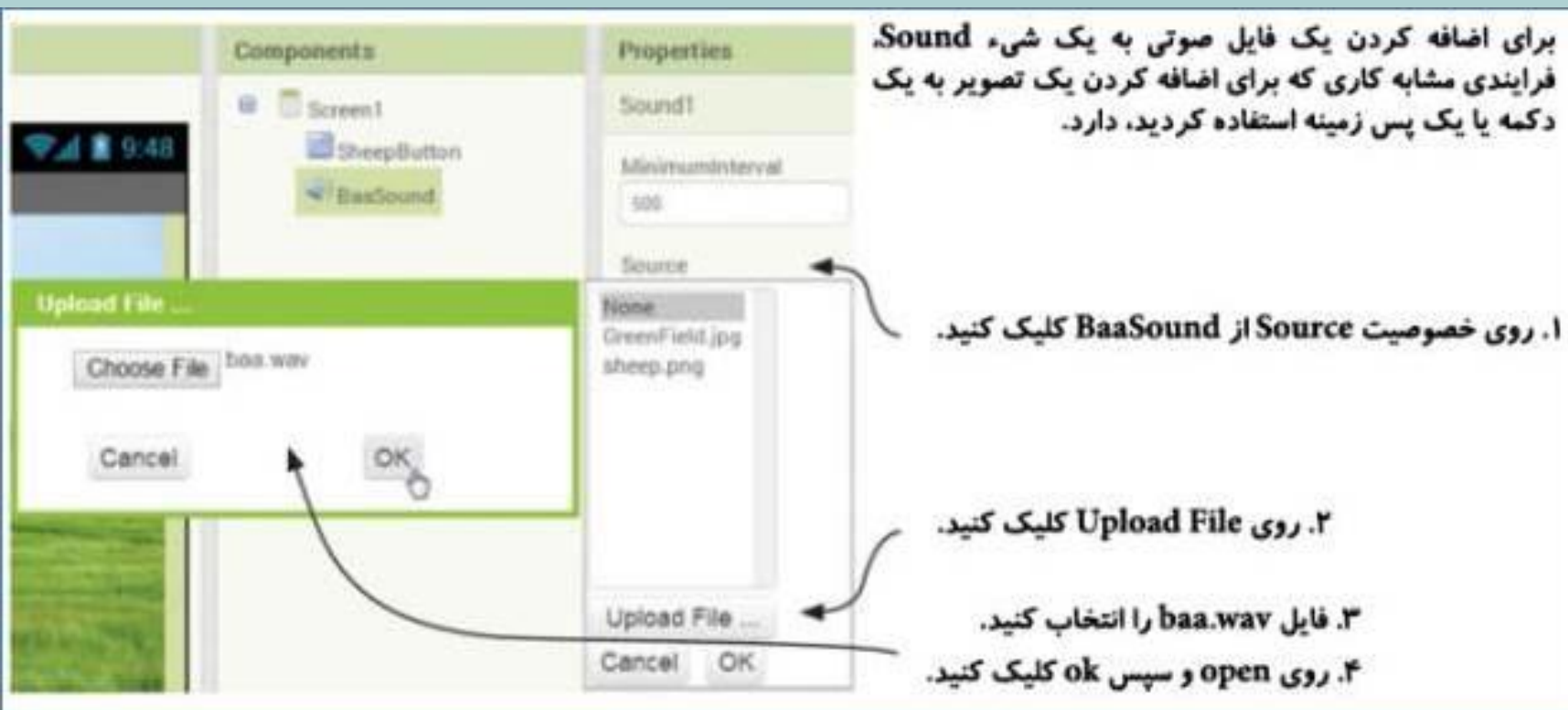
در آخر شما نیاز به وصل کردن یک فایل صوتی به کامپوننت Baasound خود دارید. App Inventor می تواند از فایل های صوتی در قالب های متنوع و مختلفی استفاده کند؛ ما به شما پیشنهاد می کنیم از فایل های PCM/WAV یا MP3 استفاده کنید، زیرا آنها در تمام گوشی ها کار می کنند.



## ۳. اضافه کردن یک صدای "بع"

هشدار

فراموش کردن این مرحله می تواند برای شما اتفاق بیافتد. اگر یک دکمه یا تصویر پس زمینه دارای یک فایل رسانه (Media) متصل به آن نباشد، در پیش نمایش صفحه مشخص خواهد بود، اما اگر یک فایل صوتی را فراموش کنید، ممکن است بعداً و در هنگام اجرای برنامه متوجه آن شوید.



برای اضافه کردن یک فایل صوتی به یک شیء Sound، فرایندی مشابه کاری که برای اضافه کردن یک تصویر به یک دکمه یا یک پس زمینه استفاده کردید، دارد.

۱. روی خصوصیت Source از BaaSound کلیک کنید.
۲. روی Upload File کلیک کنید.
۳. فایل baa.wav را انتخاب کنید.
۴. روی open و سپس ok کلیک کنید.



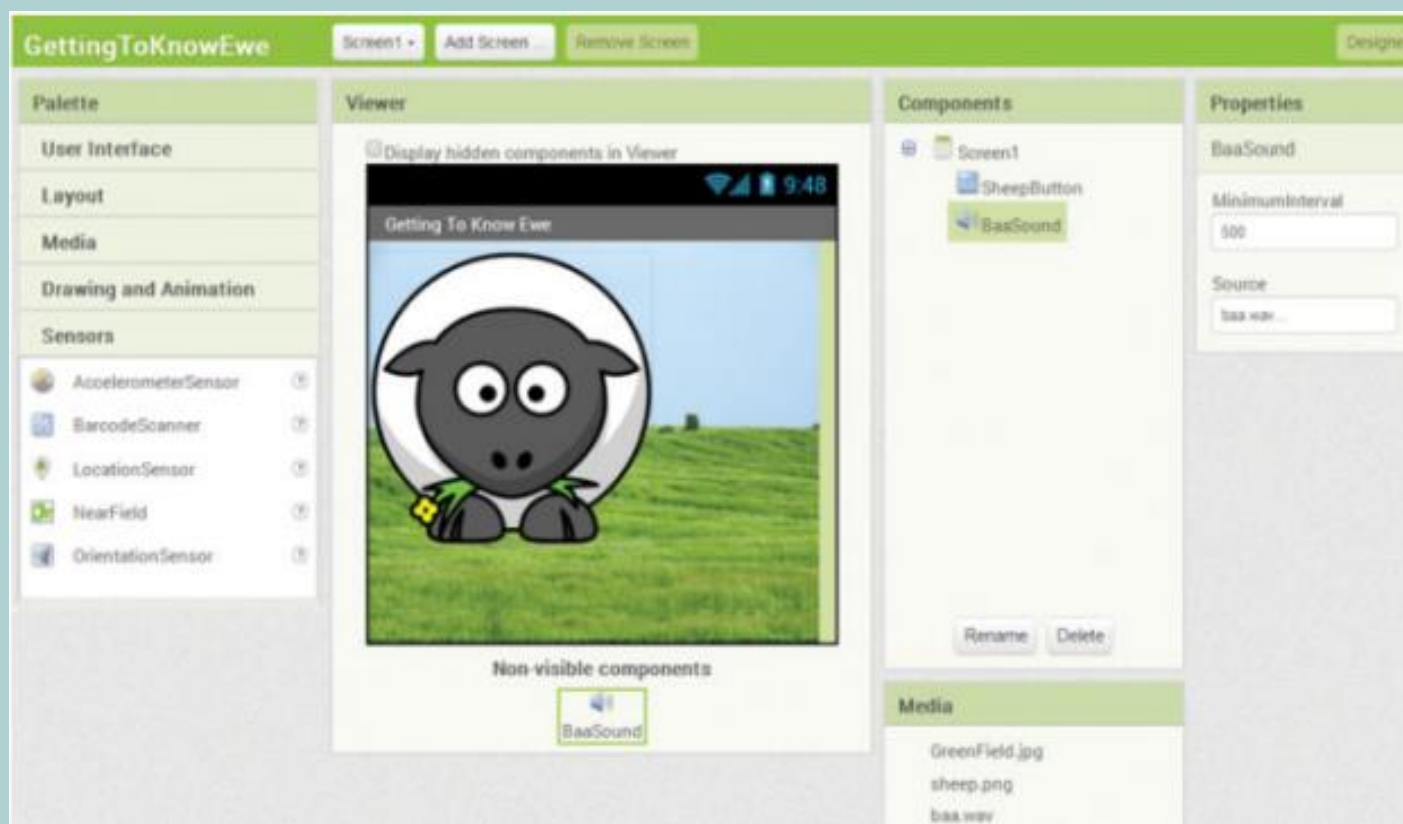
# نکته آموزشی: کامپوننت‌های غیرقابل مشاهده:

یک Sound، یک کامپوننت غیرقابل مشاهده (Nonvisible می باشد، شما آن را روی صفحه نمی بینید. سایر کامپوننت‌های غیرقابل مشاهده شامل وسایل ورودی مانند دوربین، اسکنر بارکد و شتاب سنج و سنسورهای جهتی که حرکت و جهت را تشخیص می دهند (تکان دادن یا ضربه زدن به گوشی) می باشند. هنگام طراحی UI خود، کامپوننت‌های موردنظر را درگ کرده و روی صفحه رها می کنید. خواهید دید که آنها در بخش Non-Visible Components واقع در زیر صفحه Viewer ظاهر می شوند.



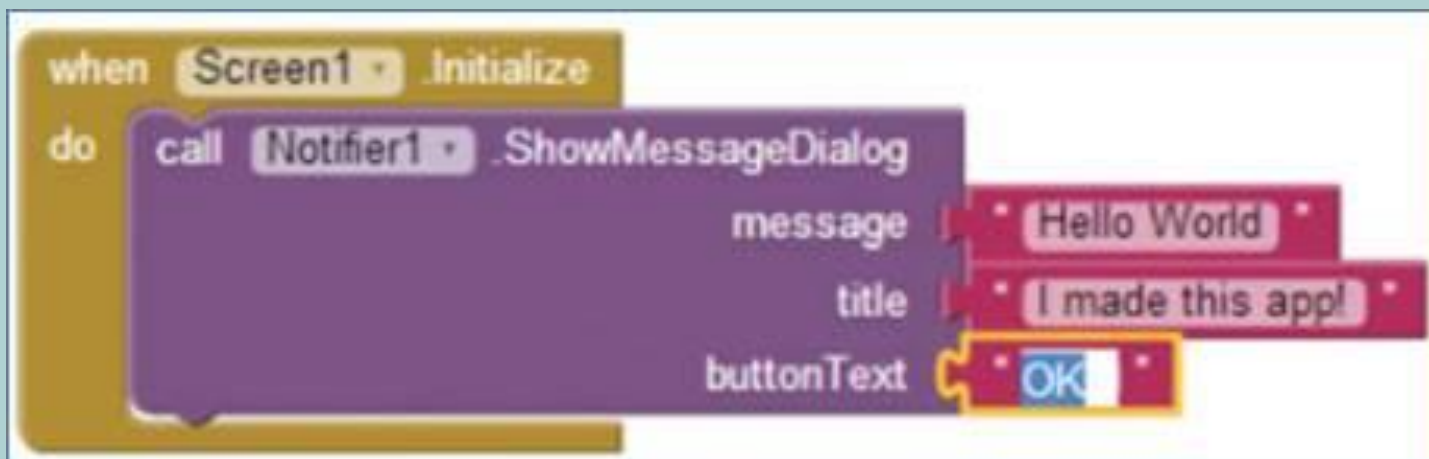
# نکته آموزشی: کامپوننت‌های غیرقابل مشاهده:

UI شما دارای تمام عناصری که برای ساختن نسخه پایه از اپلیکیشن Getting To Know Ewe نیاز دارید، می باشد. شما تمام اشیاء را انتخاب و اضافه کرده اید، رسانه‌ها را به فایل‌های مربوطه متصل کرده (تصاویر و صدا) و تعدادی از خصوصیات آنها نظیر Size و Text را تنظیم کرده اید. UI نهایی باید چیزی شبیه تصویر زیر شده باشد:



## ۴. برنامه نویسی بلوک ها، بخش ۱: بخش صدا

به یاد می آورید که چگونه در فصل ۱ از یک رویداد `Button.Click` استفاده کردید که به برنامه بگویید چه چیزی می خواهید تا انجام دهد؟ این اپلیکیشن کاملاً به همان روش کار می کند، هنگامی که یک کاربر روی گوسفند ضربه می زند، واقعاً دارد روی یک دکمه کلیک می کند.





## ۴. برنامه نویسی بلوک ها، بخش ۱: پخش صدا

به Blocks Editor برگردید، و اگر تاکنون به گوشی یا امولاتور، همانند آنچه در فصل ۱ انجام دادید، متصل نیستید، این کار را انجام دهید. بلوک ها را مانند تصویر زیر ایجاد کنید:



اکنون آن را امتحان کنید. روی گوسفند درون گوشی کلیک کنید تا جایزه کار خود را با شنیدن صدا به گوسفند دریافت کنید.





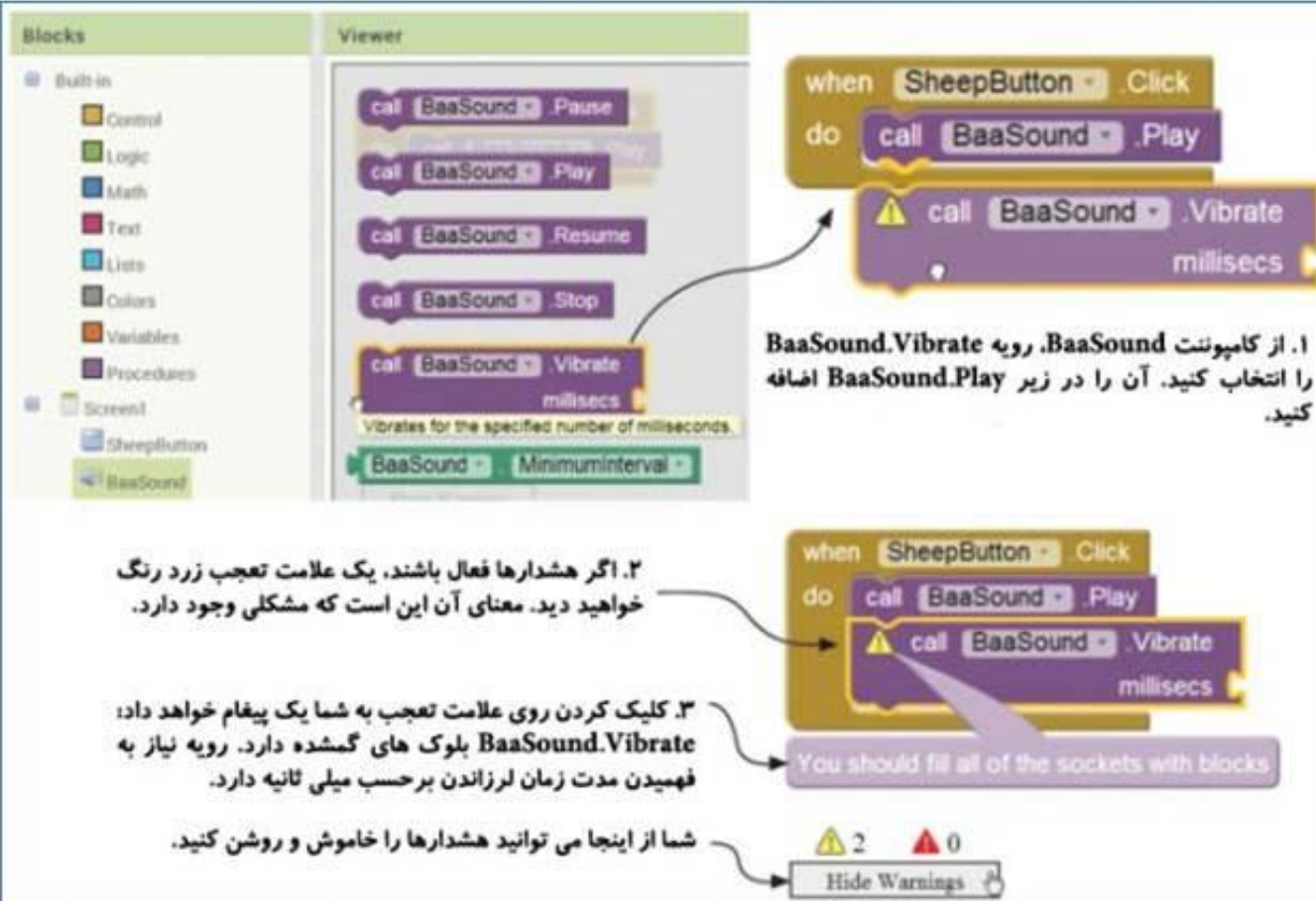
# نکته آموزشی: فراخوانی رویه‌ها (Procedures)

در این اپلیکیشن، شما با استفاده از یک رویه درون ساخت (Play)، یک صدای "بع" را پخش کردید. این کار، فراخوانی یک رویه نامیده می‌شود. شما می‌توانید رویه‌ها را تشخیص دهید، زیرا بلوک آنها بنفش بوده و شامل کلمه Call می‌باشد. در فصل ۵، یاد می‌گیرید چگونه رویه‌های خود را ساخته و فراخوانی کنید.



# ۵. برنامه نویسی بلوک ها، بخش ۲: لرزاندن (Vibrating) گوشی

روی BaaSound در Myblocks کلیک کنید. تعداد زیادی رویه مانند Pause، Resume و Stop کردن صدا خواهید دید. همچنین رویه هایی برای لرزاندن گوشی وجود دارد. از آنها استفاده خواهید کرد، طوری که به همراه پخش صدای "بع"، گوشی نیز به لرزه درآید.



The screenshot shows the Myblocks IDE interface. On the left, the 'Blocks' panel lists various categories, with 'BaaSound' highlighted under 'Built-in'. The 'Viewer' panel shows a code block for 'when SheepButton Click' with a 'do' block containing 'call BaaSound .Play' and 'call BaaSound .Vibrate milliseconds'. A warning icon is present next to the 'Vibrate' block. Below the code block, a message says 'You should fill all of the sockets with blocks'. At the bottom, a 'Hide Warnings' button is visible.

۱. از کامپوننت BaaSound، رویه BaaSound.Vibrate را انتخاب کنید. آن را در زیر BaaSound.Play اضافه کنید.

۲. اگر هشدارها فعال باشند، یک علامت تعجب زرد رنگ خواهید دید. معنای آن این است که مشکلی وجود دارد.

۳. کلیک کردن روی علامت تعجب به شما یک پیام خواهد داد: BaaSound.Vibrate بلوک های گمشده دارد. رویه نیاز به فهمیدن مدت زمان لرزاندن برحسب میلی ثانیه دارد.

شما از اینجا می توانید هشدارها را خاموش و روشن کنید.

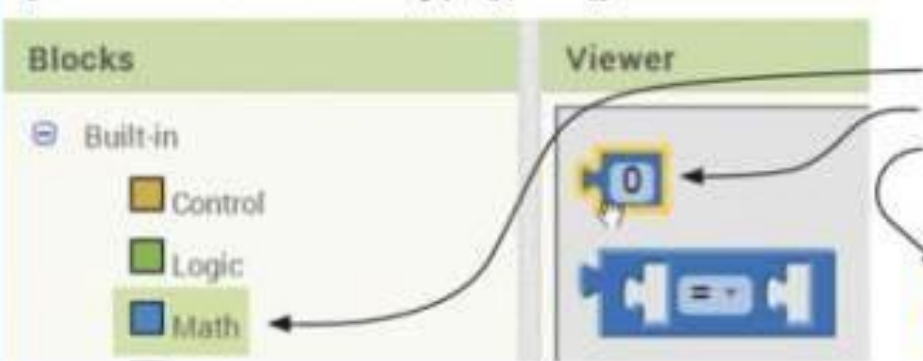


# وارد کردن یک مقدار

## وارد کردن یک مقدار

در هر ثانیه، ۱۰۰۰ میلی ثانیه وجود دارد، بنابراین ۱۰۰۰ را در بلوک Baasound.Vibrate وارد کنید. شما دو گزینه برای وارد کردن مقادیر دارید: هر کدام که ساده تر است را انتخاب کنید.


گزینه ۱: از یک بلوک Built-in > Math استفاده کنید.



برای ایجاد یک بلوک Number با مقدار ۱۰۰۰ کارهای زیر را انجام دهید:

۱. روی Built-in > Math کلیک کنید.
۲. بلوک Number که شامل یک عدد صفر می باشد را انتخاب کنید.
۳. روی ۰ کلیک کرده و آن را با ۱۰۰۰ تغییر دهید.

گزینه ۲: از یک کلید میانبر صفحه کلید استفاده کنید.

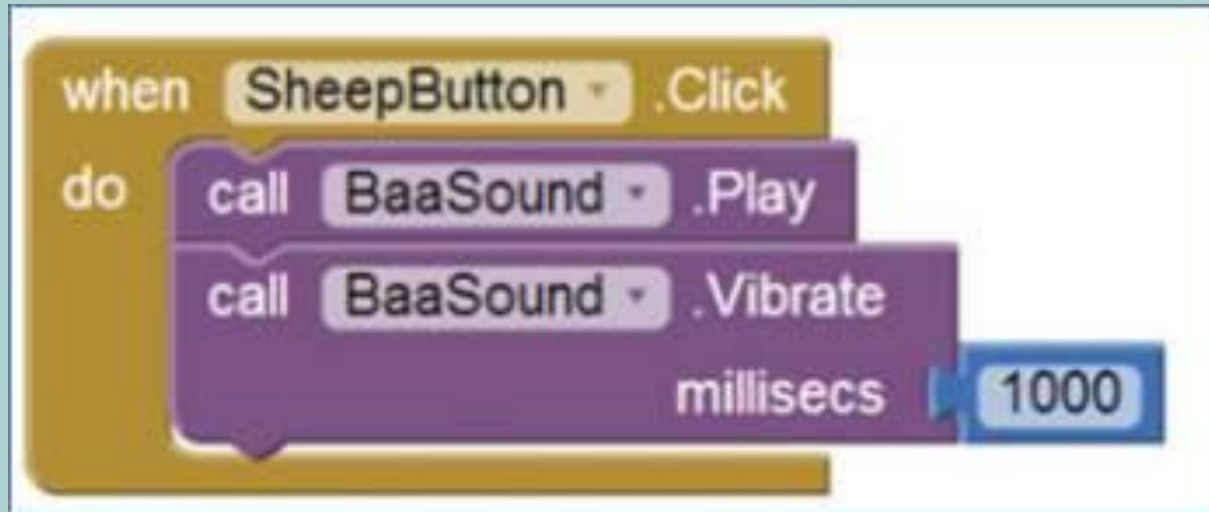


۱. در جایی داخل فضای کاری خالی از Blocks Editor کلیک کنید.
۲. عدد ۱۰۰۰ را تایپ کنید.
۳. کلید Enter از صفحه کلید خود را فشار دهید.



# وارد کردن یک مقدار

اکنون بلوک Number (با مقدار ۱۰۰۰) را داخل شکاف خالی Baasound.Vibrate قرار دهید. خواهید دید که علامت تعجب زردرنگ ناپدید شده و اپلیکیشن روی گوشی شما کار می کند. امتحان کنید!



مقادیر مختلف را برای Baasound.Vibrate امتحان کنید. سعی کنید به سوالات زیر جواب دهید:

- کوتاه ترین زمانی که می توانید وارد کنید و همچنان لرزش تلفن را احساس کنید، چه زمانی می باشد؟
- وارد کردن چه عددی برحسب میلی ثانیه باعث می شود لرزش گوشی دقیقاً به اندازه مدت زمان صدای "بع" شود؟





# چالش بزرگتر: اپلیکیشن Ewe Scared Her!

هدف از این اپلیکیشن

روی گوسفند کلیک کنید، و او بع می کند. تلفن را تکان دهید، گوسفند ناپدید شده و پیغام "Ewe scared her!" روی یک دکمه reset ظاهر می شود. کاربر روی این دکمه کلیک می کند تا بازی دوباره شروع شود.

1 درجه بازی:

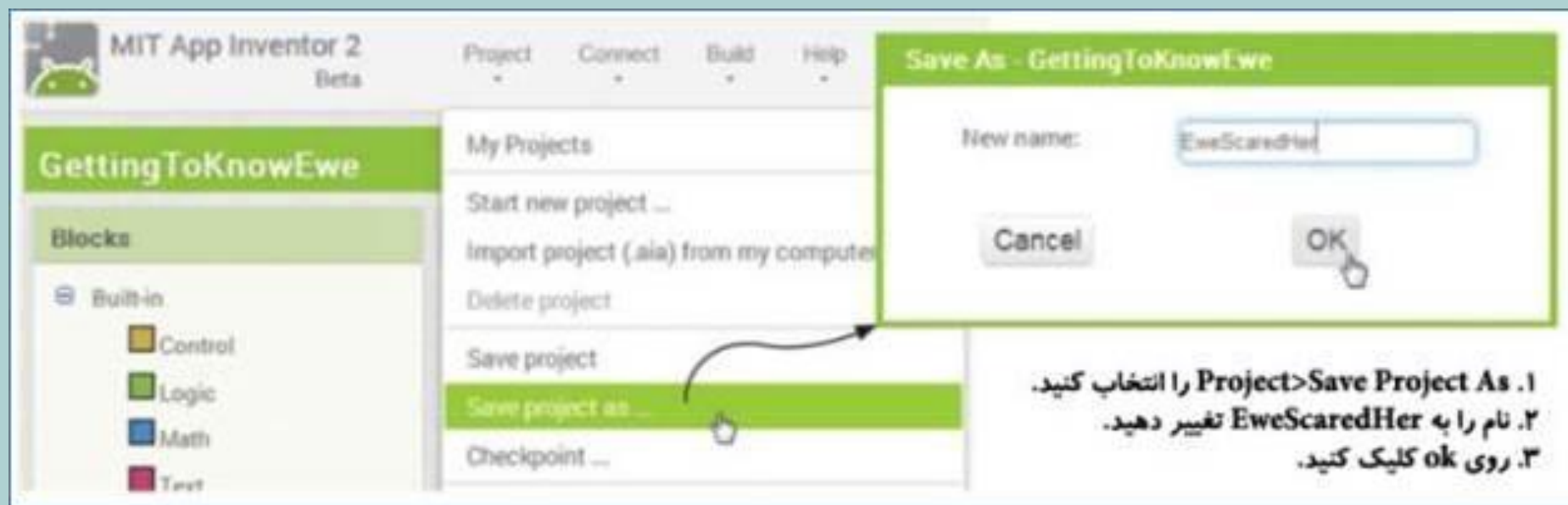
موجودیت های مورد نیاز:

یک نسخه کامل از اپلیکیشن Getting to know Ewe.



# ۱. ذخیره کردن یک پروژه جدید

برای شروع، یک کپی از Gettingtoknowewe ذخیره کنید.





## ۲. اضافه کردن کامپوننت ها: شتاب سنج، دکمه Reset و ترتیب (چیدمان) صفحه

برای ساختن این اپلیکیشن، لازم است به صفحه Designer برگشته و درباره ورودی و خروجی ها فکر کنید. ورودی ساده است: شما نیاز به فهمیدن این دارید که گوشی می لرزد. بیشتر گوشی ها یک شتاب سنج دارند که می تواند حرکت را تشخیص دهد. کامپوننت Accelerometersensor در بخش Sensord از پالت می تواند یک رویداد Shaking را تشخیص دهد. شما همچنین نیاز به یک ورودی Resetbutton دارید؛ این کامپوننت تشخیص می دهد چه زمانی یک کاربر روی دکمه Reset کلیک می کند، درست مانند Getting To Know Ewe.

حال نوبت خروجی است. هنگامی که گوشی می لرزد، Sheepbutton را مخفی کرده و Resetbutton را نمایش خواهید داد. هنگامی که دکمه Reset کلیک شود، آن را مخفی کرده و گوسفند را نمایش می دهید.

برای طراحی کلی و روی هم رفته، داشتن توانایی در مرتب کردن اشیاء در یک شبکه کمک کننده خواهد بود. برای این کار، شما می توانید با استفاده از کامپوننت Verticalarrangment دکمه را در وسط بالای صفحه و گوسفند را در وسط پایین آن قرار دهید



## ۲. اضافه کردن کامپوننت ها: شتاب سنج، دکمه Reset و ترتیب (چیدمان) صفحه

شما می توانید ببینید چگونه تمام این ها در Designer به نظر خواهند رسید.



کامپوننت ها را با درگ کردن از پالت به صفحه Designer خود اضافه کنید:



## ۲. اضافه کردن کامپوننت ها: شتاب سنج، دکمه Reset و ترتیب (چیدمان) صفحه

۱. از پالت، چهار کامپوننت زیر را اضافه کنید:

- \* یک Button
- \* یک Image
- \* یک AccelerometerSensor
- \* یک VerticalArrangement

آنها را به صفحه خود درگ کنید، اگر به ترتیب متفاوتی در صفحه ظاهر شدند، نگران شوید.

۲. دکمه جدید خود را انتخاب کرده، و نام آن را به ResetButton تغییر دهید.

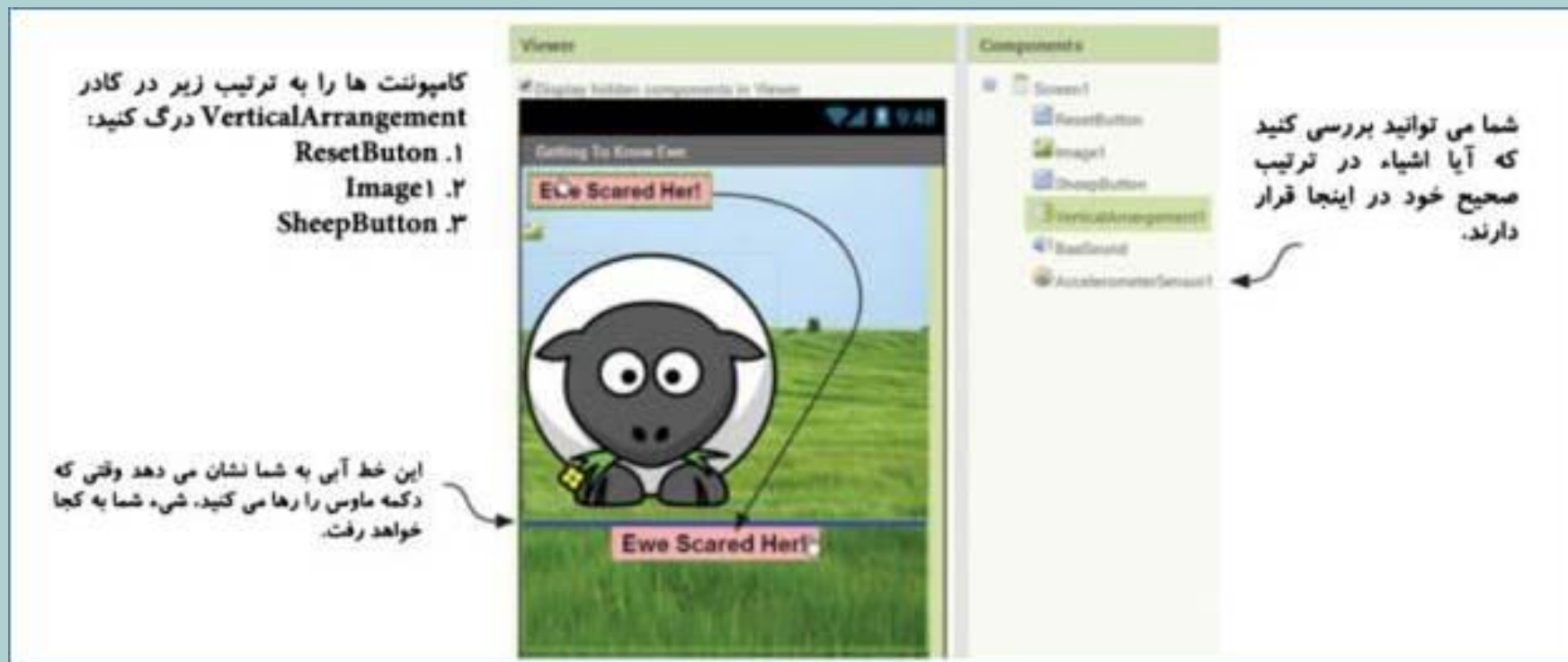
۳. خصوصیات ResetButton را تغییر دهید تا با این گزینه ها مطابقت داشته باشد.

۴. هنگامی که خصوصیت Visible را به Hidden تغییر دهید، دکمه از صفحه ناپدید می شود. برای اینکه دوباره آن را در حین کار کردن در Designer ببینید، اینجا کلیک کنید.



### ۳. تغییر چیدمان صفحه

کامپوننت Verticalarrangement را انتخاب کرده، و خصوصیات Height و Width را با Fill Parent تنظیم کنید. خصوصیت Alignhorizontal آن را با Center تنظیم کنید. کامپوننت تصویر (Image1) را انتخاب کرده و Height آن را با ۱۵۰ پیکسل تنظیم کنید؛ این یک فاصله انداز می باشد، بنابراین گوسفند به صورت ایستاده روی چمن نمایش داده می شود.





## ۴. برنامه نویسی کردن بلوک ها

با بازگشت به Blocks Editor، شما نیاز به دو بلوک جدید برای دو رویداد ورودی جدید خود دارید. از بلوک‌های Accelerometersensor1 و Resetbutton، این رویدادها را به فضای کاری خود اضافه کنید:

- Accelerometersensor1.Shaking

- Resetbutton.Click

شما باید خصوصیت Visible Sheepbutton و Resetbutton را با توجه به اینکه کدام رویداد اتفاق می افتد، تنظیم کنید. جدول زیر به شما کمک می کند:

برای تغییر نحوه نمایش یک کامپوننت روی صفحه، طوری که نمایش داده شده یا مخفی شود، خصوصیت Visible آن را با (True برای مشاهده شدن) یا (False برای مخفی شدن) تنظیم کنید.

رویداد	SheepButton	ResetButton
AccelerometerSensor1.Shaking	Hidden	Visible
ResetButton.Click	Visible	Hidden

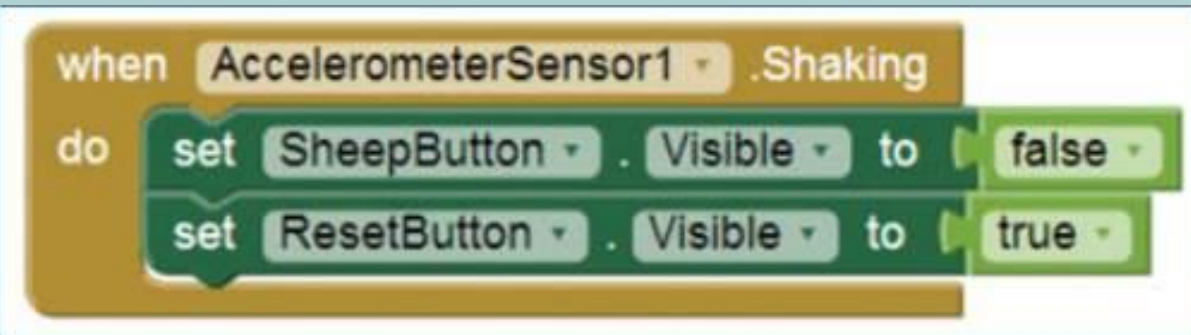
نکته: بلوک هایی که خصوصیات را تنظیم می کنند، سبز رنگ می باشند.



## ۴. برنامه نویسی کردن بلوک ها

بلوک های Set نشان داده شده از بخش های Sheepbutton و Resetbutton بلوک ها را انتخاب کنید. برای اضافه کردن بلوک های True و False، می توانید یا کلمه های True و False را در یک ناحیه خالی از Blocks Editor تایپ کرده و Enter را فشار دهید، یا به مسیر Built-In > Logic بروید.

تصویر زیر بلوک های مخفی کردن گوسفند و نشان دادن دکمه Reset را نشان می دهد. اکنون، آیا می توانید بلوک های لازم برای نشان دادن اینکه Resetbutton.Click چگونه کار می کند، را ترسیم کنید؟





## تمرین

❖ برنامه Ewe Scared Her! را تغییر دهید، طوری که به جای ناپدید شدن گوسفند، تصویر آن به یک گوسفند عصبانی تغییر یافته و متن دکمه Reset نیز به Ewe Made Her ANGRY!? تبدیل شود.



## فصل ۳: استفاده از صفحه: چیدمان‌ها و بوم

نمایش دادن اشیاء در جای مناسب بر روی صفحه مهم است، زیرا وسایل اندرویدی در اشکال و اندازه‌های متفاوتی وجود دارند. همچنین شما نمی دانید چه زمانی یک کاربر روی صفحه ضربه زده و صفحه نمایش از حالت ایستاده (Portrait) به حالت خوابیده (Landscape) تغییر حالت می دهد.

در اپلیکیشن Ewe Scared Her! در فصل دوم، از یک چیدمان عمودی برای اطمینان از اینکه آیتم‌ها به ترتیب صحیح از بالا به پایین ظاهر شوند، استفاده کردیم؛ شما همچنین از یک فاصله انداز غیرقابل نمایش برای جدا کردن گوسفند از دکمه استفاده کردید در این فصل، بیشتر به خصوصیات صفحه و کامپوننت‌های چیدمان و آرایش صفحه در پالت می پردازیم. شما این مهارت‌ها را با ساخت یک اپلیکیشن جلوه‌های صوتی تجربه خواهید کرد.



## فصل ۳: استفاده از صفحه: چیدمان ها و بوم

در نیمه دوم فصل، نگاهی می اندازیم به بوم، یک ابزار قدرتمند برای سازماندهی عناصر گرافیکی در هرجایی از صفحه. همچنین خواهید دید که بوم به شما امکان تعامل با صفحه را به روشی غیر از تنها لمس تکی که تاکنون با دکمه ها استفاده کردید، می دهد. در اپلیکیشن Graffiti Artist، شما می توانید روی صفحه نقاشی بکشید.



## چیدمان

در این بخش، کنترل صفحه را به دست گرفته و برای ساختن یک اپلیکیشن جلوه صوتی، هشت عنصر را در یک شبکه سازماندهی می کنید. به نظر ساده می رسد، اما لازم است مطمئن شوید که:

- اپلیکیشن می تواند از عهده پاسخ گویی به خم کردن یا چرخاندن گوشی توسط کاربر برآید. اگر چیدمان صفحه بچرخد، بعضی دکمه ها از پایین اپلیکیشن ناپدید خواهند شد.
- دکمه ها در یک شبکه وسط چین ظاهر شده و به صورت صحیح اندازه شوند.



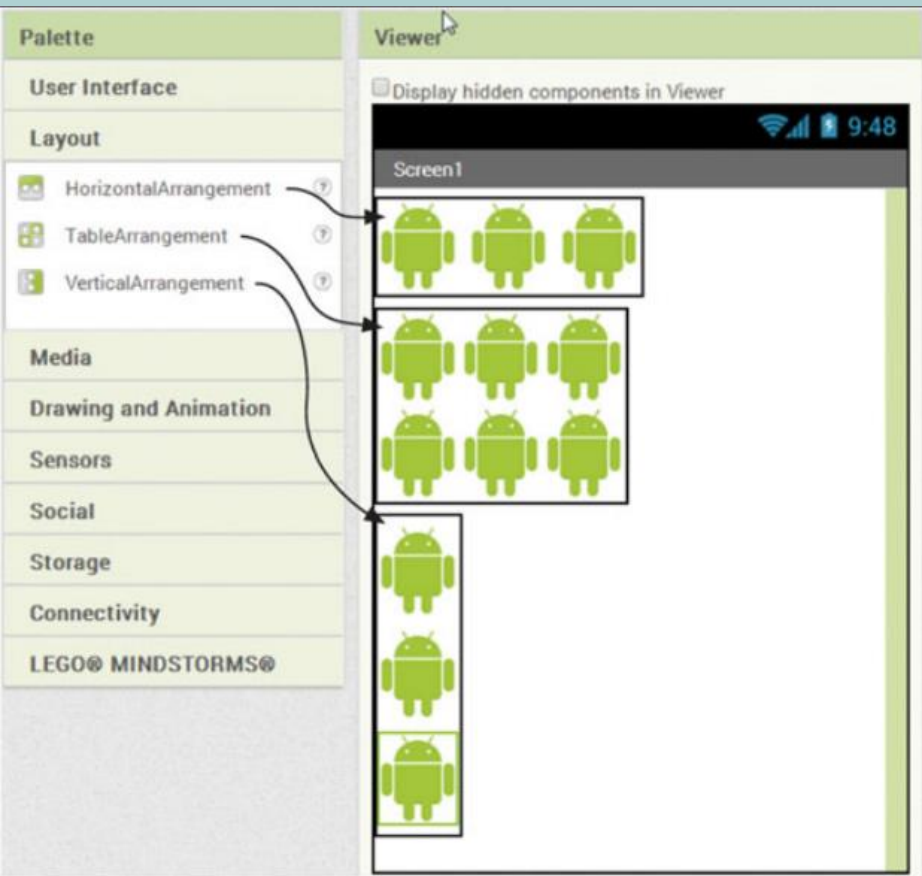
اداره کردن چرخش صفحه ساده است، گزینه ای در خصوصیات صفحه (Screen) وجود دارد که امکان چرخش را داده یا آن را متوقف می سازد. برای این اپلیکیشن، شما به صفحه نمایش اجازه چرخش را نخواهید داد.

App Inventor به طور پیش فرض کامپوننت ها را در بالای یکدیگر قرار می دهد، مگر اینکه بگویید چه کاری انجام دهد. سه شیء را در Designer درگ کنید و آنها مانند یک دسته از اجناس بر روی هم قرار می گیرند. شما می توانید ترتیب را تغییر دهید، اما اگر بخواهید اشیاء کنار یکدیگر قرار گیرند، چه کاری انجام می دهید؟



# چیدمان اشیاء کنار یکدیگر

شما می توانید از بین سه کامپوننت در بخش Layout پالت خود یکی را انتخاب کنید:



**Horizontal arrangement:** باید هنگامی استفاده شود که می خواهید کامپوننت‌ها را از چپ به راست روی صفحه قرار دهید.

**Tablearrangement** : به شما امکان مرتب کردن کامپوننت‌ها در یک شبکه را می‌دهد. شما در هر سلول تنها یک کامپوننت را می‌توانید قرار دهید. اگر تعداد بیشتری کامپوننت را قرار دهید، تنها آخرین کامپوننت نمایش داده خواهد شد.

## Vertical arrangement: به شما امکان قرار دادن کامیوننت‌ها بر روی یکدیگر را می دهد





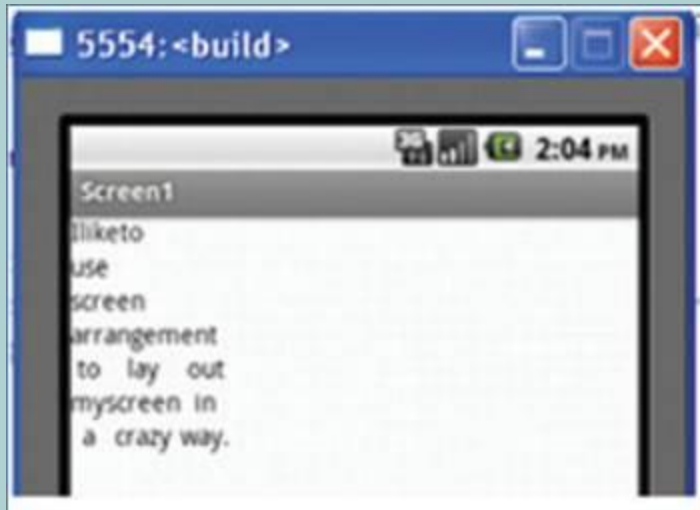
## نکته آموزشی: تست کردن چیدمان ها

همیشه چیدمان ها را روی امولاتور یا یک گوشی تست کنید. در تمام اپلیکیشن ها در این فصل، تفاوت هایی مانند خطوط در اطراف عناصر چیدمان که روی گوشی غیرقابل نمایش می باشند، خواهید دید



## چیدمان: پروژه کوچک

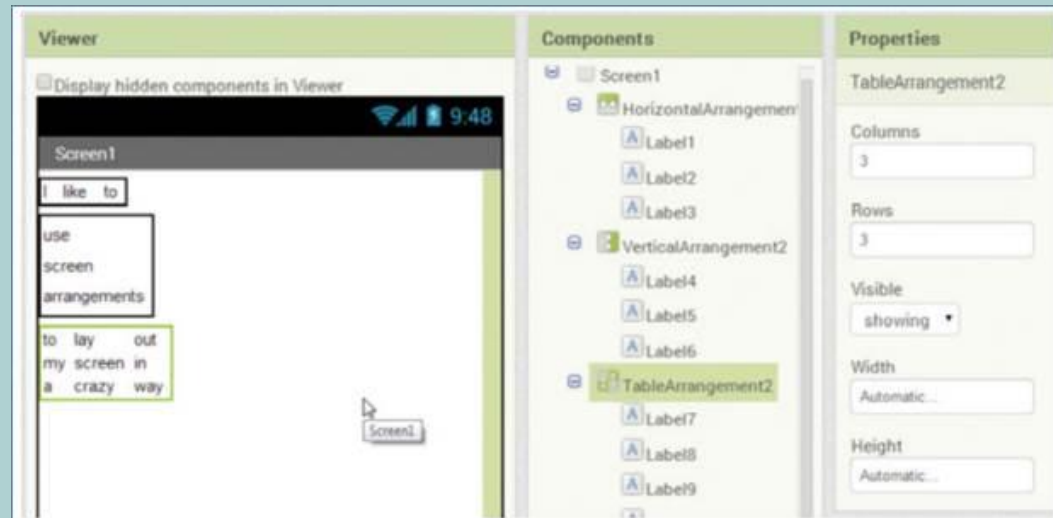
سعی کنید چیدمان جمله پایین را با استفاده از کامپوننت‌های Horizontalarrangement، Verticalarrangement و Tablearrangement دوباره تولید کنید. هر کلمه باید روی یک برچسب مجزا ایجاد شود، بنابراین شما با ۱۵ برچسب در پروژه خود آن را به پایان خواهید رساند. برای Horizontalarrangement و Verticalarrangement شما می‌توانید برچسب‌ها را با استفاده از نشان دهنده آبی رنگ برای کمک گرفتن در مشاهده اینکه برچسب در کجا ظاهر خواهد شد، در جای مناسب درگ کرده و رها کنید.



## چیدمان: پروژه کوچک



هنگام ایجاد `Tablearrangement`، عدد ستون ها و سطر ها را در `Properties` مشخص خواهید کرد. در این مورد، شما سه ستون و سه سطر می خواهید. سپس کامپوننت ها را همانند `Horizontalarrangement` و `Verticalarrangement` درگ کنید. پس از اینکه کار را تمام کردید، صفحه شما باید شبیه تصویر زیر شده باشد:



## هشدار: جلوه‌های غیر عادی گرافیکی

### جدول ها

کامپوننت Layout در App Inventor می تواند کمی عجیب و غیر عادی باشد، زیرا از سلول‌های خالی صرف‌نظر می شود. برای مثال، اگر شما یک جدول ۳x۳ ساخته و چیزی را در سلول بالا-وسط و چیزی دیگر را در سلول پایین-وسط قرار دهید، آنها در Designer از یکدیگر فاصله گرفته و دور می شوند، اما روی گوشی، فاصله موجود در وسط نادیده خواهد شد.

به علاوه، عناصر تصویر در یک جدول می خواهند با پهنای عنصر عریض تر مطابقت پیدا کنند. بنابراین اگر یک تصویر در بالا پهن بوده و دیگری در پایین باریک باشد، تصویر پایین کشیده به نظر خواهد رسید، یا ناپدید خواهد شد. شما می توانید این مشکل تصویری را با تنظیم کردن خصوصیات Height و Width هر تصویر به طور دستی برطرف کنید.



## هشدار: جلوه‌های غیر عادی گرافیکی

## شفافیت

در حالی که روی موضوع جلوه‌های غیرعادی گرافیکی قرار داریم، شفافیت می‌تواند در بعضی مواقع به صورت صحیح انجام نشود. برای مثال، اگر خصوصیت `BackgroundColor` صفحه (Screen) را با یک رنگ تنظیم کنید به جای (None) و سپس یک تصویر پس زمینه و یک کامپوننت دارای کمی شفافیت درج کنید (مانند گوسفند در فصل ۱)، روی گوشه‌های شفاف به صورت یک بلوک رنگی یکنواخت خواهند بود. خصوصیت `BackgroundColor` صفحه را دوباره با `None` تنظیم کنید تا شفافیت تصویر برگردد.



## شروع پروژه

اکنون که با سه کامپوننت چیدمان و طراحی صفحه آشنا شدیم، عنصر جدول را برای تمرین در یک اپلیکیشن جلوه‌های صوتی مورد استفاده قرار خواهیم داد. سپس، در اپلیکیشن Graffiti Artist، از یک ترتیب افقی استفاده خواهیم کرد تا به کاربر یک نوار مرتب از دکمه‌ها برای انتخاب رنگ‌ها بدهیم.





# Spooky Sound FX اپلیکیشن

### هدف از این اپلیکیشن

این یک اپلیکیشن تخته صدا است که به شما امکان پخش هشت جلوه صوتی را با لمس کردن آیکن مربوطه می دهد. شما می توانید از این اپلیکیشن برای ترساندن دوستان (یا معلم کامپیوتر خود) استفاده کنید. همچنین جلوه های صوتی را برای یک داستان ترسناک، بازی یا فیلمی که ضبط می کنید، فراهم می کند.



### درجه بازی:

### موجودیت های مورد نیاز:

## تصویر و صداهاى هشت دکمه ترسناک



## ساخت کامپوننت های مورد نیاز

برای کمک به ساخت سریع تر اپلیکیشن ها، به شما خلاصه ای از کامپوننت هایی که نیاز دارید روی صفحه درگ کنید را نشان می دهیم. مهم است که هر اپلیکیشن جدید با یک نام معنی دار شروع شود (برای مثال، Soundfx). اگر به پروژه های خود نام معنی دار ندهید، در آینده با لیستی از پروژه هایی روبرو خواهید شد که برای فهمیدن اینکه چه کاری انجام می دهند، باید یکی یکی آنها را باز کنید. این امر به سرعت شما را خسته خواهد کرد، بنابراین از الآن عادت کنید پروژه ها را به طور صحیح نام گذاری کنید.

اپلیکیشن Spooky Sound FX			
Screen1 خصوصیات	<b>AlignHorizontal:</b> Center <b>ScreenOrientation:</b> Portrait <b>Scrollable:</b> Yes (لتخاب شده) <b>Title:</b> Sound FX: Spooky <b>Icon:</b> ghost_button.png (موجود در پوشه فایل های کتاب، دانلود شده از اینترنت) <b>BackgroundColor:</b> Black (یک زمینه مشکی باعث می شود خطوط راهنما به سختی دیده شود، بنابراین پس از اینکه سایر کامپوننت های روی صفحه را مرتب کردید، آن را تنظیم کنید)		
کامپوننت ها	نام تغییر یافته آنها	چه کاری انجام می دهد؟	چه خصوصیاتی را باید تنظیم کنیم؟
TableArrangement از دسته Layout	TableArrangement1	یک شبکه ۲ ستون در ۴ سطر می سازد که دکمه های شما روی صفحه را سازماندهی می کند.	<b>Columns:</b> 2 <b>Rows:</b> 4 <b>Width:</b> Automatic <b>Height:</b> Fill Parent
Button از دسته: User Interface	GhostButton BatButton OwlButton FootstepButton CatButton WolfButton CackleButton ThunderButton	اینها دکمه هایی هستند که کاربر روی آنها کلیک می کند تا صدایی بشنود.	<b>Image:</b> فایل مرتبط با هریک را آپلود کنید: ghost_button.png برای bat_button.png, GhostButton برای BatButton و مانند آن. <b>Text:</b> هیچ متنی، متن آن را حذف کنید. <b>Width:</b> 90 pixels <b>Height:</b> 90 pixels



## ساخت کامپوننت های مورد نیاز

<p>۸ شیء <b>Sound</b> از دسته: <b>Media</b></p>	<p>GhostSound BatSound OwlSound FootstepSound CatSound WolfSound CackleSound ThunderSound</p>	<p>هشت شیء صوتی ایجاد می کنند که شما می توانید با استفاده از <b>Blocks</b> <b>Editor</b> آنها را پخش کنید، درست مانند صدای "baa" در اپلیکیشن <b>Getting to know Ewe</b>.</p>	<p><b>Source:</b> فایل مرتبط برای هر صدا را آپلود کنید: Ghost.wav برای GhostSound، bats.wav برای BatSound و مانند آن.</p>
فایل های رسانه ای			
<p>هشت فایل صوتی دانلود شده از اینترنت:</p> <p>ghost.wav bats.wav owl.wav cackling.wav thunderclap.wav cat.wav wolf.wav footstep.wav</p>		<p>خصوصیت <b>Source</b> از هر شیء صوتی با یکی از این فایل های مرتبط تنظیم خواهد شد، طوری که در زمان پخش کردن صدا توسط شما، فایل مناسب را پخش کند.</p>	
<p>هشت فایل تصویری دانلود شده از اینترنت:</p> <p>bat_button.png lightning_button.png cat_button.png owl_button.png footsteps_button.png witch_button.png ghost_button.png wolf_button.png</p>		<p>خصوصیت <b>Image</b> از هر دکمه برای نمایش دادن یکی از این تصاویر تنظیم خواهد شد.</p> 	



## ۱. تنظیم کردن خصوصیات Screen: Alignment و Scrolling

بعضی مواقع می خواهید هنگامی که گوشی می چرخد، اپلیکیشن شما جهت خود را تغییر دهد (مثال هایی را در فصل های آینده خواهید دید). اما برای این اپلیکیشن، می خواهید مطمئن شوید که اپلیکیشن در تراز عمودی (ایستاده) خود باقی بماند (حتی اگر کاربر به گوشی آهسته ضربه بزند). همچنین می خواهید در صورتی که کاربر یک صفحه نمایش اندرویدی خیلی کوچک دارد، قادر به پیمایش (اسکرول) کردن به اطراف و دیدن تمام آیکن ها باشد. در نهایت، می خواهید جدول و دکمه ها به صورت وسط چین تراز شوند. تمام این ها در خصوصیات کامپوننت Screen قابل کنترل کردن است. از جدول بالا برای تنظیم کردن تمام خصوصیات Screen1 به جز Backgroundcolor استفاده کنید. خصوصياتی که تا اینجا با آنها مواجه نشده اید، به صورت زیر می باشند:

Components	Properties	
Screen1	Screen1	خصوصیات کامپوننت Screen چگونه به نظر رسیدن و رفتار اپلیکیشن به صورت کلی (روی هم رفته) را کنترل می کند.
	AboutScreen	
	AlignHorizontal Center	۱. خصوصیت AlignHorizontal یک ترازبندی مشترک را برای تمام کامپوننت های روی صفحه تنظیم می کند.
	ScreenOrientation Portrait	۲. تنظیم کردن ScreenOrientation به Portrait جهت را ثابت نگه می دارد، حتی اگر کاربر گوشی خود را بچرخاند.
	Scrollable	۳. انتخاب خصوصیت Scrollable به این معناست که اگر بخشی از اپلیکیشن شما از صفحه خارج شد، کاربر می تواند برای دیدن آن به اطراف پیمایش کند.
Media		
ghost_button.png		





## ۲. تنظیم کردن Tablearrangement و هشت دکمه

پس از اینکه کامپوننت Tablearrangement را به صفحه اضافه کردید، مطمئن شوید که خصوصیات آن را دقیقاً مشابه آنچه در جدول اپلیکیشن در شروع این بخش توضیح داده شد، تنظیم کنید. اکنون شما یک کادر خالی با یک طرح کلی سبز رنگ را ببینید. هشت دکمه را اضافه کرده و هرکدام را در یک بخش از جدول قرار دهید. فراموش نکنید متن دکمه را حذف کرده و پهنا و ارتفاع آنها را با ۹۰

پیکسل تنظیم کنید، همان طور که در فصل ۲ انجام دادید.

پس از اینکه طرح بندی و چیدمان کامل شد، خصوصیت Backgroundcolor مربوط به Screen1 را با Black تنظیم کنید. اکنون باید یک صفحه مشکی با هشت آیکن سبزرنگ ببینید، همان طور که در تصویر واقع در بخش بعد قابل مشاهده می باشد.



### ۳. اضافه کردن ۸ کامپوننت Sound و فایل‌های مربوطه آنها

بخش پایانی این تنظیمات اضافه کردن هشت کامپوننت Sound می باشد (به یاد داشته باشید که آنها در بخش Media پالت قرار داشته و می توانید آنها را به هرجایی از صفحه درگ کنید). آنها را تغییر نام داده و خصوصیت Source آنها را تنظیم کنید تا با نام هر کدام مطابقت داشته باشد، برای مثال، فایل منبع کامپوننت WolfSound، Wolf.Wav خوانده می شود. طراحی اپلیکیشن Spooky Sound FX اکنون کامل شده و باید مانند تصویر مقابل به نظر برسد.





## ۴. کدنویسی بلوک ها

اکنون کدی که در اینجا نیاز دارید را باید شناخته باشید، دقیقاً مشابه همانی است که برای بع گفتن گوسفند در Getting To Know Ewe استفاده کردید. تفاوت این است که به جای Sheepbutton.Click، نیاز به هشت رویداد Button.Click دارید، برای هر صدا، یکی. پس از اینکه هشت رویداد Button.Click و هشت تابع Sound.Play را اضافه کردید، نتیجه نهایی باید شبیه تصویر زیر شده باشد:

برای کپی سریع بلوک های مشابه، کارهای زیر را انجام دهید:

۱. راست کلیک کرده، و Duplicate را انتخاب کنید.

۲. برای تغییر دادن بلوک ها، از فلش های پایین افتادنی استفاده کنید.



## تمرین بیشتر

در زیر تعدادی روش خلاق که توسط آنها می توانید این اپلیکیشن را تغییر داده یا دوباره بسازید، ذکر شده است:

- سطر و ستون بیشتری از دکمه‌ها اضافه کرده و تصاویر و صداها دکمه‌های موردعلاقه خود را پیدا کنید.
- یک تخته صدا با استفاده از جلوه‌های صوتی از فیلم‌ها و ژانرهای بازی موردعلاقه خود بسازید.
- یک تخته صدای دوستان و خانواده خود با استفاده از پیغام‌های ضبط شده و تصاویر آنها بسازید.
- یک تخته صدا از جوک‌های موردعلاقه خود بسازید.

### نکته

به غیر از یافتن کلیپ‌های صوتی روی وب، شما می توانید با استفاده از یک میکروفن صداها را ضبط کنید.

تعداد زیادی سایت اینترنتی وجود دارد که شما می توانید جلوه‌های صوتی و کلیپ‌های هنری برای دکمه‌های خود را از آنها دانلود کنید. برای اپلیکیشن Spooky Sound FX ما از [www.Clker.Com](http://www.Clker.Com) برای تصاویر و از [www.Freesound.Org](http://www.Freesound.Org) برای کلیپ‌های صوتی استفاده کرده ایم. کلیپ‌های صوتی را با استفاده از نرم افزار رایگان Audacity ویرایش شده است: [Http://Audacity.Sourceforge.Net](http://Audacity.Sourceforge.Net)



## نکات ضروری – هشدار: قالب‌های فایلی صدا و تصویر

کامپوننت Sound در App Inventor به صورت کلی صداها را در قالب WAV یا MP3 می‌پذیرد. بعضی مواقع با فایل‌های صوتی که حجم زیادی دارند، مشکل پیش می‌آید. بنابراین اگر با یک فایل صوتی مشکل دارید، کارهای زیر را انجام دهید:

- بررسی کنید که کوتاه باشد، کمتر از ۳۰ ثانیه.

- آن را به یک قالب دیگر تبدیل کنید. برای مثال، اگر یک فایل MP3 است، آن را به قالب WAV تبدیل کنید. شما می‌توانید از Audacity یا سرویس‌های آنلاین رایگان برای این کار استفاده کنید، مانند [Http://Media.Io](http://Media.io).

- با استفاده از Audacity و برای کم کردن حجم فایل، نرخ بیت (کیفیت) آن را کم کنید. این تنظیم را در قسمت پایین و راست صفحه پیدا خواهید کرد. ۱۱,۰۲۵ Hz یا کمتر خوب است.

- پنجره App Inventor را ببندید، اتصال گوشی را قطع کرده و سپس همه چیز را ری‌استارت کنید (اغلب از اینکه این روش چطور جواب می‌دهد، شگفت زده خواهید شد).

برای فایل‌های تصویری و گرافیکی، ما معمولاً از JPEG یا PNG استفاده می‌کنیم. تغییر دادن اندازه آنها در یک ویرایش گر گرافیکی قبل از آپلود کردن معمولاً نسبت به آپلود کردن یک تصویر بزرگ و سپس کوچک کردن آن در App Inventor بهتر است.



## معرفی کامپوننت Canvas

Canvas کامپوننتی است که کامپوننت‌های پیشرفته تر گرافیکی، انیمیشن‌ها و تعامل با کاربر را فراهم می کند. چه تفاوتی بین Canvas و Screen وجود دارد؟ همه اپلیکیشن‌ها یک صفحه ( Screen) دارند که به محض اینکه پروژه جدیدی را انتخاب می کنید، تولید می شود. شما نمی توانید نام Screen1 را تغییر دهید. Screen یک ظرف برای تمام کامپوننت هایی است که کاربر با آنها تعامل دارد. Canvas هم یک ظرف برای اشیاء و هم یک پنل مستطیلی دوبعدی حساس به لمس است که می تواند رویدادهایی مانند تماس و درگ کردن را تشخیص دهد – در فصل های بعدی اطلاعات بیشتری در این زمینه خواهیم داشت. شما می توانید روی یک Canvas ترسیم کنید، و Sprite ها (اشکال متحرک) می توانند اطراف آن حرکت کنند. در حقیقت، اگر هر انیمیشنی در اپلیکیشن خود بخواهید، باید یک کامپوننت Canvas ضمیمه کنید. به غیر از خصوصیات رنگ، پهنا، ارتفاع و تصویر پس زمینه معمول، Canvas به شما امکان تنظیم یک رنگ نقاشی و پهنای خط پیش فرض را نیز می دهد.



# اپلیکیشن Graffiti Artist

## هدف از این اپلیکیشن

هنگامی که کاربر انگشت خود (یا ماوس، در امولاتور) را روی دیوار حرکت می دهد، خطوطی را ترسیم می کند. کاربر همچنین می تواند صفحه را لمس کند تا نقاطی ایجاد کند. او همچنین می تواند رنگ نقاشی را با فشار دادن یک دکمه رنگی انتخاب کند. کلیک کردن روی Clean It Off تمام دیوارنوشته ها را پاک می کند.

3

## درجه بازی:

## موجودیت های مورد نیاز:

تصویر: Wall.png





# اپلیکیشن Graffiti Artist

یک اپلیکیشن جدید با نام Graffiti Artist ایجاد کنید.

اپلیکیشن Graffiti Artist			
Screen1	AlignHorizontal: Center AlignVertical: Center ScreenOrientation: Portrait	Scrollable: No (انتخاب نشده) Title: Graffiti Artist BackgroundColor: None	
کامپوننت ها	نام تغییر یافته آنها	چه کاری انجام می دهد؟	چه خصوصیتی را باید تنظیم کنیم؟
HorizontalArrangement از دسته Layout	ColorBar	شامل ۵ دکمه تغییر رنگ در بالای صفحه.	Width: Fill Parent Height: Automatic AlignHorizontal: Center
Button ۵ از دسته: User Interface	RedButton GreenButton WhiteButton PinkButton BlueButton	رنگ نقاشی را تغییر می دهد.	BackgroundColor: این خصوصیت را مطابق با نام دکمه تنظیم کنید. FontSize: 14 Text: مشابه یا رنگ پس زمینه (BackgroundColor) TextAlignment: Center Width: Automatic Height: Automatic
Canvas از دسته: Drawing and Animation	WallCanvas	کاربر را قادر به ترسیم و نقطه گذاشتن روی دیوار به وسیله انگشت خود می سازد.	BackgroundColor: Black BackgroundImage: Wall.png PaintColor: Red Width: Fill Parent Height: Fill Parent
Button از دسته: User Interface	WipeButton	دیوارنوشته ها را پاک می کند.	FontSize: 16 Text: "Clean It off!" TextAlignment: Center Width: Automatic Height: Automatic





# ۱. تنظیم واسط کاربری

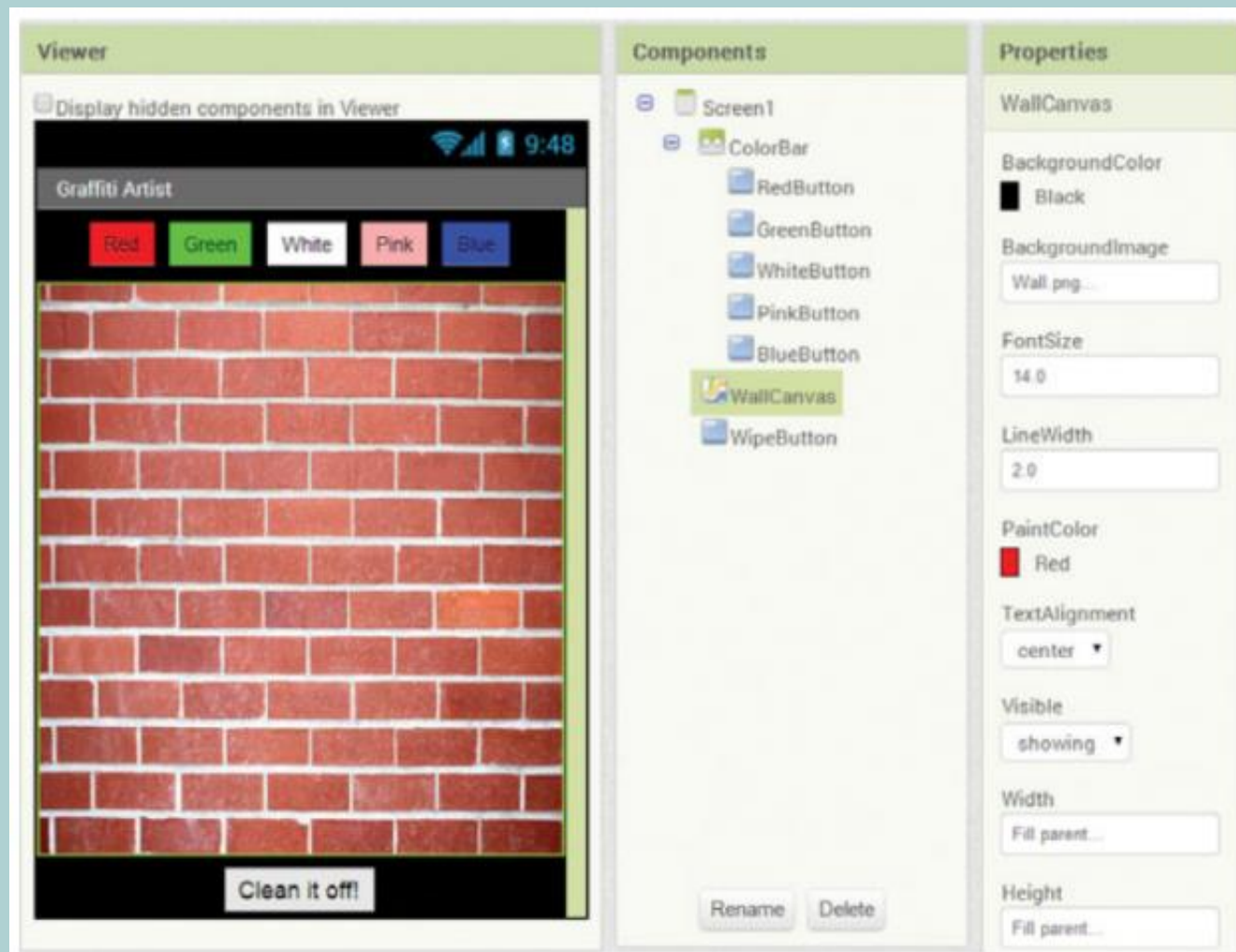
یک اپلیکیشن جدید با نام Graffiti Artist ایجاد کنید. در این اپلیکیشن، مهم است که دیوار در صورت امکان به مقدار زیاد صفحه را پر کند. اندازه فعلی استاندارد کار برای یک گوشی کوچک، ۳۲۰ پیکسل در ۴۸۰ پیکسل است (پهنا X ارتفاع). اما اندازه صفحه دستگاه‌ها (گوشی‌ها، تبلت‌ها و مانند آن) که ممکن است بخواهید اپلیکیشن خود را روی آنها استفاده کنید، متفاوت خواهد بود. به طور کلی، امتحان کردن گزینه‌های Fill Parent و Automatic برای پهنا و ارتفاع، ایده خوبی است. این بدان معناست که کامپوننت Layout تلاش می‌کند فضای آماده و در دسترس صفحه را پر کند.

از جدول اسلاید قبلی (توجه نمایید جزوه کامل را میتوانید از سایت Hrezapour.ir دریافت کنید) می‌توانید ببینید که از این گزینه‌ها برای بیشتر کامپوننت‌ها استفاده می‌کنید. نکته مهم دیگر این بخش اطمینان از این است که ۵ دکمه بالا در یک ترتیب افقی (Horizontal arrangement) و هر چیز دیگری در زیر آنها قرار بگیرند.



# ۱. تنظیم واسط کاربری

جدول دو اسلاید قبل را برای طراحی صفحه خود دنبال کنید. در نهایت، باید چیزی شبیه تصویر زیر شود:



## ۲. کدنویسی بلوک ها

اکنون که واسط کاربری خود را با Canvas، Arrangement و دکمه‌ها ایجاد کرده اید، زمان فکر کردن درباره چگونگی تعامل کاربر با اپلیکیشن می باشد. یک کاربر می تواند چهار نوع رویداد را در اپلیکیشن شما تحریک کند:

- آنها می توانند دیوار را در یک نقطه لمس کنند تا یک نقطه ترسیم شود.
- آنها می توانند انگشت خود را روی دیوار بکشند تا یک خط ترسیم شود.
- آنها می توانند رنگ نقاشی را با کلیک کردن روی یکی از دکمه‌های رنگی تغییر دهند.
- آنها برای پاک کردن دیوارنوشته‌ها روی دکمه Clean It Off! کلیک می کنند.



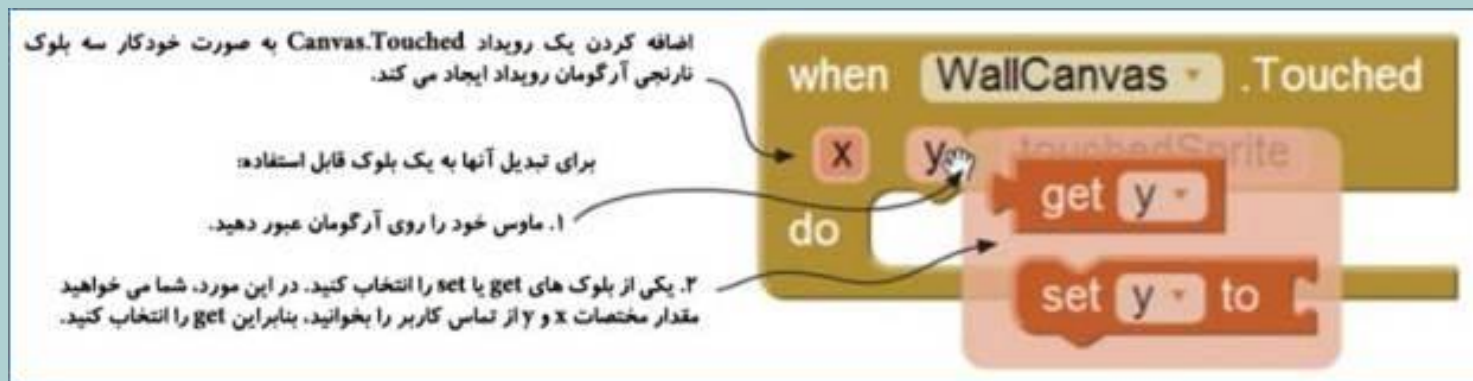
## ترسیم: ایجاد نقطه ها

زمانی که اپلیکیشن کامل شد، شما می خواهید کاربران قادر به ترسیم نقاط بر روی صفحه باشند. به منظور انجام این کار در اپلیکیشن، شما از محرک رویداد Touched استفاده می کنید. این رویداد دقیقاً مانند رویداد Click برای دکمه ها کار می کند، اما در ضمن آن به برنامه نیز می گوید که کاربر کدام نقطه صفحه را لمس کرده است. این کار مفید است، زیرا شما می توانید بگویید کاربر کجا را لمس کرده و سپس یک نقطه در آن محل ترسیم کنید. هنگامی که یک رویداد به شما اطلاعات اضافی مانند این را می دهد، به آن ها آرگومان رویداد گفته می شود. آرگومان های رویداد، متغیرهای محلی هستند، تنها آن رویداد می تواند آنها را ببیند. درباره آنها اسلاید های آینده بحث خواهیم کرد.



## ترسیم: ایجاد نقطه ها

به Blocks Editor بروید. از Wallcanvas، بلوک اداره کننده رویداد Wallcanvas.Touched را انتخاب کنید. توجه کرده اید که بلوک دارای سه مستطیل نارنجی اضافی دقیقاً در زیر نام خود می باشد؟ آنها آرگومان های رویداد می باشند، بخش های مفید اطلاعات درباره مکانی از بوم که کاربر لمس کرده است. در این اپلیکیشن، آرگومان های رویدادی که درباره آنها بحث می کنیم، محورهای X و Y می باشند، شناخت آنها بدین معناست که شما می توانید یک نقطه در یک جای مشخص ترسیم کنید. برای استفاده از این مقادیر بلوک نارنجی در سایر بخش های برنامه خود، نیاز به عبور دادن اشاره گر ماوس خود روی آنها دارید (بدون کلیک کردن) و سپس روی گزینه Get کلیک کنید. احتمالاً درباره سومین بلوک نارنجی یعنی Touchedsprite تعجب کرده اید. این به شما نام هر Sprite که کاربر لمس کرده را می دهد، اما شما در این اپلیکیشن هیچ Sprite ای ندارید، بنابراین تا فصل ۴ نیازی به نگرانی درباره آن وجود ندارد





## ترسیم: ایجاد نقطه ها

برای ترسیم یک نقطه (که در App Inventor یک دایره خوانده می شود) در جایی که کاربر لمس کرده، مراحل زیر را انجام دهید:

۱- از Wallcanvas، بلوک Wallcanvas.Drawcircle را انتخاب کرده و آن را به اداره کننده رویداد Wallcanvas.Touched وصل کنید.

۲- بلوک های Get X و Get Y را از آرگومان های نارنجی رویداد Wallcanvas.Touched انتخاب کرده و در شکاف های X و Y در بلوک بنفش رنگ Wallcanvas.Drawcircle رها کنید.

۳- مقدار (Radius: R شعاع) نقطه یا دایره ای که ترسیم کرده اید، هنوز باقی مانده است. یک بلوک Number با مقدار ۵ اضافه کنید. شما می توانید یا روی Wallcanvas.Touched

کلیک کرده و عدد ۵ را تایپ کرده و Enter را فشار دهید؛ یا از بلوک های Built-In، Math (بلوک آبی صفر) را انتخاب کرده و سپس مقدار را به ۵ تغییر دهید.

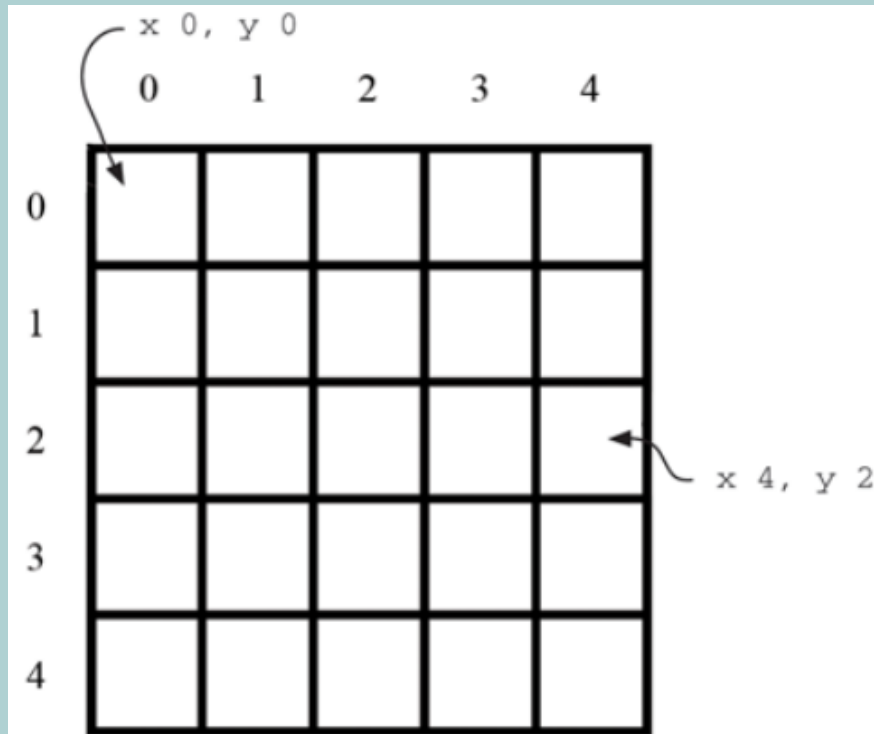
بلوک نقطه گذاری پس از اتمام باید شبیه تصویر زیر شود:



# نکته آموزشی: اصطلاحات ریاضی – مختصات و شعاع

مختصات X و Y

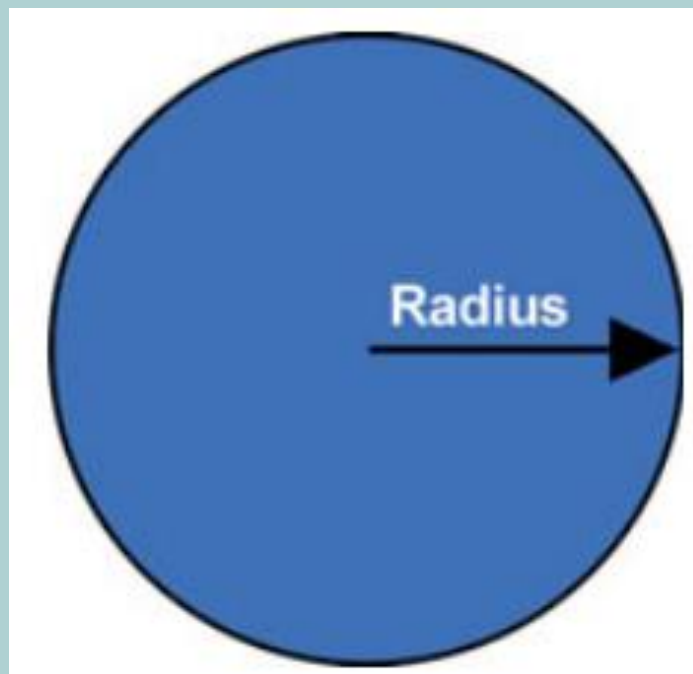
Canvas یک جدول از پیکسل‌ها است. یک پیکسل کوچک ترین نقطه ای است که می تواند روی گوشی ظاهر شود. هرکدام از این پیکسل‌ها یک موقعیت روی Canvas دارند، که با استفاده از یک سیستم مختصاتی X و Y تنظیم می شود. X نشان می دهد نقطه در چه مکانی در طرح چپ-راست (افقی) قرار دارد و Y نشان می دهد در چه مکانی در طرح بالا-پایین (عمودی) قرار دارد.



# نکته آموزشی: اصطلاحات ریاضی – مختصات و شعاع

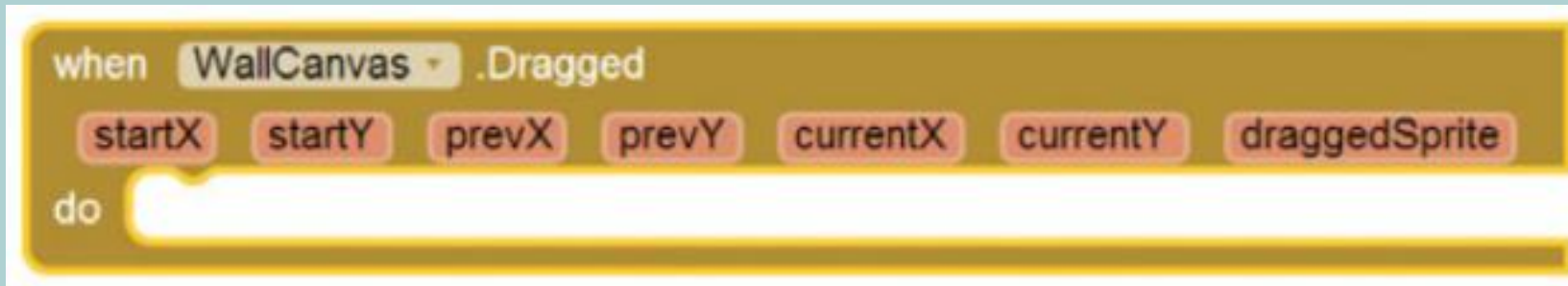
شعاع

کمی ریاضی! شعاع یک دایره، فاصله وسط دایره تا لبه آن می باشد. در این مورد، شعاع بر حسب پیکسل اندازه گیری می شود.



## ترسیم: ترسیم خطوط

هنگامی که از انگشت خود برای ترسیم خطوط یا منحنی‌ها روی یک صفحه استفاده می‌کنید، در حقیقت تعداد زیادی خط راست کوچک ترسیم می‌کنید. برای این کار، از محرک رویداد Dragged استفاده می‌کنید. این رویداد هنگامی تحریک می‌شود که کاربر انگشت خود را روی صفحه درگ کند (بلغزاند). از بلوک‌های Wallcanvas، اداره‌کننده رویداد Wallcanvas.Dragged را انتخاب کنید:

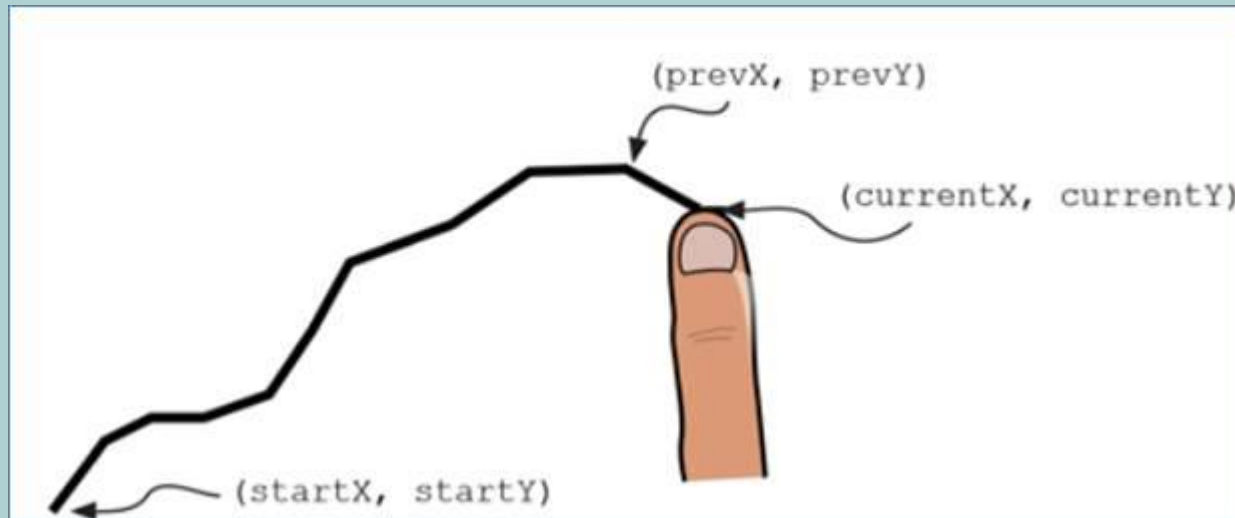


## ترسیم: ترسیم خطوط

درست مانند رویداد Touched، خواهید دید که تعداد زیادی آرگومان جدید رویداد ایجاد کرده است. درک کردن آنها نیاز به کمی مهارت دارد، زیرا درگ کردن روی گوشی ساده بوده، اما توصیف آن در یک زبان برنامه نویسی (یا حتی در انگلیسی) این طور نیست!

یک رویداد Draggd هر زمان که کاربر انگشت خود را روی صفحه حرکت دهد، تحریک می شود. برخلاف یک دکمه، که در هر بار کلیک یکی از آنها را ثبت می کند، یک رویداد Draggd تا زمانی که کاربر انگشت خود را به اطراف حرکت می دهد، تحریک شدن را حفظ می کند، درست تا لحظه ای که کاربر انگشت خود را از روی صفحه بردارد.

در آرگومان های نارنجی رنگ رویداد، `Startx` و `Starty` را خواهید دید؛ اینها ویژه هستند، زیرا تنها هنگامی تغییر می کنند که یک کاربر شروع به یک درگ جدید کند. موقعیت فعلی انگشت شما همیشه در `Currentx` و `Currenty` ذخیره می شود. هنگامی که انگشت شما متوقف شده یا تغییر مسیر می دهد (حتی برای یک لحظه)، موقعیت آن در `Prevx` و `Prevy` ذخیره می شود.





## ترسیم: ترسیم خطوط

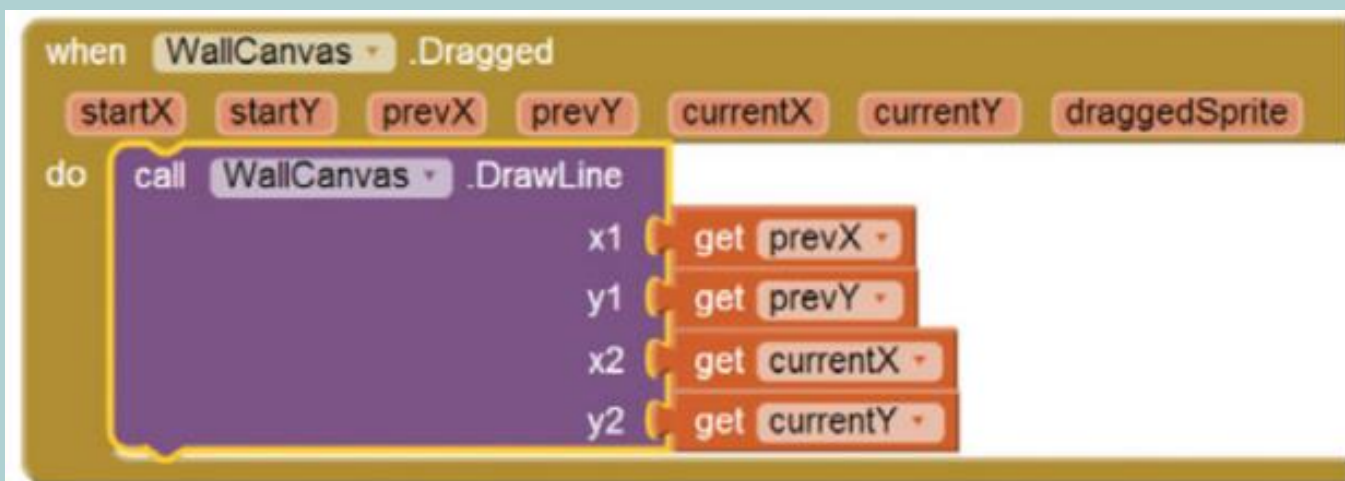
یک خط درگ شده، از تعداد زیادی خطوط راست ریز با مقادیری از PrevX و PrevY و CurrentX و CurrentY که دائماً در حال تغییر است، تشکیل شده است.

برای ترسیم یک خط در جایی که کاربر درگ می کند، مراحل زیر را دنبال کنید:

۱- از بلوک WallCanvas، بلوک WallCanvas.DrawLine را انتخاب کرده و آن را به بلوک WallCanvas.Dragged وصل کنید.

۲- از آرگومان‌های رویداد WallCanvas.Dragged، آرگومان‌های PrevX، PrevY، CurrentX و CurrentY را در شکاف‌های مرتبط روی بلوک WallCanvas.DrawLine درج کنید.

بلوک‌های شما اکنون باید به صورت زیر باشد:



## نکته آموزشی: آرگومان ها

آرگومان ها (در علم کامپیوتر) مقادیر ویژه ای هستند که یک تابع قبل از اینکه بتواند کار کند، باید آنها را بشناسد. در این اپلیکیشن، آرگومان های نقطه گذاری و درگ کردن، مختصات انگشت کاربر بر روی صفحه، در هنگام لمس کردن یا درگ کردن صفحه می باشند.

تغییر رنگ نقاشی

رنگ فعلی که روی بوم نقاشی می شود، یک خصوصیت از Canvas است که می تواند با هر رنگی تنظیم شود:

۱- از بلوک Bluebutton، رویداد Bluebutton.Click را انتخاب کنید.

۲- از بلوک Wallcanvas، بلوک Wallcanvas.Paintcolor را انتخاب کرده و آن را درون اداره کننده رویداد Bluebutton.Click قرار دهید.

۳- از رنگ های تعبیه شده، بلوک آبی را انتخاب کرده و آن را درون بلوک Set Wallcanvas.Paintcolor قرار دهید.

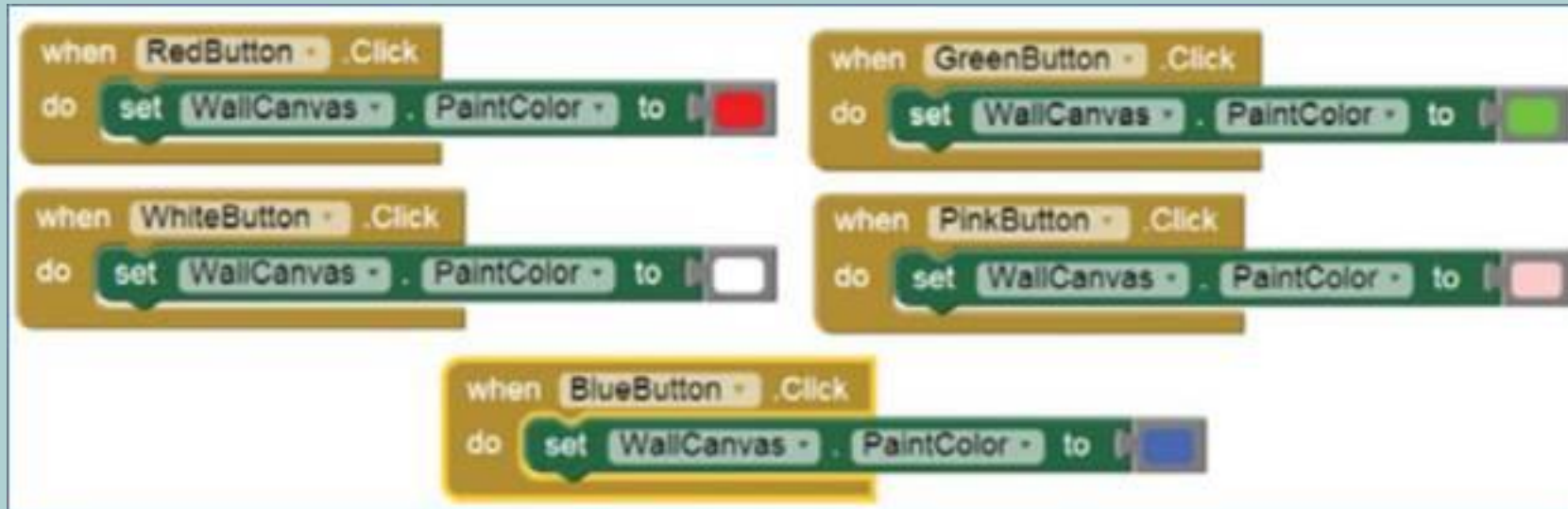
۴- مراحل بالا را برای تمام رنگ ها تکرار کنید تا کاربر قادر به تغییر دادن رنگ نقاشی شود. یک روش سریع برای این کار، تکثیر (Duplicate) بلوکی که ایجاد کرده اید، (راست

کلیک و انتخاب (Duplicate) و سپس تغییر دادن نام دکمه و رنگ می باشد.



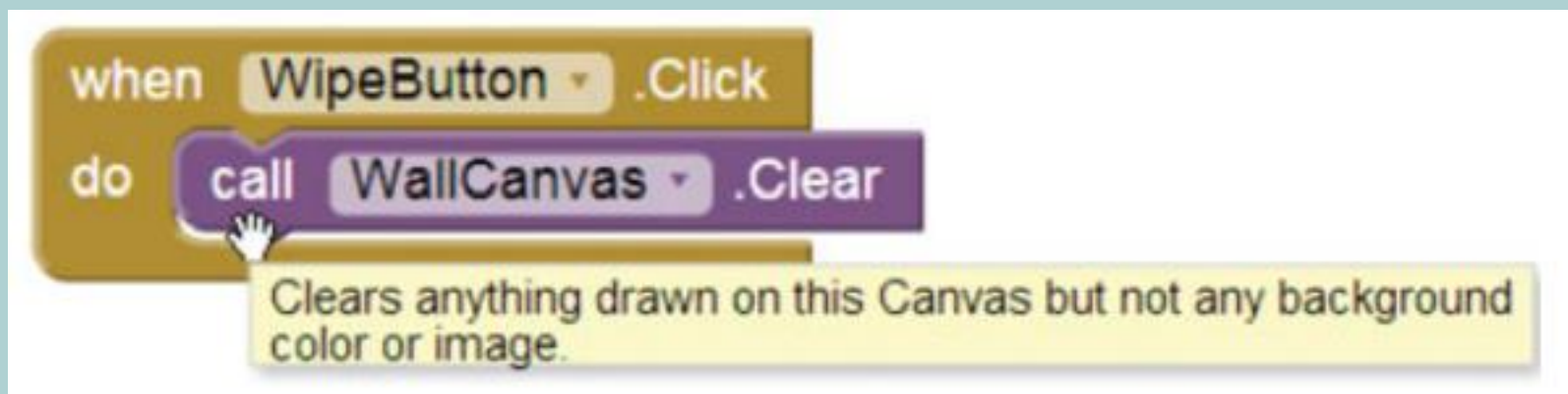
## نکته آموزشی: آرگومان ها

ما رنگ های قرمز، سبز، سفید، صورتی و آبی را انتخاب کرده ایم، البته شما می توانید آنها را تغییر داده یا اضافه کنید.



## پاک کردن دیوار

پس از اینکه کاربر نوشته‌های خود را روی سرتاسر دیوار نوشت و نقاشی کشید، احتمالاً به دنبال روشی برای پاک کردن آن و شروع دوباره خواهد بود. پاک کردن بوم ساده است، شما تنها نیاز به فراخوانی یک تابع `Canvas.Clear` دارید. در این مورد، `WallCanvas.Clear` را زمانی فراخوانی کنید که کاربر روی `Wipebutton` کلیک کرده باشد. آنها را مانند تصویر زیر داخل یکدیگر قرار دهید:



توجه کنید که چگونه عبور دادن اشاره گر ماوس روی یک بلوک به شما اطلاعات مفیدی درباره کاری که آن بلوک انجام می دهد، خواهد داد.



## سنجش

## ۱- چه اتفاقی می افتد اگر دو دکمه در سلولی مشابه از Tablearrangement قرار دهید؟

## ۲- تفاوت بین یک رویداد Dragged و یک رویداد Touched چیست؟

### ۳- چرا رویداد Dragged به مختصات شروع، قبل و فعلی نیاز دارد؟

