6133(15)

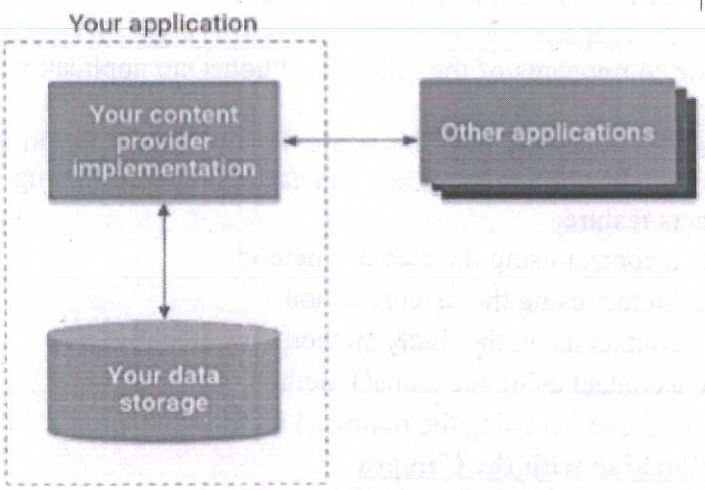
SMART DEVICE PROGRAMMING

TED (15) -6131

Qno	Scoring Indicators	Split score	Total score
I	PART A		
1	Android Studio, Eclipse,Visual Studio,NetBeans,PhoneGap(write any two).	2	2
2	-Storage :- uses SQLite -Connectivity:-supports GSM/EDGE,CDMA -Messaging, web browser, media support, multi-touch, multi-tasking	2	2
3	An activity is a window that contains the user interface of your application. An application can have zero or more activities. The main purpose of an activity is to interact with user.	2	2
4	Bundles are generally used for passing data between various Android activities. The saved data that the system uses to restore the previous state is called instance state and is a collection of key value pairs stored in a bundle object.	2	2
5	Audio,Video.	2	2
II	PART B		
1	Step-1. System Requirements -Microsoft windows/Mac OS/GNOME or KDE desktop. -Java JDK% or later version -Java Runtime Enviornment 6 -Android Studio. Step-2.Setup Android Studio -before that have to install some tools -Install the Android SDK Tools -Configure the Android SDK Manager -Create Android virtual device	6	6
2	• Src- Contains the .java source file for the project		

	• Android Library –this item contain one file ,android.jar ,which contains all the class libraries needed for an android project. • Gen- Contains the R.java file, a compiler generated file that references all the resources found in our project. • Assets-this folder contains all the resources used in our application.It also contains few other folders :drawable<resolution>,layout,and values. • AndroidManifest.xml- This is the manifest file for your android application .Here we specify the permissions needed by our application. • Basic "hello world" program contain following default files. • app/src/main/java/com.mycompany.myfirstapp/MainActivity.java • app/src/main/AndroidManifest.xml-defines components in the app • app/src/main/res/layout/activity_main.xml- Activity seen in design mode and text mode.	6	6

3	1) Implicit Intent
	Implicit intents specify the action which should be performed and optionally data which provides content for the action.
	-If an implicit intent is sent to the Android system, it searches for all components which are registered for the specific action and the fitting data type.
	-If only one component is found, Android starts this component directly.
	-If several components are identified by the Android system, the user will get a selection dialog and can decide which component should be used for the intent. This process is called intent resolution.
	-Intent filters are used for this which is specified in AndroidManifest.xml For example, the following tells the Android system to view a webpage. Intent resolution data are action,data(both uri &type) and category For example, you may write the following code to view the webpage.
	Intent intent=new Intent(Intent.ACTION_VIEW);
	intent.setData(Uri.parse("http://www.javatpoint.com"));
	startActivity(intent);
	2) Explicit Intent
	Explicit Intent specifies the component. In such case, intent provides the external class to be invoked.
	- Explicit intents explicitly define the component which should be called by the Android system, by using the Java class as identifier. -
	-Explicit intents are typically used within an application as the classes in an application are controlled by the application developer.
		6	6

	The following shows how to create an explicit intent and send it to the Android system to start an activity. Intent i = new Intent(getApplicationContext(), ActivityTwo.class); startActivity(i);		
4	A broadcast receiver is a component that receive information, broadcasted by other applications or by the system, and take some actions accordingly. For example, An app can schedule an alarm to notify the user at any specified time. It's same as we set an alarm to notify at a specific time. Another example is notify the user when battery is low. Although the user do not show user interface, it can create status bar notifications to alert the user to take a specific action. A broadcast receiver is implemented as subclass of BroadcastReceiver. public class MyReceiver extends BroadcastReceiver { public void onReceive(context,intent){} }	6	6
5	Content providers make the content of any file system, SQLite database, or on any other persistent storage, available to any other applications. i.e. you can use content providers to get a specific set of data in any application. -Means it acts as bridge between the data and the application. you can also access many native databases using content providers. - For example, you can access contacts using ContactManager. You must have proper permissions to access data of other applications. You can implement Content Providers by subclassing ContentProvider as below – public class MyProvider extends ContentProvider { public void onCreate(){} }	3+3	6
	 ▪ onCreate() This method is called when the provider is		

	started. ▪ query() This method receives a request from a client. The result is returned as a Cursor object. ▪ insert()This method inserts a new record into the content provider. ▪ delete() This method deletes an existing record from the content provider. ▪ update() This method updates an existing record from the content provider. ▪ getType() This method returns the MIME type of the data at the given URI.		
6	STYLING MOBILE PAGES WITH CSS3 Cascading Style Sheets (CSS) to define how your HTML documents will look. Style HTML for print, for web pages, and even for specific mobile devices. Creating a CSS Style Sheet CSS is made up of one or more selectors with style properties attached. For example, to change the text color of a paragraph you would write: p { color: red; } The selector is p and the style property (enclosed in the curly braces) is color: red; . To add a second selector, simply separate it with a comma: p, .redText { color: red; } After you have a style, you attach it to a web page in one of three ways: 1)Inline in the tags themselves 2)Embedded in the head of your HTML 3)In a separate document as an external style sheet:	6	6
7.	The major components of the API for a PhoneGap application are: □ **Working with Contacts** : The contacts list is a fairly ubiquitous feature, available on most smartphones. With PhoneGap, you can easily do the following with the contacts feature: ➢ Create a contact using the create() method. ➢ Save a contact using the save() method ➢ Find a contact using the find() method. ➢ Clone a contact using the clone() method. ➢ Remove a contact using the remove() method. ○ **Working with the Camera** Most smartphones have built-in cameras. The PhoneGap API provides two ways to capture images: 1. one is giving access to the camera via the camera object. 2. using the Media Capture API. **Working with Geolocation.** □ Position - The Position object contains coordinates that are	6	6

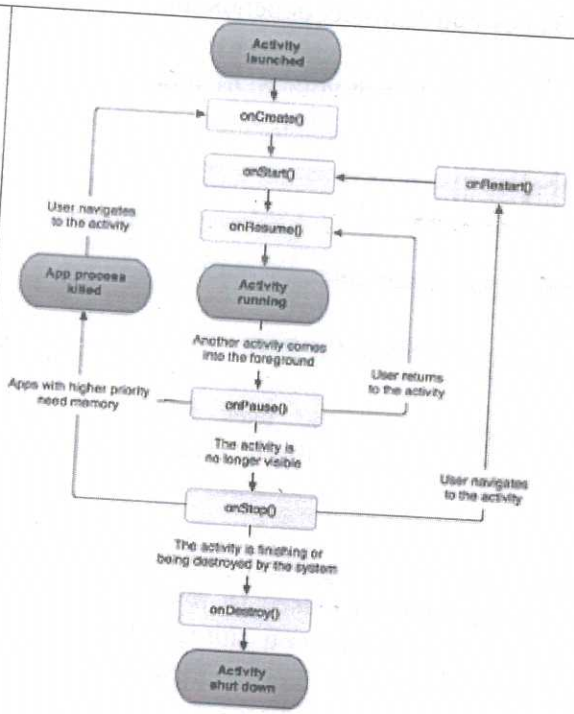
	created by the geolocation API. □ PositionError - onError callback function returns both an error code and an error message in case of a problem, encapsulated as a read-only PositionError object. □ Coordinates - the Coordinates data is attached to the position object, which is then returned to the user via the onSuccess callback function. **Working with Media Files:** **Working with Storage Options**		
	PART C		
III (a)	An Android Virtual Device (AVD) is an emulator configuration that allows developers to test the application by simulating the real device capabilities. We can configure the AVD by specifying the hardware and software options. AVD manager enables an easy way of creating and managing the AVD with its graphical interface. We can create as many AVDs as we need, based on the types of device we want to test for.. This emulator to provide a virtual device-specific environment in which to install and run Android apps. The Procedure For Creating Android Virtual Device □ In Android Studio, select Tools > Android > AVD Manager, or click the AVD Manager icon in the toolbar->Create AVD □ Or, use the command line to navigate to your SDK's tools/ directory and execute: $ android avd □ In Eclipse ADT, Below are the steps to create an AVD from AVD manager graphical interface Go to Window ->AVD Manager and select Virtual Devices. Click on New to create a Virtual Device, give it some Name and select Target Android Platform from the drop down list o Click "Create AVD".	9	9

(b)			6	6

	Android	iOS
Customizability	A lot. Can change almost anything.	Limited unless jailbroken
Developer	Google, Open Handset Alliance	Apple Inc.
Initial release	September 23, 2008	July 29, 2007
Source model	Open source	Closed, with open source components.
OS family	Linux	OS X, UNIX
Widgets	Yes	No, except in NotificationCenter
File transfer	Easier than iOS	More difficult. Media files can be transferred using iTunes desktop app.
Available on	Many phones and tablets. Major manufacturers are Samsung, Motorola, LG, HTC and Sony. Nexus and Pixel line of devices is pure Android, others bundle manufacturer software.	iPod Touch, iPhone, iPad, Apple TV (2nd and 3rd generation)
Calls and messaging	Google Hangouts, 3rd party apps like Facebook Messenger, WhatsApp, Google Duo and Skype all work on Android and iOS both.	iMessage, FaceTime (with other Apple devices only). 3rd party apps like Google Hangouts, Facebook Messenger, WhatsApp, Google Duo and Skype all work on Android and iOS both.

IV (a)	**1. Install the Java Development Kit (JDK)** Step 1:$sudo add-apt-repository ppa:webupd8team/java Step 2: $ sudo apt-get update Step 3:$ sudo apt-get install oracle-java8-installer oracle-java8-set-default Verify if java installation was successful or not by checking the version number Step4: : $ java -version 2.Download Eclipse IDE from its website. Check out your OS Type, 32-bit or 64-bit, and select Linux 32-bit or 64-bit of Eclipse Installer. 3.Run the installer wizard. Decompress the downloaded archive - **$tar xf eclipse-inst-linux64.tar.**gz Run eclipse-inst - **$./eclipse-inst** When the wizard launches, select install item and folder, and finally click INSTALL button. 4. Download the ADT plugin for eclipse 5. Configuring the ADT plugin 6. Create an Android Virtual Device (AVD) 7. Create and run the simple android example	9	9
(b)	To download and install latest android APIs and development tools from the internet, android provide us with android SDK manager. Android SDK Manager separates the APIs, tools and different platforms into different packages. □ **launch Android SDK Manager - -** • You can select which package you want to download by selecting the checkbox and then click Install to install those		

	packages. • By default SDK Manager keeps it up to date with latest APIs and other packages. Gradle can automatically download missing SDK packages that a project depends on. • Once you download the SDK, following packages are available but first three are necessary to run your SDK and others are recommended.	6	6
V (a)	Method Description **onCreate** -Called when activity is first created. Used to initialize the activity, for example create the user interface. onStart Called when activity is becoming visible to the user. **onResume** -Called if the activity get visible again and the user starts interacting with the activity again. Used to initialize fields, register listeners, bind to services, etc. **onPause** -Called once another activity gets into the foreground. Always called before the activity is not visible anymore. Used to release resources or save application data. For example you unregister listeners, intent receivers, unbind from services or remove system service listeners. **onStop** -Called once the activity is no longer visible. Time or CPU intensive shut-down operations, such as writing information to a database should be down in the onStop() method. **onRestart-** Called after your activity is stopped, prior to start. **onDestroy-** Called before the activity is destroyed.	Fig 8 Exp 7	15



VI (a)	There are 4 main components of any android application. They are 1. Activities 2. Services 3. Broadcast Receivers 4. Content Providers 1.Activity An activity is single screen with user interface. For example, A activity can have user interface for signup process, Another activity can have user interface for login process An activity provides following features : 1. It keeps track of the what the user currently using. 2. It keeps track of the previously visited activities or other things so that user may return to it whenever needed. 3. It also provides a way for apps to implement user flows between each other 2.Services A service is component that runs in background to perform long running operations or to perform work for remote processes. public class MyService extends Service { } 3.Broadcast Receivers A broadcast receiver is a component that receive information, broadcasted by other applications or by the system, and take some actions accordingly. public class MyReceiver extends BroadcastReceiver { public void onReceive(context,intent){} }	4 7	7

	4.Content Providers
	Content providers make the content of any file system, SQLite database, or on any other persistent storage, available to any other applications.
	
	public class MyProvider extends ContentProvider {
	public void onCreate(){}
	}		

(b)	This method is the most elegant one because it uses all the android features. Intent in android is an action description. Intent supports three ways to pass data:
	• Direct: put our data into intents directly
	• Bundle: create a bundle and set the data here
	• Parcelable: It is a way of "serializing" our object
	Passing data: Direct
	-Intent has several method called **putExtra(String name, ...)** that allows us to save inside the Intent our information.
	- Reading the API Doc we know that we can add string, longs, CharSequence, int and so on.
	-It behaves like a Map where there's the key and the value. In the caller **(EditActivity)** we can pass data in this way:
	
	Intent i = new Intent(EditActivity.this, ViewActivity.class);
	i.putExtra("name",edtName.getText().toString();
	i.putExtra("surname",edtSurname.getText().toString();
	i.putExtra("email", edtEmail.getText().toString();
	startActivity(i);
	Where "name", "surname" and "email" are the keys.
	
	In other word first we create the intent (line 1) defining the caller activity **(EditActivity)** and the destination activity **(ViewActivity),** then we use **putExtra** to add our data. In the destination activity to retrieve the data sent we have:
	protected void onCreate(Bundle savedInstanceState)
	{ Intent i = getIntent();
	String name = i.getStringExtra("name"));
	String surname = i.getStringExtra("surname"));
	String email = i.getStringExtra("email")); . . . }
		8	8

VII (a)	**SMS SERVICE IN ANDROID** SMS messaging is one of the main killer applications on a mobile phone. — for some users as necessary as the phone itself.		

Sending SMS Messages Programmatically

Android uses a permission-based policy where all the permissions needed by an application need to be specified in the AndroidManifest.xml file.

-By doing so, when the application is installed it will be clear to the user what specific access permissions are required by the application. For example, as sending SMS messages will potentially incur additional cost on the user's end, indicating the SMS permissions in the AndroidManifest.xml file will let the user decide whether to allow the application to install or not.

In the AndroidManifest.xml file, we can add following two permissions – SEND_SMS and RECEIVE_SMS whichever is needed.:

```
<usespermissionandroid:name="android.permission.SEND_SMS">
</uses-permission>

<usespermissionandroid:name="android.permission.RECEIVE_S
MS">
  </uses-permission>
```

In the main.xml file located in the res/layout folder, add code so that the user can enter a phone number as well as a message to send using edittexts and button.

Next, in the SMS activity, using the Button view on click, we will check to see that the phone number of the recipient and the message is entered before we send the message using the sendSMS() function

```
btnSendSMS.setOnClickListener(new       View.OnClickListener()
{
public void onClick(View v)
    {
                String phoneNo = txtPhoneNo.getText().toString();
Stringmessage=txtMessage.getText().toString();
if
(phoneNo.length()>0&&message.length()>0)
sendSMS(phoneNo, message);
                else
Toast.makeText(getBaseContext(),"Please    enter    both    phone
number                  and                  message.",
Toast.LENGTH_SHORT).show();
    }
  });
```

Where sendSMS is defined as below:

```
private void sendSMS(String phoneNumber, String message)
{
```

9 9

	SmsManager sms=SmsManager.getDefault();
	sms.sendTextMessage(phoneNumber, null, message, null, null);
	}		

(b)	Advantages of content providers □ Content providers offer granular control over the permissions for accessing data. You can choose to restrict access to a content provider from solely within your application, grant blanket permission to access data from other applications, or configure different permissions for reading and writing data. □ You can use a content provider to abstract away the details for accessing different data sources in your application. Dusadvantages of content providers □ content providers can only use SQLite as a storage mechanism. □ This may change in the future, though, because there are no hard-coded SQLite constraints. □ If using object-oriented databases, a proxy could be used between the content provider and the database interface. Or a temporary SQLite database could be a solution □ Resources are crucial on a mobile environment and you might be better off using SQLite. □ More complicated queries, with joins and distinct, are not possible with content provider	6	6
VIII (a)	1. Create an account. 2. Familiarise yourself with Developer Console 3. Fill in the necessary account details. 4. Link your merchant account 5. Upload your app 6. Alpha and beta testing of app 7. Provide details for store listing 8. Add pricing and distribution details 9. Publishing the application 10. Device Filtering option	Step 5 Ex 10	15
IX (a)	JavaScript is a programming language that you can use to affect the user interface and create dynamic websites. -To add JavaScript to a web page, you use the <script> tag. -You can place scripts in the <head> or in the <body> of your document.		

	-You also can trigger script functions in any event attribute, such as onmouseover or onload. Eg: helloworld application using javascript and html <!DOCTYPE html> <html> <head> <title>Simple JavaScript</title> <script> function hello() { alert("Hello World"); } </script> </head> <body> Click Me <noscript> <p>Hello World <p>This text is not written with JavaScript. </noscript> </body> </html> The <noscript> tag can contain any HTML that you need to provide an alternative to your script contents. Then, when people come to your page with JavaScript turned off or with a user agent that doesn't support the script, they see the contents of the <noscript> tag.	9	6
(b)	* <audio>—Embedded sound files. * <canvas>—Embedded dynamic graphics. * <embed>—To add other technologies that don't have a specific HTML5 element. * <source>—The source files for embedded sound and video. * <track>—Supplementary media tracks for embedded sound and video. * <video>—Embedded video files	6	6
X (a)	-An external style sheet is ideal when the style is applied to many pages. -With an external style sheet, you can change the look of an entire Web site by changing one file. -Each page must link to the style sheet using the <link> tag. -To create an external style sheet: 1. Open a new document. 2. Write your styles as in the earlier embedded style sheet example,	9	9

	but with- out the <style> tag surrounding them.
	3. Save that file as a style sheet with a .css extension, such as styles.css .
	-Here is a style sheet with some additional styles along with the red paragraphs:
	html, body
	{
	margin:0;
	padding:0;
	border:0;
	}
	body {
	font: 1em/1.25 Arial, Helvetica, sans-serif;
	}
	p
	{
	color: red;
	}
	Writing the style sheet document is not enough; you have to also attach it to your web page.
	- To do this, add a <link> tag to the head of your document that points to the style sheet.
	For example:
	<head>
	<title>My web page</title>
	<link href="styles.css" rel="stylesheet">
	</head>		

(b)	-An attribute is written in HTML after the tag name, separated by a space, inside the greaterthan and less-than signs. -If the attribute can have a value, that value is attached to the attribute by an equal sign. -If spaces exist in the value, then you should surround the whole value with quotation marks. For example: <elementname attributename=value> Event attributes indicate an event that might happen when the page is loaded. -The new HTML5 event attributes include: * onabort —Fires when an action is aborted. * onbeforeonload , onbeforeonunload , and onunload —Fires just before an element loads or unloads and as an element unloads. * oncontextmenu —Fires when the context menu is triggered. * ondrag , ondragend , ondragenter , ondragleave , ondragstart , and ondrop —These fire when various drag-and-drop actions occur. * onerrror and onmessage —These fire when errors or messages are triggered.	6	6

	* onscroll —This fires when the user scrolls the browser scroll bar. * onresize —Fires when an element is resized.		