

LESSON 5 NOTES

Mobile App Development Part I



shawacademy

Storing Persistent Data

A **Database** is used to Store Data about the App even after the App has been closed. It is known as a collection of related data that can be organized.

Persistent Data is information that is infrequently accessed – such as Username, Password, Preferences, Images or Files – that are unlikely to be modified or changed. Data that we would like to keep even after the End-User has shut off the mobile device.

Local Storage This is a Database that can be stored locally on the Machine without needing to be backed-up to a Server online. In App Inventor, we are able to use **TinyDB** to store values like a Login Password or Search Term to store a User's Personal Information Securely.

TinyDB uses a **Tag/Value** Pairing System to store the information locally on the device. There is no inherent limit of the number of tags the TinyDB can store, only space limitations involved are whether your device has sufficient storage.

There is also a **TinyWebDB** that can be used for interacting with a Test Server to make your App available online. This Test Server is great during Development & Debugging Stages for your App however, upon publishing your app, you will need to update this Test Server to your own Server for Storing User Information Securely such as SQLite or a Content Provider.

Project Sample from App Inventor 2

To get the most from any programming language, the best thing to do is practice. The following examples will get you up and running with a Database in your App:

A Personal Assistant App:

This App uses speech-to-text API to Record your Thoughts. It can only be accessed through a password-protected screen.

Screen 1 Properties

BackgroundImage:	yourLogo.png
Icon:	yourIcon.png
Title:	My PA App

Components for Screen 1:

TextBox	PasswordInput	BackgroundColor:	Cyan
		FontSize:	20
		TextColor:	Blue
		Width:	100 pixels
		Hint:	Password
Button	SubmitBtn	BackgroundColor:	Blue
		FontSize:	14
		Shape:	Rounded
		Text:	Submit
Notifier	Error404Notifier		

Screen 2 Properties

BackgroundColor: Cyan
Title: ThoughtsScreen

Components for Screen 2

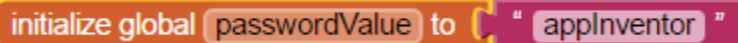
Button	ThoughtsBtn	BackgroundColor: Blue
		FontSize: 20
		Text: "Talk to Me"
Label	ThoughtDesc	FontSize: 14
		Text: "-Your Thoughts Here-"

Speech-Recognizer

TinyDB

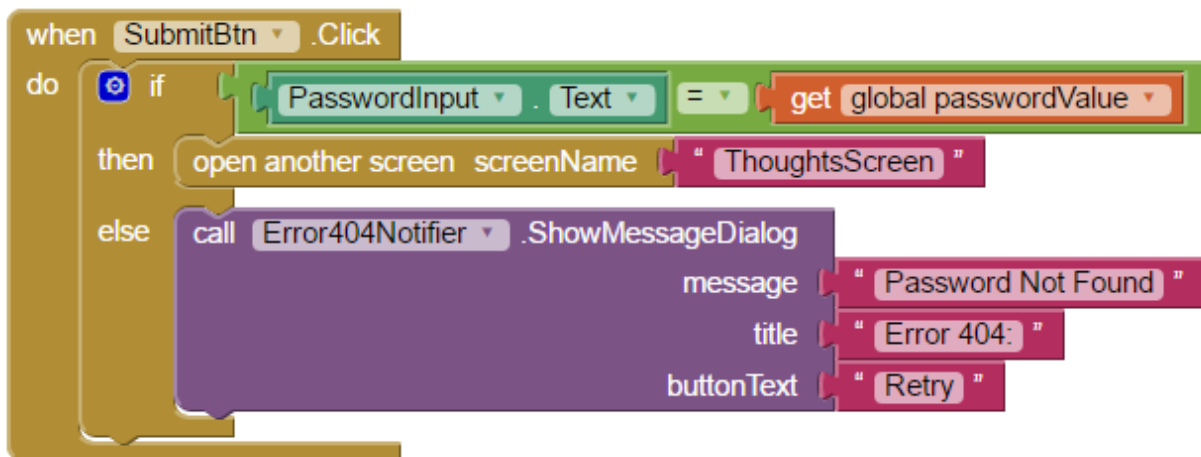
Adding Functionality:

Step 1: Create the Password:



initialize global passwordValue to "applInventor"

Step 2: Check the Password:



```
when SubmitBtn .Click
do
  if PasswordInput .Text = get global passwordValue
  then open another screen screenName "ThoughtsScreen"
  else call Error404Notifier .ShowMessageDialog
        message "Password Not Found"
        title "Error 404:"
        buttonText "Retry"
```

Next Open the SecondScreen in your App to Record your Input.

Step 3: Record your Thoughts:

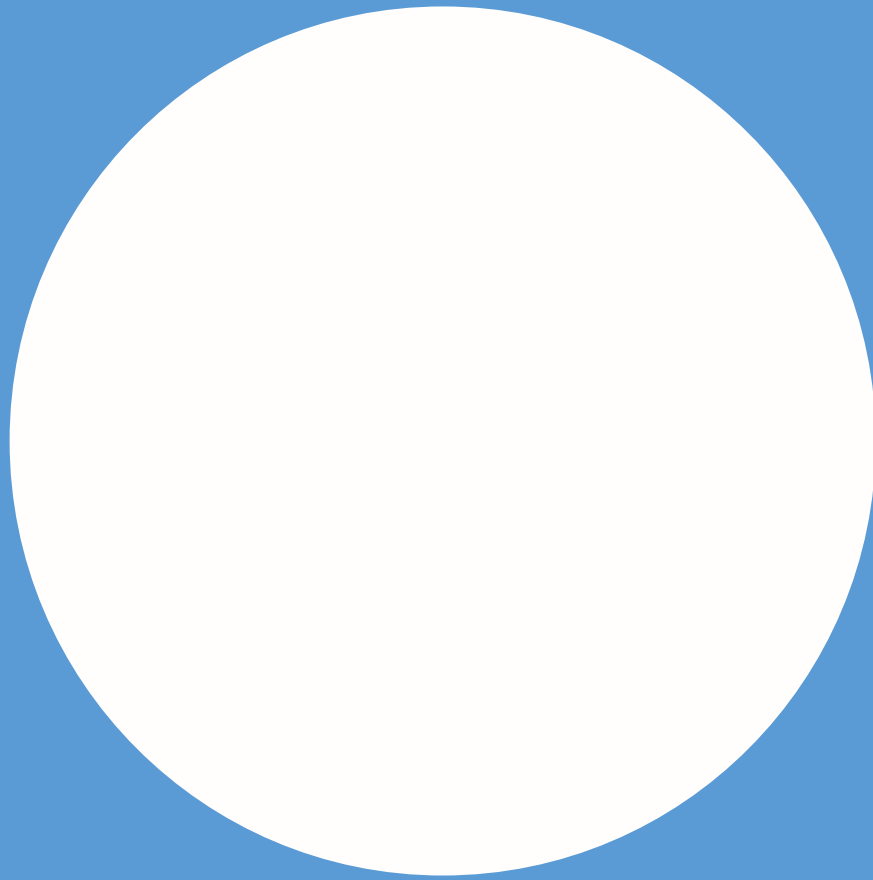
```
when ThoughtsBtn .Click
do call SpeechRecognizer1 .GetText
```

Step 4: Convert Speech to Text & Store the Data:

```
when SpeechRecognizer1 .AfterGettingText
result
do set ThoughtsDesc . Text to get result
   call TinyDB1 .StoreValue
      tag "Thoughts"
      valueToStore get result
```

Step 5: Set Up the Database for First Use:

```
when ThoughtsScreen .Initialize
do set ThoughtsDesc . Text to call TinyDB1 .GetValue
   tag "Thoughts"
   valueIfTagNotThere ""
```



mobile.app@shawacademy.com



shawacademy