**Cook Book for Toolies**



**Gov Hack 2017**

Table of Contents

[Chapter 1 3](#_Toc489135429)

[Objective 3](#_Toc489135430)

[Target Audience 3](#_Toc489135431)

[In Short 3](#_Toc489135432)

[Chapter 2 3](#_Toc489135433)

[Pre-Requisite 3](#_Toc489135434)

[Chapter 3 6](#_Toc489135435)

[Toolies Demo In Action 6](#_Toc489135436)

[Chapter 4 15](#_Toc489135437)

# 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.

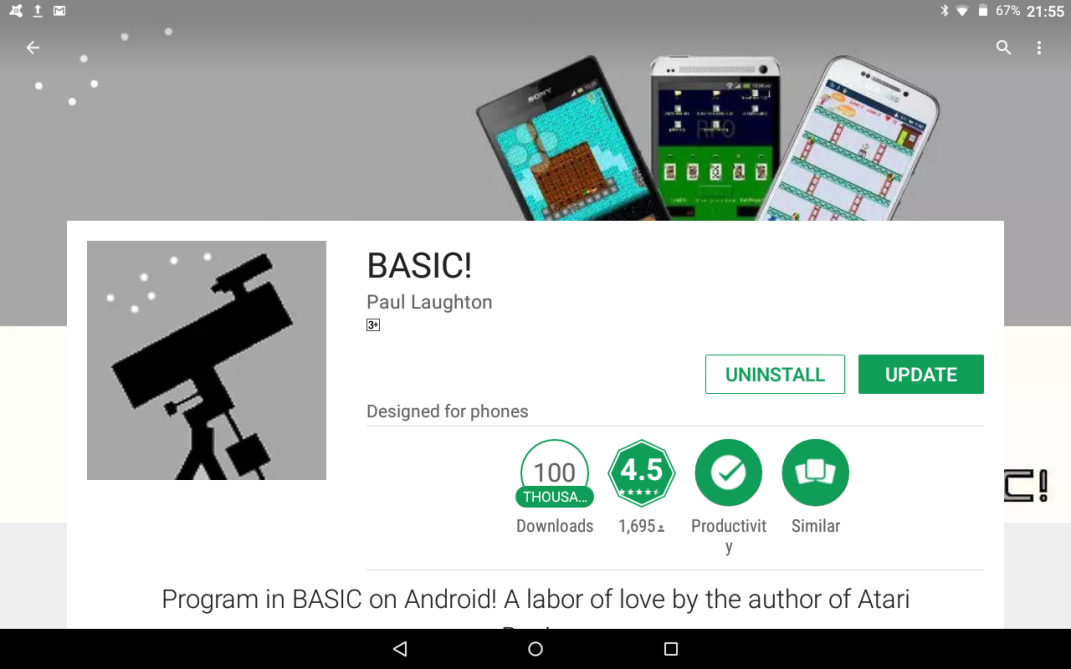
In Short

This Cook Book offers step by step instructions for successfully installing and running the sample application [Toolies\_Demo.bas](https://github.com/viru48/Toolies/blob/master/Toolies_Demo.bas) which was developed on an Android tablet using RFO BASIC as part of the GovHack 2017 Competition.

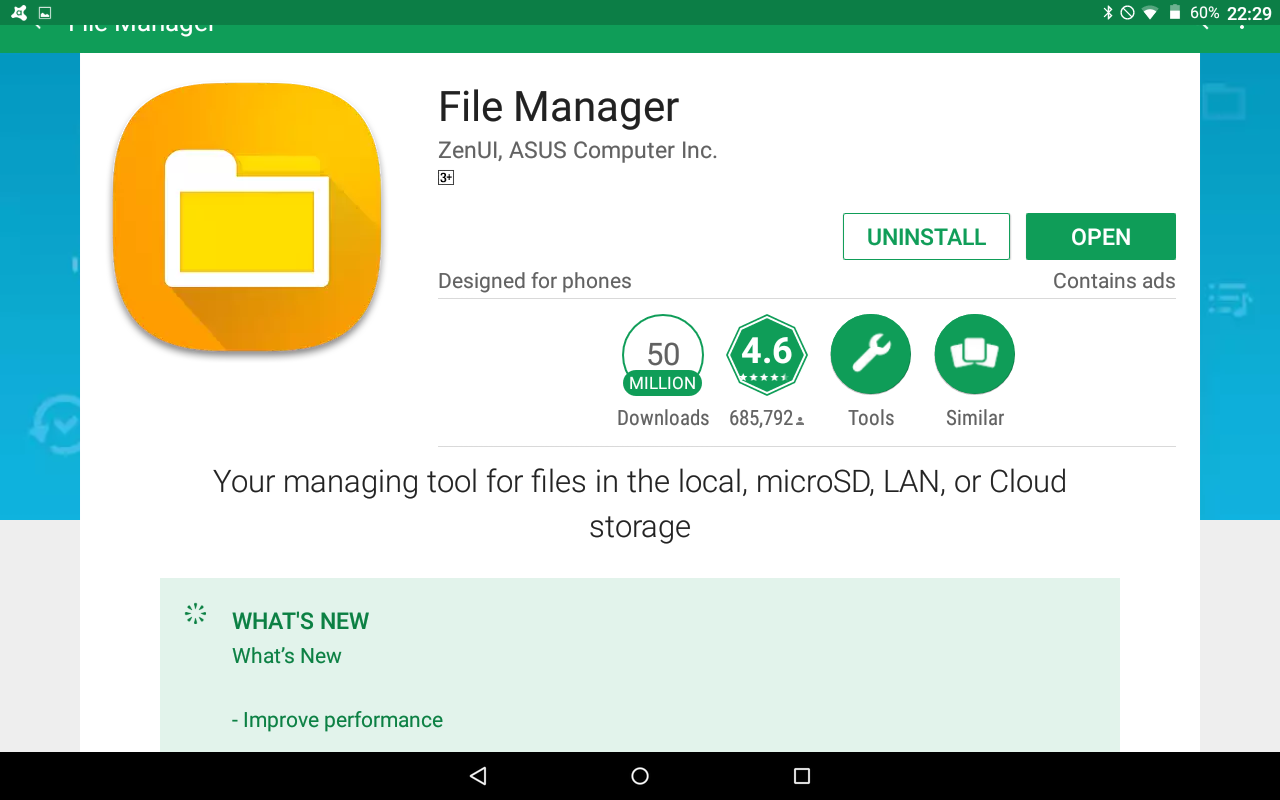
# Chapter 2

## Pre-Requisite

1. An Android Phone or Tablet.
2. Download BASIC! (Paul Laughton) from Google Play (for free).



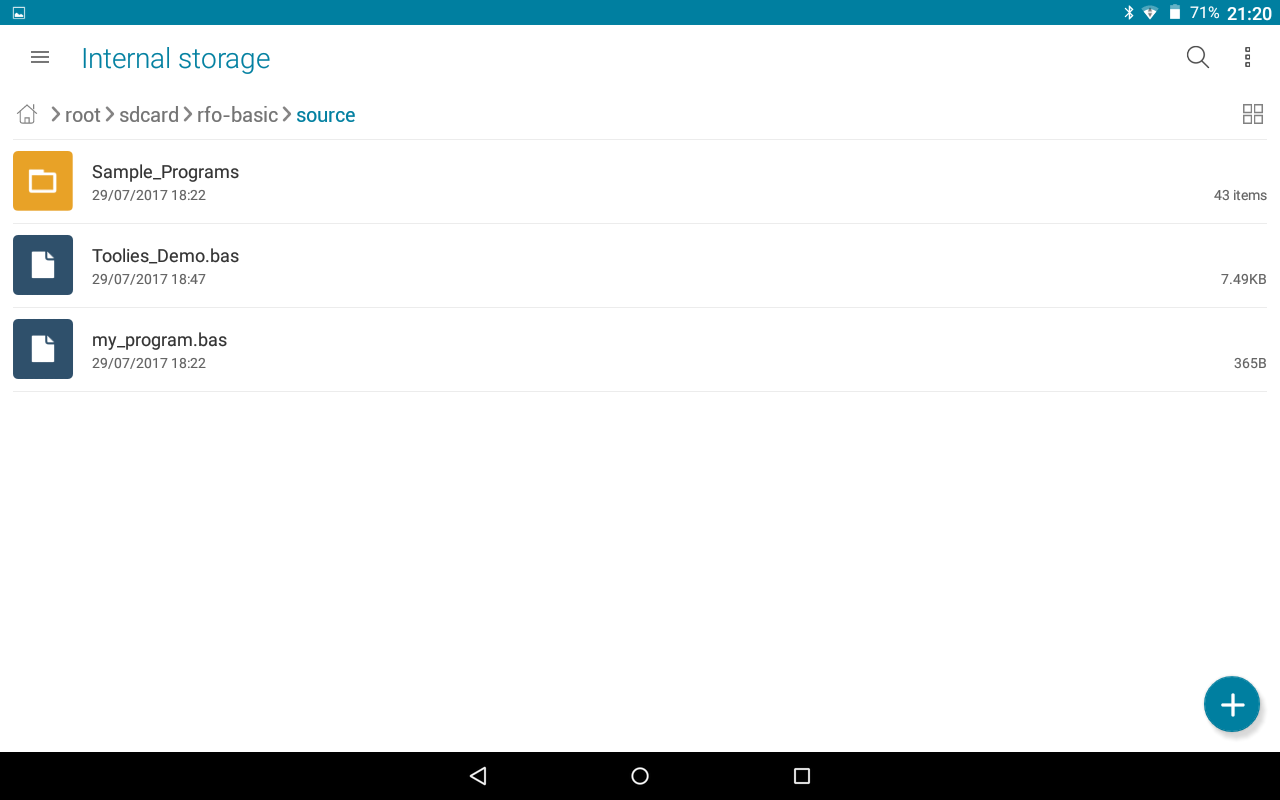
1. We recommend downloading File Manager (ZenUI, ASUS Comptuer Inc.) from Google Play (for free).



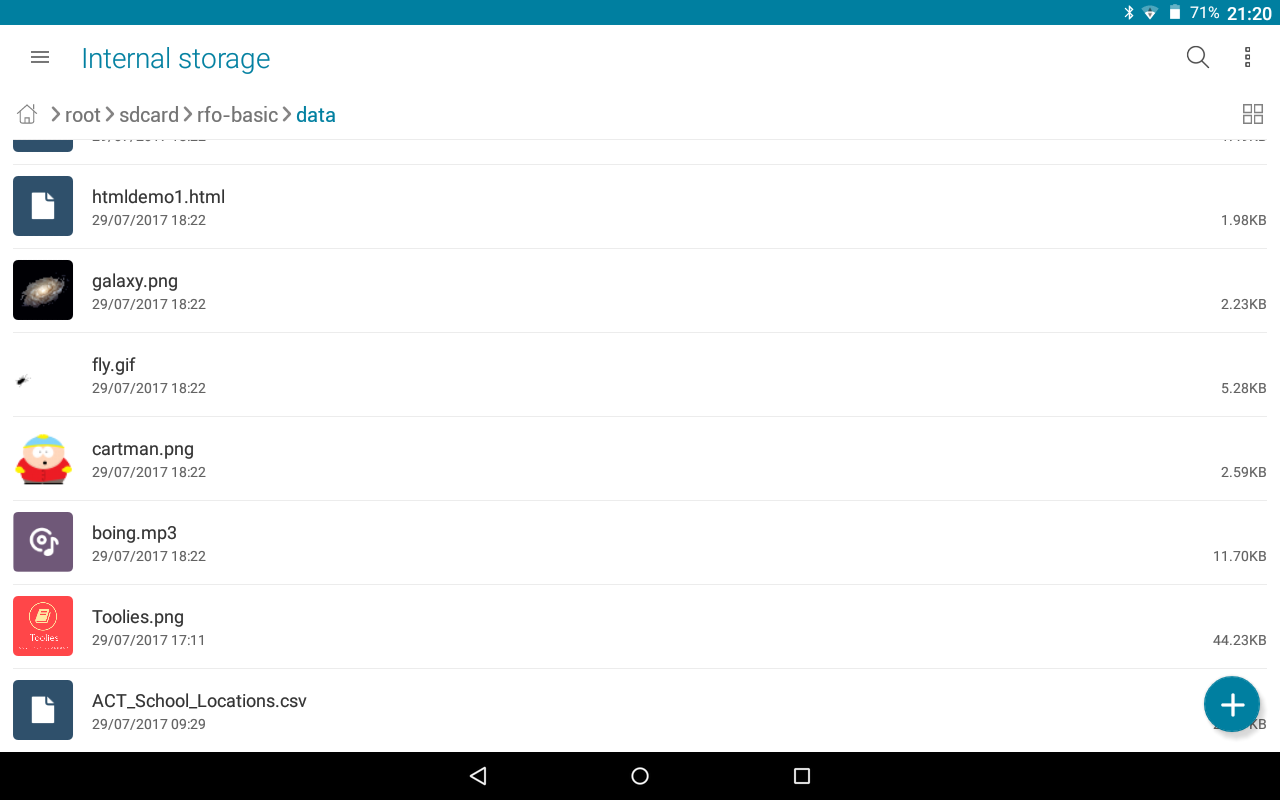
1. Dowload the following files from [**https://github.com/viru48/Toolies/**](https://github.com/viru48/Toolies/) :

* [**ACT\_School\_Locations.csv**](https://github.com/viru48/Toolies/blob/master/ACT_School_Locations.csv)
* [**Toolies\_Demo.bas**](https://github.com/viru48/Toolies/blob/master/Toolies_Demo.bas)

1. Copy [**Toolies\_Demo.bas**](https://github.com/viru48/Toolies/blob/master/Toolies_Demo.bas) to your Android device directory: rfo-basic/source



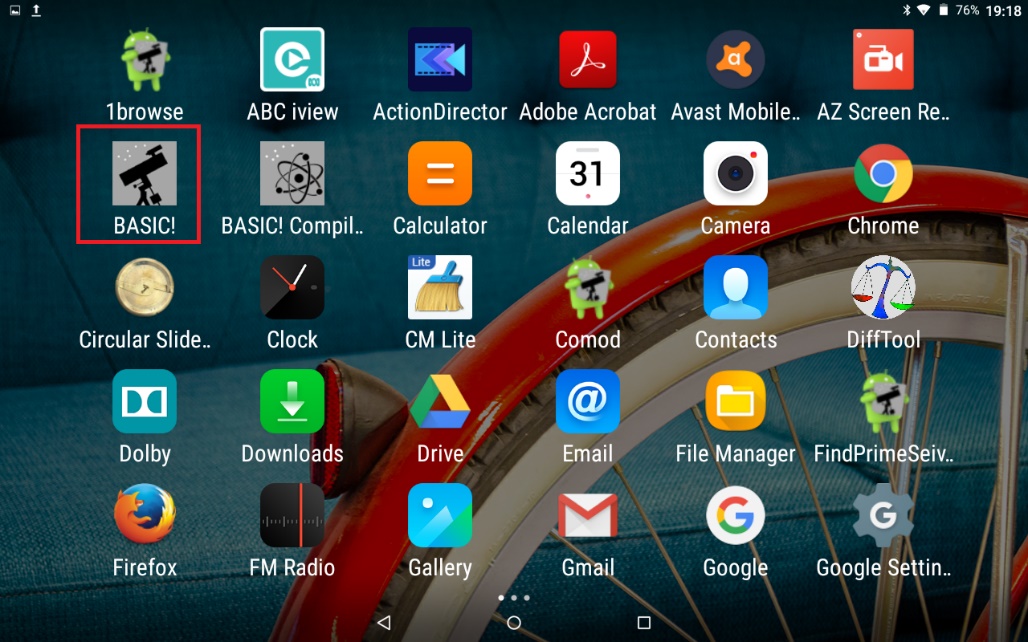
1. Copy **ACT\_School\_Locations.csv** to your Android device directory: rfo-basic/data



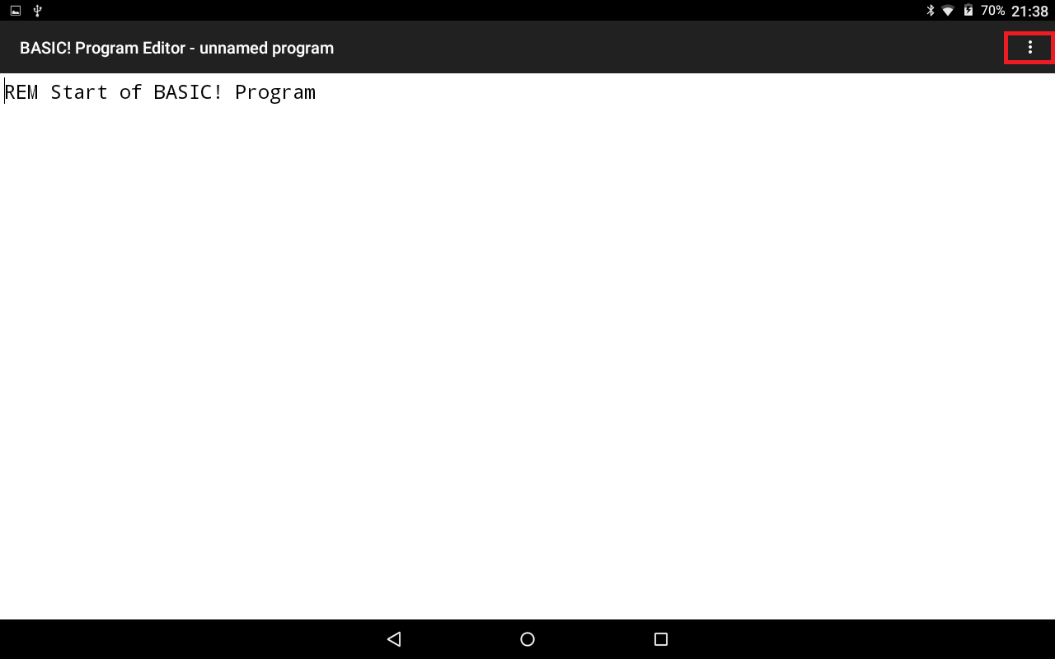
# Chapter 3

## Toolies Demo In Action

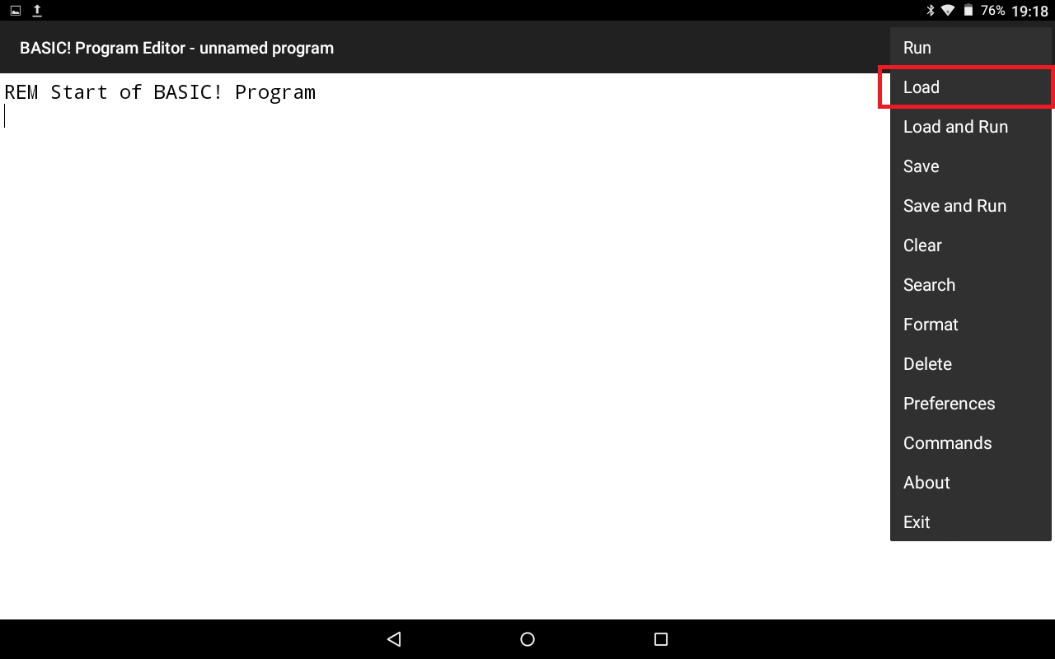
1. Open RFO BASIC



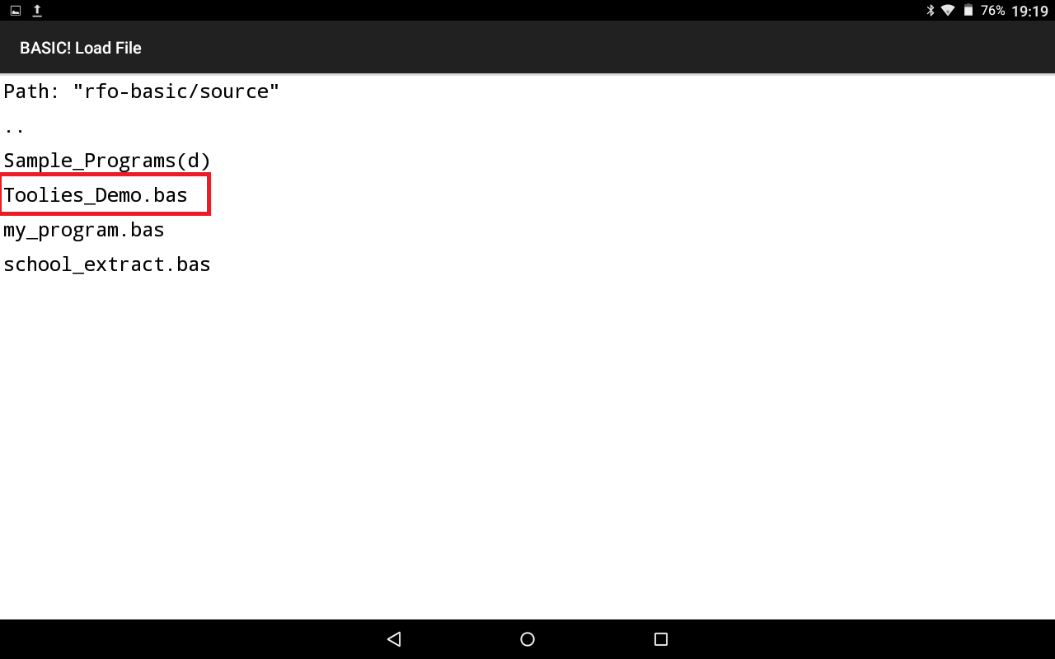
1. Click on Menu



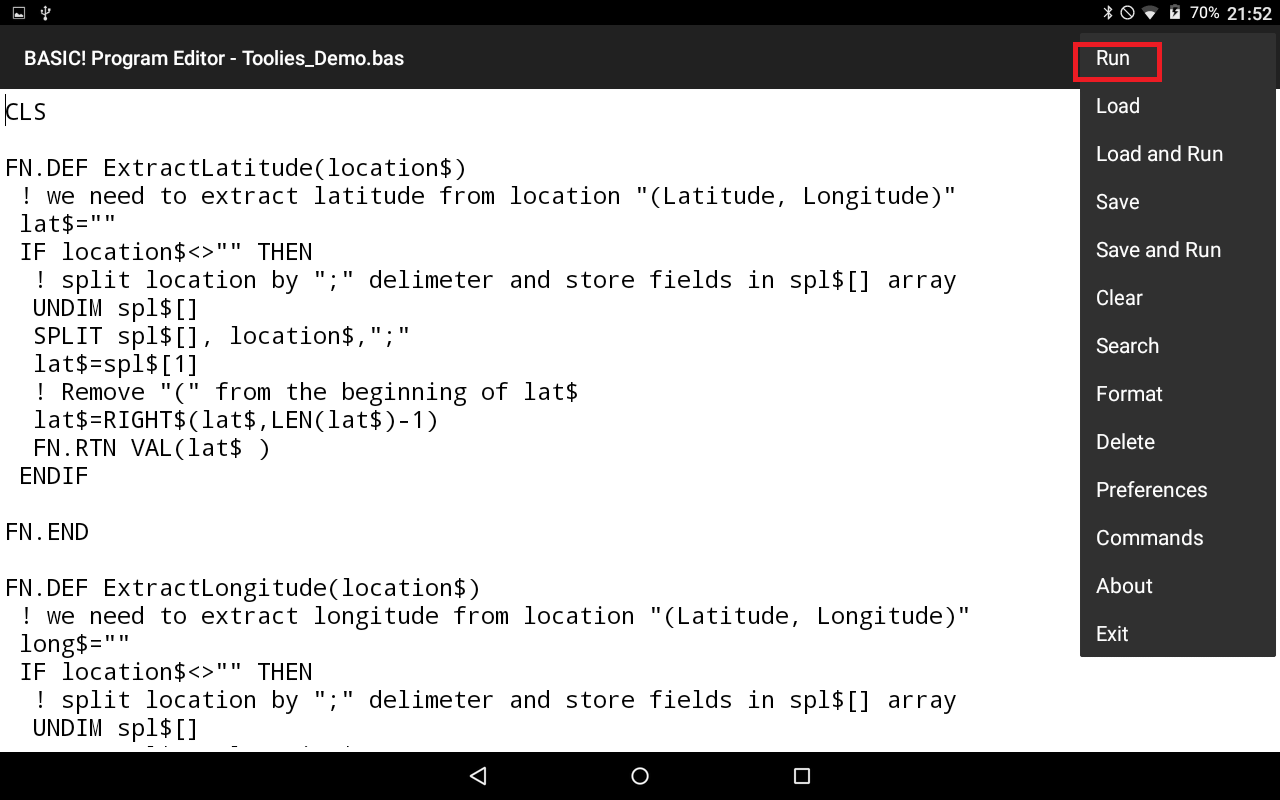
1. Click on Load



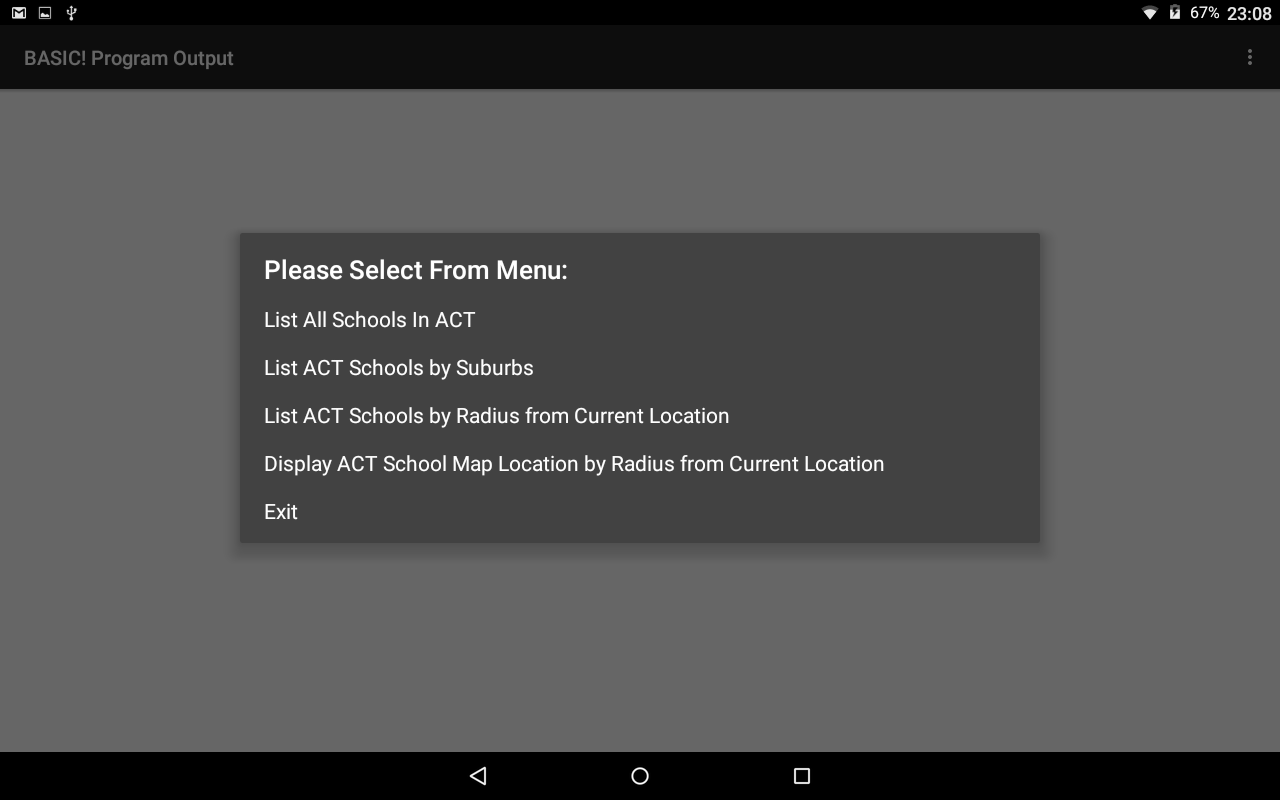
1. Select [Toolies\_Demo.bas](https://github.com/viru48/Toolies/blob/master/Toolies_Demo.bas)

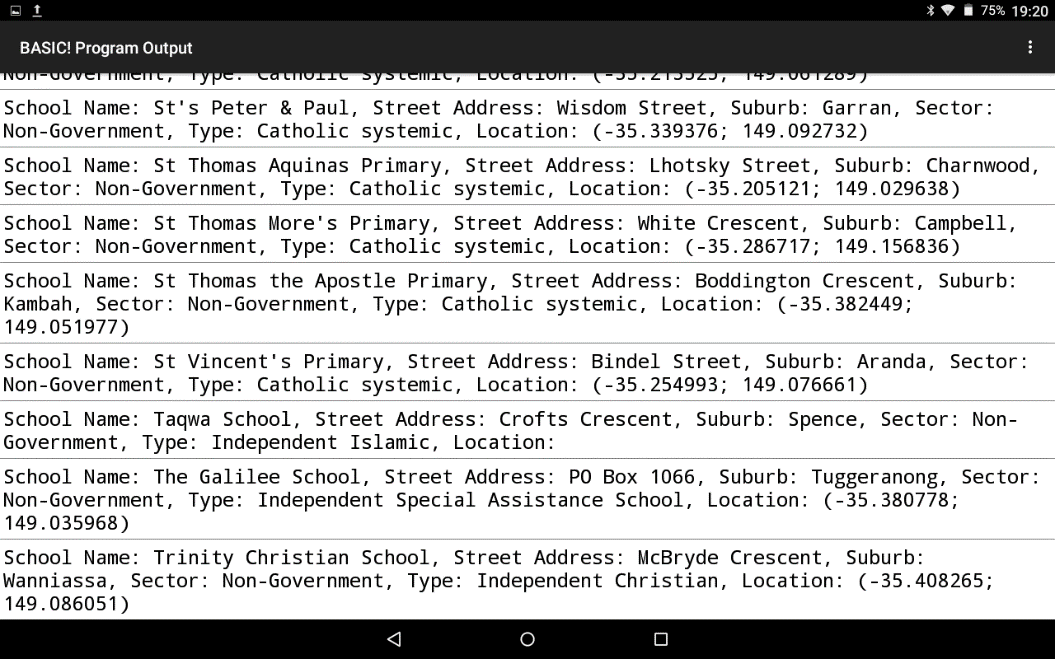


1. Select [Run](https://github.com/viru48/Toolies/blob/master/Toolies_Demo.bas)

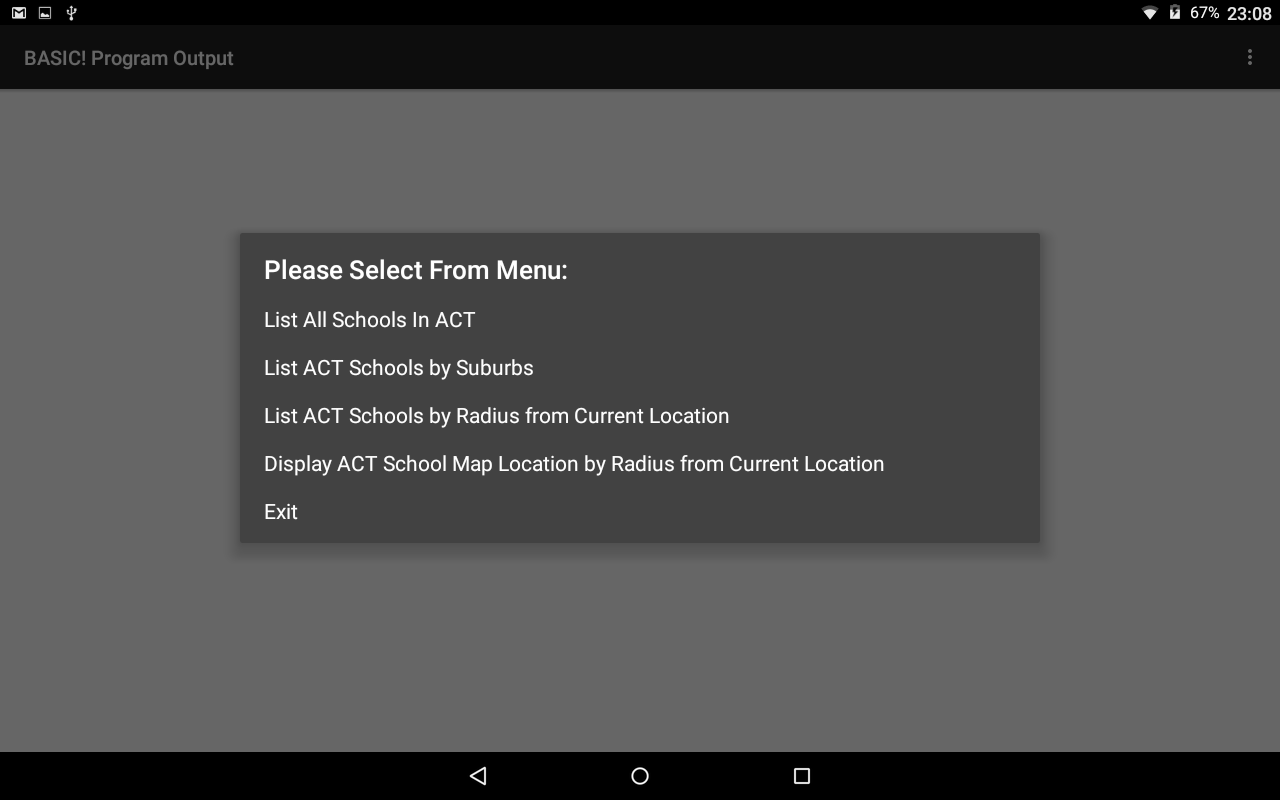


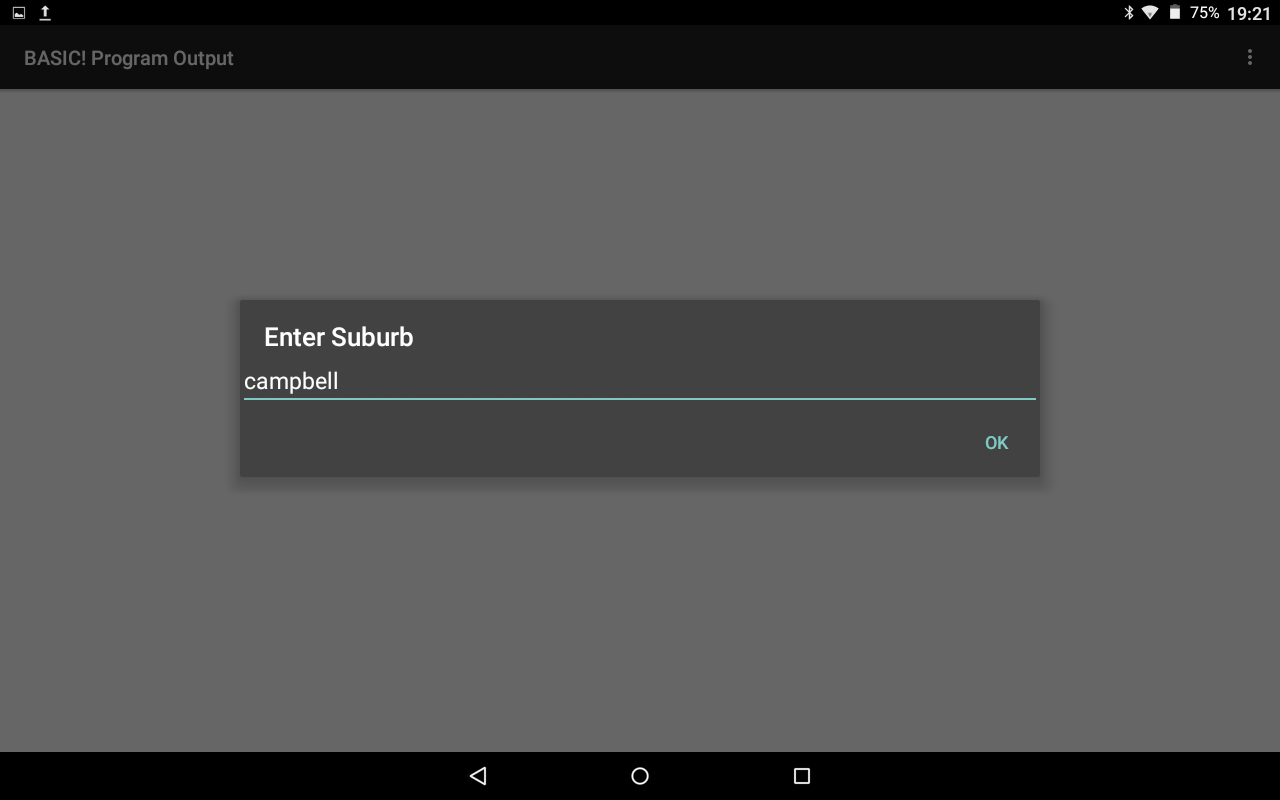
1. By Selecting List All Schools in ACT: program displays all ACT schools with their details. (Note by clicking the Back Key, program returns to Main Menu).

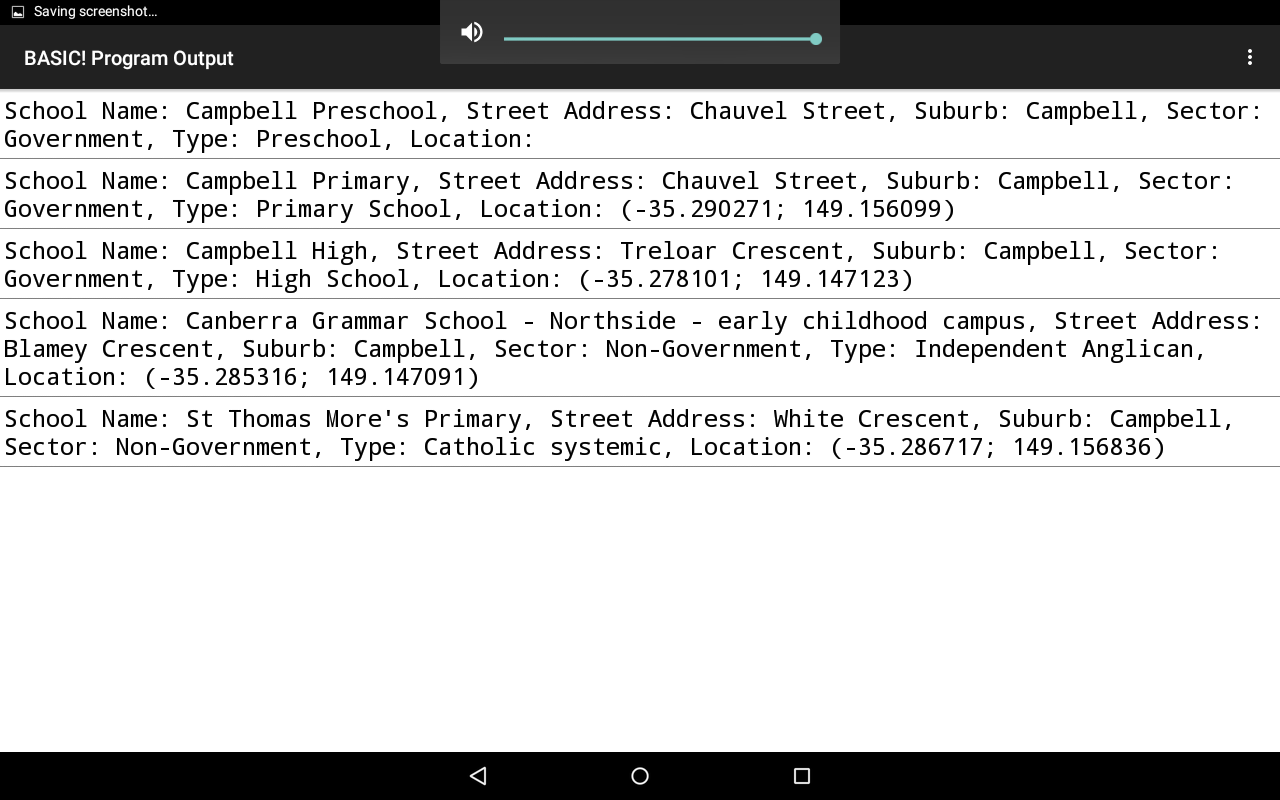




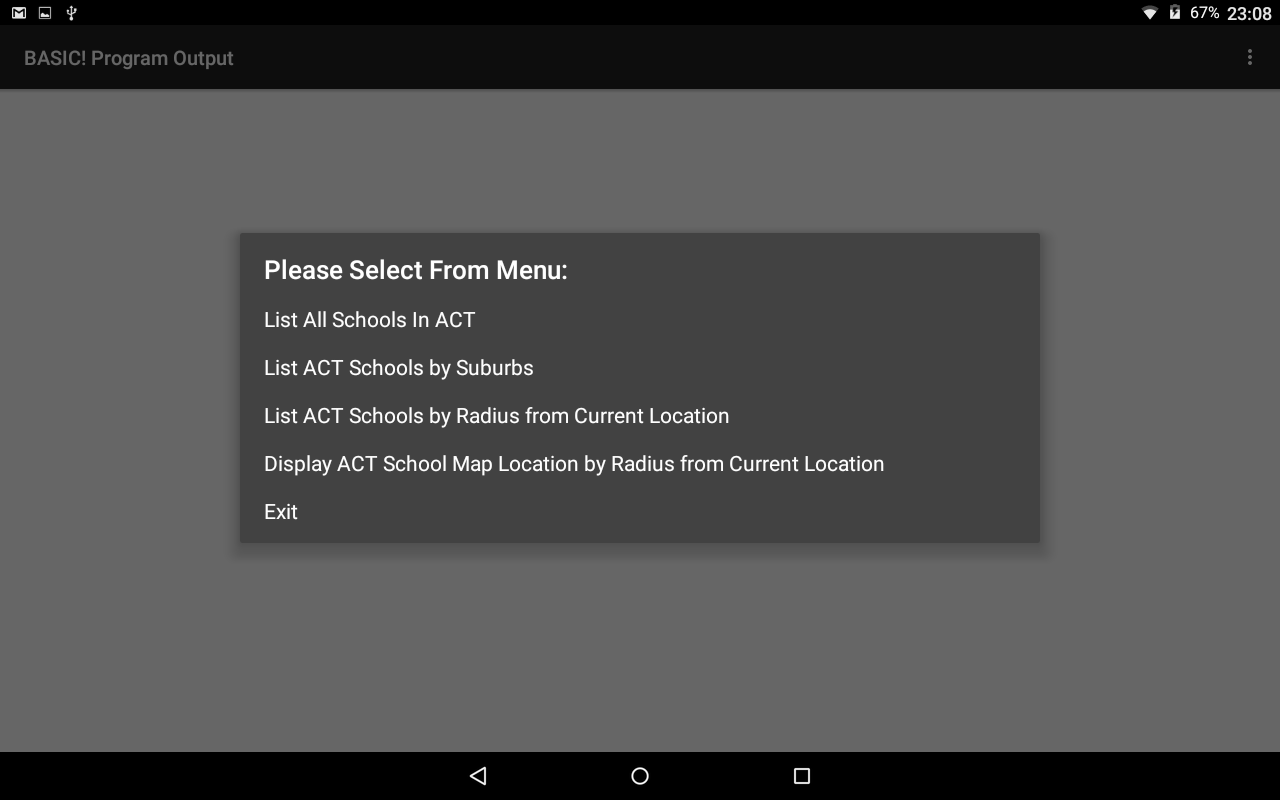
1. By Selecting List ACT Schools by Suburbs: program will prompt for a Suburb then displays all schools in Suburb (if any). (Note by clicking the Back Key, program returns to Main Menu).

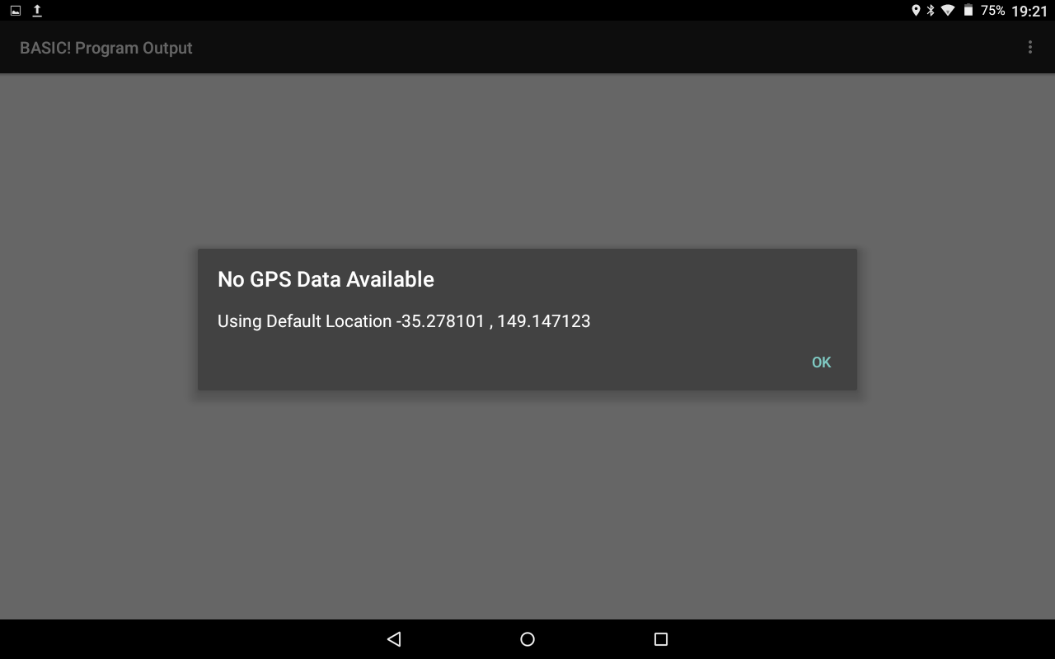


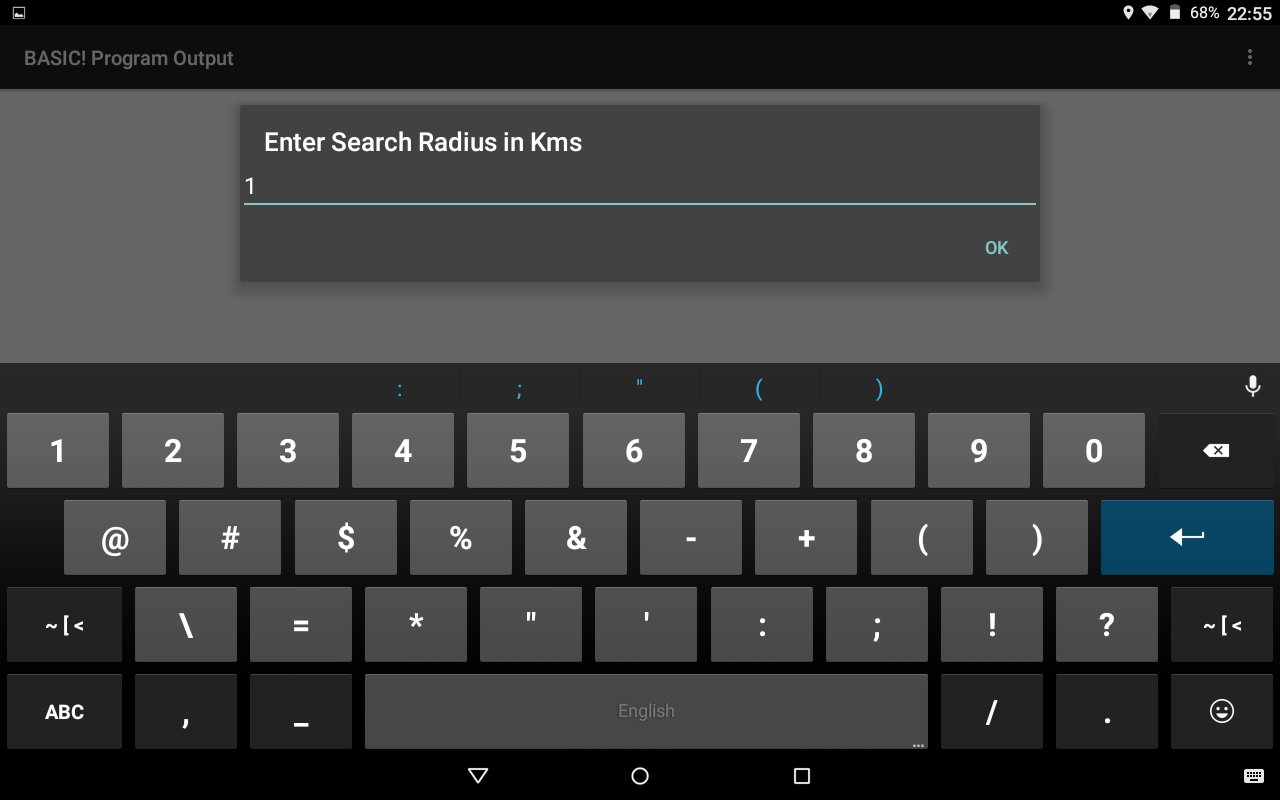


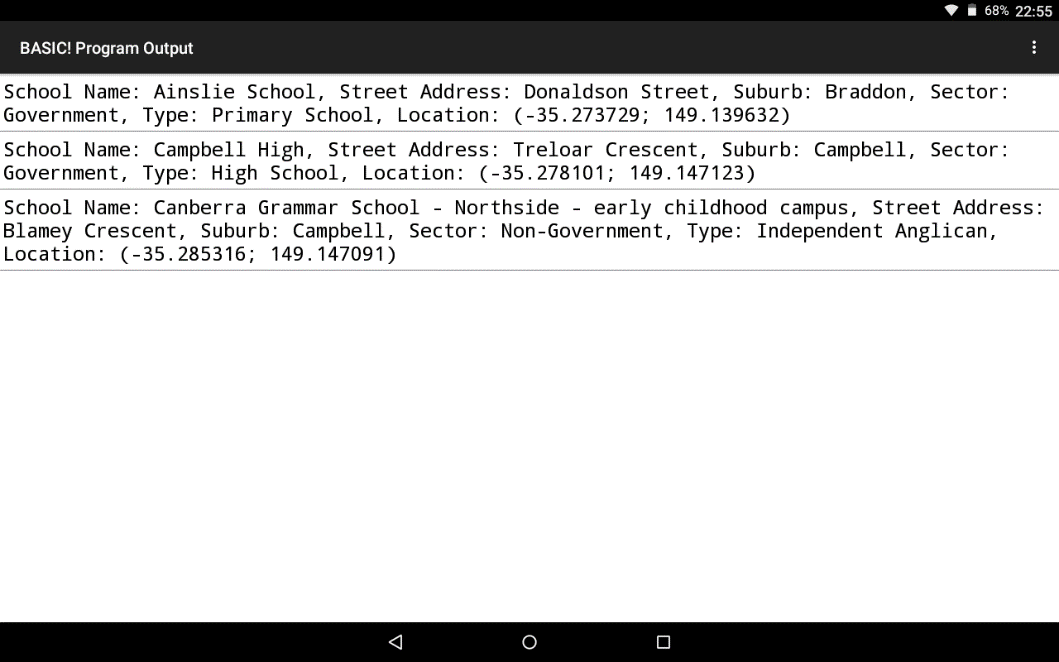


1. By Selecting List ACT Schools by Radius from Current Location: program will prompt for a Radius in Kms and then lists all schools in Area (Note, if Device GPS Location is not On, the program defaults to a Canberra Location).

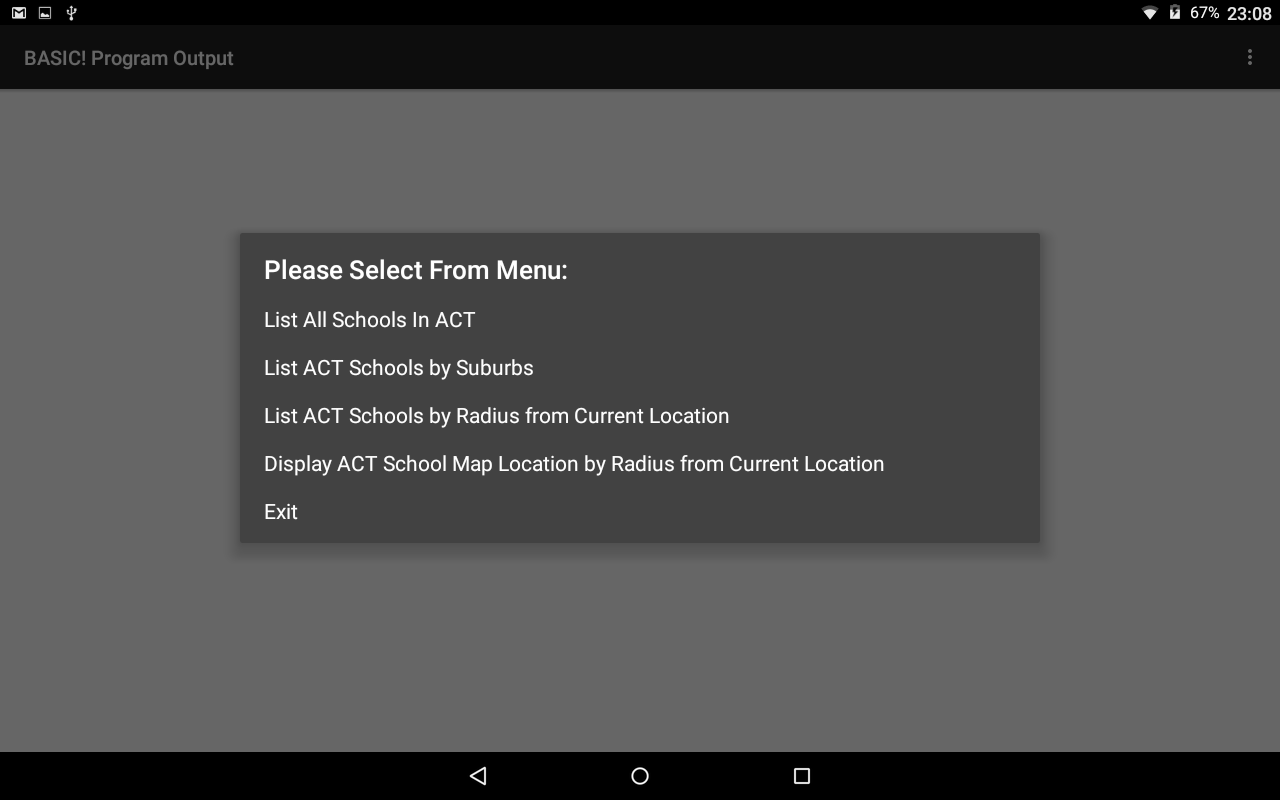


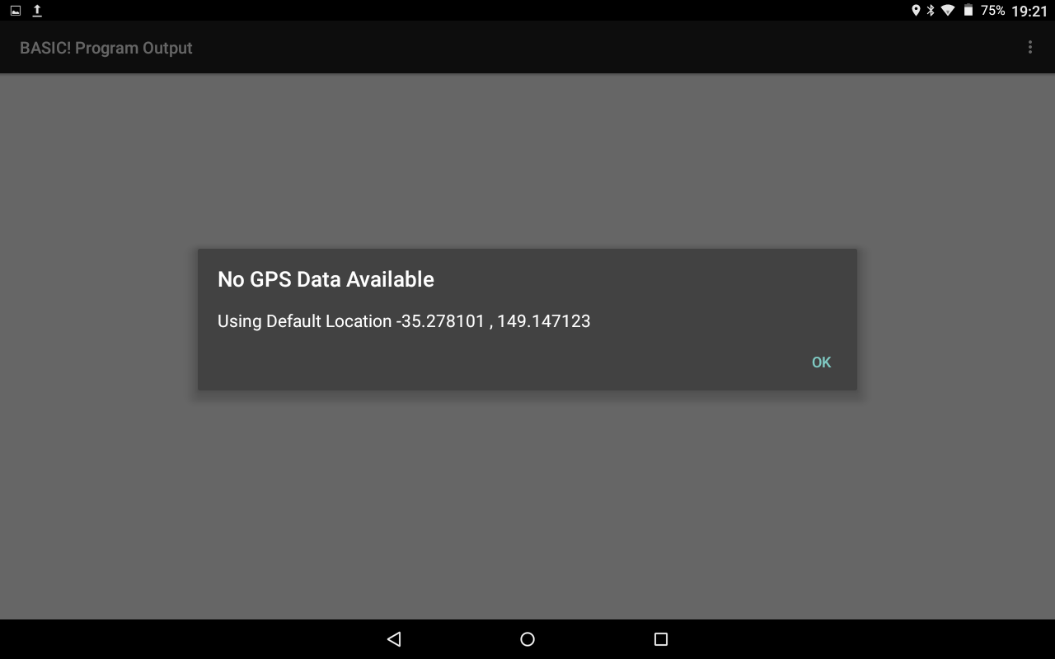


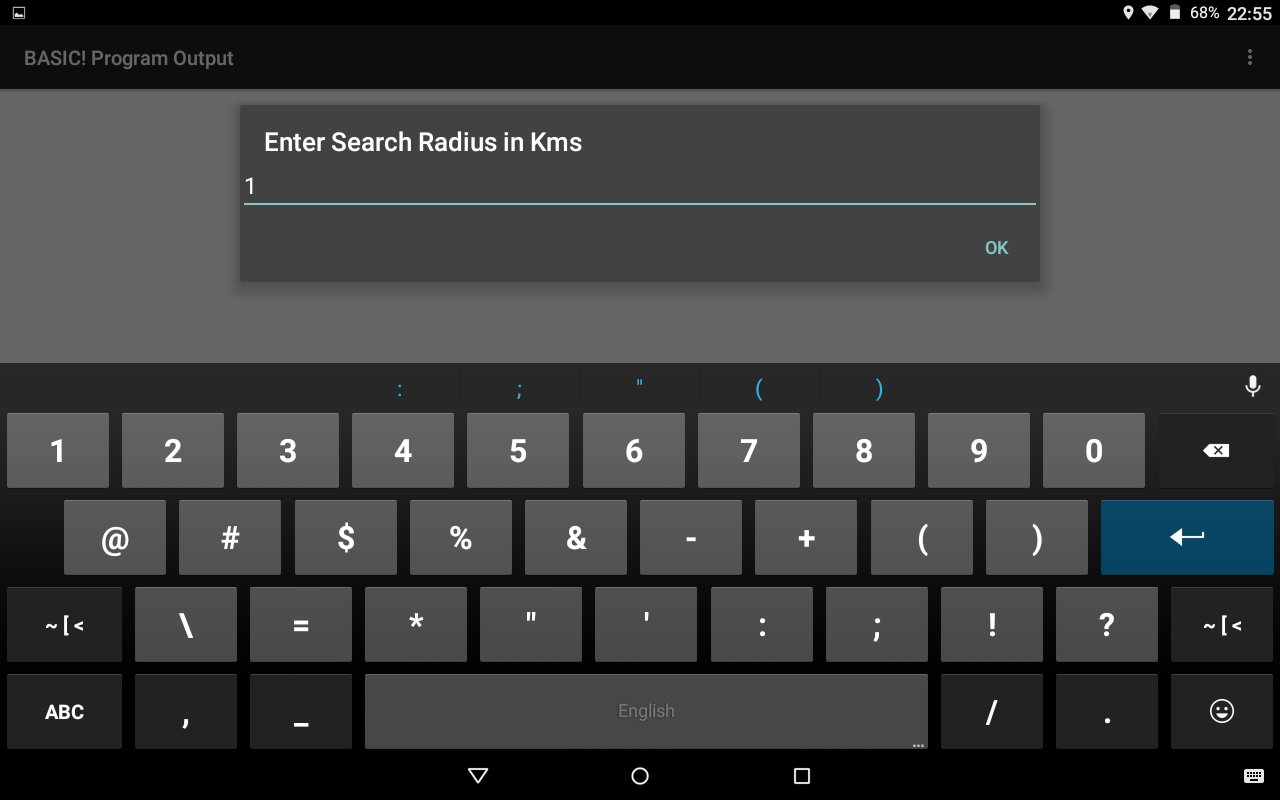


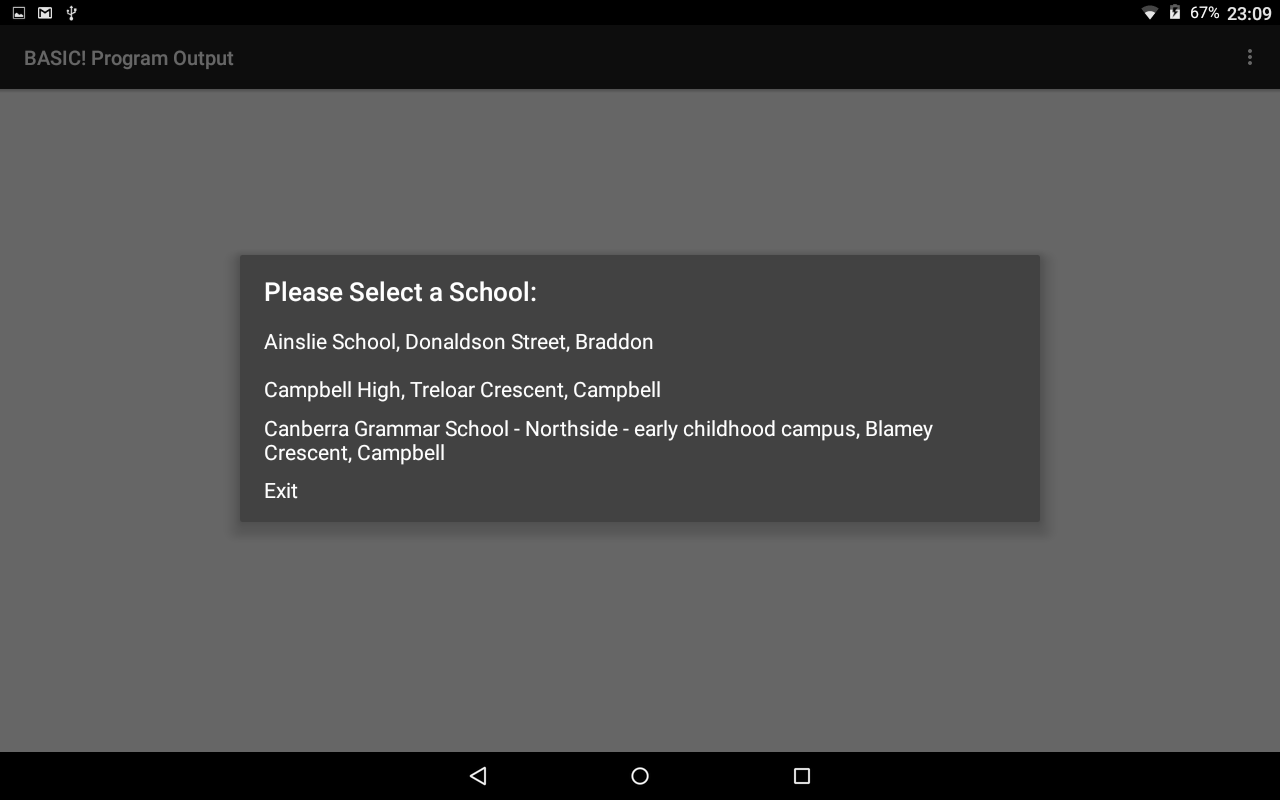


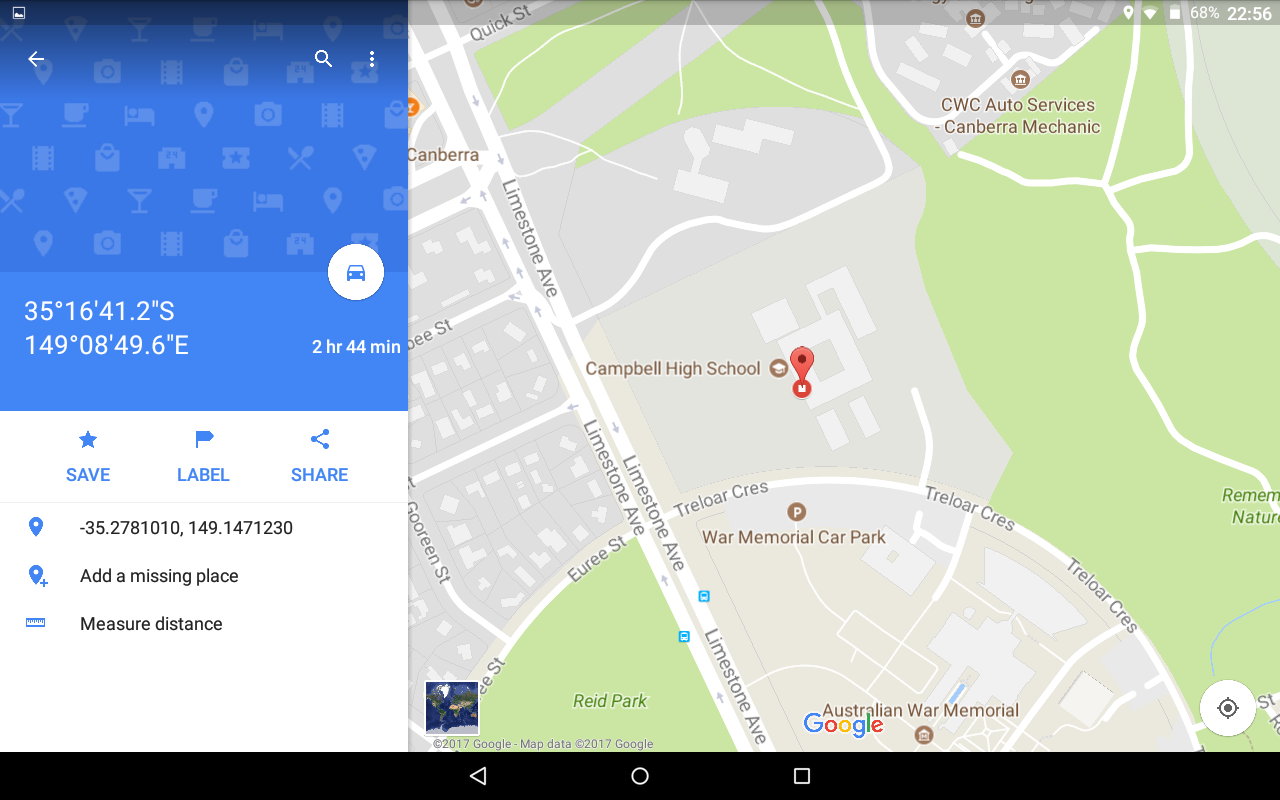
1. By Selecting Display ACT School Map Location from Current Location: program will prompt for a Radius in Kms and then lists all schools in Area (Note, if Device GPS Location is not On, the program defaults to a Canberra Location). Once a school is selected, Google Maps will display with the school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











# 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

! rec$[1,5] will store the Type for the first record

! rec$[1,6] will store the Location for the first record

! rec$[2,1] will store the School Name for the first record

! rec$[2,2] will store the Street Address for the first record

! rec$[2,3] will store the Suburb for the first record

! rec$[2,4] will store the Sector for the first record

! rec$[2,5] will store the Type for the first record

! rec$[2,6] will store the Location for the first record

FOR j =1 TO fld\_num

IF j > flds THEN

! If the last column is blank we are filling the corresponding array entry with blank

rec$[I,j] =""

ELSE

rec$[I,j] =spl$[j]

ENDIF

NEXT j

NEXT I

! close file

TEXT.CLOSE FN1

RETURN

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)

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\_Radius2:

INPUT "Enter Search Radius in Kms",KMS$

IF !IS\_NUMBER(KMS$) THEN GOTO Enter\_Radius2

km=VAL(kms$)

! Create a new Array to hold all schools in the selected radius

DIM schools$[rec\_num,5]

I=0

FOR j = 1 TO rec\_num

! we need to extract latitude and longitude from rec$[i,6]

IF rec$[j,6]<>"" THEN

position$=rec$[j,6]

lat = ExtractLatitude(position$)

long = ExtractLongitude(position$)

IF ABS(GpsDistance(latitude, longitude, lat,long) ) <= km THEN

I=I+1

schools$[I,1]=rec$[j,1]

schools$[I,2]=rec$[j,2]

schools$[I,3]=rec$[j,3]

schools$[I,4]=STR$(lat)

schools$[I,5]=STR$(long)

ENDIF

ENDIF

NEXT j

GPS.CLOSE

DIM map$[i+1]

FOR j = 1 TO I

map$[j] = schools$[j,1] +", "+schools$[j,2]+", "+schools$[j,3]

NEXT j

map$[j]="Exit"

MAP\_MENU:

IF I=0 THEN

DIALOG.MESSAGE "No Schools In This Area", " ",go,"ok"

ELSE

msg2$ ="Please Select a School: "

DIALOG.SELECT map, map$[], msg2$

IF map = j THEN

RETURN

ELSEIF map > O & map < j THEN

url$="http://maps.google.com/?q=" + schools$[map,4]+","+schools$[map,5]

BROWSE url$

GOTO MAP\_MENU

ENDIF

ENDIF

RETURN

WaitLoop:

w=0

DO

PAUSE 10

UNTIL w=1

ONBACKKEY:

v\_error\_back =1

RETURN