

Brief Inventory of ApplInventor Example Projects

*Note: some project require texting enabled devices, which will not be used at workshop

No Texting While Driving (requires texting-enabled device)

Concepts/Components introduced: Texting, Inputting Data, TinyDB, Location Sensor, Text to Speech, Screen Initialize

Commentary: This program has four main event blocks, which makes it very readable once completed. It is a good way to practice passing information along from function to function, within a program. Also good practice for database management and SMS text processing. The tutorial will have you construct the program by building one piece of its functionality at a time. Moderate amount of code, and focusing on one function at a time makes the program easy to understand and build. User interface/components are simple.

Ladybug Chase

Concepts/Components introduced: Detecting Collisions, Orientation Sensor (to control ImageSprite), Clock (to control events), variables (to keep track of numbers without showing the end user), parameters (in procedures)

(Assumes you know: procedure creation, random-number generation, ifelse block, ImageSprite, Canvas, Sound and Clock....from MoleMash)

Commentary: This app is very fun, because the end product is a game. However, there is large amount of code, which gets a bit confusing and complicated as you are in the building process, but the main concepts involved are quite simple. This program is probably the most intricate use of basic concepts, and extension of MoleMash.

Map Tour/Paris Map Tour

Concepts/Components introduced: Activity Starter (and its properties, including DataUri's), List Picker, defining variables as lists

Commentary: The code for this app is very short and simple. The main purpose of this app is to introduce the Activity Starter component, along with the List Picker component, both of which can prove very useful in future creation of apps. These concepts, however, are much more complicated than the basic concepts presented in MoleMash.

Where's My Car

Concepts/Components introduced: Location Sensor, TinyDB, Activity Starter

Commentary: More complicated implementation of Activity Starter than that in ParisMapTour. Also, uses TinyDB to store data persistently and directly on phone database, so info can be recalled once the app is relaunched after closing it. Good practice of manipulating labels on user interface and memory management on the phone. The thinking process behind the coding takes longer than the actual construction of it. Short amount of code.

Presidents Quiz

Concepts/Components introduced: Indices (to sequence through lists), conditional behaviors (if)

Commentary: Assumes you are familiar with all of the basics of AppInventor and the fundamental concepts of programming. This program introduces index sequencing, an important and useful programming skill. The final code is clear and moderate in length. This concepts introduced in this program bridge well with the concepts of other programming languages, like C.

Xylophone

Concepts/Components introduced: Sound (one component to play more than one sound), Clock (to measure and enforce delays), Procedures (deciding when to create them, call to itself), Lists (advanced use), Refactoring

Commentary: This program is very rewarding once completed. Large amount of code involved, but overall the code is quite repetitive. Introduces refactoring, a very useful programming skill. Great practice for creating and managing procedures within a program.

MakeQuiz and TakeQuiz

Concepts/Components introduced: TinyWebDB (storing and retrieving info from WebDB, management of user interface

Commentary: This program is a more generalized version of the PresidentsQuiz App, as the Question and Answers are entered dynamically by the user. Good for management of user interface. Large amount of code, but having done the PresidentsQuiz app, there are very few adds on's and edits.

Broadcast Hub

(requires texting-enabled device)

Concepts/Components introduced: Looping ("foreach" and embedding), TinyDB, Texting (SMS processing)

Commentary: The concepts aren't too advanced, and having practice with the components, this program shouldn't be difficult. Moderate amount of code, however it gets a bit sticky with the embedded loops.

Amazon at the Bookstore

Concepts/Components introduced: Barcode Scanner, TinyWebDB (accessing and processing web info), API's

Commentary: Very good practice of using the TinyWebDB component, and familiarization of API's. The blocks and code are very clean and discrete. Uses four variables and embedded loops.