

Developing an App

One of the most popular forms of programming in recent times is the creation of mobile applications or apps. These run on mobile devices such as phones and tablets. You probably use a lot of different apps in your everyday life. Would not it be awesome to create one of your own?



Let's Learn About

- Using tools to develop an app
- Using an app inventor to develop an app

Mobile apps are developed for a certain platform, and the two most popular platforms today are iOS and Android. The iOS platform powers apps built for the iPhone and iPad. On the other hand, Android apps are developed for Android phones. These apps are developed with the Java programming language and the Android Software Development Kit. There are other mobile platforms such as Windows, but iOS and Android are the most popular. App development is the process of designing and coding mobile applications in a software development environment that provides some tools to develop the apps.

TOOLS USED TO DEVELOP APPS

The tools used to develop apps are as follows.

Design window with components

In design window, you develop the interface and the look of the app with the help of controls such as buttons, textboxes, drawing, media controls and so on.

Code window

In code window, you write or give instructions to the computer.

What is the interface of
button etc that help us to
create an app.

Programming language

Programming language enables you to give instructions to the computer to do a task by writing code.

Emulator

A software that appears as a window on computers to simulate a mobile device.

MIT APP INVENTOR

Among many app developing software environments, App Inventor is widely used to develop apps for the Android operating system. It is an open-source web application. It is a joint venture of Google and MIT (Massachusetts Institute of Technology). It is an easy and innovative platform to introduce programming and app creation to beginners.



Info Box

MIT is a research university in Cambridge imparting advance knowledge to students in various fields such as science, technology, humanities, architecture and management.

App Inventor uses a graphical interface that resembles the Scratch Programming interface where you just drag and drop components and blocks in different windows to design an application.

Developing an app with App Inventor

Objective: Develop an app to display the message 'It's easy to develop apps in App Inventor' as soon as you run the app and the app screen appears.

1. Open the MIT App Inventor home page www.appinventor.mit.edu in a web browser. Click on the Create apps! button at the top-right corner (Fig. 7.1).



Fig. 7.1 First step of creating an app

2. A new window appears (Fig. 7.2). Enter your existing email id and password. If you do not have an email account, create it and use the credentials to sign in. Click on the NEXT button.

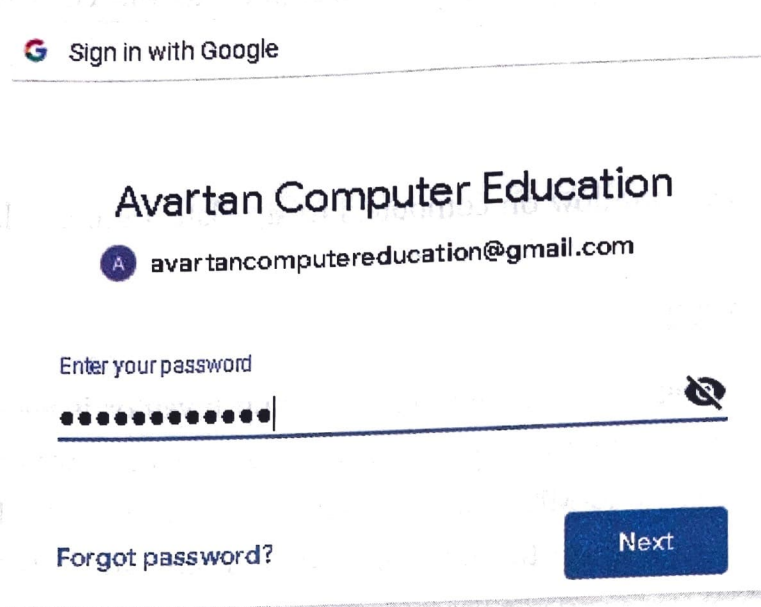


Fig. 7.2 Sign in to App Inventor

3. After you sign in, you will be taken to the Terms of Service page. To reach the Terms of Service page you have to click on the Allow button of the Google Accounts web page. After reaching the Terms of Service page, click on the I accept the terms of service! button (Fig. 7.3).

To use App Inventor for Android, you must accept the following terms of service.

Terms of Service

MIT App Inventor Privacy Policy and Terms of Use

MIT Center for Mobile Learning

Welcome to MIT's Center for Mobile Learning's App Inventor website (the "Site"). The Site runs on Google's App Engine service. You must read and agree to these Terms of Service and Privacy Policy (collectively, the "Terms") prior to using any portion of this Site. These Terms are an agreement between you and the Massachusetts Institute of Technology. If you do not understand or do not agree to be bound by these Terms, please immediately exit this Site.

MIT reserves the right to modify these Terms at any time and will publish notice of any such modifications online on this page for a reasonable period of time following such modifications, and by changing the effective date of these Terms. By continuing to access the Site after notice of such changes have been posted, you signify your agreement to be bound by them. Be sure to return to this page periodically to ensure familiarity with the most current version of these Terms.

Description of MIT App Inventor

From this Site you can access MIT App Inventor, which lets you develop applications for Android devices using a web browser and either a connected phone or emulator. You can also use the Site to store your work and keep track of your projects. App Inventor was originally developed by Google. The Site also includes documentation and educational content, and this is being licensed to you under the Creative Commons Attribution 4.0 International license (CC BY 4.0).

Account Required for Use of MIT App Inventor

In order to log in to MIT App Inventor, you need to use a Google account. Your use of that account is subject to Google's Terms of Service for

I accept the terms of service!

Fig. 7.3 Terms of Service

- After agreeing to the Terms of Service, a pop up window appears (Fig. 7.4a). Read the instructions of the Welcome to MIT App Inventor window and click on the Continue button. The App Inventor Projects View will open.
- Click on the Start new project button (Fig. 7.4b).

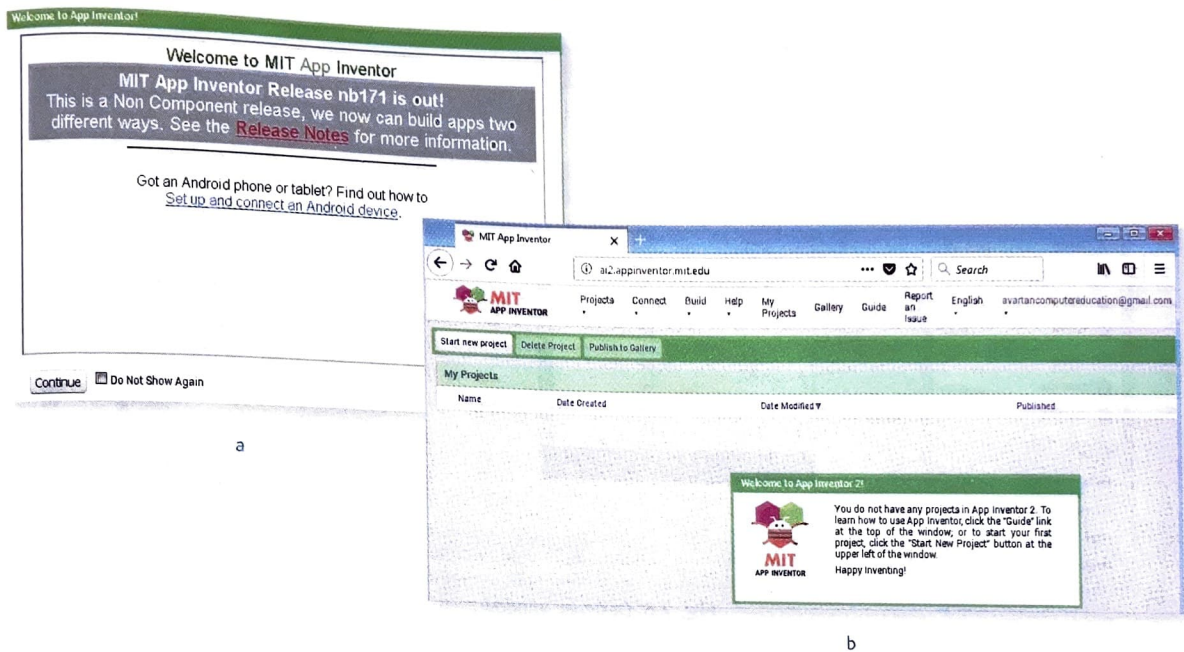


Fig. 7.4 Starting a new project

- The Create new App Inventor project dialog box appears (Fig. 7.5). Type the Project name as Display_Message. (You cannot give a space in the project name but you can use an underscore.) Click on OK.



Fig. 7.5 Naming the project

The Designer window will appear (Fig. 7.6). This is the window in which you will be working now. The Designer button at the top-right corner appears dull green. This indicates that the window is active.

2. A new window appears (Fig. 7.2). Enter your existing email id and password. If you do not have an email account, create it and use the credentials to sign in. Click on the NEXT button.

Sign in with Google

Avartan Computer Education

avartancomputereducation@gmail.com

Enter your password

.....

Forgot password? Next

Fig. 7.2 Sign in to App Inventor

3. After you sign in, you will be taken to the Terms of Service page. To reach the Terms of Service page you have to click on the Allow button of the Google Accounts web page. After reaching the Terms of Service page, click on the I accept the terms of service! button (Fig. 7.3).

To use App Inventor for Android, you must accept the following terms of service.

Terms of Service

MIT App Inventor Privacy Policy and Terms of Use

MIT Center for Mobile Learning

Welcome to MIT's Center for Mobile Learning's App Inventor website (the "Site"). The Site runs on Google's App Engine service. You must read and agree to these Terms of Service and Privacy Policy (collectively, the "Terms") prior to using any portion of this Site. These Terms are an agreement between you and the Massachusetts Institute of Technology. If you do not understand or do not agree to be bound by these Terms, please immediately exit this Site.

MIT reserves the right to modify these Terms at any time and will publish notice of any such modifications online on this page for a reasonable period of time following such modifications, and by changing the effective date of these Terms. By continuing to access the Site after notice of such changes have been posted, you signify your agreement to be bound by them. Be sure to return to this page periodically to ensure familiarity with the most current version of these Terms.

Description of MIT App Inventor

From this Site you can access MIT App Inventor, which lets you develop applications for Android devices using a web browser and either a connected phone or emulator. You can also use the Site to store your work and keep track of your projects. App Inventor was originally developed by Google. The Site also includes documentation and educational content, and this is being licensed to you under the Creative Commons Attribution 4.0 International license (CC BY 4.0).

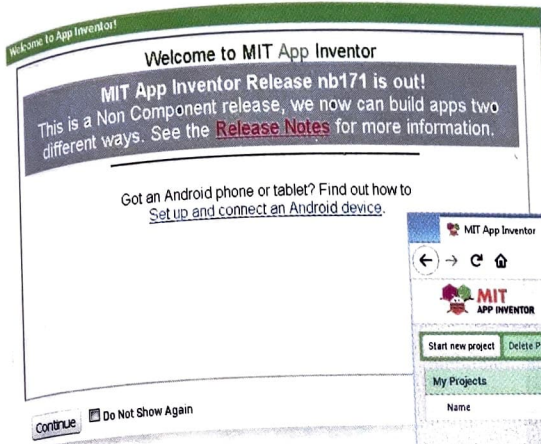
Account Required for Use of MIT App Inventor

In order to log in to MIT App Inventor, you need to use a Google account. Your use of that account is subject to Google's Terms of Service for

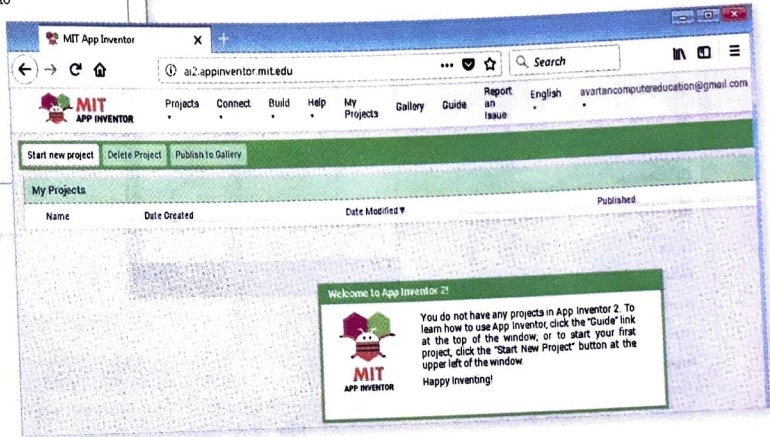
I accept the terms of service!

Fig. 7.3 Terms of Service

4. After agreeing to the Terms of Service, a pop up window appears (Fig. 7.4a). Read the instructions of the Welcome to MIT App Inventor window and click on the Continue button. The App Inventor Projects View will open.
5. Click on the Start new project button (Fig. 7.4b).



a



b

Fig. 7.4 Starting a new project

6. The Create new App Inventor project dialog box appears (Fig. 7.5). Type the Project name as Display_Message. (You cannot give a space in the project name but you can use an underscore.) Click on OK.

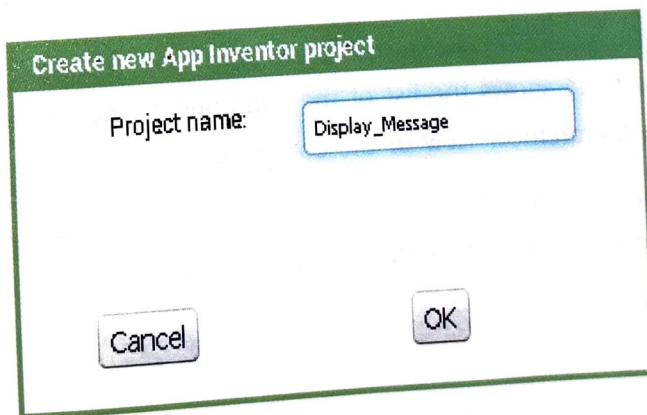


Fig. 7.5 Naming the project

The Designer window will appear (Fig. 7.6). This is the window in which you will be working now. The Designer button at the top-right corner appears dull green. This indicates that the window is active.

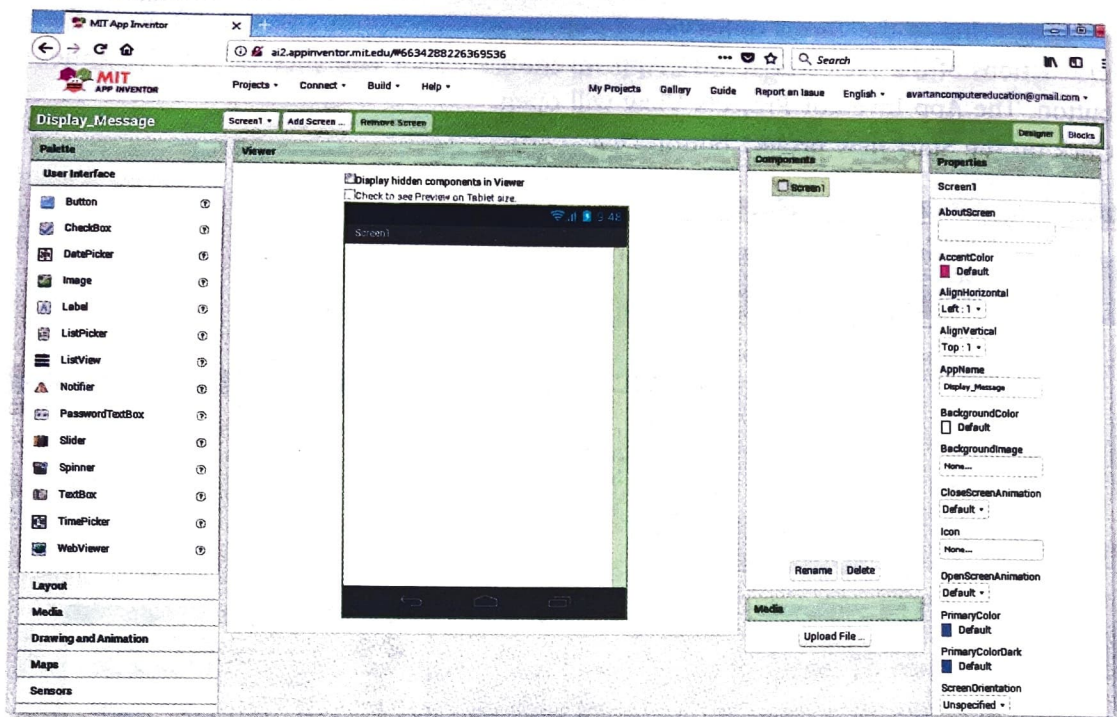


Fig. 7.6 Designer window

The Designer window is divided into four panes: Palette, Viewer, Components and Properties.

- **Palette:** This pane contains the components that are required to design a graphical user interface. It includes options such as User Interface, Layout, Media, Drawing and Animation and so on.
- **Viewer:** This pane has a screen where the components are arranged to show how the app you design will look.
- **Components:** This pane shows the list of the components that are arranged on the app screen. It shows a hierarchical view of the components.
- **Properties:** This pane shows various properties of the selected component. These properties are used to change the settings of the selected component.



Info Box

WhatsApp, a messaging app, is a global application with a strong user base of around one billion users worldwide. Among these, India is the largest country in terms of monthly active users. In May 2014, India had 50 million active users on WhatsApp. This number increased drastically by 150 million within a duration of 3 years. By 2018, the total number of Indian WhatsApp users will cross over 210 million.

- From the User Interface section of the Palette pane, click on the Label component and drag it to Viewer Screen1. Figure 7.7 shows how the screen will look like when you do this.

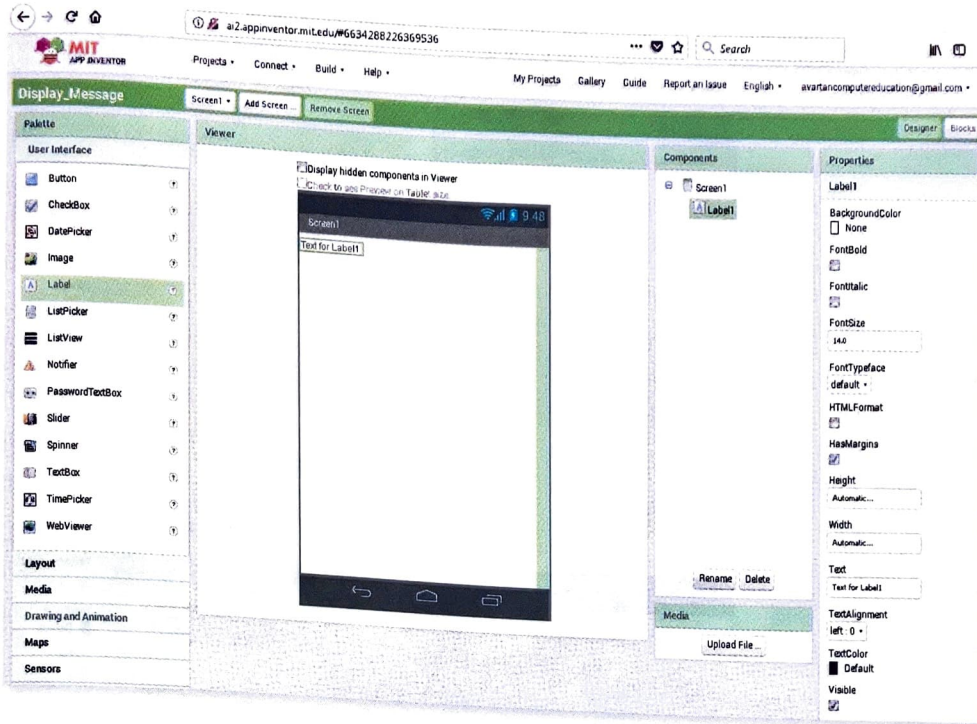


Fig. 7.7 Working with Palette pane



Info Box

In 2015, most of the Internet traffic observed was from social networking applications such as Facebook, Twitter, Instagram and so on.

- Select the Label1 component from the Palette pane.
- In the Properties pane, locate the 'Text' property and delete its contents, 'Text for Label1' (Fig. 7.8). The Label component in the Viewer pane becomes small after its contents are removed. The Components pane shows Label1 as selected. The Properties pane shows the blank Text property of the Label component.



Info Box

42% of app developers choose iOS as a mobile app development platform, while 31% choose Android.

- When you click on Screen1 from the Blocks pane. The corresponding blocks are displayed in the Viewer pane. Locate the block when Screen1.Initialize do and drag it to the workspace (Fig. 7.11).

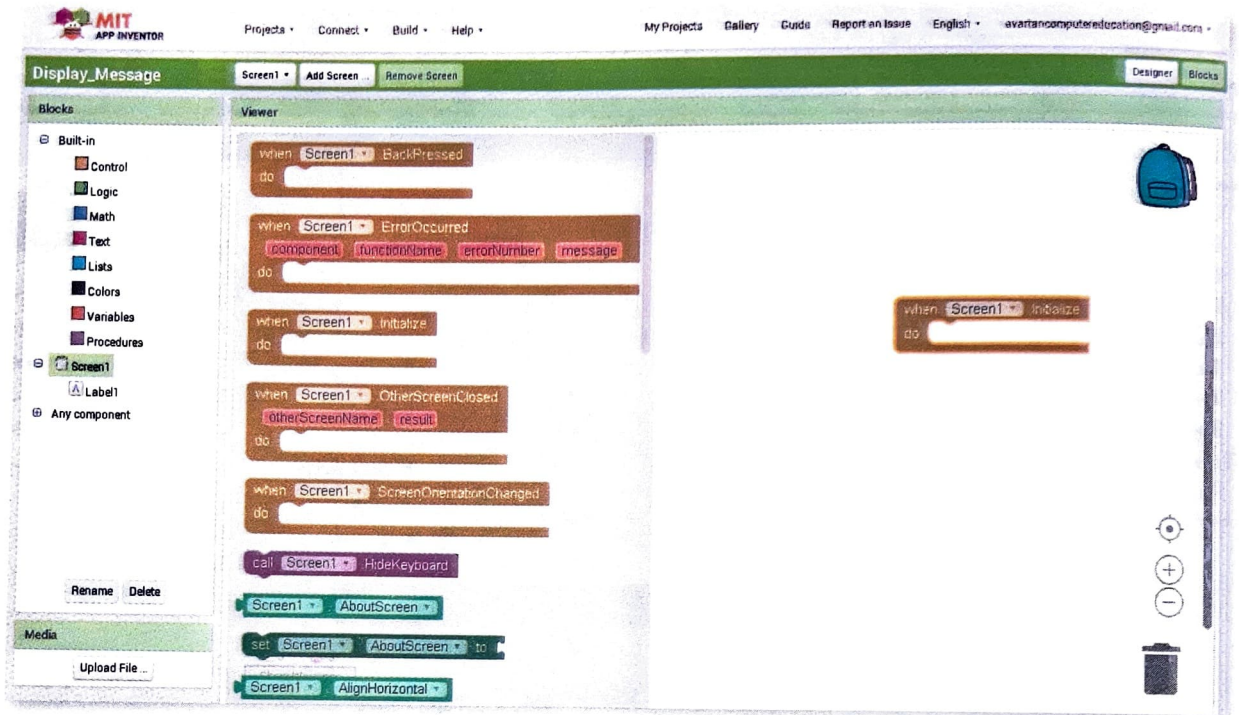


Fig. 7.11 Dragging Screen from Blocks pane

- Click on Label1 under Screen1. A screen like this appears (Fig. 7.12).

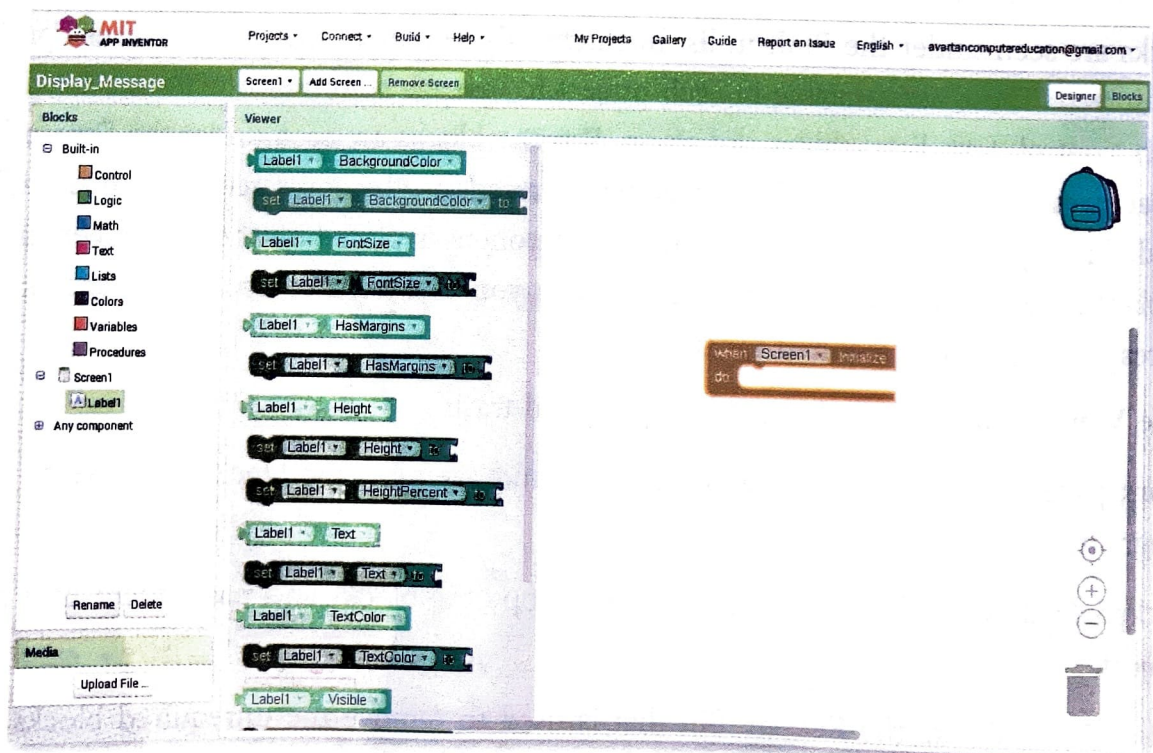
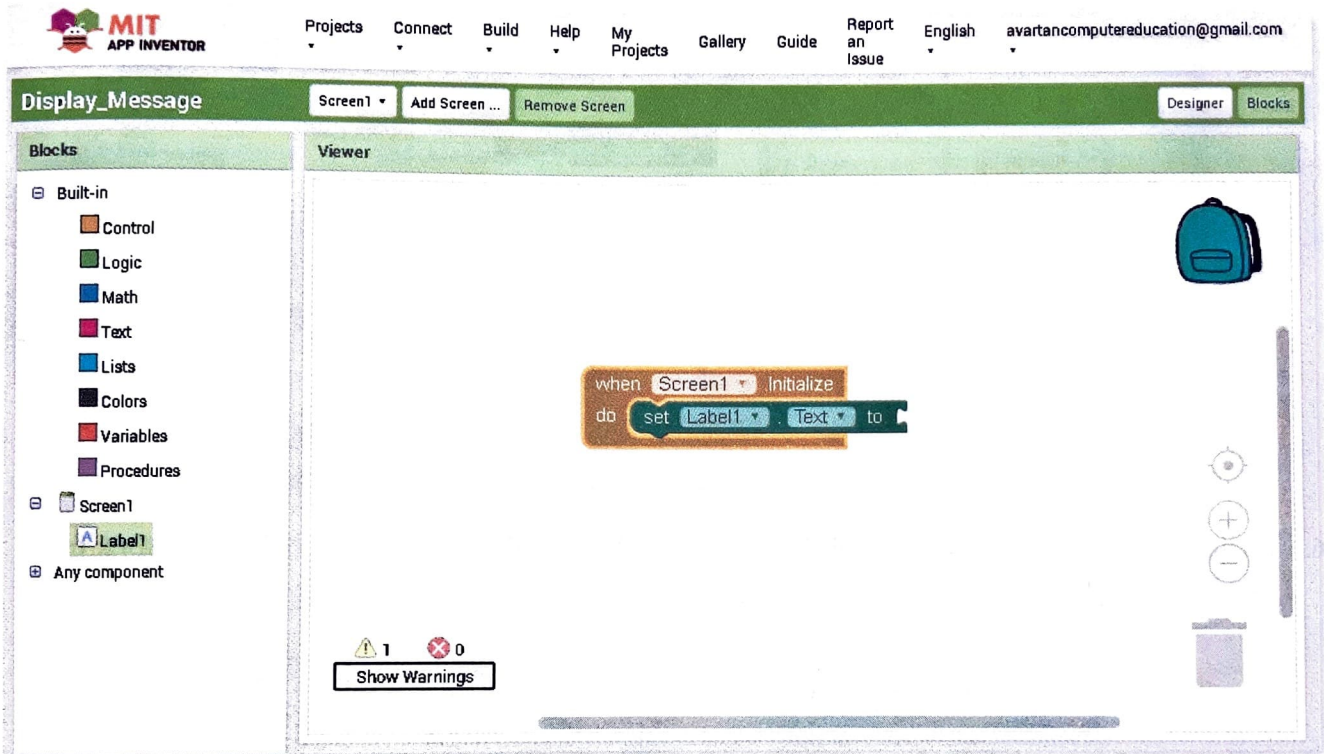


Fig. 7.12 Clicking Label under Screen1

In the screen shown in Figure 7.13, locate the block set Label1.Text to. Drag it to fit inside the when Screen1.Initialize do block on the workspace. The block set Label1.Text to will show some text inside the label. As this block is placed inside the when Screen1.Initialize do block, the app will display this text inside the label.



5. Click on Text under the Built-in block and locate the empty block of text string as shown in Figure 7.14.

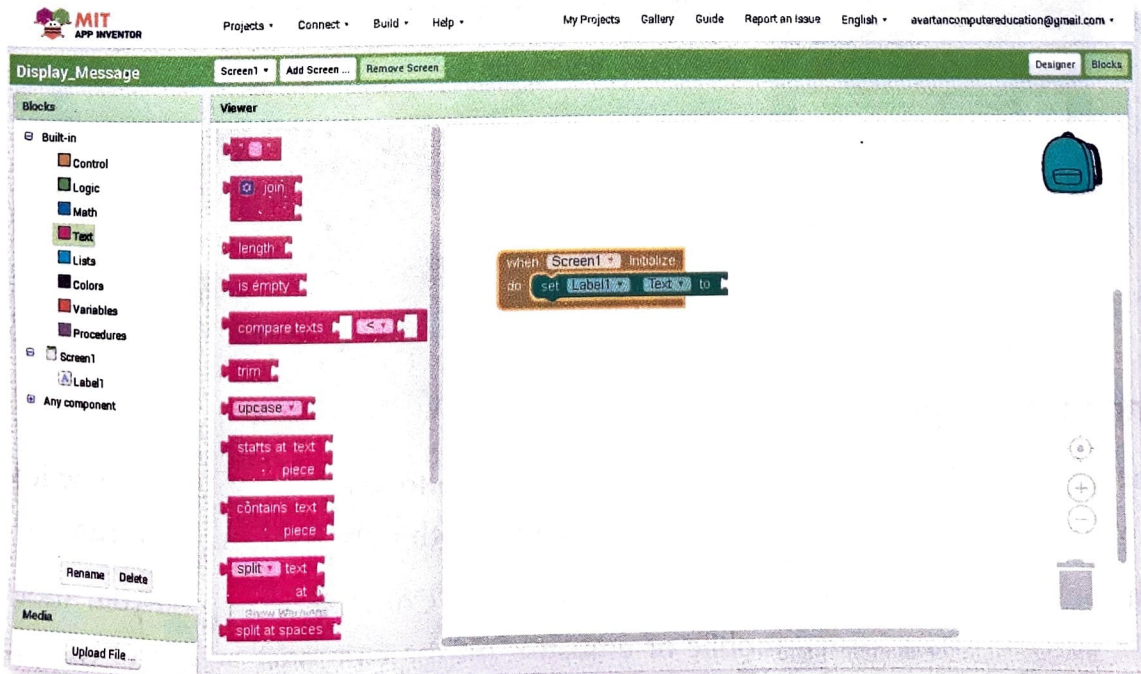


Fig. 7.14 Adding Text block

Drag it to fit inside the set Label1.Text to block on the workspace (Fig. 7.15). The empty block of text string is called a text block. You can type a message in it.

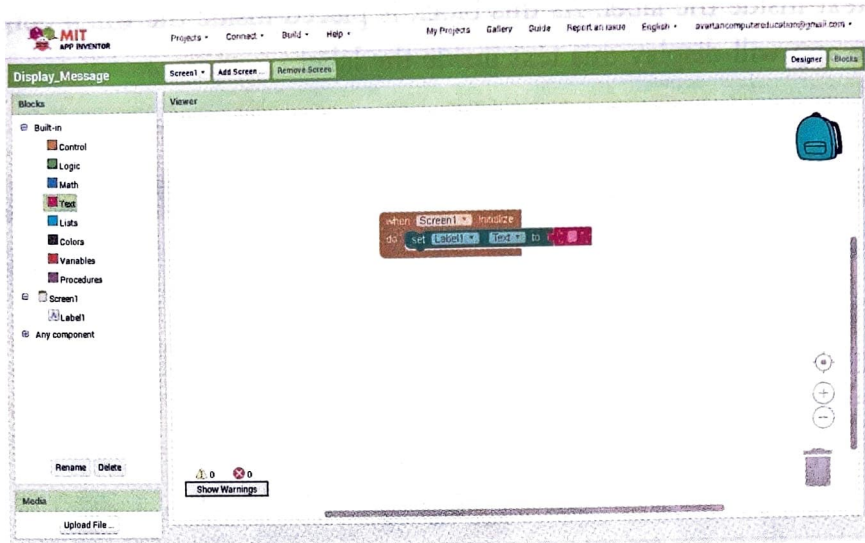


Fig. 7.15 Empty text block added

6. Write this message inside the pink text block (Fig. 7.16): 'It is easy to develop Apps in App Inventor.'

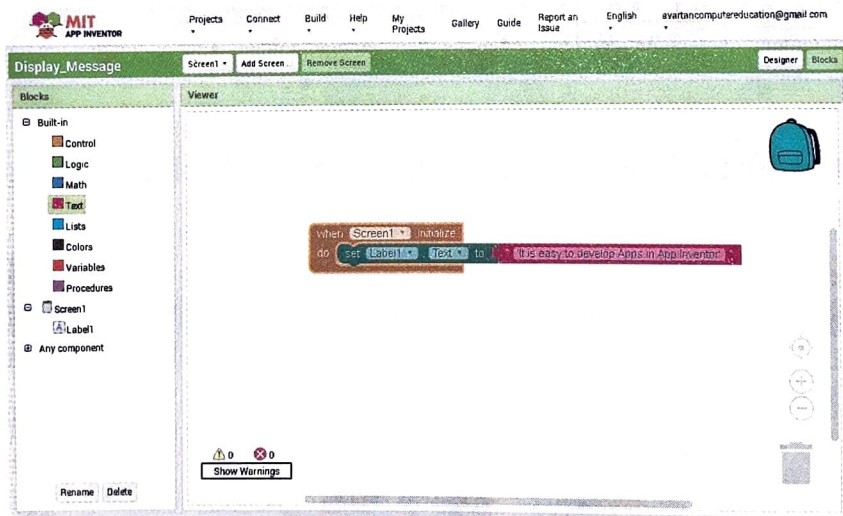


Fig. 7.16 Inserting text

Running the app

When the design and code of your app is final, you have to test the app to see how it works in a real or virtual mobile device. App Inventor provides both options to test and run the app.

To test the app in a real mobile device, download the AI Companion app on your mobile device. Or, you can use a virtual mobile or emulator on the computer screen to test the app.

You can download this from: <http://appinventor.mit.edu/explore/ai2/setup.html>.

To run the app, follow these steps.

1. Click on the Designer button to open the Designer window. From the Connect drop-down list, click on Emulator (Fig. 7.17). An emulator is hardware or software that enables a computer system (called the host) to behave like another computer system.

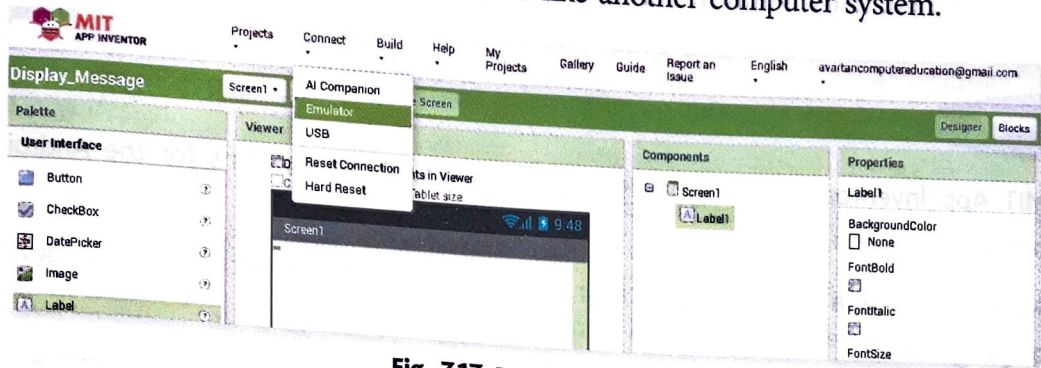


Fig. 7.17 Running Emulator

2. App Inventor will start connecting to the Emulator. A virtual mobile screen will appear (Fig. 7.18). It will start the Companion app to test your app.

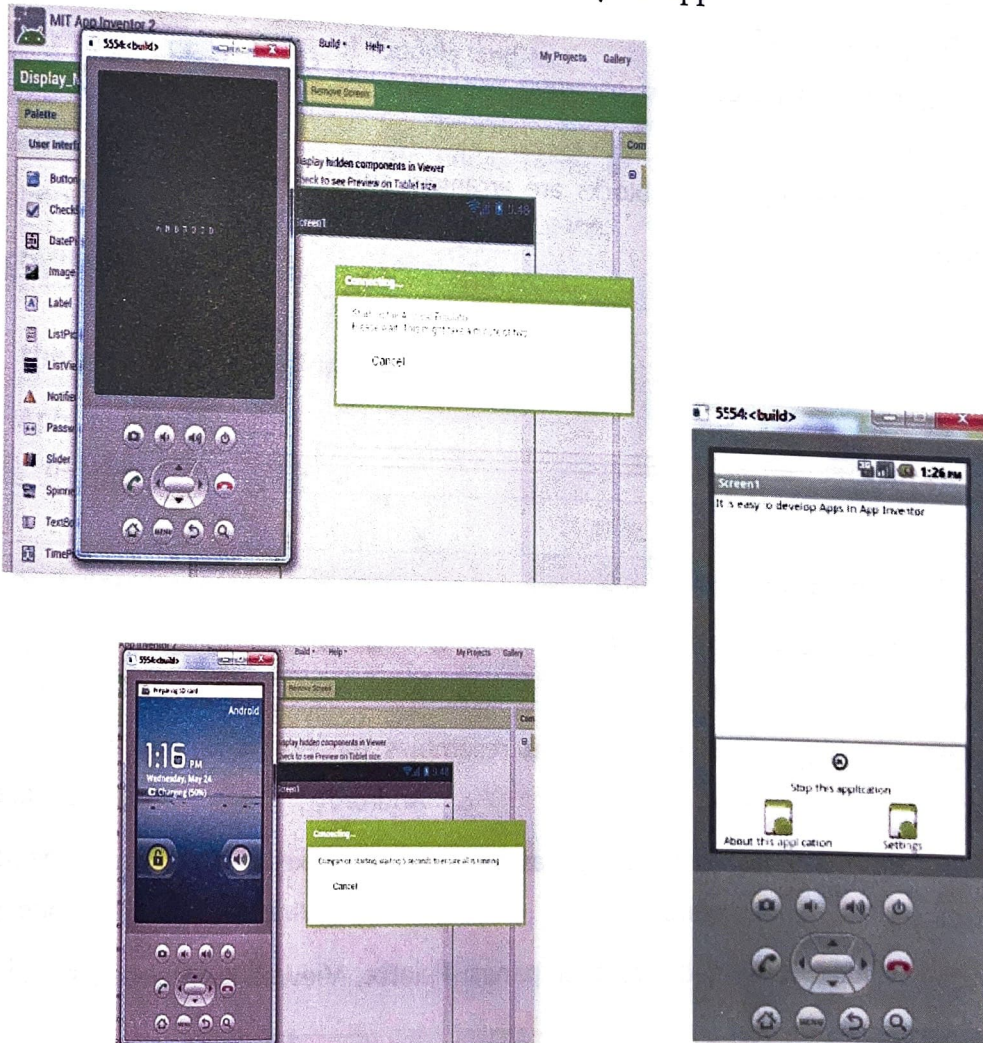


Fig. 7.18 Virtual mobile screen

3. The app starts running on the emulator and shows the final output. For the app named Display_Message, the emulator will display the message 'It is easy to develop Apps in App Inventor' as soon as the emulator window starts (Fig. 7.19).

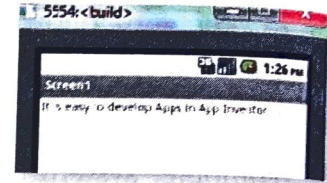


Fig. 7.19 App displaying message

Flashback

- MIT App Inventor helps beginners to create software applications for the Android operating system.
- An emulator is a hardware device or program that pretends to be another device or program.
- Trash is used to remove the unrequired blocks from the workspace just like a Recycle Bin.
- Palette helps in designing of an app and includes user interface, layout, media, drawing and animation.
- Blocks is a pane that contains common working tools for designing an application.
- Workspace is an area where blocks are arranged and assembled to make an app.

