

برنامه نویسی

# ANDROID

از مبتدی تا پیشرفته

ترجمه و تألیف: مهندس سید علیرضا قمصری

انتشارات پندار پارس

سرشناسه : قمصری جوینانی، سیدعلیرضا، ۱۳۶۱ -، گردآورنده، مترجم  
 عنوان و نام پدیدآور : برنامه‌نویسی ANDROID از مبتدی تا پیشرفته/ ترجمه و تالیف سیدعلیرضا قمصری.  
 مشخصات نشر : تهران : پندار پارس، ۱۳۹۲.  
 مشخصات ظاهری : ۳۸۰ ص.  
 شابک : ۱۸۸۰۰۰ ریال : 978-600-6529-33-2  
 وضعیت فهرست نویسی : فیبا  
 موضوع : اندروید (منبع الکترونیکی)  
 موضوع : نرم‌افزار کاربردی -- طراحی و توسعه  
 موضوع : تلفن‌های هوشمند -- برنامه‌نویسی  
 موضوع : سیستم‌های عامل (کامپیوتر)  
 رده بندی کنگره : QA۷۶/۷۶ ۱۳۹۲ ۸ق۸۴الف/  
 رده بندی دیویی : ۲۷۶۸/۰۰۵  
 شماره کتابشناسی ملی : ۳۱۲۶۱۸۵

#### انتشارات پندارپارس



دفتر فروش: انقلاب، ابتدای کارگر جنوبی، کوی رشتچی، شماره ۱۴، واحد ۱۶ [www.pendarepars.com](http://www.pendarepars.com)  
 تلفن: ۶۶۵۷۲۳۳۵ - تلفکس: ۶۶۹۲۶۵۷۸ همراه: ۰۹۱۲۲۴۵۲۳۴۸  
[info@pendarepars.com](mailto:info@pendarepars.com)



نام کتاب : برنامه‌نویسی ANDROID از مبتدی تا پیشرفته  
 ناشر : انتشارات پندار پارس  
 ترجمه و تالیف : سید علیرضا قمصری جوینانی  
 چاپ نخست : فروردین ۹۲  
 شمارگان : ۱۰۰۰ نسخه  
 طرح جلد : فرزانه روزبهبانی  
 لیتوگرافی، چاپ، صحافی : ترام‌سنج، فرشویه، خیام

قیمت : ۱۸۸۰۰ تومان به همراه DVD : شابک : ۹۷۸-۶۰۰-۶۵۲۹-۳۳-۲  
 \*هرگونه کپی برداری، تکثیر و چاپ کاغذی یا الکترونیکی از این کتاب بدون اجازه ناشر تخلف بوده و پیگرد قانونی دارد \*



## پیش‌گفتار

مدت‌هاست که تلفن‌های همراه به ابزاری جداناپذیر از زندگی امروزی تبدیل شده‌اند. گسترش و تنوع روزافزون و مجهزتر شدن این ابزار، خود دلیلی بر این مدعاست. توسعه‌ی تلفن‌های همراه، نه تنها صنعت بخش سخت‌افزار، بلکه نرم‌افزار آن را نیز در بر گرفته است. گستردگی مشتریان، تنوع نیازها، فرهنگ‌های متفاوت و علاقمندی‌های مشترک، همه و همه دست به دست هم داده‌اند تا تولیدکنندگان، پی در پی به دنبال نوآوری و درنوردیدن مرزهای تکنولوژی باشند.

با ورود تلفن‌های هوشمند به بازار و مجهز شدن گوشی‌های همراه به سیستم‌عامل، عملاً صنعت تلفن همراه وارد عرصه‌ی جدیدی شد که به واسطه‌ی آن، محدودیت برنامه‌نویسان برای ورود به این صنعت، از میان رفت. تا پیش از این، انتخاب برنامه برای گوشی‌های موبایل منحصر در اختیار شرکت تولیدکننده‌ی آن بود، ولی اینک برنامه‌نویسان نیز می‌توانستند برای گوشی‌های موبایل برنامه‌نویسی کنند.

تجارت جدیدی شکل گرفت و حجم انبوهی گردش مالی ایجاد شد. با این وجود به دلیل تخصصی بودن این سبک برنامه‌نویسی، همه قادر به تولید و ارائه‌ی ایده‌های خود در قالب یک برنامه برای گوشی‌های هوشمند نبودند و این، مانع گسترش این صنعت می‌شد. شرکت اپل با توجه به این حقیقت، با ارائه‌ی آیفون توانست سهم به‌سزایی از بازار را به خود اختصاص دهد. البته این پایان کار نبود و تلاش‌های فراوانی در جهت رفع این نقطه ضعف انجام گرفت و در نهایت، شرکت گوگل با ارائه‌ی آندروید، راه حلی برای آن ارائه داد. این سیستم‌عامل به سرعت توسط تولیدکنندگان مورد پذیرش قرار گرفت. سادگی برنامه‌نویسی و همچنین گستره‌ی وسیعی از ابزارهای هدف که توسط این سیستم‌عامل پشتیبانی می‌شوند، امتیاز بزرگی برای آن محسوب می‌شود. پس از خواندن این کتاب، شما نیز این موضوع را تأیید خواهید کرد که اکنون برنامه‌نویسان نیمه حرفه‌ای و حتی مبتدی نیز قادرند به برنامه‌نویسی موبایل بپردازند.

هرچند تلاش من بر این بوده که ترجمه روان و قابل فهم باشد، ولی آن را خالی از نقص نمی‌بینم که بابت آن پیشاپیش پوزش می‌خواهم و مشتاقانه منتظر دریافت نظرات و پیشنهادات شما خواننده‌ی گرامی هستم.

در پایان با شکر خداوند که به سبب توفیق ترجمه‌ی این کتاب، بر من منت نهاد، از همه‌ی دوستانی که مرا در این امر یاری رساندند سپاسگزارم.

قمصری / بهار 92

programmer@qamsari.com

تقديم به بی نهایت عشق....، به همسر م

## فهرست

1	فصل 1. شروع برنامه‌نویسی با آندروید.....
1	آندروید چیست؟ .....
2	نگارش‌های آندروید .....
3	ویژگی‌های آندروید .....
3	معماری آندروید .....
5	ابزارهای آندروید در فروشگاه‌ها .....
6	آندروید مارکت .....
6	تهیه‌ی ابزارهای مورد نیاز .....
7	JAVA JDK .....
7	Eclipse .....
8	SDK آندروید .....
8	Android Development Tools (ADT) .....
11	تولید شبیه‌ساز مجازی آندروید (AVD) .....
14	تولید نخستین برنامه‌ی آندرویدی .....
14	تمرین: تولید نخستین برنامه‌ی آندرویدی .....
18	مناسب‌ترین AVD برای تست برنامه .....
19	برنامه چگونه کار می‌کند .....
20	تشریح ساختار یک برنامه‌ی آندرویدی .....
24	خلاصه .....
25	فصل 2. ACTIVITY ها و INTENT ها.....
25	درک Activity ها .....
28	تمرین: درک سیکل زندگی یک Activity .....
31	اعمال Style و تم به Activity .....
31	مخفی نمودن عنوان Activity .....
32	نمایش یک دیالوگ .....
33	تمرین: نمایش یک پنجره‌ی دیالوگ، با استفاده از Activity .....
36	آبجکت Context .....
36	نمایش دیالوگ Progress .....
37	تمرین: نمایش یک دیالوگ Progress، با استفاده از Activity .....
40	اتصال Activity ها با استفاده از Intent ها .....
40	تمرین اتصال Activity با استفاده از Intent ها .....
44	رفع مشکل تصادم فیلتر Intent .....
46	بازگرداندن مقادیر از یک Intent .....
46	تمرین: به‌دست آوردن نتیجه از یک Activity .....
50	ارسال اطلاعات به‌وسیله‌ی شیء Intent .....
51	تمرین: ارسال اطلاعات به Activity هدف .....
53	فراخوانی برنامه‌های توکار، با استفاده از INTENT ها .....
53	تمرین: فراخوانی برنامه‌های توکار، با استفاده از Intent ها .....
60	درک شیء Intent .....
62	استفاده از فیلترهای Intent .....
62	تمرین: مشخص کردن فیلتر Intent با جزئیات بیشتر .....

67	..... اضافه کردن دسته‌بندی‌ها
69	..... نمایش پیام‌ها
70	..... تمرین: نمایش پیام در نوار وضعیت
75	<b>فصل 3. درک واسط کاربری آندروید</b>
76	..... View و ViewGroup ها
77	..... LinearLayout
79	..... واحد اندازه‌گیری
81	..... Absolutelayout
82	..... Tablelayout
83	..... Relativelayout
84	..... Framelayout
87	..... Scrollview
89	..... تطابق با جهت نمایش
90	..... Anchoring یا اتصال viewها
92	..... تغییر اندازه و تغییر موقعیت
95	..... مدیریت تغییرات مربوط به جهت صفحه
95	..... تمرین: درک رفتار Activity در زمان تغییر جهت صفحه
98	..... نگهداری اطلاعات وضعیت در هنگام تغییر در تنظیمات
100	..... شناسایی تغییر جهت
101	..... کنترل جهت Activity
102	..... تولید واسط کاربری با استفاده از کد
102	..... تمرین: تولید UI با استفاده از کد
104	..... گوش دادن به پیام‌های UI
105	..... بازنویسی متدهای تعریف شده در یک Activity
105	..... تمرین: بازنویسی متدهای Activity
110	..... ثبت رویدادها برای viewها
113	<b>فصل 4. طراحی واسط کاربری با استفاده از VIEWها</b>
113	..... viewهای پایه
114	..... TextView
114	..... viewهای RadioButton, ToggleButton, CheckBox, EditText, ImageButton, Button
115	..... تمرین: استفاده از viewهای پایه
120	..... تمرین: مدیریت رویدادهای View
123	..... progressBar view
123	..... تمرین: استفاده از ProgressBar
126	..... تمرین: سفارشی کردن ProgressBar
128	..... AutoCompleteTextView
128	..... تمرین: استفاده از AutoCompleteTextView
131	..... Picker هایView
131	..... TimePicker
131	..... تمرین: استفاده از TimePicker
133	..... نمایش TimePicker در قالب یک دیالوگ
133	..... تمرین استفاده از دیالوگ، برای نمایش TimePicker
135	..... DatePicker
135	..... تمرین: استفاده از DatePicker

138	نمایش DatePicker در قالب دیالوگ
138	تمرین: استفاده از دیالوگ، برای نمایش DatePicker
141	Viewهای لیست
141	ListView
141	تمرین: نمایش لیستی طولانی از آیتمها، با استفاده از ListView
143	سفارشی کردن ListView
143	تمرین: سفارشی سازی ListView
145	تمرین: ذخیره‌ی آیتمها در فایل strings.xml
146	تمرین: استفاده از View Spinner برای نمایش یک آیتم در هر لحظه
149	<b>فصل 5. نمایش تصاویر و منوها با VIEWها</b>
149	استفاده از Viewهای تصاویر برای نمایش عکس
149	Viewهای Gallery و ImageView
150	تمرین: استفاده از Gallery
156	ImageSwitcher
156	تمرین: استفاده از ImageSwitcher
161	GridView
161	تمرین: استفاده از GridView
164	استفاده از منو به همراه Viewها
165	تولید متدهای کمک
165	تمرین: تولید متدهای کمکی منو
167	منوی Options
167	تمرین: نمایش منوی Options
169	منوی محتوا
169	تمرین: نمایش یک منوی زمینه
171	برخی Viewهای دیگر
171	viewهای AnalogClock و DigitalClock
172	WebView
177	<b>فصل 6. نگهداری اطلاعات</b>
177	ذخیره‌سازی و بارگذاری User Preferences
178	استفاده از getSharedPreferences()
178	تمرین: ذخیره‌ی اطلاعات به‌وسیله‌ی شیء SharedPreferences
182	استفاده از getPreferences()
183	نگهداری اطلاعات در فایل
183	ذخیره در انباره‌ی داخلی
183	تمرین: ذخیره‌ی داده‌ها در انباره‌ی داخلی
188	ذخیره در انباره‌ی خارجی
190	انتخاب بهترین انباره برای ذخیره‌سازی
191	استفاده از منابع ایستا
192	تولید و استفاده از پایگاه داده
192	تولید کلاس کمکی DBAdapter
193	تمرین: تولید کلاس کمکی پایگاه داده
198	استفاده از پایگاه داده در برنامه
198	افزودن تماس‌ها
198	تمرین: افزودن تماس‌ها به جدول

199	بازگرداندن همه‌ی تماس‌ها
199	تمرین: بازگرداندن همه‌ی تماس‌ها از جدول
201	بازیابی یک تماس
201	تمرین: بازیابی یک تماس از جدول
202	ویرایش یک تماس
202	تمرین: به‌روزرسانی یک تماس در جدول
203	حذف یک تماس
203	تمرین: حذف یک تماس از جدول
204	به‌روزرسانی پایگاه داده
204	پایگاه داده‌های از پیش ساخته
205	اتصال پایگاه داده به یک برنامه
206	تمرین: اتصال یک پایگاه داده
209	<b>فصل 7. ارائه‌دهنده‌های محتوا</b>
209	اشتراک اطلاعات در آندروید
211	استفاده از ارائه‌دهنده‌ی محتوا
211	تمرین: استفاده از ارائه‌دهنده‌ی محتوای Contacts
214	رشته‌های پرس و جوی از پیش تعریف شده
218	Projection
218	فیلتر کردن
218	مرتب‌سازی
219	ایجاد یک ارائه‌دهنده‌ی محتوای
219	تمرین: تولید یک ارائه‌دهنده‌ی محتوای
228	استفاده از ارائه‌دهنده‌ی محتوا
228	تمرین: استفاده از ارائه‌دهنده‌ی محتوای تولید شده
233	<b>فصل 8. ارسال پیام و شبکه</b>
233	ارسال SMS
234	ارسال SMS به‌وسیله‌ی کد
234	تمرین: ارسال یک پیام SMS
237	دریافت بازخورد پس از ارسال پیام
239	ارسال پیام SMS با استفاده از Intent
240	دریافت پیام‌های SMS
240	تمرین: دریافت پیام‌های SMS
242	به‌روزنمایی یک Activity به‌وسیله‌ی یک BroadcastReceiver
242	تمرین: تولید یک برنامه‌ی مبتنی بر View
247	فراخوانی یک Activity به‌وسیله‌ی یک BroadcastReceiver
247	تمرین فراخوانی یک Activity
250	هشدارها و پیام‌ها
250	ارسال E-mail
251	تمرین: ارسال ایمیل به‌وسیله‌ی کد
252	شبکه
253	تمرین: تولید پروژه
255	دانلود اطلاعات باینری
255	تمرین: تولید پروژه
258	دانلود فایل‌های متنی



258	تمرین: دانلود فایل‌های متنی
259	برنامه چگونه کار می‌کند
259	دسترسی به وب سرویس‌ها
260	تمرین: استفاده از وب سرویس‌ها
265	انجام فراخوانی به صورت غیر هم‌زمان
267	<b>فصل 9. سرویس‌های مبتنی بر موقعیت</b>
267	نمایش Maps
268	تولید پروژه
268	تمرین: تولید پروژه
269	به دست آوردن همه‌ی API‌های مربوط به Maps
271	نمایش نقشه
272	تمرین: نمایش Google Maps
273	نمی‌توانید نقشه را ببینید؟
274	نمایش کنترل بزرگ‌نمایی
274	تمرین: نمایش کنترل‌های توکار بزرگ‌نمایی
275	تمرین: بزرگ‌نمایی و کوچک‌نمایی نقشه با استفاده از کد
276	تغییر نماها
278	هدایت به یک مکان مشخص
278	تمرین: هدایت نقشه برای نمایش یک موقعیت مشخص
280	افزودن نشان‌ها
280	تمرین: افزودن نشان‌ها به نقشه
283	دریافت موقعیت با لمس کردن نقشه
284	ژئوکدینگ و ژئوکدینگ معکوس
286	دریافت اطلاعات یک موقعیت
287	تمرین انتقال نقشه به یک مکان مشخص با استفاده از کلاس مدیریت موقعیت
291	مانیتور کردن موقعیت
293	<b>فصل 10. تولید سرویس‌های آندروید</b>
293	تمرین: تولید یک سرویس ساده
298	انجام کارهای زمان‌بر در داخل یک سرویس
298	تمرین: کاربردی کردن سرویس ایجاد شده
299	تمرین: انجام کارها در یک سرویس، به صورت غیر هم‌زمان
303	انجام کارهای تکرار در یک سرویس
303	تمرین: اجرای کارهای تکراری، با استفاده از کلاس Timer
305	اجرای کارها به صورت غیر هم‌زمان در Threadهای مجزا، با استفاده از IntentService
306	تمرین: استفاده از کلاس IntentService، برای یک سرویس با توقف خودکار
308	تعامل میان یک سرور و یک Activity
309	تمرین: فراخوانی یک Activity از یک سرویس
312	اتصال Activity به سرویس
314	تمرین: دسترسی مستقیم به یک خصوصیت به وسیله‌ی اتصال
321	<b>فصل 11. انتشار برنامه‌های آندرویدی</b>
321	آماده‌سازی برای انتشار
322	نگارش‌های برنامه
324	امضای دیجیتال برای برنامه‌های آندرویدی‌تان
325	تمرین: Export و امضا کردن یک برنامه‌ی آندرویدی

329	انتشار فایل‌های APK
330	استفاده از ابزار adb.exe
332	استفاده از یک سرور وبی
335	انتشار آندروید مارکت
335	تولید یک پروفایل Developer
336	ارسال برنامه‌تان
341	<b>پیوست الف. استفاده از ECLIPSE برای برنامه‌نویسی آندروید</b>
341	گردش در Eclipse
341	Workspaceها
343	Package Explorer
343	استفاده از پروژه‌های موجود در دیگر Workspaceها
345	ویرایش گرها
347	نماها
347	اضافه کردن اتوماتیک فضای نام‌ها
348	کامل کننده‌ی کد
349	Refactoring
350	خطایابی
350	تنظیم Breakpointها
352	خطاها
353	<b>پیوست ب. استفاده از شبیه‌ساز آندروید</b>
353	استفاده از شبیه‌ساز آندروید
354	نصب Avdهای سفارشی
357	شبیه‌سازی ابزارهای واقعی
358	شبیه‌سازی کارت SD
360	شبیه‌سازی توانایی‌های فیزیکی
361	ارسال پیام SMS برای شبیه‌ساز
361	کلیدهای میان‌بر
363	برقراری تماس تلفنی
364	انتقال فایل‌ها به و یا از شبیه‌ساز
366	بازگرداندن شبیه‌ساز به حالت ابتدایی

# فصل 1

## شروع برنامه‌نویسی با اندروید

در این فصل خواهید آموخت:

- اندروید چیست،
- نسخه‌های مختلف اندروید و ویژگی‌های هر کدام،
- معماری اندروید،
- دستگاه‌های اندرویدی موجود در بازار،
- چگونگی به‌دست آوردن ابزار و نرم‌افزارهای مورد نیاز برای برنامه‌نویسی اندروید،
- نوشتن نخستین برنامه‌ی اندرویدی.

خوش آمدید. این واقعیت که این کتاب در دستان شماست، نشان می‌دهد که علاقه‌مندید تا با چگونگی برنامه‌نویسی برای سیستم‌عامل اندروید آشنا شوید. بازار برنامه‌های تلفن همراه، دچار یک تحرک جدید شده است و تحقیق‌های جدید نشان می‌دهد که اندروید با سبقت از iPhone، دومین جایگاه را در بازار تلفن‌های هوشمند آمریکا به‌خود اختصاص داده است. جایگاه نخست به Research In Motion (RIM) و جایگاه سوم در اختیار iPhone است. هنگامی که این کتاب را می‌خوانید این احتمال وجود دارد که اندروید، جایگاه نخست را در بازار تلفن‌های همراه هوشمند آمریکا به‌دست آورده باشد.

چه چیزی این سیستم‌عامل گمنام را که به‌وسیله‌ی گوگل در سال 2005 خریداری شد، به شهرت و محبوبیت امروزی رسانده است و چه ویژگی‌هایی به‌وسیله‌ی آن ارائه می‌شود؟ در این فصل خواهید آموخت که اندروید چیست و به چه دلیل توانسته برنامه‌نویسان و تولیدکنندگان را، به استفاده از خود قانع کند. همچنین نخستین برنامه‌ی اندرویدی خود را تولید خواهید کرد. در پایان این فصل، به دانش پایه‌ای که برای یادگیری تکنیک‌های پیچیده و فنی که برای تولید برنامه‌های اندروید مورد نیاز است، دست خواهید یافت.

## اندروید چیست؟

اندروید، یک سیستم عامل تلفن همراه می‌باشد که بر اساس یک نسخه‌ی ویرایش شده‌ی Linux تولید شده است. این سیستم‌عامل در سال 2005 در شرکتی با همین نام تولید شد. گوگل به عنوان بخشی از استراتژی ورود به بازار موبایل، آن را خریداری و اقدام به گسترش این سیستم عامل نمود.

گوگل، قصد این را داشت که آندروید رایگان و متن باز باشد. از این رو بخش زیادی از کدهای آندروید، تحت لیسانس متن باز مربوط به Apache ارائه گردید. این کار به این مفهوم بود که هر کسی که می‌خواست با آندروید کار کند، می‌توانست سورس کد آن را دریافت نماید. همچنین فروشندگان (که بیشتر، تولیدکنندگان سخت‌افزار بودند) می‌توانستند افزودنی‌های مورد نیاز خود را به آن اضافه و با سفارشی کردن آن، محصول خود را از دیگر نسخه‌های موجود، متمایز کنند. این مدل ساده، سبب جذابیت آندروید و تحریک تولیدکنندگان به استفاده از آن شد. شرکت‌هایی که از محصول آیفون اپل، که یک محصول بسیار موفق بود و انقلابی در صنعت گوشی‌های هوشمند ایجاد کرده بود، آسیب دیده بودند از آندروید استقبال کردند.

حتی کمپانی‌هایی همانند موتورولا و سونی‌اریکسون، که برای سال‌ها از سیستم‌عامل مخصوص به خود استفاده می‌کردند نیز در این فهرست وجود داشتند. هنگامی که آیفون ارائه شد، بسیاری از این شرکت‌ها به فکر پیدا کردن راهی جدید برای تجدید قوا افتادند و متوجه شدند که آندروید یک راه‌حل مناسب است. آنها به تولید سخت‌افزار خود ادامه دادند و از آندروید به‌عنوان سیستم‌عاملی برای قدرت بخشیدن به آن استفاده کردند.

دستاورد اصلی قبول کردن آندروید، ارائه‌ی یک رویکرد یکسان برای تولید برنامه‌ها بود. برنامه‌نویسان، باید تنها برای آندروید برنامه‌نویسی می‌کردند و برنامه‌ی آنها قابلیت اجرا بر روی تعداد زیادی از ابزارهای موجود را، که از سیستم‌عامل آندروید استفاده می‌نمود، پیدا می‌کرد. در دنیای گوشی‌های هوشمند، برنامه‌های کاربردی، مهم‌ترین حلقه‌ی زنجیر موفقیت است. تولیدکنندگان ابزارها متوجه شدند که مهم‌ترین راه برای برون رفت از چالش ایجاد شده به‌وسیله‌ی آیفون، استفاده از آندروید است که هم‌اکنون پایگاهی بزرگ برای تولید برنامه‌های کاربردی است.

## نگارش‌های آندروید

از زمان ارائه‌ی نخستین نسخه از آندروید، نگارش‌های متفاوتی از آن ارائه شده است. جدول 1-1، نسخه‌های متفاوت موجود آندروید و همچنین نام کد آنها را نشان می‌دهد.

جدول 1-1 تاریخچه‌ی مختصر از نگارش‌های مختلف آندروید

نام کد	تاریخ انتشار	نگارش
	9 February 2009	1.1
Cupcake	30 April 2009	1.5
Donut	15 September 2009	1.6
Eclair	26 October 2009	2.0/2.1
Froyo	20 May 2010	2.2

نام کد	تاریخ انتشار	نگارش
Gingerbread	6 December 2010	2.3
Honeycomb	22 February 2011	3.0
Ice Cream Sandwich	19 October 2011	4.0.1
Jelly Bean	9 July 2012	4.1.1

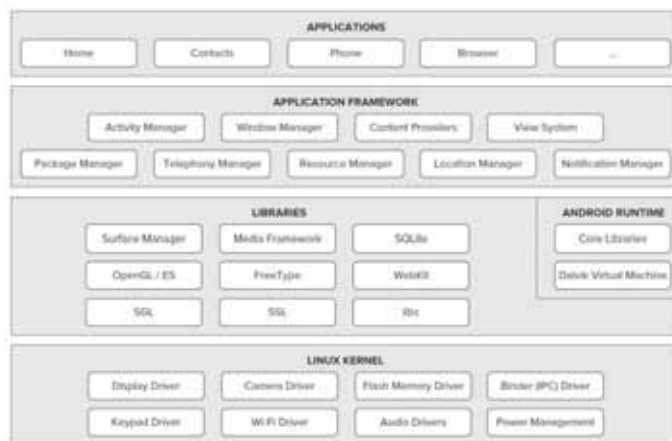
## ویژگی‌های اندروید

از آنجا که اندروید متن باز است و برای سفارشی کردن در اختیار تولیدکنندگان قرار دارد، یک ترکیب ثابت و مشخص برای سخت‌افزار و نرم‌افزار در مورد آن وجود ندارد، اما روی هم، خود اندروید از این ویژگی‌ها پشتیبانی می‌کند:

- **ذخیره‌سازی:** استفاده از SQLite، یک پایگاه داده‌ی رابطه‌ای سبک، برای ذخیره‌ی اطلاعات. در فصل 6 با جزئیات بیشتری درباره‌ی ذخیره‌ی اطلاعات بحث خواهیم کرد.
- **اتصال:** از GSM/EDGE، UMTS، EV-DO، CDMA، IDEN، بلوتوث (شامل A2DP و AVRCP)، WiFi، LTE و WiMAX پشتیبانی می‌کند. در فصل 8 در مورد شبکه، با جزئیات بیشتری آشنا خواهیم شد.
- **پیام:** هم SMS و هم MMS را پشتیبانی می‌کند. در فصل 8، بیشتر درباره‌ی پیام صحبت خواهیم کرد.
- **مرورگر وب:** بر مبنای WebKit متن باز و به همراه کروم با موتور جاوا اسکریپت V8.
- **پشتیبانی از مدیا:** از این فرمت‌ها پشتیبانی می‌کند: H.263، H.264 (در قالب MP4 و یا 4GP)، MPEG-4، AMR، AMR-WB (در قالب 3GP)، AAC، HE-AAC (در قالب MP4 یا 3GP)، MP3، MIDI، Ogg Vorbis، WAV، JPEG، PNG، GIF و BMP.
- **پشتیبانی سخت‌افزاری:** سنسورهای شتاب سنج، دوربین، قطب‌نمای دیجیتال، سنسور تقریب و GPS.
- **چند لمسی:** از صفحه‌های Multi-touch پشتیبانی می‌کند.
- **چند وظیفگی:** از برنامه‌های Multi-tasking پشتیبانی می‌کند.
- **پشتیبانی از Flash:** اندروید 2.3، از فلش 10.1 پشتیبانی می‌کند.
- **Tethering:** پشتیبانی از اتصالات اینترنت، به صورت بدون سیم و باسیم.

## معماری اندروید

برای آشنایی بیشتر با شیوه‌ی کار اندروید، به شکل 1-1 نگاهی بیندازید. این شکل، لایه‌های مختلف تشکیل دهنده‌ی سیستم‌عامل اندروید را نشان می‌دهد.



شکل 1-1

سیستم عامل آندروید، تقریباً به پنج بخش در چهار لایه تقسیم می‌شود.

**هسته‌ی لینوکس:** هسته‌ای است که بخش اصلی آندروید را تشکیل می‌دهد. این لایه، همه‌ی درایورهای سطح پایین مربوط به ابزارها و سخت‌افزارهای مختلف آندروید را دربرمی‌گیرد.

**کتابخانه‌ها:** این بخش، دربرگیرنده‌ی همه‌ی کدهایی است که ویژگی‌های اصلی سیستم عامل آندروید را ایجاد می‌کند. برای نمونه، کتابخانه‌ی SQLite پشتیبانی از پایگاه داده را آماده می‌کند که به‌وسیله‌ی یک برنامه، می‌تواند از آن برای ذخیره‌ی اطلاعات استفاده کند.

**آندروید Runtime:** همانند لایه‌ی کتابخانه، آندروید Runtime مجموعه‌ای از کتابخانه‌های پایه‌ای را ارائه می‌دهد که برنامه‌نویسان را قادر می‌سازد با استفاده از زبان برنامه‌نویسی Java، برای آندروید برنامه‌نویسی کنند. آندروید Runtime همچنین دارای ماشین مجازی با نام Dalvik می‌باشد که این امکان را می‌دهد تا هر برنامه‌ی آندروید، در فرایند مستقل و در داخل نسخه‌ی ماشین مجازی Dalvik مربوط به خود اجرا گردد. (در واقع برنامه‌ی آندرویدی باید در فرمت Dalvik قابل اجرا، کامپایل شود). Dalvik، یک ماشین مجازی ویژه است که مخصوص آندروید طراحی و برای ابزارهایی با منبع تغذیه‌ی باتری و حافظه و CPU محدود، بهینه سازی شده است.

**فریم ورک:** قابلیت‌های سیستم عامل آندروید را در اختیار برنامه‌نویسان قرار می‌دهد و به این ترتیب آنها قادر خواهند بود از این توانایی‌ها، در برنامه‌های خود استفاده کنند.

**برنامه‌های کاربردی:** در این لایه‌ی سطح بالا، برنامه‌هایی وجود دارند که همراه با ابزارهای آندرویدی فروخته می‌شوند. (همانند تلفن، دفترچه تلفن و مرورگر و غیره) همچنین شامل برنامه‌هایی می‌شود که از آندروید مارکت، دانلود و نصب می‌کنید. برنامه‌هایی که می‌نویسید نیز در این لایه قرار می‌گیرد.

## ابزارهای اندروید در فروشگاهها

ابزارهای اندرویدی، به شکل‌ها و اندازه‌های متفاوتی وجود دارند. با توجه به آمار مربوط به تحقیقاتی که در اواخر نوامبر سال 2010 انجام شده است، سیستم‌عامل اندروید در ابزارهای زیر مورد استفاده قرار گرفته است.



- گوشی‌های هوشمند
- Tabletها
- ابزارهای E-reader
- Netbookها
- پخش کننده‌های MP4
- تلویزیون‌های اینترنتی

این احتمال وجود دارد که شما دست کم یکی از این ابزارها را داشته باشید. شکل 1-2 به ترتیب (در جهت عقربه‌های ساعت) Galaxy S سامسونگ، HTC Desire HD، LG Optimus One که همگی گوشی‌های هوشمند هستند را نشان می‌دهد.

شکل 1-2

دسته‌ی محبوب دیگری که به وسیله‌ی تولیدکنندگان در حال مجهز شدن به اندروید است، تبلت می‌باشد. سائز تبلت‌ها معمولا از هفت اینچ آغاز می‌شود. شکل 1-3، Galaxy Tab سامسونگ و Streak دل را در یک tablet پنج اینچی، نشان می‌دهد.

در کنار گوشی‌های هوشمند و تبلت‌ها، اندروید در حال ورود به ابزارهای اختصاصی مانند e-book readers می‌باشد. شکل 1-4، NOOKcolor مربوط به Noble را نشان می‌دهد که یک e-book reader با سیستم‌عامل اندروید است.



شکل 1-3



شکل 1-4

افزون بر این ابزارها، آندروید به آرامی در حال راه یافتن به اتاق نشیمن شماست. People of Lava محصول یک شرکت سوئدی، یک تلویزیون مبتنی بر آندروید است که با نام Android TV Scandinavia شناخته می‌شود. (شکل 1-5)

گوگل همچنین برای تولید یک تلویزیون هوشمند اختصاصی مبتنی بر آندروید، در حال همکاری با شرکت‌هایی مانند Intel، Sony و Logitech است. شکل 1-6، Google TV سونی را نشان می‌دهد.



شکل 1-5



شکل 1-6

## آندروید مارکت

همان‌گونه که پیش از این اشاره شد، یکی از فاکتورهای نشان دهنده موفقیت یک سیستم‌عامل تلفن هوشمند، برنامه‌هایی است که آن را پشتیبانی می‌کند. این امری روشن است که موفقیت آیفون، به‌خاطر نقش حیاتی است که برنامه‌های فراوان مربوط به آن بازی می‌کنند. همچنین دسترسی همگانی به تولید این نوع برنامه‌ها بسیار مهم است.

به‌همین دلیل گوگل در آگوست 2008، آندروید مارکت را ارائه داد که یک فروشگاه آنلاین نرم‌افزارهای مربوط به ابزارهای آندرویدی است. با استفاده از نرم‌افزار Market که بر روی ابزار کاربران نصب شده است، کاربران می‌توانند ابزارهای تولیدی به‌وسیله شرکت‌های مختلف را به‌صورت مستقیم بر روی ابزار خود نصب کنند. نرم‌افزارهای رایگان نیز در کنار برنامه‌های تجاری، به‌وسیله آندروید مارکت پشتیبانی می‌شوند. البته نرم‌افزارهای پولی، به‌دلیل برخی مسائل حقوقی، تنها در برخی از کشورها در دسترس قرار دارند.


## تهیه ابزارهای مورد نیاز

اکنون که می‌دانید آندروید چیست و چه ویژگی‌هایی دارد، شاید بخواهید مشغول شوید و شروع به برنامه‌نویسی کنید. پیش از نوشتن نخستین برنامه‌ی خود، باید ابزار و SDKهای مورد نیاز خود را دانلود کنید. برای برنامه‌نویسی آندروید، می‌توانید از Mac، Windows PC و یا یک ماشین Linux استفاده کنید. همه‌ی آنچه که نیاز دارید رایگان و از وب، قابل دانلود است. بیشتر مثال‌های موجود در این کتاب به جز آن دسته که نیاز به دسترسی سخت‌افزاری



## فصل نخست / شروع برنامه‌نویسی با اندروید ----- 7

دارند، به خوبی با شبیه‌ساز قابل اجرا هستند. در این کتاب، ما از Windows 7 برای انجام همه‌ی مثال‌ها استفاده خواهیم کرد. در صورتی که از کامپیوترهای Mac و یا Linux استفاده می‌کنید، تصاویر یکسان خواهند بود. البته تفاوت‌هایی جزئی وجود خواهد داشت، اما می‌توانید بدون هیچ مشکلی کار را ادامه دهید. اکنون اجازه دهید که کار را آغاز کنیم.

 البته بیان این نکته لازم است که به دلیل تحریم علمی ایران، امکان دریافت همه‌ی موارد زیر به صورت مستقیم از اینترنت وجود ندارد. برای تهیه‌ی این قبیل موارد، باید از نرم‌افزارهای جانبی برای عبور از تحریم استفاده کنید. همچنین می‌توانید به‌وسیله‌ی سایت ProgrammerF1.com، آخرین نسخه‌ی برنامه‌های مورد نیاز را تهیه کنید.

### JAVA JDK

SDK مربوط به اندروید، از JDK استفاده می‌کند. به همین دلیل در صورتی که کامپیوتر شما دارای JDK به صورت نصب شده نباشد، ابتدا باید آن را از آدرس زیر دانلود و پیش از رفتن به بخش بعدی، آن را نصب نمایید:

[www.oracle.com/technetwork/java/javase/downloads/index.html](http://www.oracle.com/technetwork/java/javase/downloads/index.html)

### Eclipse

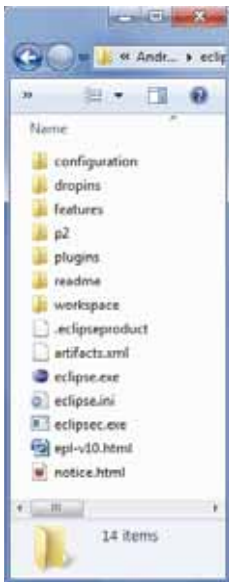
نخستین گام در تولید هر نوع برنامه، تهیه‌ی یک محیط یکپارچه‌ی برنامه‌نویسی یا IDE است. در مورد اندروید، IDE مورد توصیه‌ی Eclipse است؛ یک محیط توسعه‌ی یکپارچه‌ی چند زبانه، که از سیستم plug-in به صورت گسترده‌ای پشتیبانی می‌کند. این محیط می‌تواند برای تولید انواع مختلفی از برنامه‌ها، با استفاده از زبان‌هایی مانند Java، Ada، C، C++، COBOL، Python و غیره مورد استفاده قرار گیرد. برای برنامه‌نویسی اندروید باید این IDE را برای برنامه‌نویسی Java EE، از آدرس زیر دانلود نمایید:

[www.eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/heliosr1](http://www.eclipse.org/downloads/packages/eclipse-ide-java-ee-developers/heliosr1)

شش نگارش متفاوت وجود دارد. Windows (64 و 32 بیتی)، Mac OS X (64 و 32 Cocoa) و Linux (64 و 32 بیتی). تنها باید نسخه‌ای را انتخاب کنید که با سیستم‌عامل شما، هم‌خوانی داشته باشد. همه‌ی مثال‌های موجود در این کتاب، با نسخه‌ی 32 بیتی Eclipse در سیستم‌عامل Windows تست شده است.

هنگامی که IDE مربوط به Eclipse دانلود شد، آن را از حالت فشرده خارج سازید و حاصل را که در پوشه‌ی eclipse قرار دارد در مسیر C:\Android کپی کنید. شکل 7-1 محتوای پوشه‌ی eclipse را نشان می‌دهد.

## SDK آندروید



شکل 1-7

بخش مهم بعدی که باید دانلود کنید SDK آندروید است. SDK آندروید، دارای خطایاب، کتابخانه‌ها، شبیه‌ساز، مستندها، نمونه کد، راهنما و موارد آموزشی است. می‌توانید SDK آندروید را از آدرس زیر دانلود کنید:

<http://developer.android.com/sdk/index.html>

وقتی که SDK دانلود شد، محتوای آن را، که پوشه‌ای با نام android-sdk-windows است، از حالت فشرده، خارج کنید و در داخل پوشه‌ی C:\Android\ و یا پوشه‌ی دلخواه خود کپی نمایید.

## Android Development Tools (ADT)

ADT، یک plug-in برای Eclipse و یک گسترش‌دهنده‌ی Eclipse IDE است که امکان تولید و خطایابی برنامه‌های آندرویدی را در این محیط افزایش می‌دهد. با استفاده از ADT، قادر به انجام کارهای زیر خواهید بود:

- تولید یک پروژه‌ی آندرویدی.
- دسترسی به ابزارهایی برای شبیه‌سازی‌های دستگاه‌های آندرویدی مختلف.
- کمپایل و خطایابی پروژه‌های آندرویدی.
- تبدیل برنامه‌های آندرویدی به پکیج‌های آندروید (APK).
- تولید گواهی‌نامه‌های دیجیتال، برای امن کردن APKها (code-signing).

برای نصب ADT، ابتدا با دوبار کلیک کردن بر روی فایل eclipse.exe از پوشه‌ی eclipse، برنامه‌ی Eclipse را اجرا کنید.

هنگامی که Eclipse برای نخستین بار اجرا می‌شود از شما خواسته می‌شود که محل پوشه‌ی workspace خود را مشخص کنید. در داخل Eclipse، پوشه‌ی workspace محلی برای ذخیره کردن همه‌ی پروژه‌های شما است. محل پیش فرض را قبول و بر روی OK کلیک کنید. هنگامی که Eclipse به‌طور کامل بارگذاری شد از منوی Help، گزینه‌ی Install New Software... را همانند شکل 1-8 انتخاب کنید.

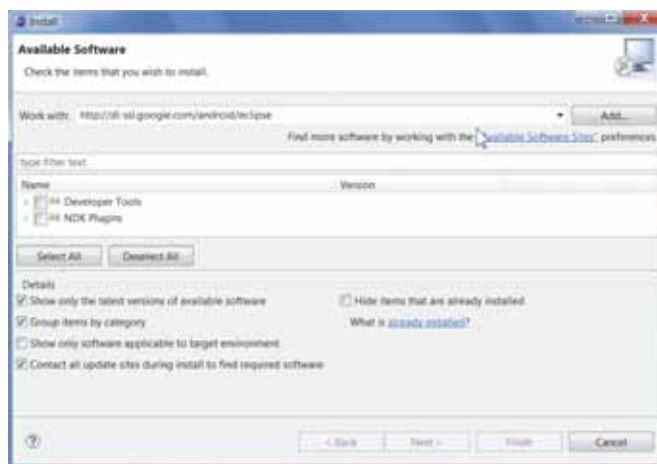
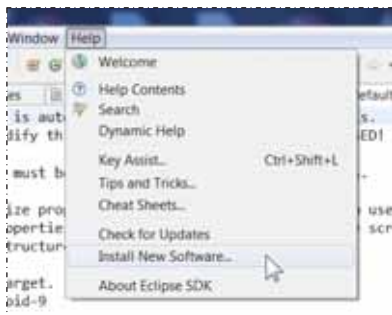
پس از باز شدن پنجره‌ی Install، آدرس زیر را در داخل textbox همانند شکل 1-9 وارد و دکمه‌ی Add را کلیک کنید:

<http://dl-ssl.google.com/android/eclipse>

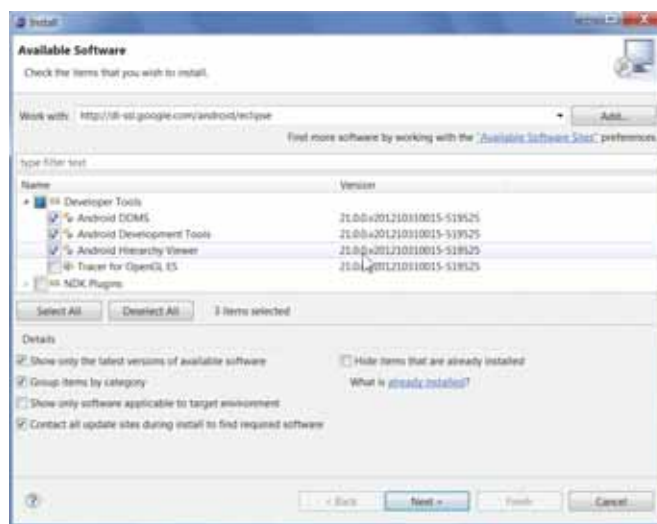
فصل نخست / شروع برنامه نویسی با اندروید ----- 9

پس از چند لحظه خواهید دید که آیتم Developer Tools در بخش میانی پنجره، همانند شکل 1-10 نمایان می شود. آن را باز کنید، باید گزینه های Android DDMS، Android Development Tools، Hierarchy Viewer Android دیده شوند. هر سه را انتخاب و بر روی دکمه ی Next کلیک کنید.

شکل 1-8

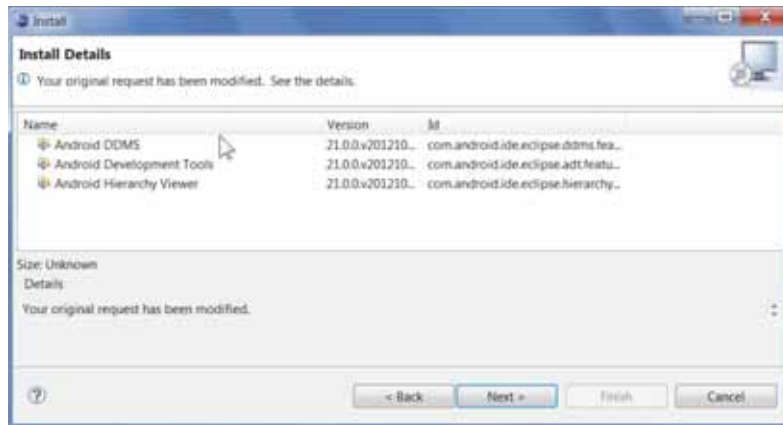


شکل 1-9



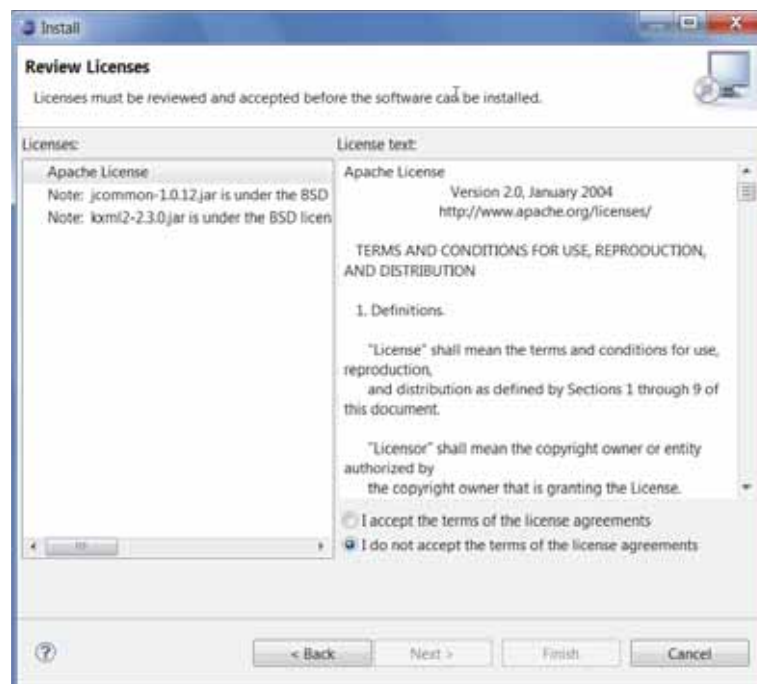
شکل 1-10

هنگامی که جزئیات نصب همانند شکل 1-11 نمایان شد بر روی Next کلیک کنید.



شکل 1-11

از شما خواسته می‌شود که لیسانس مربوط به این ابزار را مطالعه کنید. گزینه‌ی مربوط به موافقت با لیسانس را همانند شکل 1-12 انتخاب و بر روی دکمه Finish کلیک کنید.



شکل 1-12

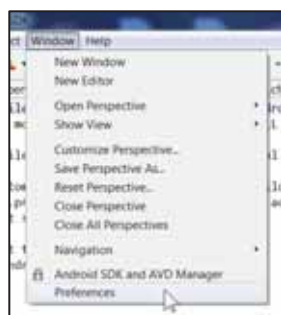


اکنون Eclipse شروع به دانلود ابزارهای انتخابی به‌وسیله‌ی اینترنت و نصب آنها خواهد کرد (شکل 1-13). این کار ممکن است زمان‌بر باشد.

شکل 1-13

در صورتی‌که در دانلود ADT، به مشکلی برخورد کردید راهنمای گوگل موجود در آدرس زیر را برای حل مشکل بخوانید:

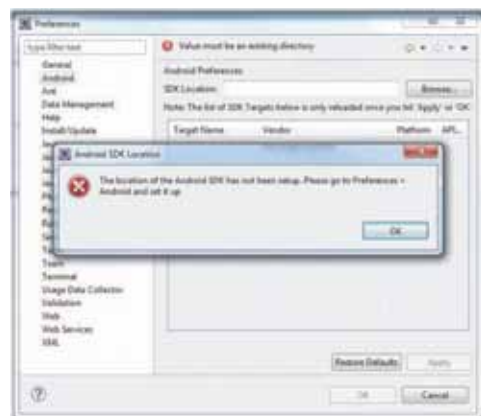
<http://developer.android.com/sdk/eclipse-adt.html#installing>



هنگامی که ADT نصب شد، از شما درخواست می‌شود Eclipse را ببندید و دوباره اجرا کنید. پس از بازکردن دوباره‌ی برنامه، به منوی Window زیرمنوی Preferences همانند شکل 1-14 بروید.

شکل 1-14

در پنجره‌ی Preferences از لیست سمت چپ، Android را انتخاب کنید. خطایی را مبنی بر اینکه SDK تنظیم نشده است، همانند شکل 1-15 دریافت خواهید کرد. بر روی OK کلیک کنید.



سیر قرارگیری پوشه‌ی SDK را مشخص کنید که در مثال ما `C:\Android\android-sdk-windows` می‌باشد. سپس بر روی OK کلیک کنید.

شکل 1-15

### تولید شبیه‌ساز مجازی اندروید (AVD)

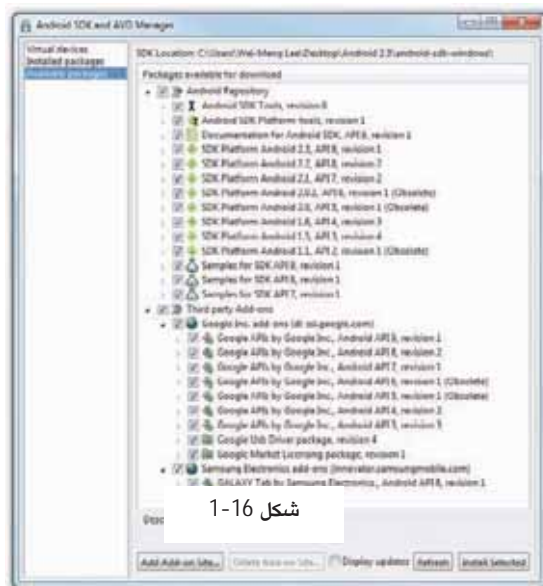
گام بعدی، تولید یک AVD برای استفاده در طول تست برنامه‌های اندرویدی است. یک AVD، یک نمونه‌ی شبیه‌سازی است که شما را قادر می‌سازد تا ابزار واقعی را مدل‌سازی کنید. هر AVD، دارای یک پروفایل ساخت‌افزار و یک نگاشت به یک image سیستم، برای شبیه‌سازی یک فضای ذخیره‌سازی همانند کارت SD، می‌باشد.

می‌توانید AVDهای گوناگونی را بنا به نیاز خود برای تست برنامه بر روی پیکربندی‌های متفاوت، ایجاد کنید. این گونه تست‌ها برای تثبیت رفتار برنامه در زمان اجرا، بر روی ابزارهای متفاوت با توانایی‌های متفاوت لازم و ضروری است.

در پیوست ب برخی از توانایی‌های شبیه‌ساز آندروید توضیح داده شده است.



برای تولید یک AVD بوسیله‌ی منوی Windows، گزینه‌ی Android SDK and AVD Manager را انتخاب کنید. از



لیست سمت چپ، گزینه‌ی Available packages را انتخاب کنید و در لیست نمایان شده در سمت راست، پکیج‌های در دسترس را باز کنید. شکل 1-16، پکیج‌های مختلفی را که برای تولید AVD در دسترس قرار دارد و با استفاده از آنها می‌توانید ابزارهای مختلف آندروید را شبیه‌سازی کنید، نشان می‌دهد.

اسناد، ابزارهای مربوطه و سکوی کاری که برای پروژه‌ی خود به آنها نیاز دارید را مطالعه کنید. هنگامی که آیتم مورد نظر را انتخاب کردید، بر روی دکمه‌ی Install Selected کلیک کنید تا مورد انتخابی دانلود شود.

به‌خاطر اینکه عملیات دانلود از سرورهای گوگل

انجام می‌شود، بهترین حالت این است که گزینه‌ای که هم‌اکنون مورد نیاز است را دانلود و دیگر موارد را بعداً هنگامی که زمان بیشتری در اختیار داشتید، دانلود کنید.

در آغاز، دست‌کم باید آخرین نسخه‌ی SDK را انتخاب کنید. برای نوشتن این کتاب، از نسخه‌ی SDK مربوط به Android 2.3، API 9 نگارش 1 استفاده شده است.



هر یک از نگارش‌های سیستم‌عامل آندروید، با استفاده از شماره‌ی سطح API شناسایی می‌شود. برای نمونه، Android 2.3، سطح 9 یا API 9 می‌باشد در حالی که Android 2.2، سطح 8 یا API 8 است و به‌همین ترتیب دیگر سطوح نیز مشخص می‌شود. برای هر سطح، دو سکوی کاری وجود دارد. برای نمونه، سطح 9 دو مورد زیر را ارائه می‌دهد:

## فصل نخست / شروع برنامه‌نویسی با اندروید ----- 13

- SDK Platform Android 2.3
- Google APIs by Google Inc

تفاوت بین این دو در این است که سکوی کاری Google APIs، دارای کتابخانه‌ی Google Maps می‌باشد. به این ترتیب در صورتی که برنامه‌ای می‌نویسید که نیاز به استفاده از Google Maps داشته باشد، باید AVD را با استفاده از سکوی کاری Google APIs که در فصل 9 با جزئیات بیشتر مورد بحث قرار می‌گیرد، ایجاد کنید.



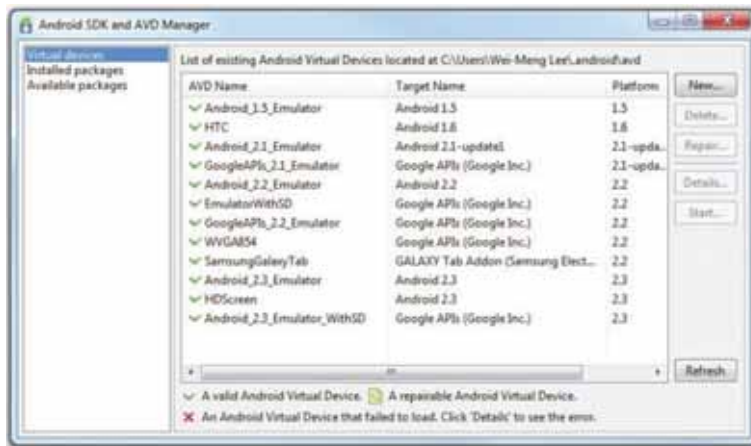
از لیست موجود در بخش سمت چپ، گزینه‌ی Virtual Device را انتخاب کنید. سپس بر روی دکمه‌ی New... که در سمت راست پنجره قرار گرفته است کلیک کنید. در پنجره‌ی جدیدی که باز می‌شود، آیتم‌هایی را که در شکل 1-17 می‌بینید وارد و بر روی دکمه‌ی Create AVD کلیک کنید.

شکل 1-17

به این ترتیب یک AVD که یک شبیه‌ساز ساده‌ی اندروید است را ایجاد کرده‌اید. ابزار شبیه‌سازی شده، دارای سیستم عامل اندروید نگارش 2.3 خواهد بود. در هنگام ایجاد، می‌توانستید تنظیم‌های مربوط به کارت SD و همچنین وضوح صفحه‌ی نمایش را نیز مشخص کنید.

در پیوست ب توضیح خواهیم داد که چگونه می‌توان ابزارهای مختلف اندروید را شبیه‌سازی کرد.

بهتر است تعدادی AVD با سطح APIهای متفاوتی ایجاد کنید. به این ترتیب می‌توانید برنامه‌ی خود را بر روی ابزارهای مختلفی تست کنید. نمونه‌ی نشان داده شده در شکل 1-18، لیستی از AVDهای ایجاد شده برای تست برنامه‌های تولیدی بر روی سکوهایی کاری مختلف اندرویدی را نشان می‌دهد.



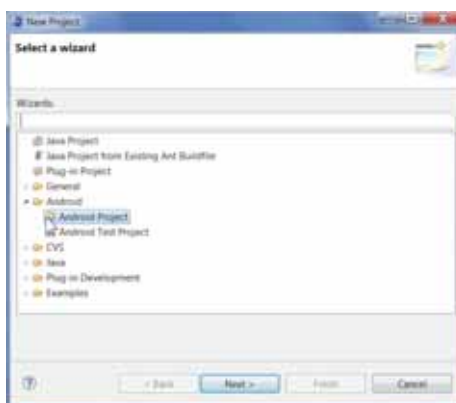
شکل 1-18

### تولید نخستین برنامه‌ی آندرویدی

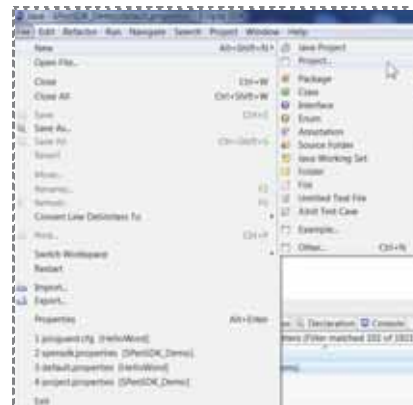
هنگامی که همه‌ی ابزارها و SDK، دانلود و نصب گردید، زمان آن می‌رسد که کار خود را آغاز کنید. همانند کتاب‌های آموزش برنامه‌نویسی، نخستین مثال برنامه، Hello World است. این مثال به شما کمک خواهد کرد که با بخش‌های مختلفی که یک پروژه‌ی آندرویدی را در برمی‌گیرد، آشنا شوید.

### تمرین: تولید نخستین برنامه‌ی آندرویدی

1. با استفاده از Eclipse همانند شکل 1-19، به‌وسیله‌ی منوی File گزینه‌ی Project...، یک پروژه‌ی جدید ایجاد کنید.



شکل 1-20



شکل 1-19

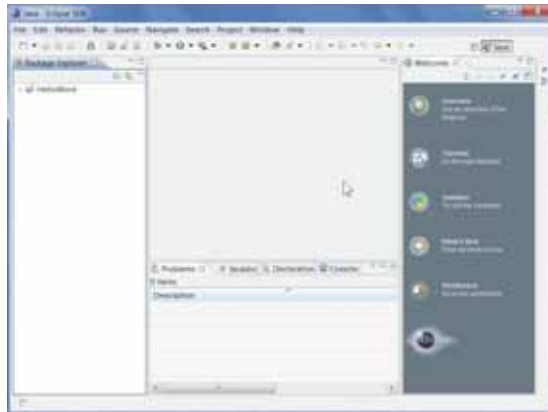


## فصل نخست / شروع برنامه نویسی با اندروید ----- 15

پس از ایجاد نخستین پروژه‌ی اندرویدی، زیرمجموعه‌ی پروژه‌های اندروید به وسیله‌ی انتخاب **File** → **New** → **Android Project** در دسترس خواهند بود.

2. پوشه‌ی **Android** را باز کنید و همانند شکل 1-20، گزینه‌ی **Android Project** را انتخاب کنید.

3. نام پروژه‌ی اندرویدی را همانند شکل 1-21 مشخص و بر روی **Finish** کلیک کنید.



شکل 1-22



شکل 1-21

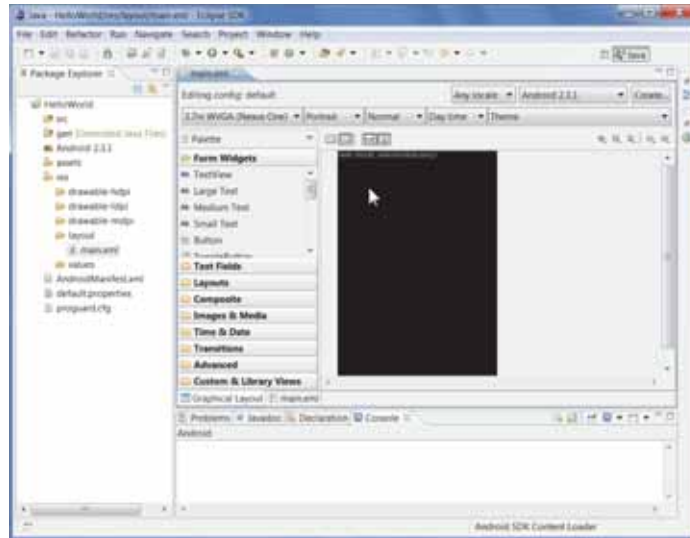
باید دست کم از یک کاراکتر نقطه (.) در نام پکیج استفاده کنید. مدل پیشنهادی برای نام گذاری پکیج‌ها، استفاده از نام دامنه با ترتیب عکس است که در نهایت به نام پروژه ختم می‌شود. برای نمونه، اگر نام دامنه‌ی شرکت شما `net.learn2develop.net` باشد، نام پکیج تان به صورت `net.learn2develop.HelloWorld` خواهد بود.

4. اکنون Eclipse IDE باید همانند شکل 1-22 باشد.

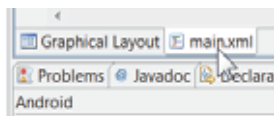
5. در **Package Explorer** که در سمت چپ IDE قرار گرفته است، با کلیک بر روی فلش‌هایی که در سمت چپ

هر یک از آیت‌های پروژه قرار گرفته است آنها را باز کنید. در داخل پوشه‌ی `res/layout` بر روی فایل

`main.xml` همان‌طور که در شکل 1-23 نشان داده شده است، دوبار کلیک کنید.



شکل 1-23



شکل 1-24

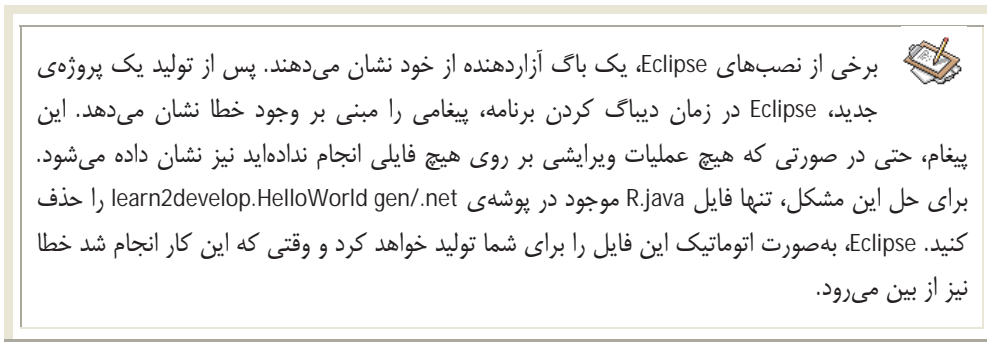
6. فایل main.xml، واسط کاربر یا UI برنامه را تعریف می‌کند. نمای پیش فرض، نمایش پوسته‌ی view می‌باشد که به شما امکان می‌دهد به صورت گرافیکی به تنظیم UI بپردازید. برای ویرایش UI، بر روی زبانه‌ی main.xml که در پایین، همانند شکل 1-24 قرار گرفته است کلیک کنید.
7. کدهای پر رنگ شده‌ی زیر را به فایل main.xml بیافزایید.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="@string/hello" />
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="This is my first Android Application!" />
    <Button
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="And this is a clickable button!" />
</LinearLayout>
```

8. برای ذخیره‌ی تغییرها، Ctrl+S را فشار دهید.

## فصل نخست / شروع برنامه‌نویسی با اندروید ----- 17

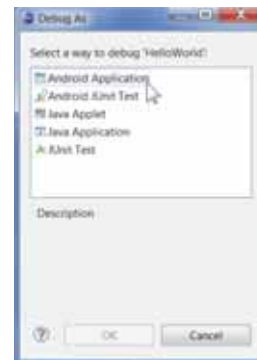
9. اکنون آماده‌ی تست برنامه‌ی خود در شبیه‌ساز هستید. نام پروژه را در Eclipse، از package explorer انتخاب و کلید F11 را فشار دهید. در مورد شیوه‌ی خطایابی برنامه پرسیده می‌شود که باید گزینه‌ی Application Android را همانند شکل 1-25 انتخاب و سپس بر روی OK کلیک کنید.



10. سپس شبیه‌ساز اندروید اجرا خواهد شد. (در صورتی که شبیه‌ساز قفل باشد ابتدا دکمه‌ی unlock را که بر روی صفحه ظاهر می‌شود بکشید، تا از حالت قفل خارج شود.) شکل 1-26، خروجی اجرای برنامه را در محیط شبیه‌ساز اندروید نشان می‌دهد.



شکل 1-26



شکل 1-25

11. بر روی دکمه‌ی Home که دارای تصویر خانه است و در گوشه‌ی پایین و سمت چپ (در بالای صفحه کلید) قرار گرفته است کلیک کنید. اکنون صفحه‌ی خانه همانند شکل 1-27 نشان داده می‌شود.



شکل 1-27

12. برای مشاهده لیست برنامه‌های نصب شده، بر روی آیکن Launcher کلیک کنید. همان‌طور که در شکل 1-28 مشاهده می‌کنید، برنامه‌ی HelloWorld بر روی ابزار نصب شده است.



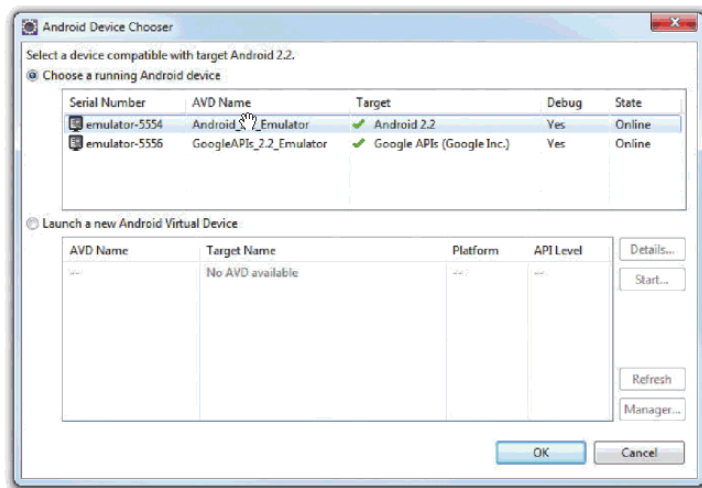
شکل 1-28

### مناسب‌ترین AVD برای تست برنامه

به یاد دارید که کمی پیش‌تر، تعدادی AVD را با استفاده از مدیریت‌کننده‌ی AVD تولید کردید. اما کدام یک باید به‌وسیله‌ی Eclipse در هنگام اجرای یک برنامه‌ی آندرویدی، مورد استفاده قرار گیرد؟ Eclipse مشخصاتی را، که در هنگام تولید پروژه مشخص کرده‌اید، با لیست AVDهای موجود مقایسه می‌کند و نخستین گزینه‌ای که با برنامه تناسب دارد، اجرا می‌کند.

## فصل نخست / شروع برنامه‌نویسی با اندروید ----- 19

در صورتی که بیش از یک AVD مناسب با برنامه یافت شد، یک پنجره همانند شکل 1-29 برای انتخاب ابزار برای عملیات دیباگ به وسیله‌ی Eclipse نمایش داده می‌شود.



شکل 1-29

### برنامه چگونه کار می‌کند

برای تولید یک پروژه‌ی اندروید به وسیله‌ی Eclipse، باید اطلاعات لیست شده در جدول 1-2 را مشخص کنید.

توضیحات	خصوصیت
نام پروژه	Project name
نام کاربر پسند برنامه‌ی شما	Application name
نام پکیج شما - باید به صورت معکوس نام دامنه‌ی خود را در آن استفاده کنید.	Package name
نام نخستین Activity درون برنامه	Create Activity
کمینه‌ی نگارش SDK مورد نیاز برای اجرای برنامه	Min SDK Version

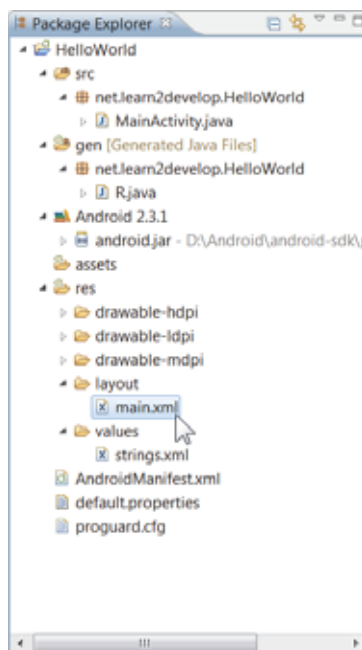
در اندروید، یک Activity پنجره‌ای است که دارای واسط کاربر یا UI مربوط به برنامه می‌باشد (چیزی همانند فرم‌ها در برنامه‌های رومیزی یا صفحه‌ها در برنامه‌های مبتنی بر وب). هر برنامه می‌تواند صفر یا چند Activity داشته باشد. در مثال پیش، برنامه دارای یک Activity با نام MainActivity می‌باشد. این MainActivity، مدخل ورود برنامه است که در زمان اجرای برنامه نمایش داده می‌شود. در فصل 2، در مورد Activity‌ها با جزئیات بیشتر

صحبت خواهد شد. در مثال پیش، فایل main.xml را برای نمایش عبارت " This is my first Android Application!" و اضافه کردن یک دکمه، ویرایش کردید.

هنگامی که برنامه را در داخل شبیه‌ساز آندروید اجرا کردید، برنامه به‌صورت خودکار در شبیه‌ساز نصب شد و به این ترتیب نخستین برنامه‌ی آندرویدی خود را تولید کردید. در بخش بعدی به توضیح این نکته خواهیم پرداخت که چگونه فایل‌های مختلف پروژه‌ی آندرویدی، در کنار یکدیگر سبب اجرای برنامه می‌گردند.

## تشریح ساختار یک برنامه‌ی آندرویدی

اکنون که نخستین برنامه‌ی آندرویدی خود را تولید کرده‌اید، زمان آن رسیده است که به کالبدشکافی قسمت‌های داخل یک پروژه‌ی آندرویدی و توضیح همه‌ی بخش‌هایی که سبب اجرای برنامه می‌گردند، بپردازیم. در ابتدا توجه داشته باشید که همان‌طور که در شکل 1-30 نشان داده شده است، فایل‌های گوناگونی در داخل یک پروژه‌ی آندرویدی وجود دارد. این مجموعه، در داخل Package Explorer دیده می‌شود.



شکل 1-30

پوشه‌ها و فایل‌های موجود به‌شرح زیر است: