Cook Book for Toolies

Gov Hack 2017

Table of Contents

Chapter 1

Objective

To showcase the ease in using own/open Data Sets (.csv,.xls) with Mobile Applications.

Target Audience

High School IT students / individuals with entry level programming experience / interest to unleash Open Data Sets.

In Short

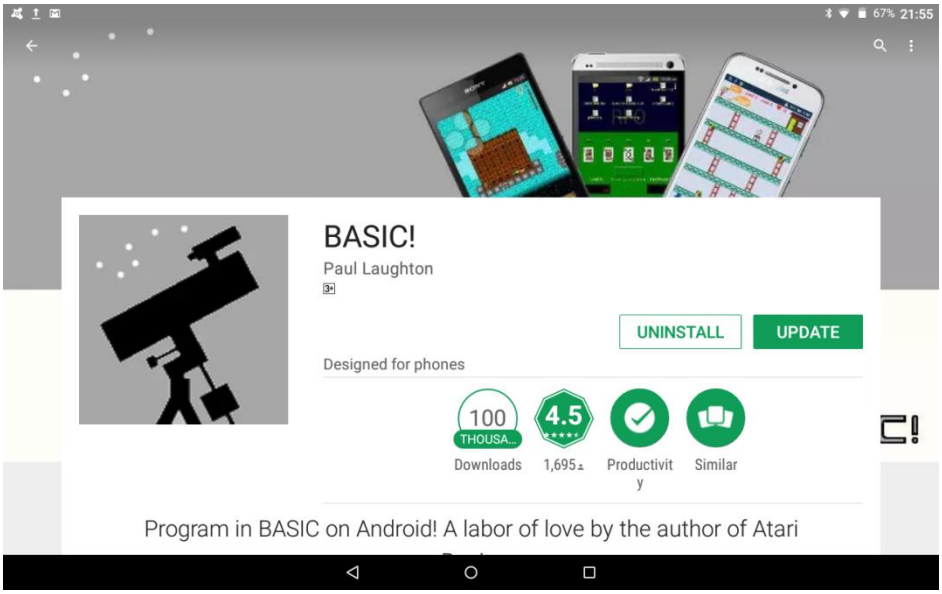
This Cook Book offers step by step instructions for successfully installing and running the sample application Toolies_Demo.bas which was developed on an Android tablet using RFO BASIC as part of the GovHack 2017 Competition.

Chapter 2

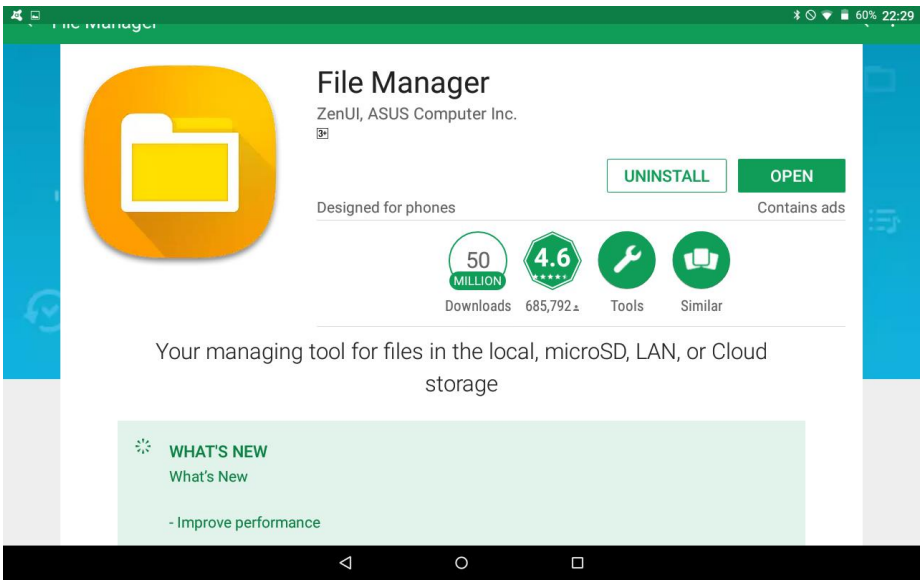
Pre-Requisite

1- An Android Phone or Tablet.

2- Download BASIC! (Paul Laughton) from Google Play (for free).



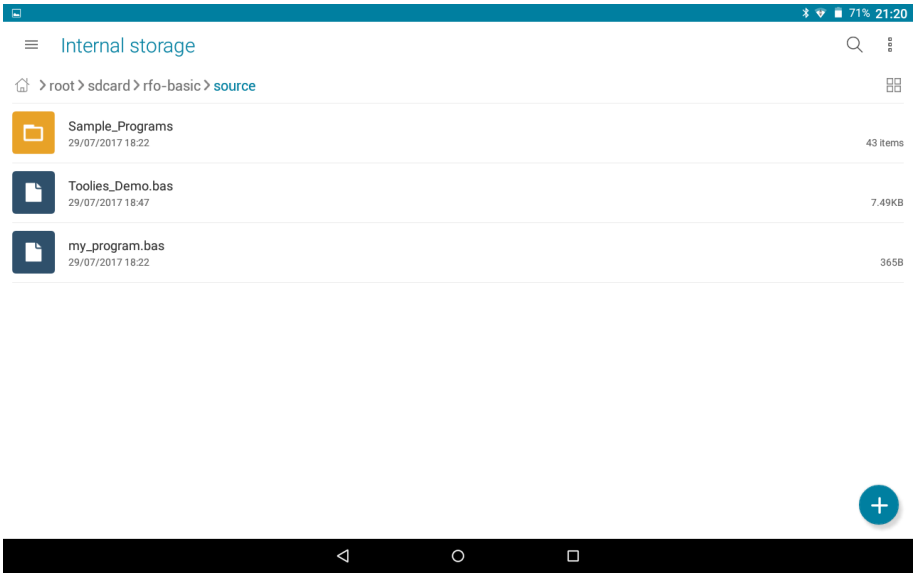
3- We recommend downloading File Manager (ZenUI, ASUS Comptuer Inc.) from Google Play (for free).



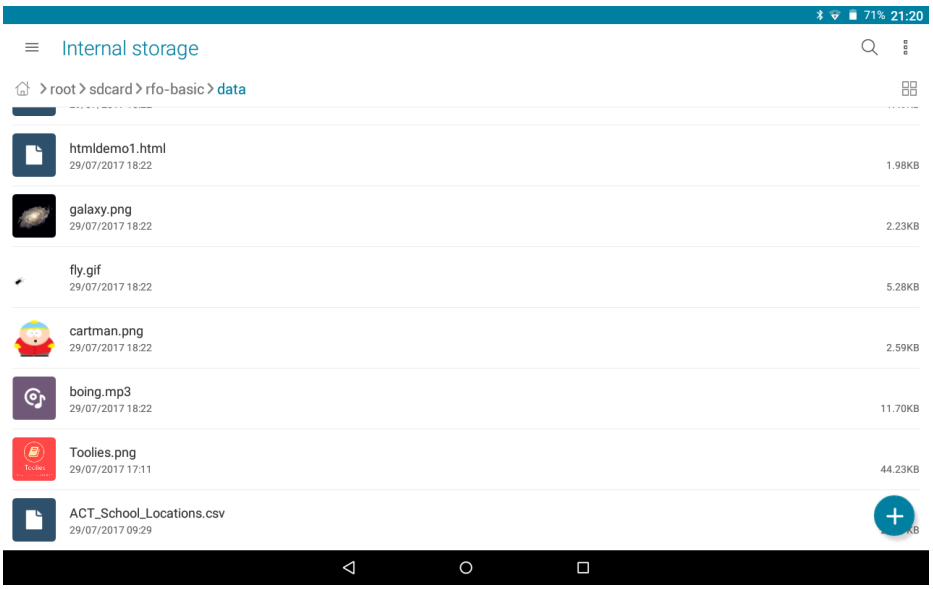
4- Dowload the following files from **https://github.com/viru48/Toolies/** :

- **ACT_School_Locations.csv**
- **Toolies_Demo.bas**

5- Copy **Toolies_Demo.bas** to your Android device directory: rfo-basic/source



6- Copy **ACT_School_Locations.csv** to your Android device directory: rfo-basic/data



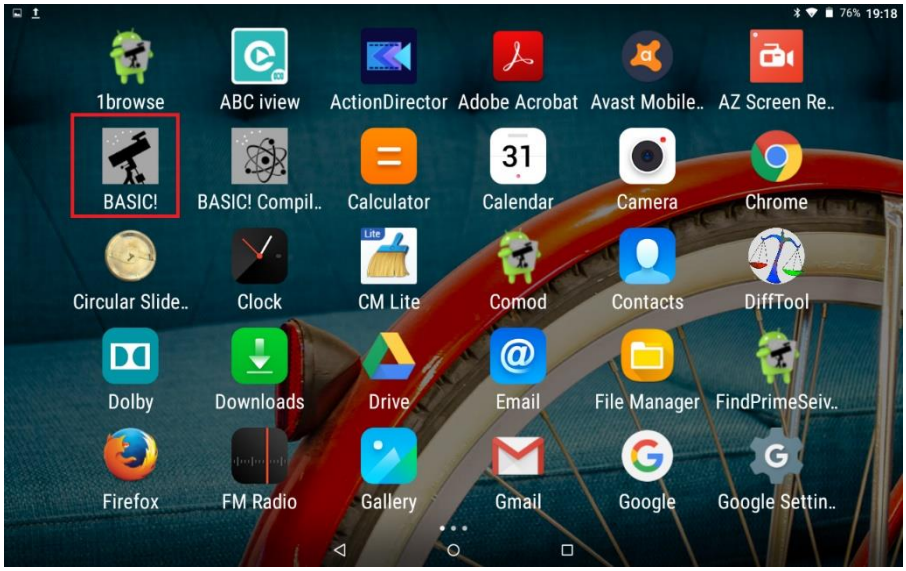
Expert Tips

1. A csv format is a “Comma Separated Values” text file.
2. When offered the choice between a “csv” or “Excel csv” file download format, its recommended to chose the earlier (csv format).
3. Always open the file first with a spreadsheet compatible software (eg. Excel, WpsOffice, …) and view the data layout.
4. Check if the first row is a column header row.
5. Ensure that the data is properly aligned along the different columns.
6. Count the number of actual columns in use.
7. Understand what each column stands for.
8. Check if any data in the (opened spreadsheet) contains any commas.
9. Replace all commas occurences in your spreadsheet with a “;” or a blank (“ “) or any other relevant character.

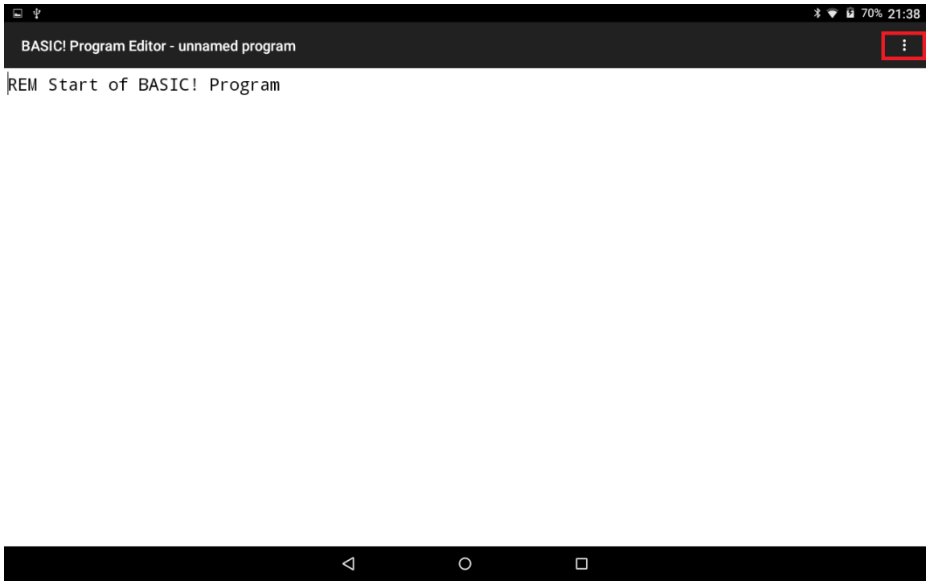
Chapter 3

Toolies Demo In Action

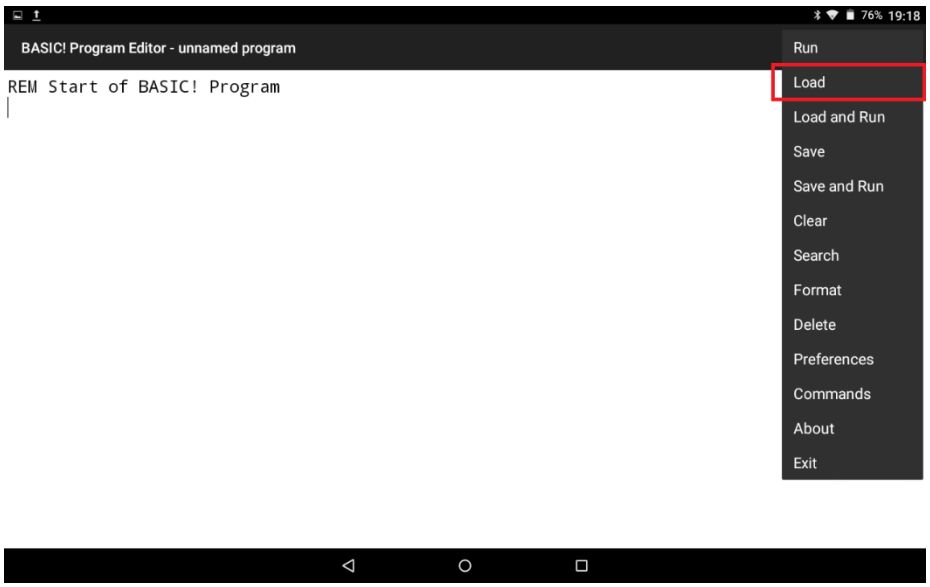
1- Open RFO BASIC



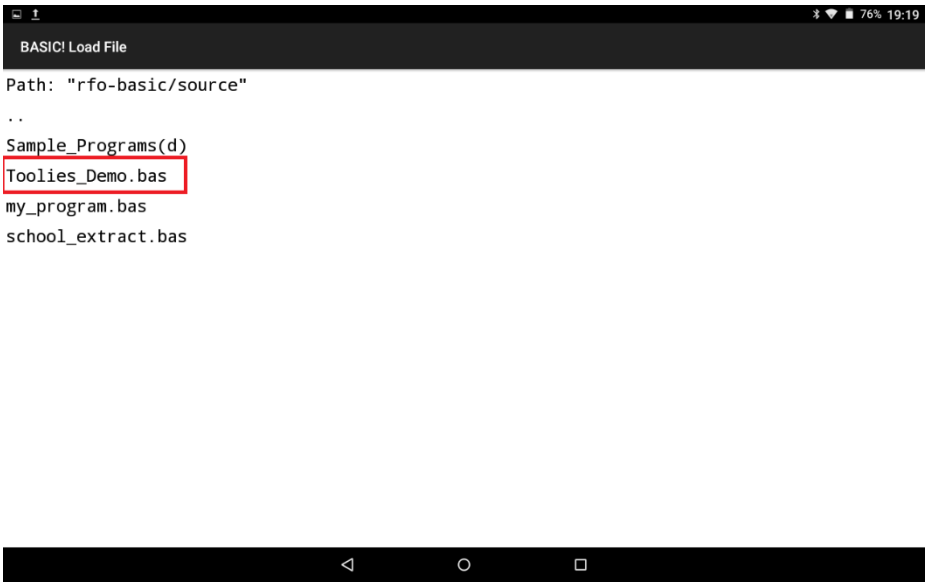
2- Click on Menu



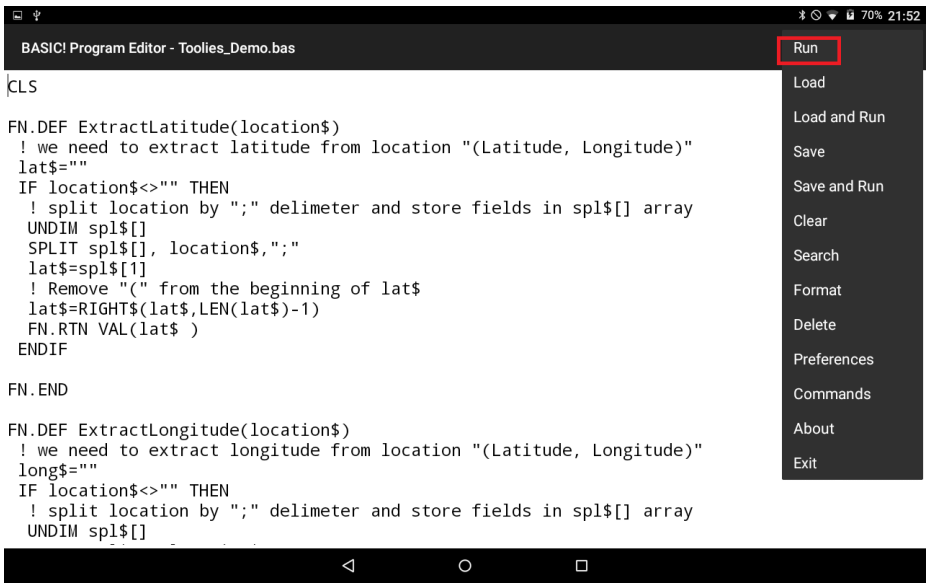
3- Click on Load



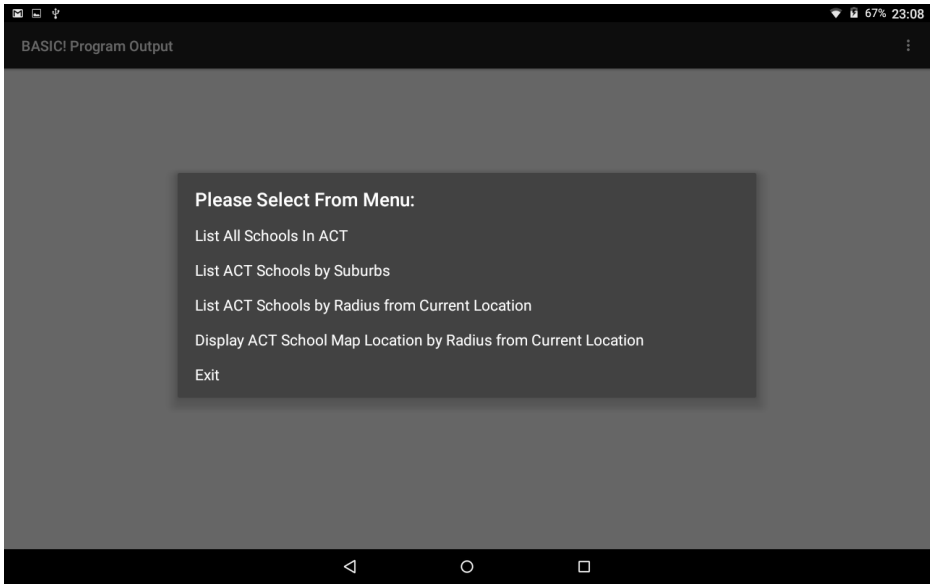
4- Select Toolies_Demo.bas



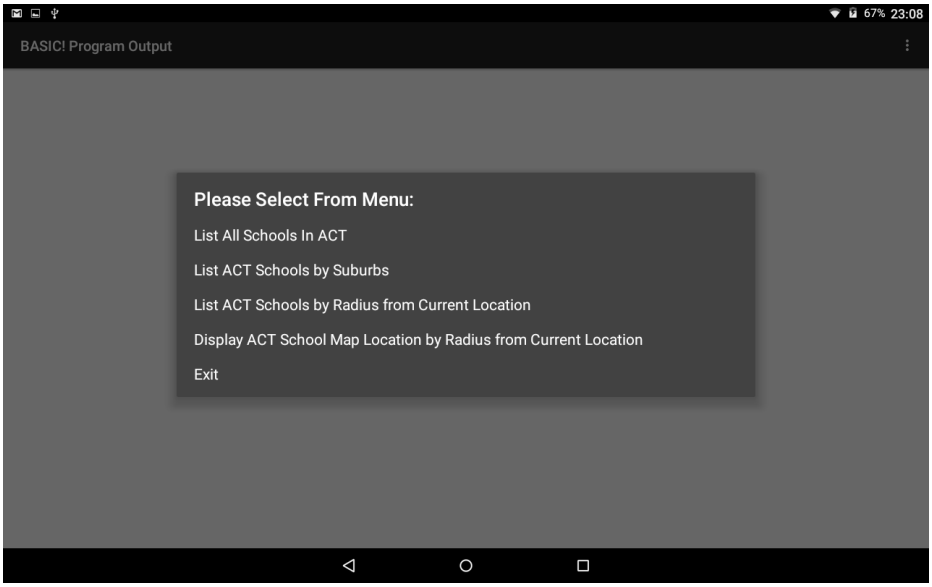
5- Select Run

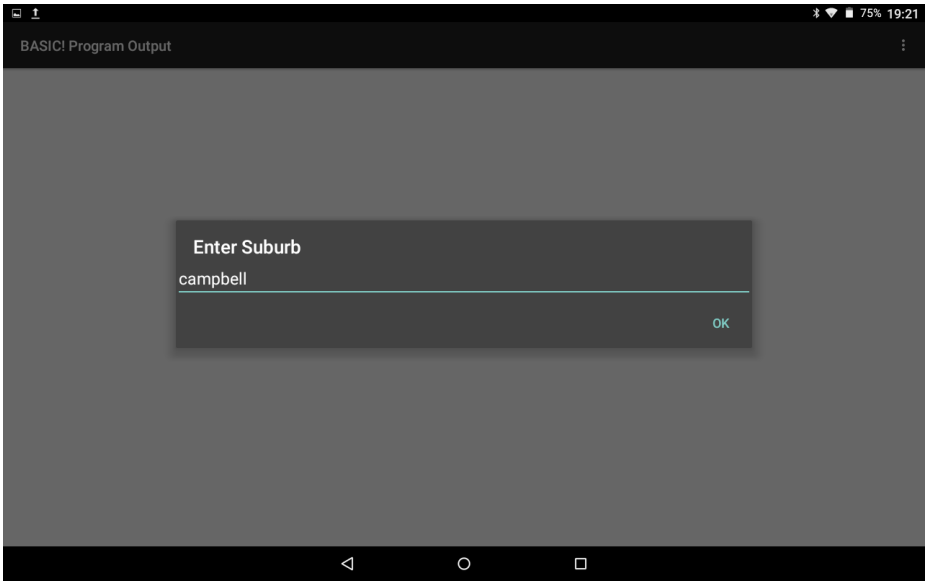


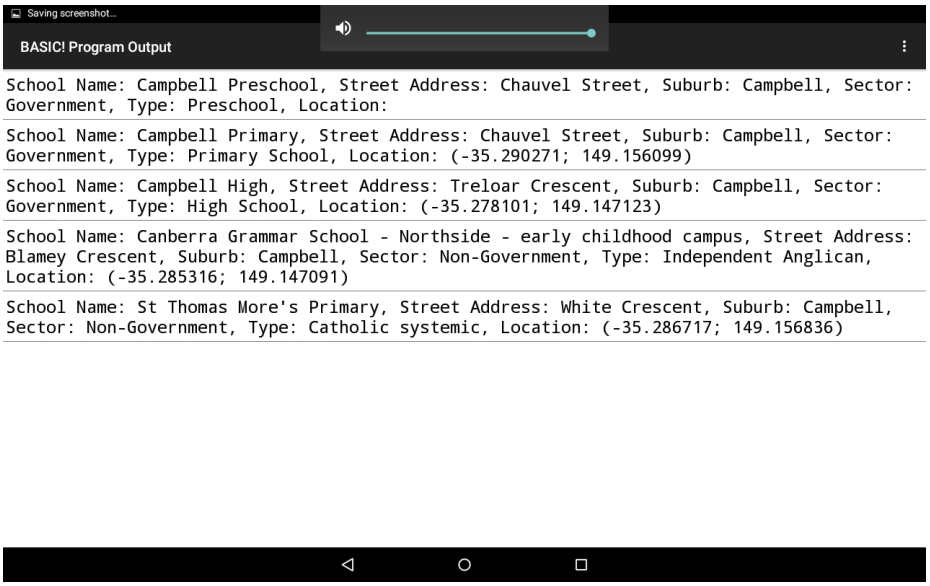
6- By Selecting List All Schools in ACT: program displays all ACT schools with their details. (Note by clicking the Back Key, program returns to Main Menu).



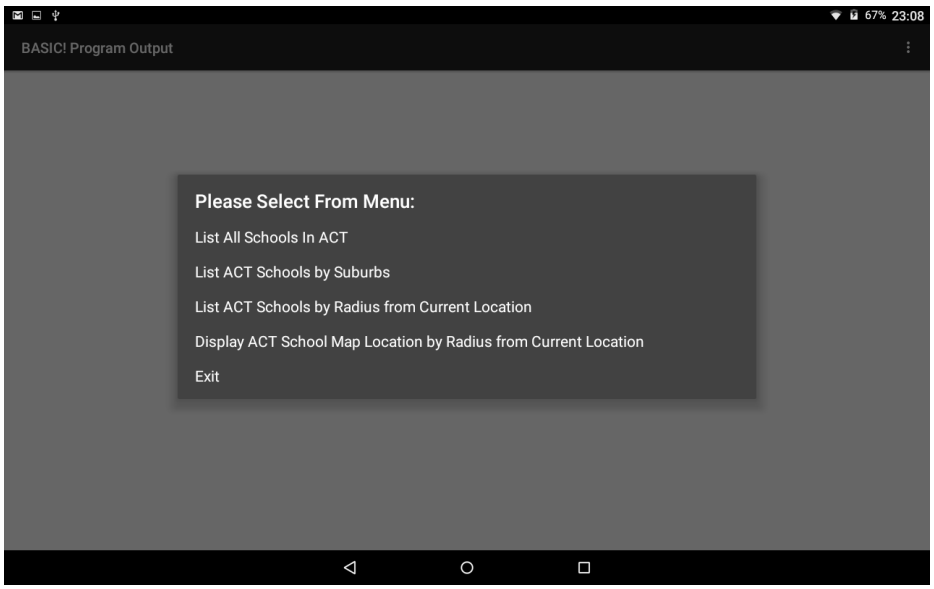
7- By Selecting List ACT Schools by Suburbs: program will prompt for a Suburb then displays all schools in Suburb (if any). (Note by clicking the Back Key, program returns to Main Menu).

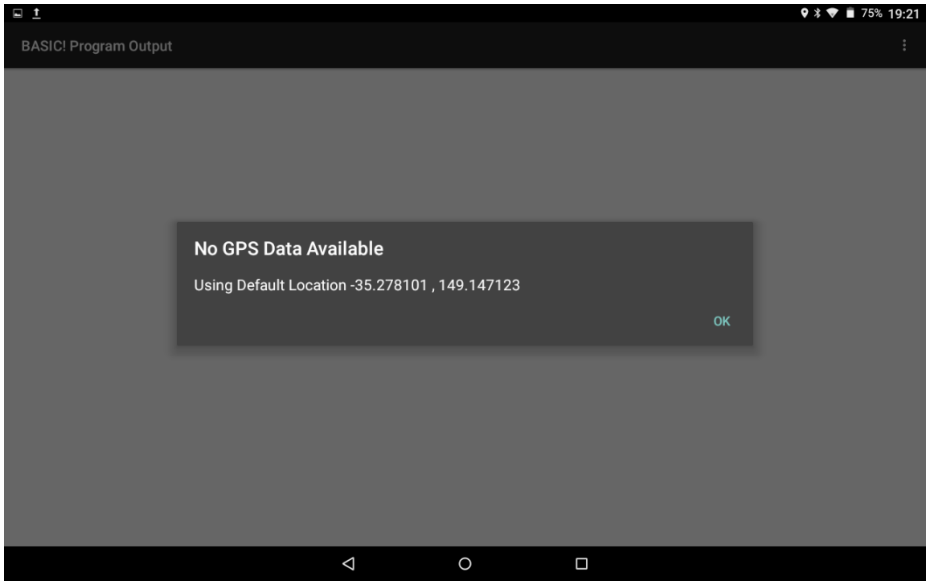


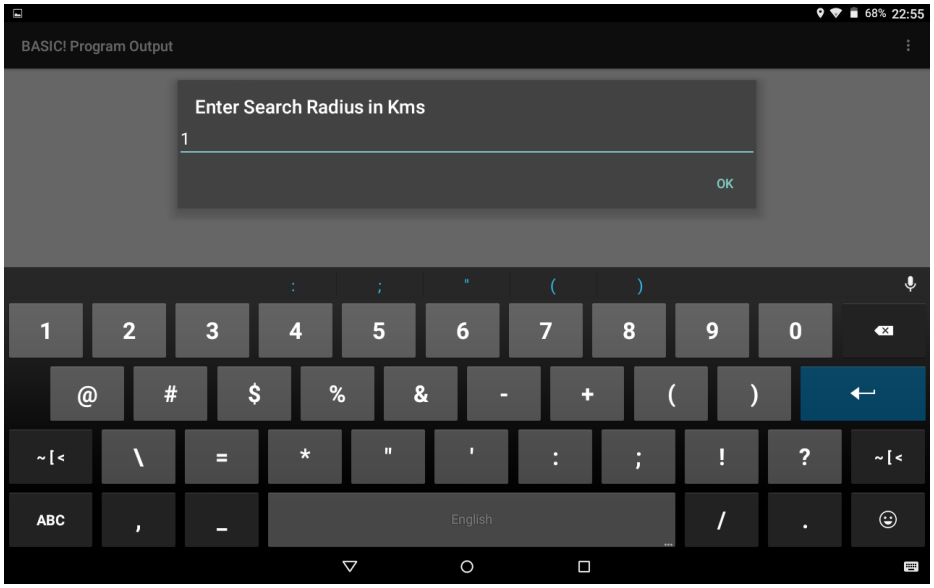


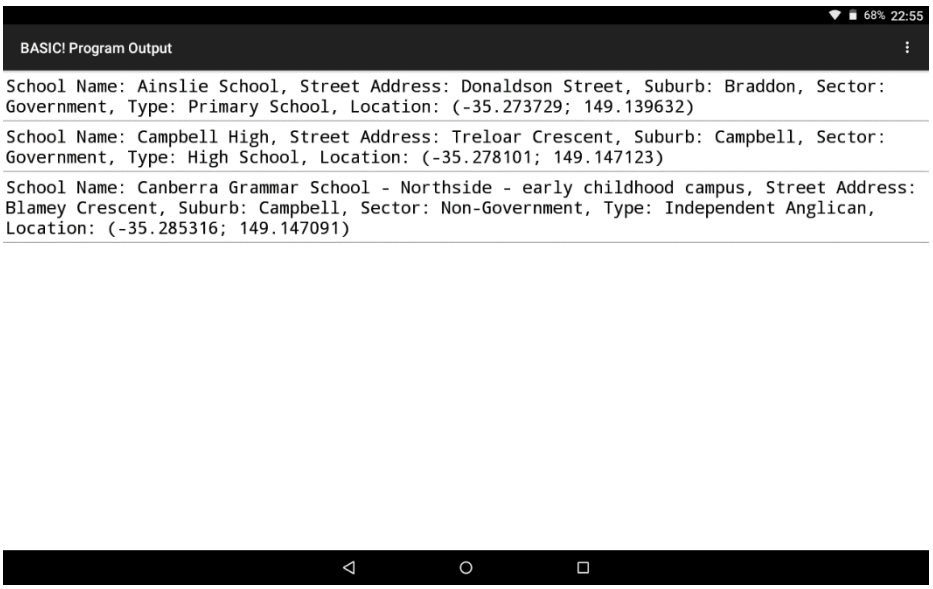


8- By Selecting List ACT Schools by Radius from Current Location: program will prompt for a Radius in Kms and then lists all schools in Area (Note, if Device GPS Location is not On, the program defaults to a Canberra Location).

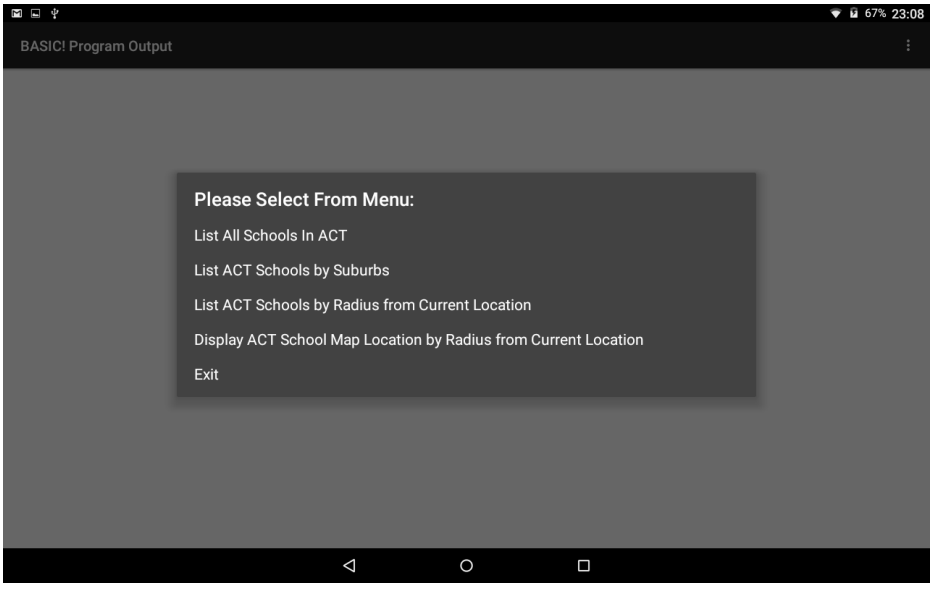

67% 23:08
BASIC! Program Output
Please Select From Menu:
List All Schools In ACT
List ACT Schools by Suburbs
List ACT Schools by Radius from Current Location
Display ACT School Map Location by Radius from Current Location
Exit

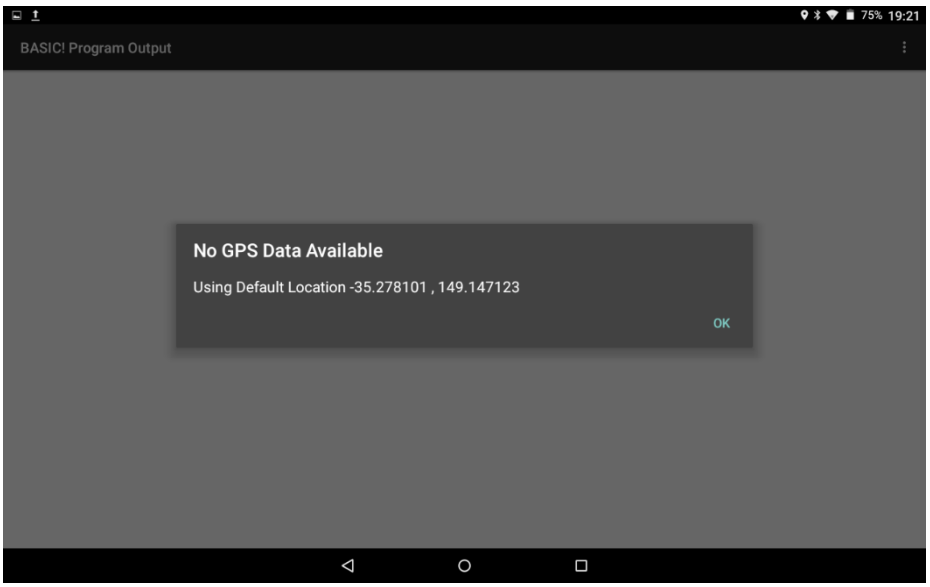

75% 19:21
BASIC! Program Output
No GPS Data Available
Using Default Location -35.278101 , 149.147123
OK

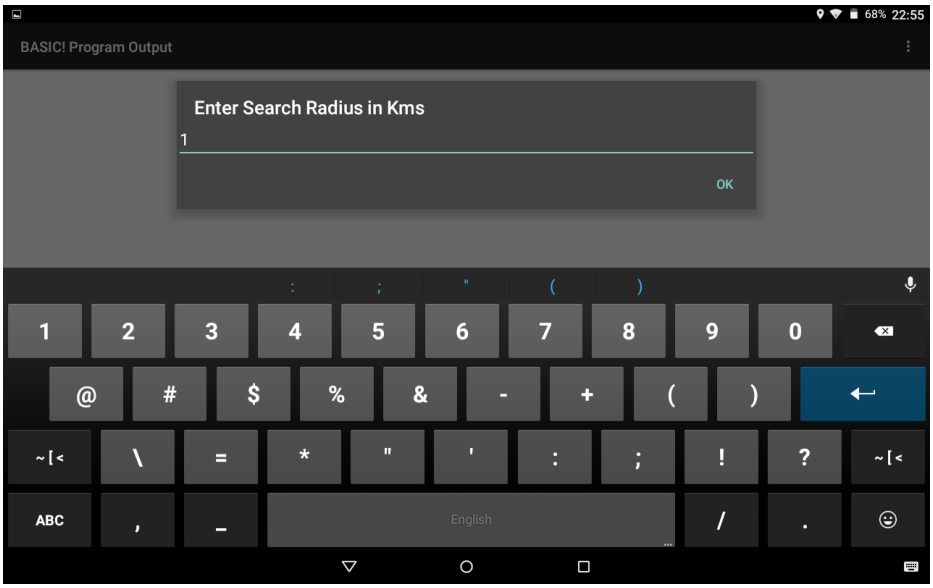

68% 22:55
BASIC! Program Output
Enter Search Radius in Kms
1
OK
1 2 3 4 5 6 7 8 9 0
@ # $ % & - + ( )
~[< \ = * " ' : ; ! ? ~[<
ABC , _ English / .

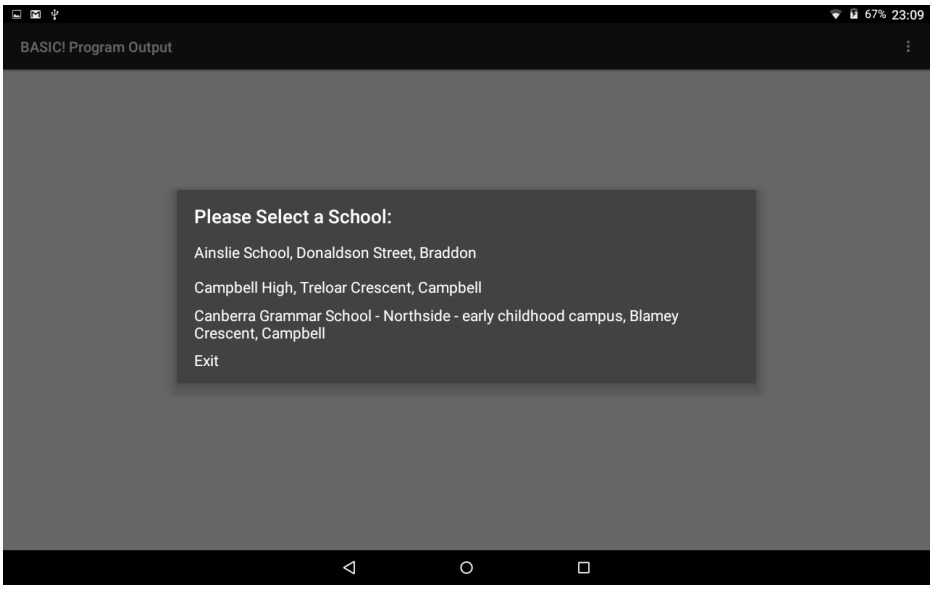


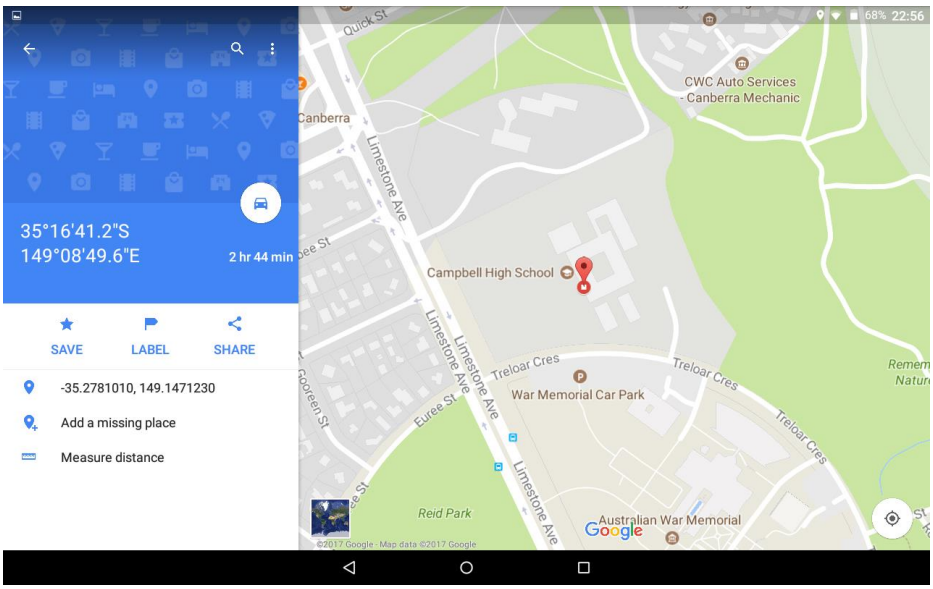
9- By Selecting Display ACT School Map Location from Current Location: program will prompt for a Radius in Kms and then lists all schools in Area (Note, if Device GPS Location is not On, the program defaults to a Canberra Location). Once a school is selected, Google Maps will display with the school location.




75% 19:21
BASIC! Program Output
No GPS Data Available
Using Default Location -35.278101 , 149.147123
OK


68% 22:55
BASIC! Program Output
Enter Search Radius in Kms
1
OK
1 2 3 4 5 6 7 8 9 0
@ # $ % & - + ( )
~ [< \ = * " ' : ; ! ? ~ [<
ABC , _ English / .


67% 23:09
BASIC! Program Output
Please Select a School:
Ainslie School, Donaldson Street, Braddon
Campbell High, Treloar Crescent, Campbell
Canberra Grammar School - Northside - early childhood campus, Blamey Crescent, Campbell
Exit


35°16'41.2"S
149°08'49.6"E
2 hr 44 min
SAVE
LABEL
SHARE
-35.2781010, 149.1471230
Add a missing place
Measure distance
Quick St
Canberra
Limestone Ave
Campbell High School
CWC Auto Services
- Canberra Mechanic
Treloar Cres
War Memorial Car Park
Euree St
Reid Park
Australian War Memorial
Google
©2017 Google - Map data ©2017 Google
68% 22:56

Chapter 4

Toolies Demo Source Code

```
CLS
FN.DEF ExtractLatitude(location$)
 ! we need to extract latitude from location "(Latitude, Longitude)"
 lat$=""
 IF location$<>"" THEN
  ! split location by ";" delimeter and store fields in spl$[] array
  UNDIM spl$[]
  SPLIT spl$[], location$,";"
  lat$=spl$[1]
  ! Remove "(" from the beginning of lat$
  lat$=RIGHT$(lat$,LEN(lat$)-1)
  FN.RTN VAL(lat$ )
 ENDIF

FN.END
```

```
FN.DEF ExtractLongitude(location$)
 ! we need to extract longitude from location "(Latitude, Longitude)"
 long$=""
 IF location$<>"" THEN
  ! split location by ";" delimeter and store fields in spl$[] array
  UNDIM spl$[]
  SPLIT spl$[], location$,";"
  long$=spl$[2]
  ! Remove ")" from the end of long$
  long$=LEFT$(long$,LEN(long$)-1)
  FN.RTN VAL(long$)
 ENDIF

FN.END
```

```
FN.DEF degreesToRadians(degrees)
 FN.RTN degrees * PI() / 180
FN.END
```

```
FN.DEF GpsDistance(lat1, lon1, lat2, lon2)
 earthRadiusKm = 6371

 dLat = degreesToRadians(lat2-lat1)
 dLon = degreesToRadians(lon2-lon1)

 lat1 = degreesToRadians(lat1)
 lat2 = degreesToRadians(lat2)

 a = SIN(dLat/2) * SIN(dLat/2) + SIN(dLon/2) * SIN(dLon/2) * COS(lat1) *
COS(lat2)
 c = 2 * ATAN2(SQR(a), SQR(1-a))
 FN.RTN earthRadiusKm * c

FN.END
```

```
v_error_back=0

! Create and load rec$[]
GOSUB GET_DATA

START:

CLS
! Setup your Menu
ARRAY.LOAD menus$[] ~
"List All Schools In ACT"~
"List ACT Schools by Suburbs"~
"List ACT Schools by Radius from Current Location"~
"Display ACT School Map Location by Radius from Current Location"~
"Exit"


! Set the Popup Message
msg$ ="Please Select From Menu: "

! Shows the list and waits for the user
! to make the selection.
DIALOG.SELECT menu, menus$[], msg$

SW.BEGIN menu

 SW.CASE 1
  GOSUB LIST_ALL
  SW.BREAK

 SW.CASE 2
  GOSUB LIST_SUBURBS
  SW.BREAK

 SW.CASE 3
  GOSUB LIST_RADIUS
  SW.BREAK

 SW.CASE 4
  GOSUB SCHOOL_MAP_RADIUS
  SW.BREAK

 SW.CASE 5
  EXIT

 SW.DEFAULT
  GOTO start

SW.END


GOTO start


END
```

```
GET_DATA:

! open file

fl$="ACT_School_Locations.csv"

TEXT.OPEN R, FN1, fl$

! first line which is the header line
TEXT.READLN FN1, a_line$

! find out how may fields in the header
! store value in fld_num

UNDIM spl$[]
SPLIT spl$[], a_line$,","
ARRAY.LENGTH fld_num, spl$[]


! let us find out how many records we have in the file
! store value in rec_num

rec_num=0
TEXT.READLN FN1, a_line$
WHILE a_line$ <> "EOF"
 rec_num = rec_num +1
 TEXT.READLN FN1, a_line$
REPEAT

! close file
TEXT.CLOSE FN1


!!

create array rec$[] to capture the following data
Field 1: School Name
Field 2: Street Address
Field 3: Suburb
Field 4: Sector
Field 5: Type
Field 6: Location

!!

DIM rec$[rec_num,fld_num]

! open file
TEXT.OPEN R, FN1, fl$

! skip first line which is title line
TEXT.READLN FN1, a_line$

FOR I = 1 TO rec_num
 TEXT.READLN FN1, a_line$

 ! clear array spl$[]
 UNDIM spl$[]
 ! split each record line by comma delimeter and store fields in spl$[] array
 ! eg school_code will be stored in spl$[1],  school_name will be stored in
spl$[3]


 SPLIT spl$[], a_line$,","
 ! If the last column is blank, we will be short on a column
 ! The next command will return the number of columns

 ARRAY.LENGTH flds ,spl$[]

 ! load each record in rec$[I,j]
 ! rec$[1,1] will store the School Name for the first record
 ! rec$[1,2] will store the Street Address for the first record
 ! rec$[1,3] will store the Suburb for the first record
 ! rec$[1,4] will store the Sector for the first record
```

```
LIST_ALL:

FOR i = 1 TO rec_num
 PRINT "School Name: "; rec$[i,1];", "; "Street Address: "; rec$[i,2];",
";"Suburb: "; rec$[i,3];", ";"Sector: "; rec$[i,4];", ";"Type: "; rec$[i,5];",
";"Location: "; rec$[i,6]
NEXT i


IF v_error_back =1 THEN BACK.RESUME

GOTO WaitLoop

RETURN

LIST_SUBURBS:

INPUT "Enter Suburb",PSUBURB$

FOR i = 1 TO rec_num
 IF UPPER$(rec$[i,3])=UPPER$(PSUBURB$) THEN
  PRINT "School Name: "; rec$[i,1];", "; "Street Address: "; rec$[i,2];",
";"Suburb: "; rec$[i,3];", ";"Sector: "; rec$[i,4];", ";"Type: "; rec$[i,5];",
";"Location: "; rec$[i,6]
 ENDIF
NEXT i

IF v_error_back =1 THEN BACK.RESUME

GOTO WaitLoop


RETURN
```

```
LIST_RADIUS:

GPS.OPEN

GPS.LATITUDE latitude
! PRINT  "Latitude:   " + FORMAT$("##%.#####", latitude)

GPS.LONGITUDE longitude
! PRINT"Longitude:  " + FORMAT$("##%.#####", longitude)


IF ABS(Latitude) = 0  | ABS(longitude) =0 THEN

 latitude= -35.278101
 longitude=149.147123
!!
 latitude= -33.81571            %-33.780193
 longitude=151.00050           %150.905175
!!
 DIALOG.MESSAGE "No GPS Data Available", "Using Default Location -35.278101 ,
149.147123 ",go,"ok"

ENDIF


Enter_Radius1:
INPUT "Enter Search Radius in Kms",KMS$
IF !IS_NUMBER(KMS$) THEN GOTO Enter_Radius1
km=VAL(kms$)

v_CNT=0

FOR i = 1 TO rec_num
 ! we need to extract latitude and longitude from rec$[i,6]
 IF rec$[i,6]<>"" THEN
  position$=rec$[i,6]

  lat = ExtractLatitude(position$)
  long =  ExtractLongitude(position$)

  IF ABS(GpsDistance(latitude, longitude, lat,long) ) <= km THEN
   PRINT "School Name: "; rec$[i,1];", "; "Street Address: "; rec$[i,2];",
";"Suburb: "; rec$[i,3];", ";"Sector: "; rec$[i,4];", ";"Type: "; rec$[i,5];",
";"Location: "; rec$[i,6]
   v_CNT=v_CNT+1
  ENDIF
 ENDIF


NEXT i



GPS.CLOSE



IF v_cnt =0 THEN
 DIALOG.MESSAGE "No Schools In This Area", " ",go,"ok"
 IF v_error_back =1 THEN BACK.RESUME
ELSE
 IF v_error_back =1 THEN BACK.RESUME
 GOTO WaitLoop
ENDIF

SCHOOL_MAP_RADIUS:
GPS.OPEN

GPS.LATITUDE latitude
! PRINT  "Latitude:   " + FORMAT$("##%.#####", latitude)

GPS.LONGITUDE Longitude
! PRINT"Longitude:  " + FORMAT$("##%.#####", longitude)
```

```
WaitLoop:

w=0
DO
 PAUSE 10
UNTIL w=1
```

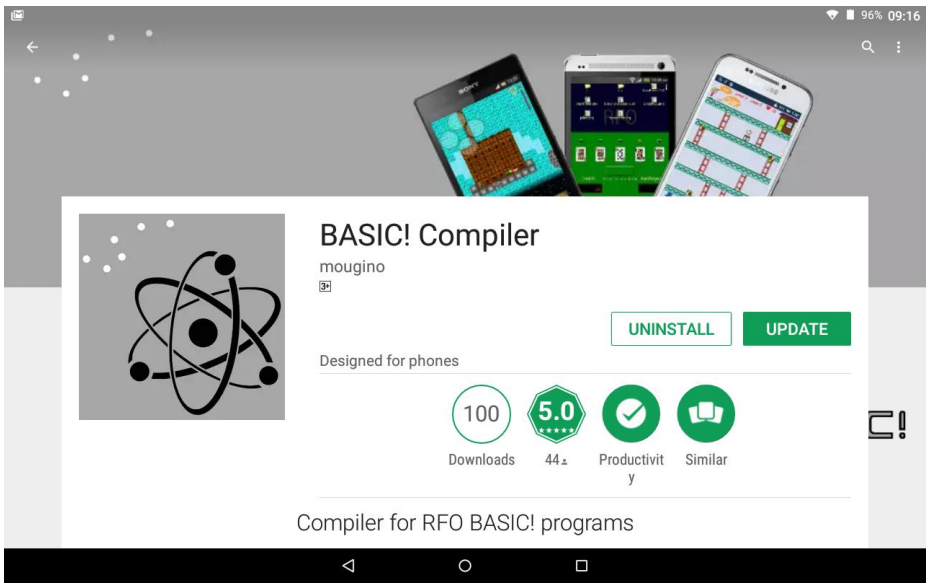
```
ONBACKKEY:
v_error_back =1
RETURN
```

Chapter 5

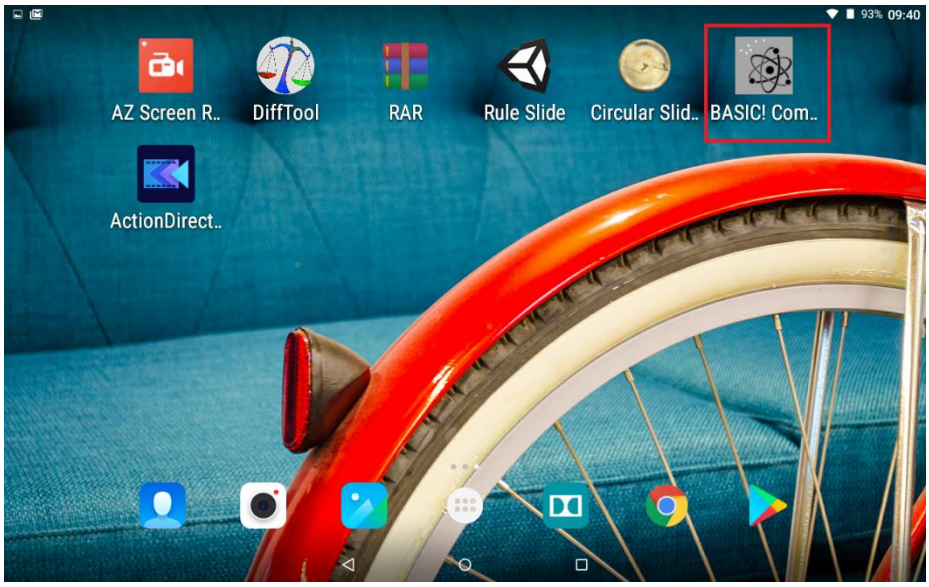
Optional, But Interesting

Now that you have developed your Android App using **RFO BASIC**, you may be interested in compiling it into an APK and installing it like any other Android App on you device. All what you have to do is to follow these simple steps:

1. You may need to consider investing $4 and purchase **BASIC! Compiler** from the Google Play Store (Note: Check if your Android Device Processor is compatible with this program via a list provided by the BASIC! Compiler Developers)

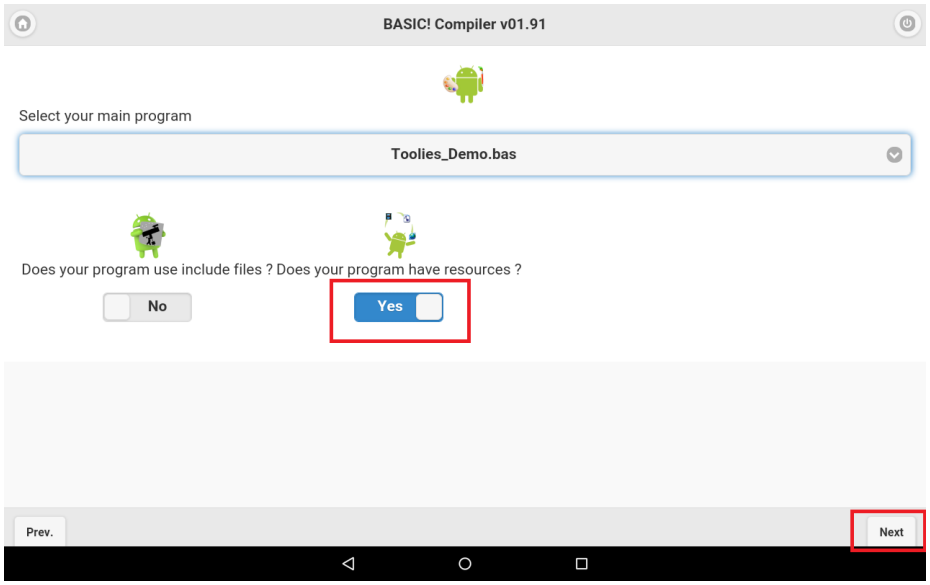


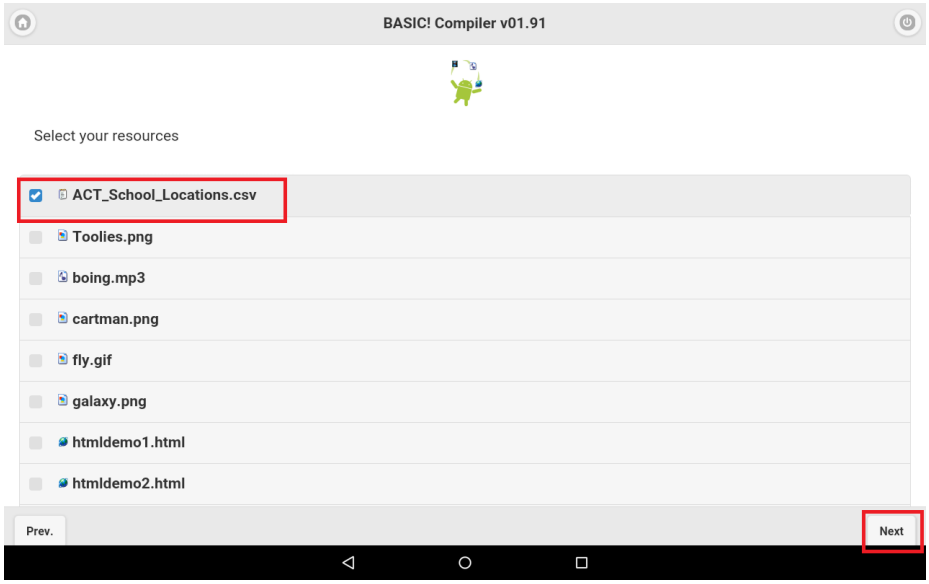
2. Open **BASIC! Compiler**



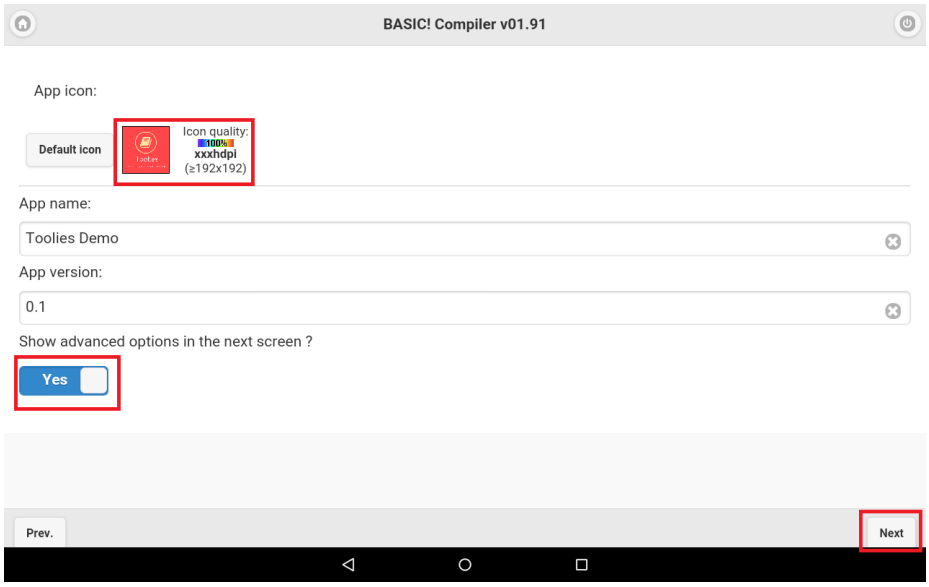
3. Click on New APK

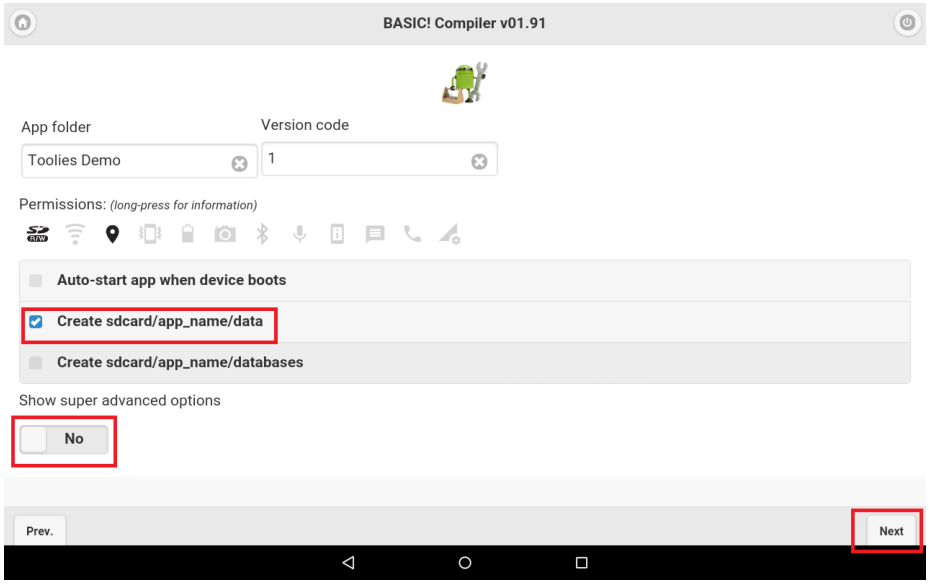
4. Select Toolies_Demo.bas

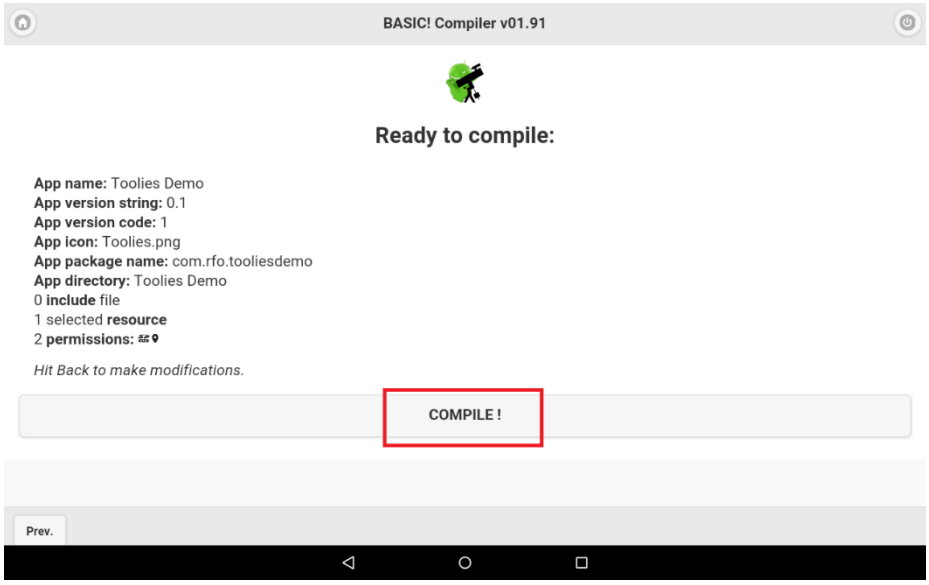


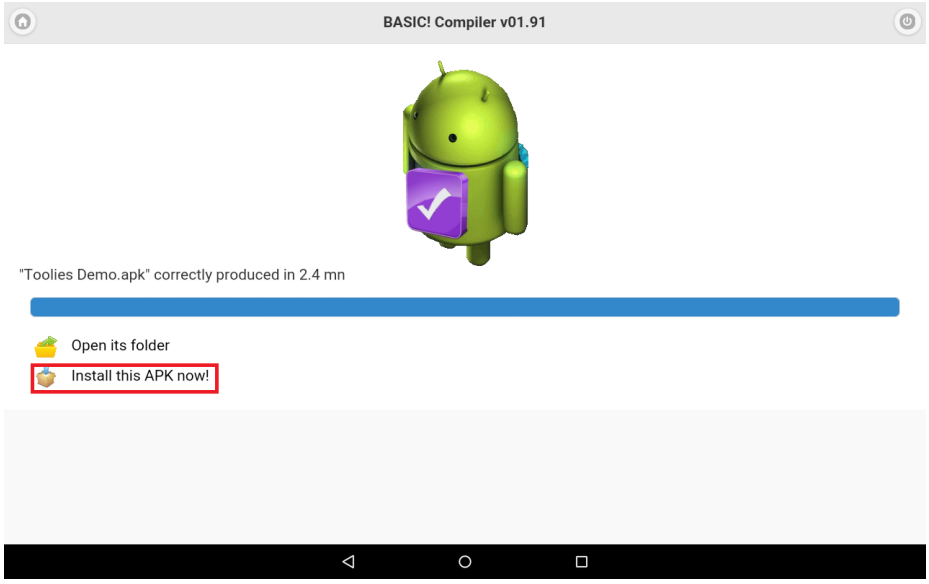


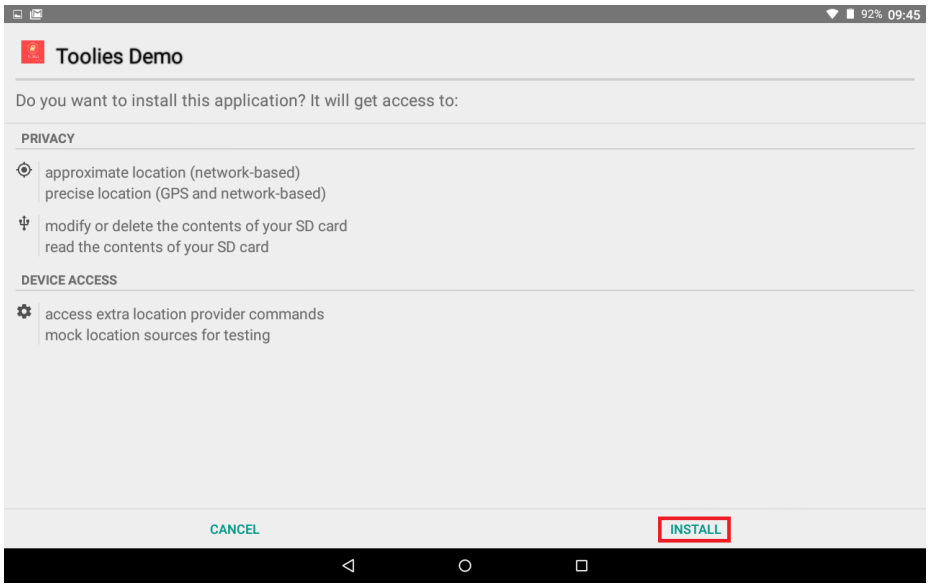
5. Select an Icon



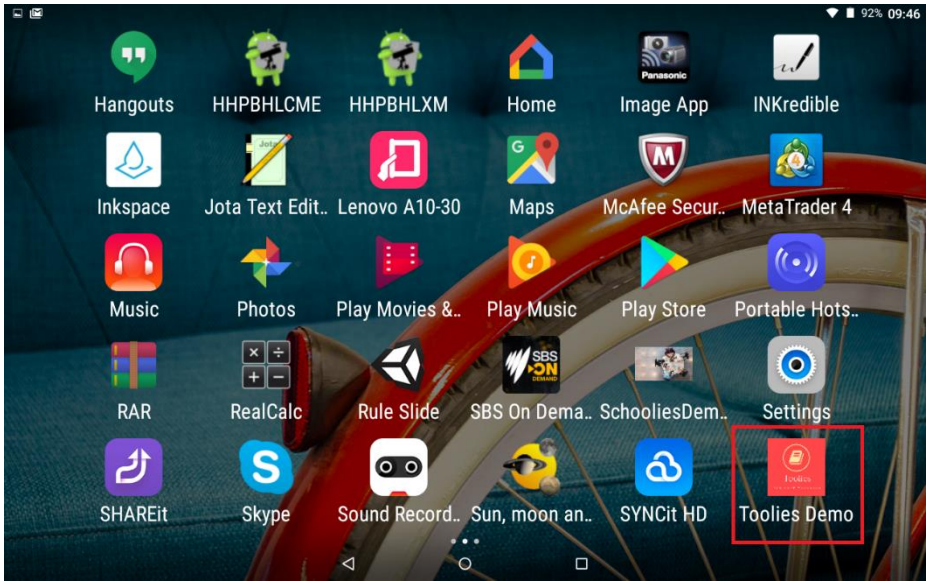

BASIC! Compiler v01.91
App folder
Toolies Demo
Version code
1
Permissions: (long-press for information)
Auto-start app when device boots
Create sdcard/app_name/data
Create sdcard/app_name/databases
Show super advanced options
No
Prev.
Next

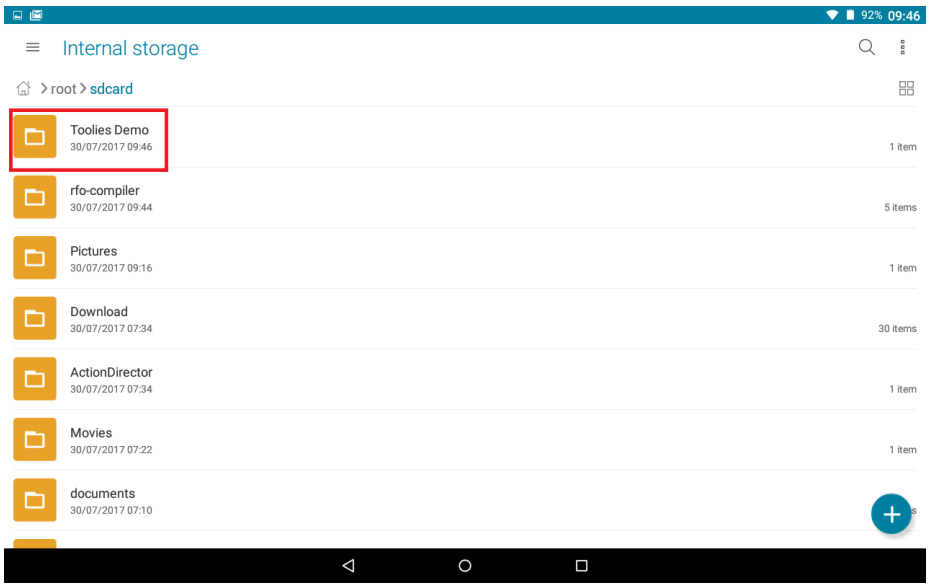

BASIC! Compiler v01.91
Ready to compile:
App name: Toolies Demo
App version string: 0.1
App version code: 1
App icon: Toolies.png
App package name: com.rfo.tooliesdemo
App directory: Toolies Demo
0 include file
1 selected resource
2 permissions:
Hit Back to make modifications.
COMPILE !
Prev.


BASIC! Compiler v01.91
"Toolies Demo.apk" correctly produced in 2.4 mn
Open its folder
Install this APK now!

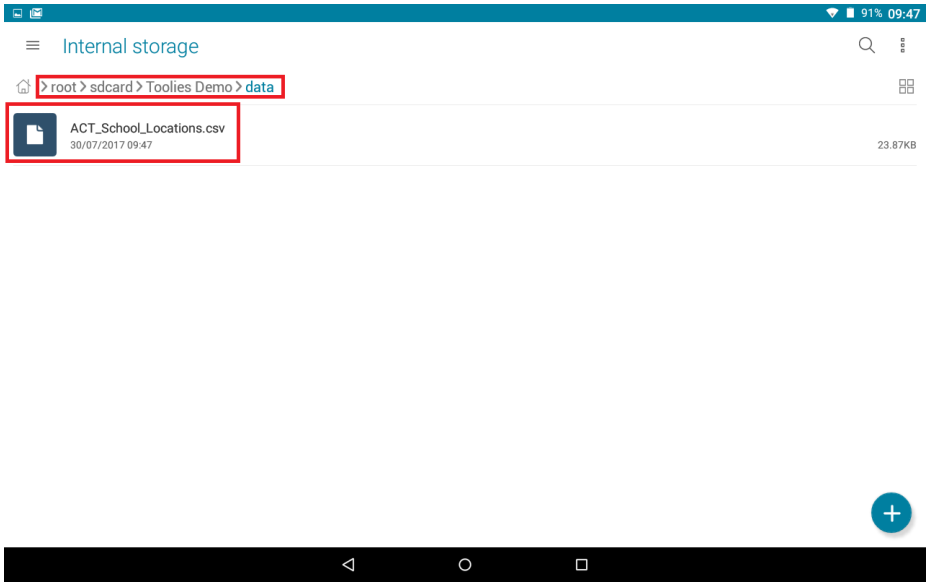


6. Click on **Toolies Demo**, this wil create the Toolies Demo directory on your device.





7. Copy ACT_School_Locations.csv to **/TooliesDemo/data** directory.



8. Now **Toolies Demo** is ready to run.