



SETTING UP YOUR DEVELOPMENT ENVIRONMENT

Jestin James M
Assistant Professor, Dept of Computer Science
Little Flower College, Guruvayoor

1. Downloading JDK 6

- JAVA_HOME environment variable to point to the JDK install folder
- For Windows Vista and Windows 7, the steps to get to the Environment Variables screen are a little different.
- Choose Start ➤ Computer, right-click, choose Properties, click the link for Advanced System Settings, and click Environment Variables

1. Downloading JDK 6

- click New to add the variable or Edit to modify it if it already exists.
- The value of JAVA_HOME is something like C:\Program Files\Java\jdk1.6.0_27.

2. Downloading Eclipse 3.6

- download the Eclipse IDE for Java Developers
- You can download all versions of Eclipse from www.eclipse.org/downloads/.
- When you first start up Eclipse, it asks you for a location for the workspace.
- To make things easy, you can choose a simple location such as C:\android or a directory under your home directory.

3. Downloading the Android SDK

- You can download the Android SDK from <http://developer.android.com/sdk>



File Home Share View

Clipboard: Pin to Quick access, Copy, Paste, Copy path, Paste shortcut

Organize: Move to, Copy to, Delete, Rename

New: New folder, New item, Easy access

Open: Properties, Open, Edit, History

Select: Select all, Select none, Invert selection

< > > This PC > Acer (C:) > Android > Search: Android

Quick access	Name	Date modified	Type	Size
OneDrive	.metadata	12/29/2016 2:36 PM	File folder	
This PC	android-sdk	1/1/2017 4:02 PM	File folder	
Desktop	build-tools	12/30/2016 10:06 ...	File folder	
Documents	java	12/27/2016 2:34 PM	File folder	
Downloads	temp	12/30/2016 10:40 ...	File folder	
Music	test	12/29/2016 10:53 ...	File folder	

- Pictures
- Videos
- Acer (C:)
- DVD RW Drive (D:) I
- Microsoft Office Cl
- Network
- Homegroup

6 items 1 item selected

android-sdk

File Home Share View

Clipboard: Pin to Quick access, Copy, Paste, Copy path, Paste shortcut

Organize: Move to, Copy to, Delete, Rename

New: New folder, New item, Easy access

Open: Properties, Edit, History

Select: Select all, Select none, Invert selection

< > > This PC > Acer (C:) > Android > android-sdk > Search android-sdk

Quick access	Name	Date modified	Type	Size
OneDrive	.metadata	12/31/2016 1:03 AM	File folder	
This PC	add-ons	12/30/2016 6:11 PM	File folder	
Desktop	build-tools	1/3/2017 11:50 AM	File folder	
Documents	extras	12/31/2016 10:21 ...	File folder	
Downloads	Firstapp	12/31/2016 8:28 PM	File folder	
Music	Helloworld	1/1/2017 4:02 PM	File folder	
Pictures	platforms	12/30/2016 11:42 ...	File folder	
Videos	platform-tools	12/30/2016 11:08 ...	File folder	
Acer (C:)	SecondAPP	12/31/2016 2:01 AM	File folder	
DVD RW Drive (D:) I	sources	12/30/2016 11:46 ...	File folder	
Microsoft Office Cl	system-images	12/31/2016 1:29 PM	File folder	
Network	temp	1/3/2017 11:50 AM	File folder	
Homegroup	testing	12/31/2016 12:35 ...	File folder	
	tools	12/31/2016 12:39 ...	File folder	
	AVD Manager	10/14/2015 7:21 AM	Application	216 KB
	SDK Manager	10/14/2015 7:21 AM	Application	216 KB
	SDK Readme	10/14/2015 7:21 AM	Text Document	2 KB
	uninstall	12/30/2016 6:12 PM	Application	70 KB

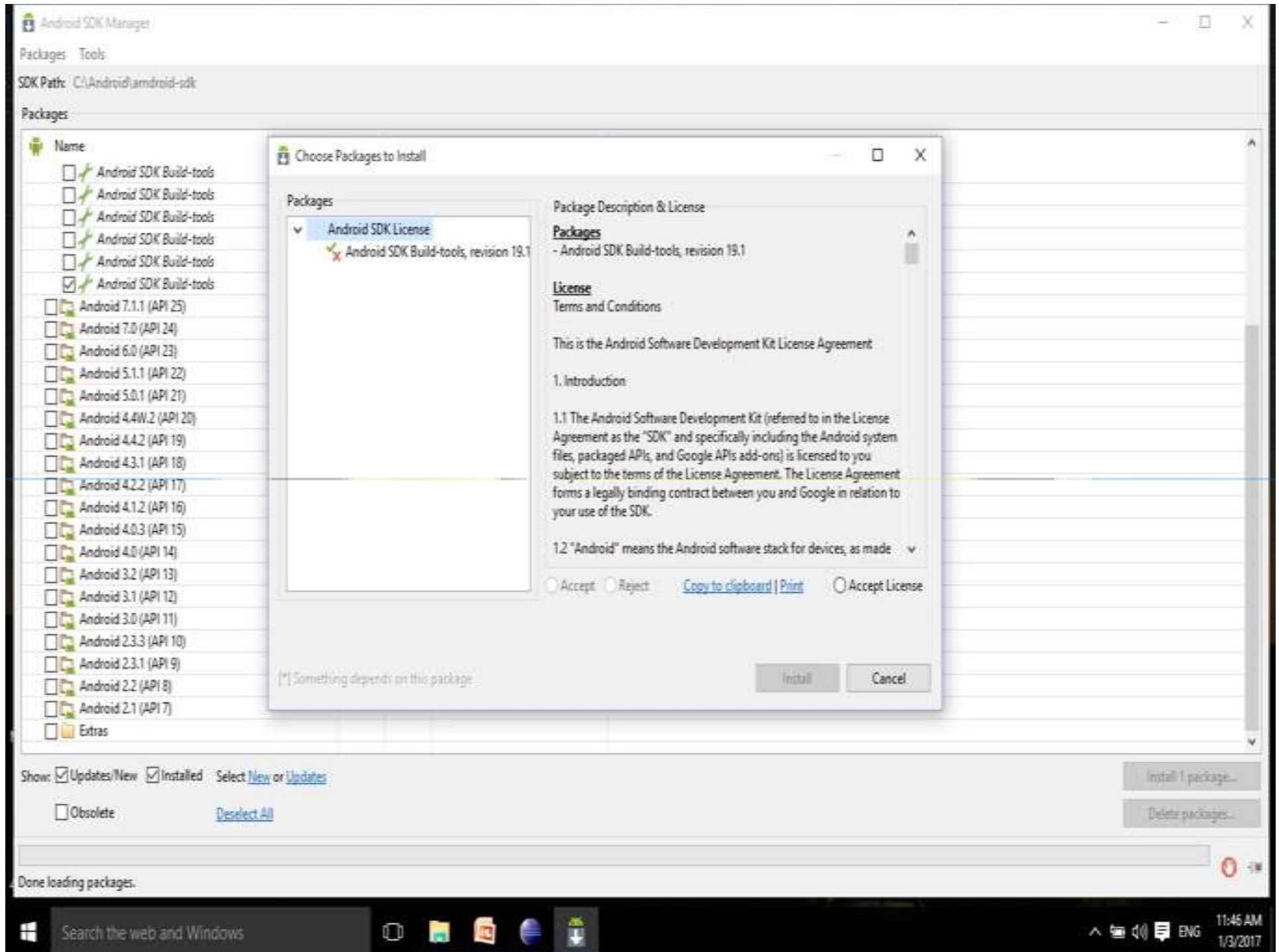
Packages

Name	API	Rev.	Status
Tools			
Android SDK Tools		25.2.4	Installed
Android SDK Platform-tools		25.0.3	Installed
Android SDK Build-tools		25.0.2	Installed
Android SDK Build-tools		25.0.1	Not installed
Android SDK Build-tools		25	Not installed
Android SDK Build-tools		24.0.3	Not installed
Android SDK Build-tools		24.0.2	Not installed
Android SDK Build-tools		24.0.1	Not installed
Android SDK Build-tools		24	Not installed
Android SDK Build-tools		23.0.3	Not installed
Android SDK Build-tools		23.0.2	Not installed
Android SDK Build-tools		23.0.1	Not installed
Android SDK Build-tools		22.0.1	Not installed
Android SDK Build-tools		21.1.2	Not installed
Android SDK Build-tools		20	Not installed
Android SDK Build-tools		19.1	Not installed
Android 7.1.1 (API 25)			
Android 7.0 (API 24)			
Android 6.0 (API 23)			
Android 5.1.1 (API 22)			
Android 5.0.1 (API 21)			
Android 4.4W.2 (API 20)			
Android 4.4.2 (API 19)			
Android 4.3.1 (API 18)			
Android 4.2.2 (API 17)			

Show: Updates/New Installed [Select New or Updates](#)
 Obsolete [Deselect All](#)

[Install packages...](#)
[Delete packages...](#)

Done loading packages.



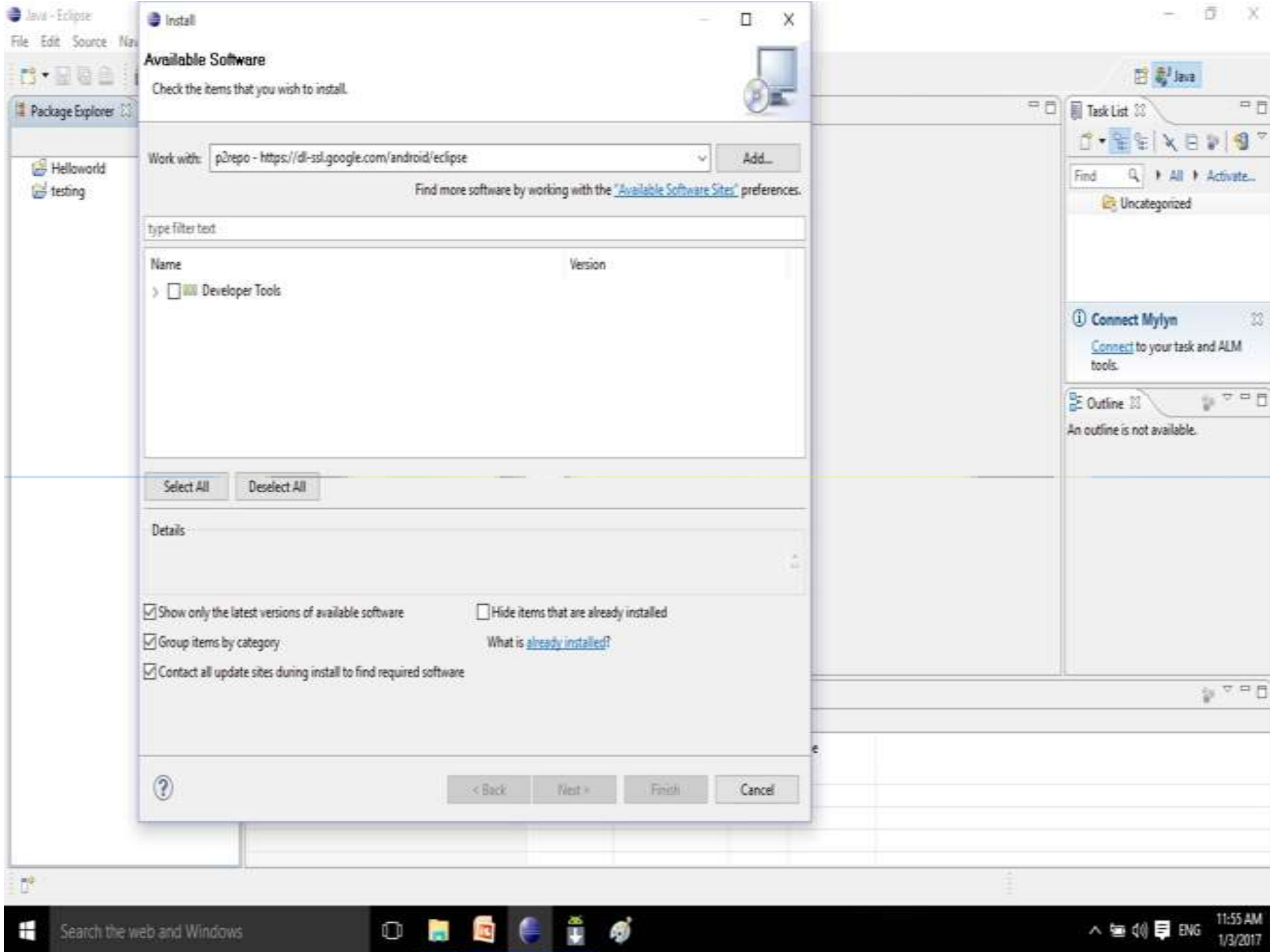
4. Updating Your PATH Environment Variable

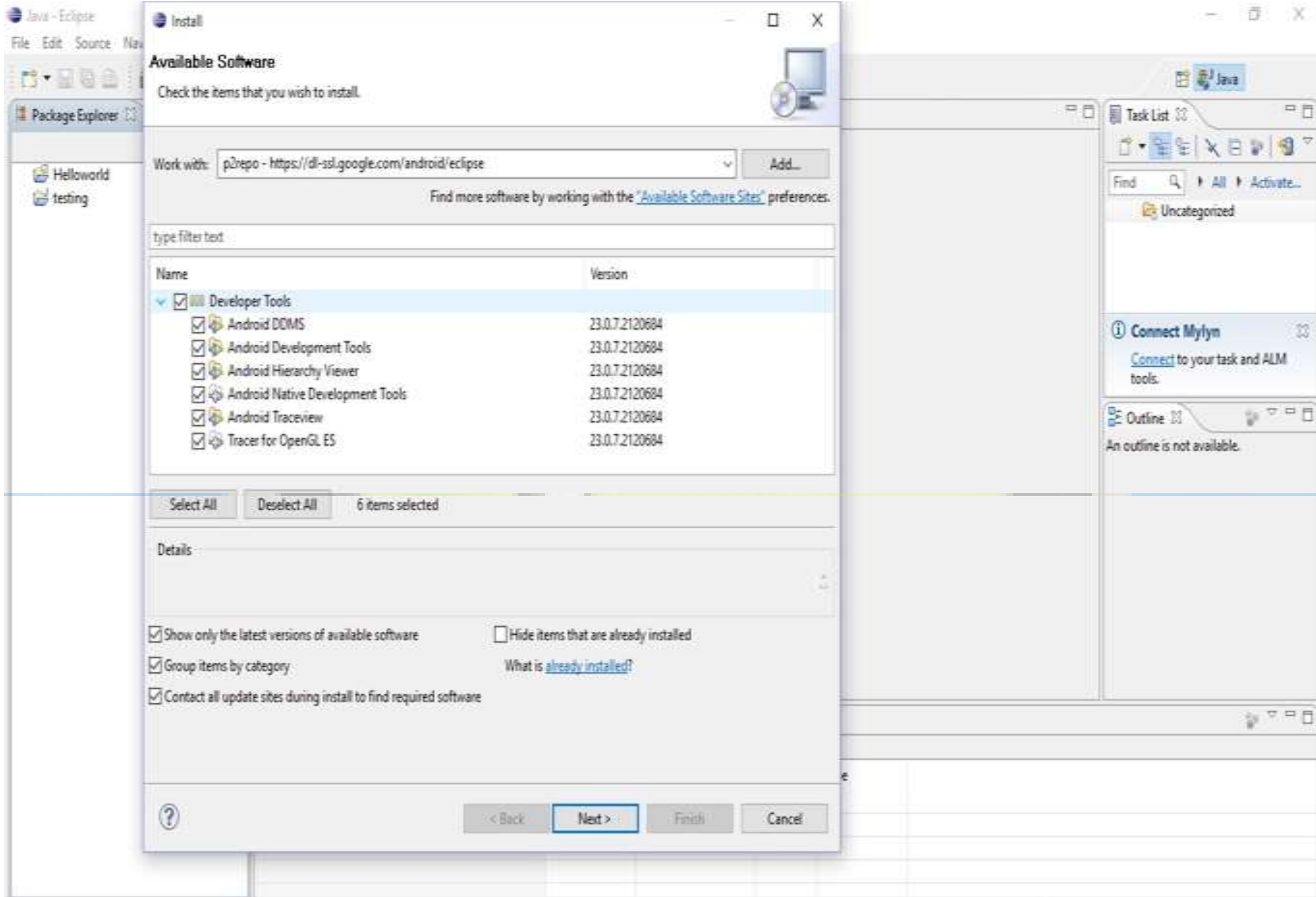
- Edit the PATH variable and add a semicolon (;) on the end,
- followed by the path to the Android SDK tools folder,
- followed by another semicolon,
- followed by the path to the Android SDK platform-tools folder,

The screenshot shows the Eclipse IDE interface with the following components:

- Package Explorer:** Located on the left, it shows a project structure with two packages: "Helloworld" and "testing".
- Task List:** Located in the top right, it displays a search bar with "Find" and "All" options, and a section for "Uncategorized" tasks. Below this is a "Connect Mylyn" section with a "Connect" link and the text "Connect to your task and ALM tools."
- Outline:** Located below the Task List, it displays the message "An outline is not available."
- Problems:** Located at the bottom, it shows a table with the following columns: "Description", "Resource", "Path", "Location", and "Type". The table currently contains 0 items.

The screenshot shows the Eclipse IDE interface. The 'Help' menu is open, displaying the following options: Welcome, Help Contents, Search, Dynamic Help, Key Assist... (with shortcut Ctrl+Shift+L), Tips and Tricks..., Report Bug or Enhancement..., Cheat Sheets..., Check for Updates, Install New Software..., Eclipse Marketplace..., and About Eclipse. The Package Explorer on the left shows a project named 'HelloWorld' with a sub-package 'testing'. The Task List on the right shows 'Uncategorized' and a 'Connect Mylyn' button. The Outline on the right shows 'An outline is not available.' The Problems, Javadoc, and Declaration tabs are visible at the bottom of the IDE.







Package Explorer

- > Helloworld
- > testing

Main editor area (currently blank)

Task List

Find [Search Icon] All Activate...

Uncategorized

Connect Mylyn
[Connect](#) to your task and ALM tools.

Outline
An outline is not available.

Problems Javadoc Declaration

0 items

Description	Resource	Path	Location	Type



Package Explorer

- helloworld
- testing

- New Window
- New Editor
- Open Perspective >
- Show View >
- Customize Perspective...
- Save Perspective As...
- Reset Perspective...
- Close Perspective
- Close All Perspectives
- Navigation >
- Android SDK Manager
- Android Virtual Device Manager
- Run Android Lint >
- Preferences**

Main editor area (currently blank)

Task List

Find All Activate...

Uncategorized

Connect Mylyn

Connect to your task and ALM tools.

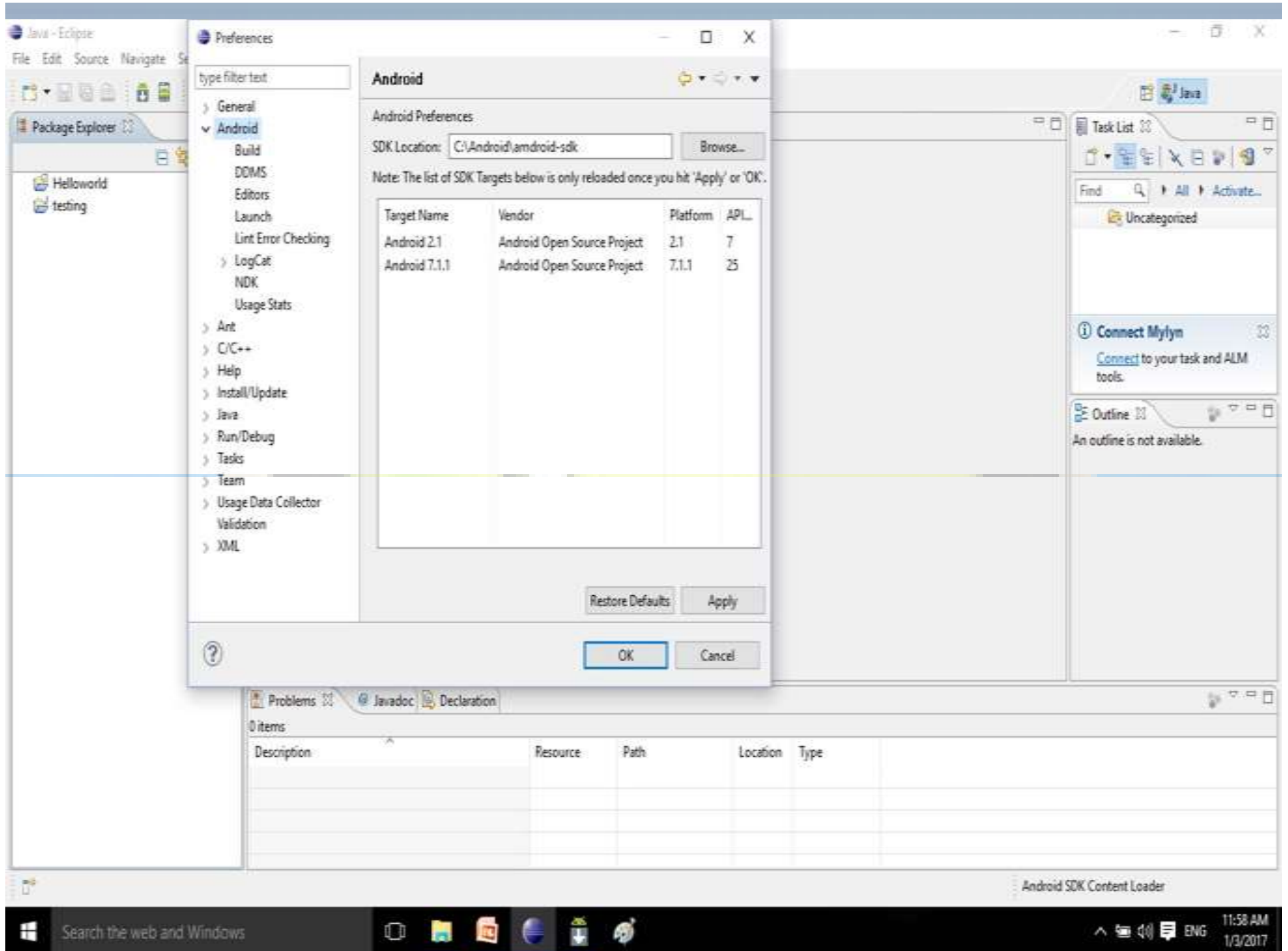
Outline

An outline is not available.

Problems Javadoc Declaration

0 items

Description	Resource	Path	Location	Type



A green Android robot is shown from the chest up, wearing a pair of brown headphones. The robot has two small antennae on its head and two circular eyes. The background is a solid light blue color. The text "LEARNING THE FUNDAMENTAL COMPONENTS" is overlaid on the lower left side of the image in a bold, black, sans-serif font.

**LEARNING THE FUNDAMENTAL
COMPONENTS**

1. View

- *Views are user interface (UI) elements that form the basic building blocks of a user interface.*
- A view can be a button, a label, a text field, or many other UI elements
- Views are also used as containers for views, which means there's usually a hierarchy of views in the UI.

2. Activity

- *An activity is a UI concept that usually represents a single screen in your application*
- something that helps the user do one thing, which could be viewing data, creating data, or editing data

3. Fragment

- When a screen is large, it becomes difficult to manage all of its functionality in a single activity.
- *Fragments are like sub-activities, and an activity can display one or more fragments on the screen at the same time.*
- When a screen is small, an activity is more likely to contain just one fragment, and that fragment can be the same one used within larger screens.

4. Intent

An intent generically defines an “intention” to do some work

- Broadcast a message.
- Start a service.
- Launch an activity.
- Display a web page or a list of contacts.
- Dial a phone number or answer a phone call

4. Intent

- Intents can be explicit or implicit.
- to display a URL, the system decides what component will fulfill the intention.
- Intents loosely couple the action and action handler.

5. Content Provider

- Data sharing
- Android provide standard mechanism for applications to share data
- Through content providers, you can expose your data and have your applications use data from other applications.

6. Service

- *Services in Android resemble services you see in Windows or other platforms*
- Android defines two types of services: local services and remote services
- Local services are components that are only accessible by the application that is hosting the service

6. Service

- remote services are services that are meant to be accessed remotely by other applications running on the device.
- An example of a service is a component that is used by an e-mail application to poll for new messages.

7. AndroidManifest.xml

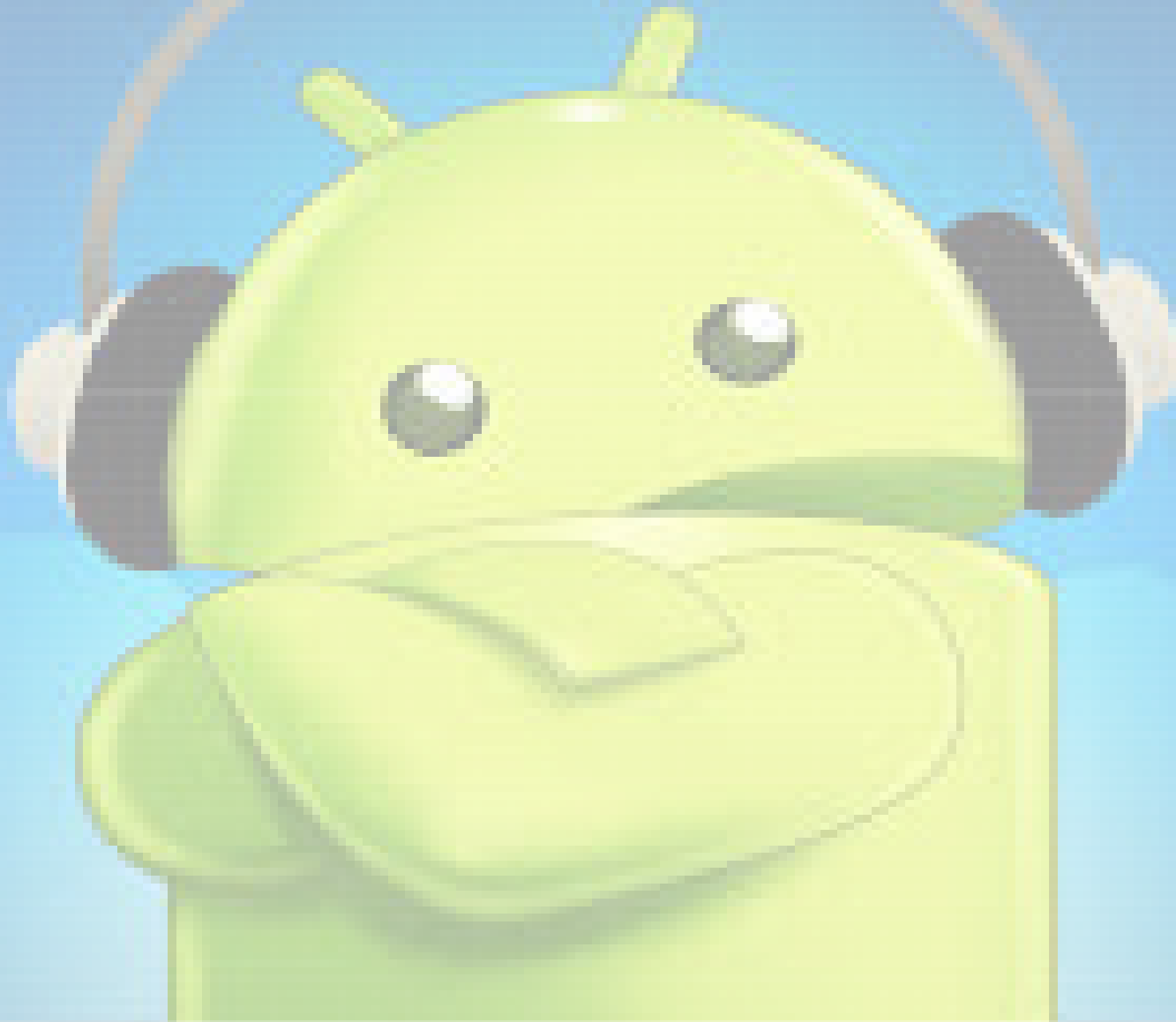
- defines the contents and behavior of your application.
- example, it lists your application's activities and services, along with the permissions and features the application needs to run.



8. Android Virtual Devices

- An Android Virtual Device (AVD) allows developers to test their applications without hooking up an actual Android device
- AVDs can be created in various configurations to emulate different types of real devices.

Hello world program



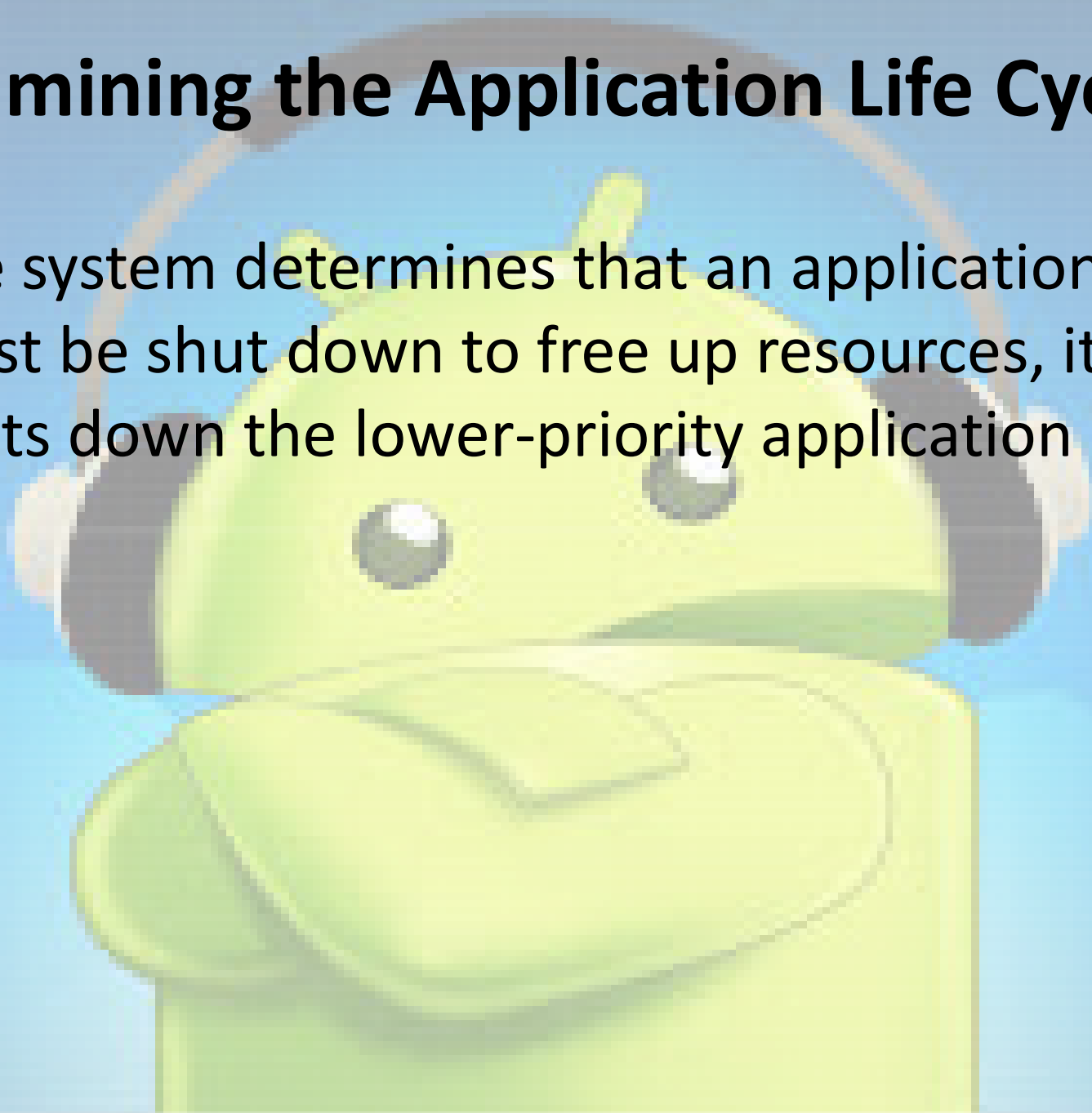
Examining the Application Life Cycle



- The life cycle of an Android application is strictly managed by the system
- A user may want to launch a web browser,
- but the system ultimately decides whether to start the application
- If the user is currently working with an activity, the system gives high priority to that application.

Examining the Application Life Cycle

- The system determines that an application must be shut down to free up resources, it shuts down the lower-priority application



J2EE



- J2EE apps are loosely managed by the container they run in.
- a J2EE container can remove an application from memory if it sits idle for a predetermined time period
- J2EE container usually has sufficient resources to run lots of applications at the same time
- With Android, resources are more limited, so Android must have more control and power over applications

Examining the Application Life Cycle

- Android runs each application in a separate process,
- each of which hosts its own virtual machine.
- This provides a protected-memory environment.
- By isolating applications to an individual process, the system can control which application deserves higher priority.
- For example, a background process that's doing a CPU-intensive task can't block an incoming phone call.

Examining the Application Life Cycle

- The concept of application life cycle is logical,
- Example : A user is talking to someone on the phone and needs to open an e-mail message to answer a question
- In the background, however, the system is saving and restoring application state.
- For instance, when the user clicks the link in the e-mail message
- the system saves metadata on the running e-mail message activity before starting the browser-application activity to launch a URL

Life-Cycle Methods of an Activity

- `protected void onCreate(Bundle savedInstanceState);`
- `protected void onStart();`
- `protected void onRestart();`
- `protected void onResume();`
- `protected void onPause();`
- `protected void onStop();`
- `protected void onDestroy();`

