



Hello App Inventor!

Errata

Thank you for purchasing [Hello App Inventor](#). Please post any errors, other than those listed below, in the book's [Author Online Forum](#). We'll compile a comprehensive list and publish it here for everyone's convenience. Thank you!

Last updated December 22, 2015

Errata from the first print run

1. Replacing GIF image files with PNG files

Some recent Android devices do not render GIF transparency correctly. This particularly affects the Creepy Spider, Hungry Spider and A-Mazeing penguin apps. Alternative PNG graphics have been provided and references to GIF files in the book will be amended to PNG. This affects:

- Page 177 - Add this text to the end of the Exploding GIFs Learning Point "Once you have extracted your separate GIFs it is always a good idea to convert them to PNG files because some Android devices don't correctly display GIFs that have transparent areas, but PNGs work fine."
- Page 176 and 177 - Replace all filenames ending in .gif with .png
- Page 178 top image - Replace all occurrences of . gif with .png
- Page 193 and 194 - Replace Fly.Gif with Fly.png
- Page 194 image - Change the Picture property to read "Spider1frame.png"
- Page 295 - Replace all filenames ending in .gif with .png
- Page 298 - Replace all filenames ending in .gif with .png and replace the ".gif" block in the top image with a ".png" block
- Page 300 - Replace the ".gif" block in the top image with a ".png" block
- Page 302 - Replace the ".gif" block in the top image with a ".png" block
 - Page 304 - Replace the ".gif" block in the top image with a ".png" block

2. Creepy Spider faces left, but Hungry Spider must face right

The spider images used in Hungry spider need to be mirror images of those in Creepy spider. Therefore:

Page 193 - Add to Assets You'll Need - "10 new spider images because we need our spider to face right, replace spider1frame.png to spider10frame.png with those in the Chapter 9 resources file.

3. Weather Api Changed

The third party WWO weather API used in chapter 12 has been updated, requires additional parameters, and now returns its data in 3 rows.

- Page 243 - Replace both occurrences of "maximum and minimum temperatures" with "average temperature"

- Page 245 - Add a new row to the table between "cc" and "show_comments" that reads Parameter "includelocation" Value "no" Explanation "Provides the location of the weather station where readings were taken. To keep things simple we set it to No. Append new row to table as follows: Parameter "tp" Value "24" Explanation "Specifies we want average weather over a 24 hour period."
- Page 246 - Replace api call (2nd paragraph) with: "api.worldweatheronline.com/free/v2/weather.ashx?q=orlando&format=csv&num_of_days=1&cc=no&includelocation=no&show_comments=no&tp=24&key= your key goes here"
Add new bullet point after "&cc=no" which reads "&includelocation=no - Don't send the location of the weather station."
Add new bullet point after "&show_comments=" which reads "&tp=24 - gets the average weather over a 24 hour period."
Replace the image with the output of the new API call
Replace "You'll tell the user a decription of the weather (weatherDesc) the minimum and maximum temperatures in Celsius (tempMinC, tempMaxC) and Farenheit (tempMinF, tempMaxF) and the rainfall in mm (precipMM)."
With "You'll tell the user a decription of the weather (weatherDesc) the average temperature in Celsius (tempC) and Farenheit (tempF) and the rainfall in mm (precipMM)."
Replace the api address at the bottom of the page with http://api.worldweatheronline.com/free/v2/weather.ashx?q=Orlando&ormat=CSV&num_of_days=1&cc=no&includelocation=no&show_comments=no&tp=24&key= your key goes here
- Page 247 - Replace the top image with the output of the new API call
- Page 248 - Replace the Api call blocks in all three illustrations (top and bottom block in each) with the new Api call
- Page 249 - Replace the two illustrations with those below:

The screenshot shows the MIT AI2 Companion app interface on the right, displaying a list of weather data for 2015-01-15. The list includes values for 24, 22, 72, 9, 15, and 341. On the left, the App Inventor code blocks are shown with several annotations:

- Initialize global:** "WeatherList" to "create empty list". Annotation: "You store any data received in a list called WeatherList."
- When Web1 - GotText:** A block with "uri", "responseCode", "responseType", and "responseContent" fields. Annotation: "The list of data sent is stored in a local variable called responseContent."
- Do:** A block containing "set global WeatherList to select list item list list from csv table text" and "get responseContent".
- Set:** "WeatherListPicker" to "Elements" and "get global WeatherList".
- Annotation:** "Setting WeatherListPicker to the contents of WeatherList means you can click it to see what has been received."
- Annotation:** "These blocks convert responseContent (the 3 rows of data we received from the API) into an App Inventor list using a list from csv table block. The 3rd row of data is stored in WeatherList."

- Page 259 - Replace the illustration as below:

```

when Web1 . GotText
url responseCode responseType responseContent
do
set global WeatherList to select list item list list from csv table text get responseContent
index 3
set WeatherListPicker . Elements to get global WeatherList
set WeatherLabel . Text to join " Today the weather is ...n "
select list item list get global WeatherList
index 11
" n "
set RainfallLabel . Text to join " Rainfall - "
select list item list get global WeatherList
index 12
" mm/n "
set FahrenheitLabel . Text to join " Fahrenheit: "
select list item list get global WeatherList
index 4
set CelsiusLabel . Text to join " Celsius: "
select list item list get global WeatherList
index 3

```

These blocks select the data items we want to display from the elements in the WeatherList (11 = weatherDesc, 12 = precipMM, 4 = tempF, 3 = tempC. The line breaks and explanation text are added to help the user make sense of the data.