

Лекция № 3
Разработка приложений на
платформе Android

План

- ▶ **Описание установки:**
- ▶ Среда разработки Eclipse
- ▶ Adroid SDK
- ▶ Плагин ADT
- ▶ Android API **для элементов UI:**
- ▶ Кнопки
- ▶ Поля текстового ввода
- ▶ Простейшие списки
- ▶ Размещение элементов с помощью layouts
- ▶ **Отладочная сессия в Eclipse и консоли**



Описание установки

- ▶ Среда разработки Eclipse
- ▶ <http://www.eclipse.org/downloads/>
- ▶ Установить JDK — Java Development Kit,
- ▶ <http://www.oracle.com/technetwork/java/javase/downloads/index.html>



Android SDK

- ▶ Android SDK представляет средство разработки приложений под Android. В него входят:
- ▶ **android.jar** - Файл архива Java, содержащий все классы SDK Android, необходимые для создания приложений.
- ▶ **documentation.html** и каталог **docs** - Документация SDK предоставляется локально и через Интернет.
- ▶ **Каталог инструментов** - Содержит все инструменты командной строки для создания Android-приложений.
- ▶ **USB-driver** - Каталог, содержащий все необходимые драйверы для подключения среды разработки к поддерживающим Android устройствам.

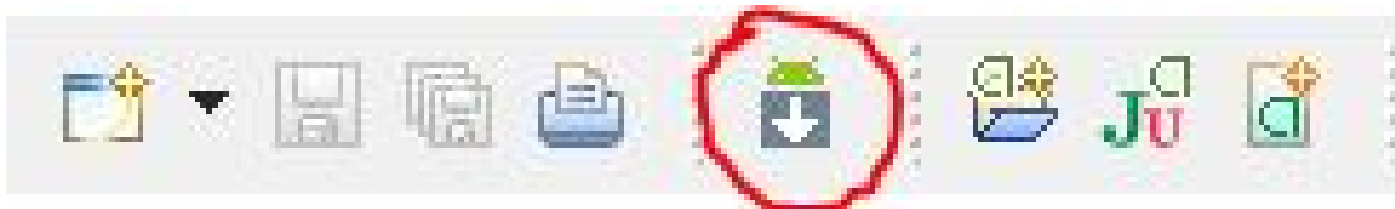
Последняя версия Android SDK

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



Android ADT

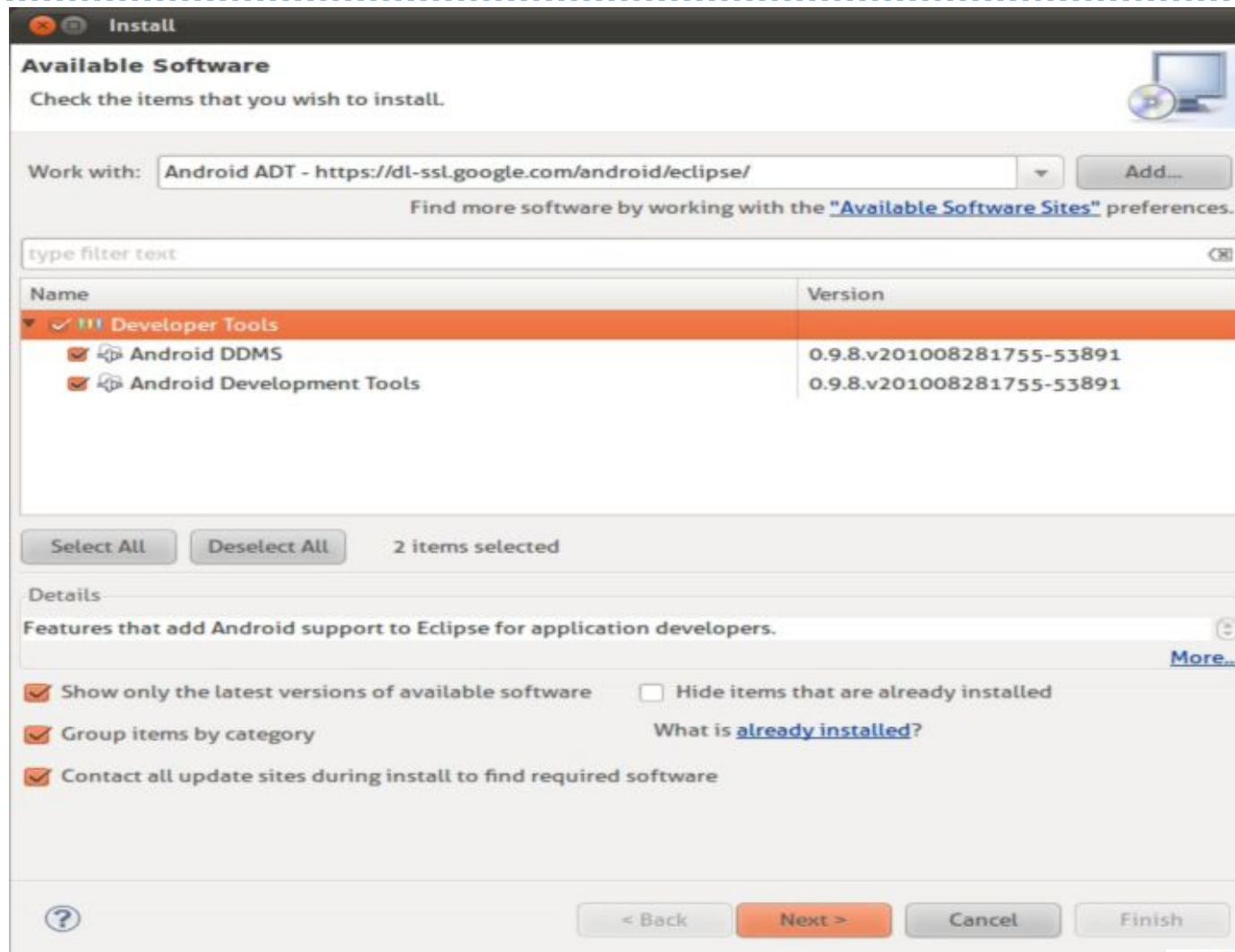
- ▶ Android предлагает плагин ADT для Eclipse IDE, который обеспечивает разработчиков эффективными инструментами создания Android приложений. К примеру, установив ADT, на панелях toolbar и menubar появятся контролы, позволяющие быстро создавать проект.



Установка Android ADT

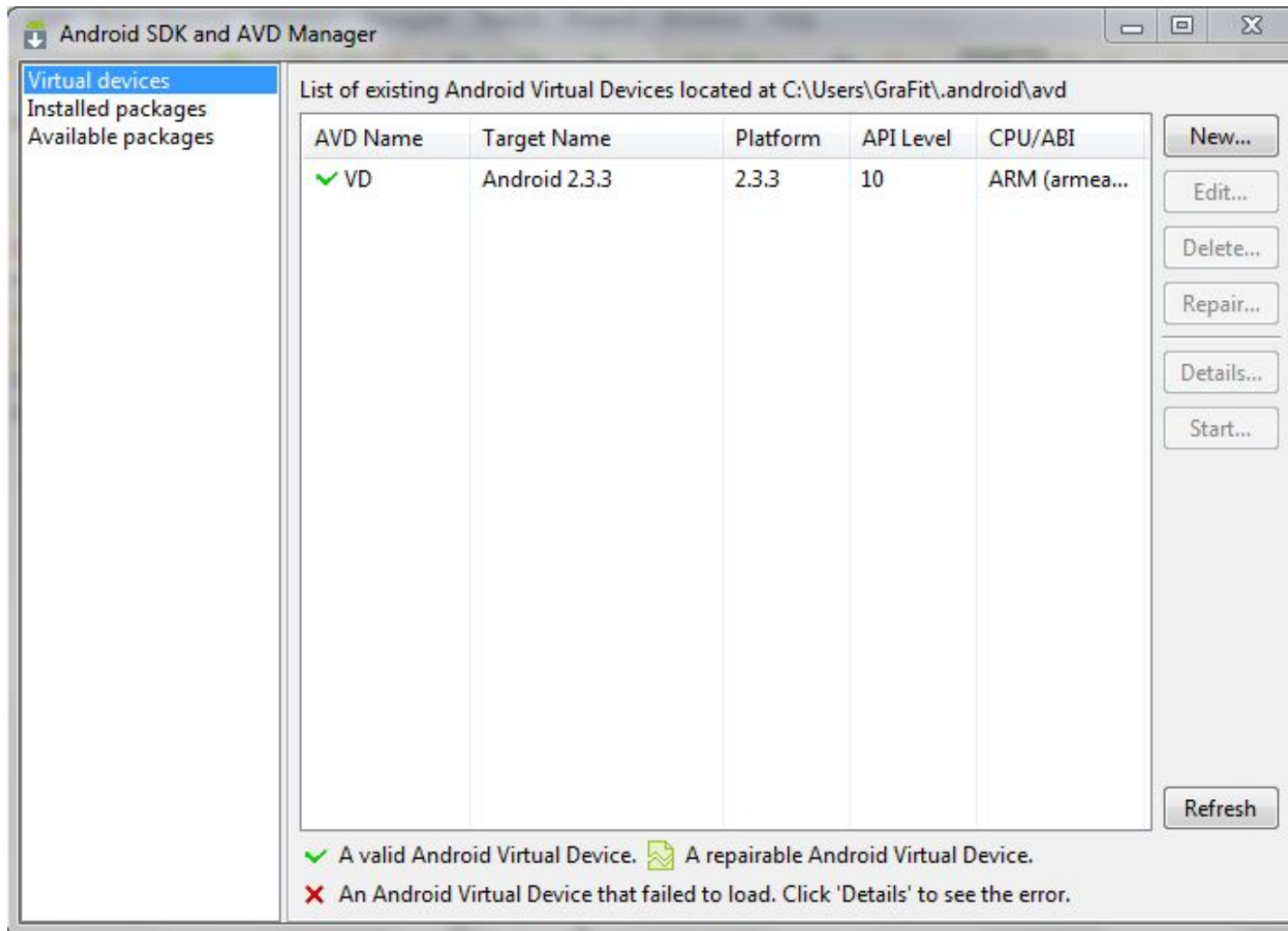
- ▶ Чтобы установить ADT, необходимо открыть панель расширения среды (Install New Software), затем в появившемся окне добавить новый путь - <https://dl-ssl.google.com/android/eclipse/>
- ▶ Далее Eclipse выполнит поиск требуемых данных, и в выпадающем списке вы сможете выбрать ADT и установить его. После установки у вас появится AVD Manager. Его можно запустить прямо из Eclipse (Window-> Android SDK and AVD Manager)

Установка Android ADT

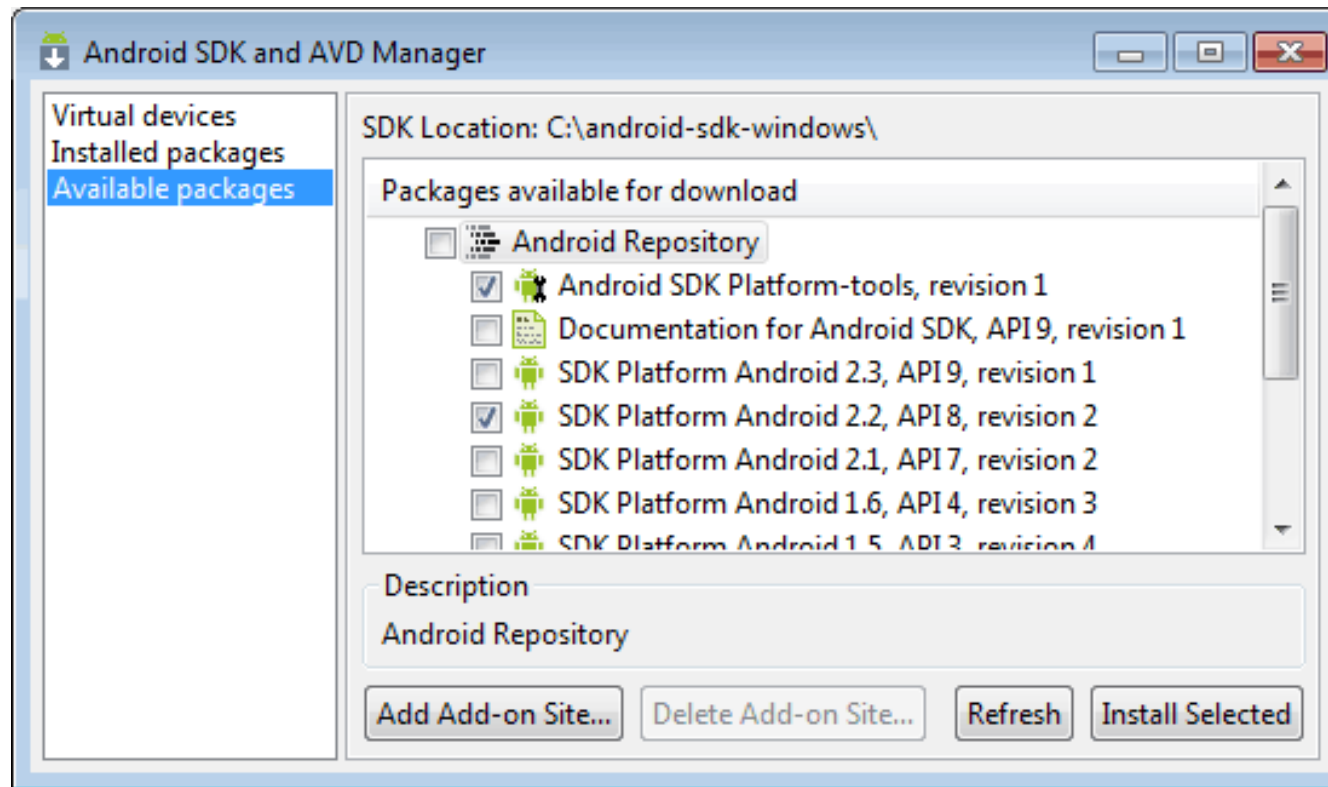


Разработка приложений для нетбуков планшетников. Лекция 3

Настройка Virtual Device



Установка версий API



Кнопки

- ▶ Кнопка описана классом android.widget.Button, который является суперклассом для всевозможных типов кнопок (флажковая-кнопка, радио-кнопка, переключатель)
- ▶ У кнопки определено несколько конструкторов:
 - **Button(Context context);**
 - **Button(Context context, AttributeSet attr);**
 - **Button(Context context, AttributeSet attr, int defStyle);**

Надо сказать что, Context - класс содержащий информацию о среде приложения и предоставляющий доступ к различным системным ресурсам и классам

Поля текстового ввода

- ▶ Android.widget.EditText — является небольшой облицовкой вокруг TextView, позволяющей создавать редактируемый текст.
- ▶ EditText имеет несколько конструкторов:
 - **EditText(Context context);**
 - **EditText(Context context, AttributeSet attr);**
 - **EditText(Context context, AttributeSet attr, int defStyle);**

Списки

- ▶ Android.widget.ListView — класс отображающий набор элементов в виде списка с возможностью прокрутки. Элементы ассоциированные с этим представлением описываются классом ListAdapter.
- ▶ ListAdapter — наследует базовый класс Adapter и служит мостом между данными и ListView. Удобство в том, что ListView может отображать любые данные, лишь бы они были завернуты в ListAdapter.

Layout

▶ **LinearLayout**

- ▶ Линейный менеджер размещения используется для отображения элементов в виде линейных списков, горизонтальных или вертикальных.
- ▶ У `LinearLayout` определено несколько конструкторов:
- ▶ → `LinearLayout(Context context)`
- ▶ → `LinearLayout(Context context, AttributeSet)`
- ▶ → `LinearLayout(Context context, AttributeSet , int defStyle)`

Layout

▶ TabLayout

- ▶ Чтобы организовать несколько отображений в одном окне и обеспечить между ними быстрое и комфортное переключение используется вкладочный менеджер размещения (tab layout)
- ▶ Чтобы создать tab layout необходимо выполнить следующее:
- ▶ Создать TabHost - этот элемент является контейнером для вкладок (табов) оконного вида. TabHost должен иметь два дочерних элемента TabWidget и FrameLayout.
TabWidget — элемент представляющий список вкладок.
FrameLayout, используемый для отображения содержимого каждой вкладки.

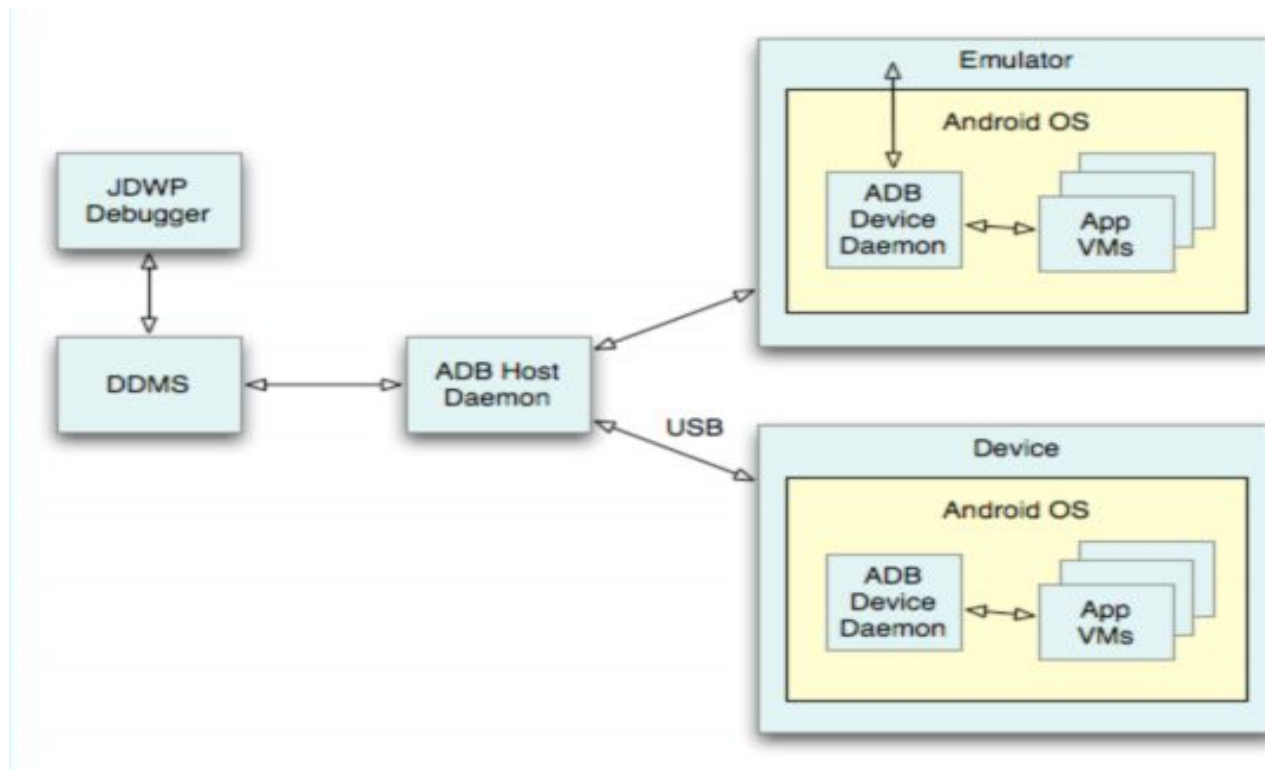
Отладка

Основные компоненты среды отладки Android:

- ▶ ADB (Android Debug Bridge) - выступает в качестве посредника между устройством и программой.
- ▶ Dalvik Debug Monitor Service(DDMS) - это элемент плагина Android, используемый для анализа работы VM. Последовательно отображает поток информации, которая описывает каждый шаг работы виртуальной машины.
- ▶ Device or Android Virtual Device(AVD). Ваше приложение должно выполняться на устройстве или в AVD так чтобы оно могло быть отлажено. Демон adb устройства работает на устройстве или эмуляторе, и предоставляет средства для демона adb хоста для предоставления связи с устройством или эмулятором.

Отладка

- ▶ Взаимодействие средств отладки между собой



Отладка

- ▶ *ADB* – клиент-серверное приложение состоящее из 3х компонентов:
- ▶ Клиент – запускается на машине разработчика. Клиент можно запустить из командной строки при помощи команд посылаемых *adb*. Другие инструменты Android вроде плагина ADT и DDMS тоже создают *adb*-клиенты.
- ▶ Сервер – запускается на машине разработчика в виде фонового процесса. Сервер управляет соединениями между клиентами и *adb*-сервисом запущенным на эмуляторе или устройстве.
- ▶ Сервис – фоновый процесс, который запускается на каждом эмуляторе или устройстве.

Отладка

```
Administrator: Command Prompt

F:\Mobile and PDA\Android\dev kits\android-sdk_r07-windows\android-sdk-windows\t
ools>adb devices
List of devices attached
HT9CLP802105    device

F:\Mobile and PDA\Android\dev kits\android-sdk_r07-windows\android-sdk-windows\t
ools>adb remount
remount succeeded

F:\Mobile and PDA\Android\dev kits\android-sdk_r07-windows\android-sdk-windows\t
ools>adb pull /system/fonts/DroidSans.ttf DroidSans.ttf
1271 KB/s (190044 bytes in 0.146s)

F:\Mobile and PDA\Android\dev kits\android-sdk_r07-windows\android-sdk-windows\t
ools>adb shell mv /system/fonts/DroidSans.ttf /system/fonts/DroidSans.ttf.orig

F:\Mobile and PDA\Android\dev kits\android-sdk_r07-windows\android-sdk-windows\t
ools>adb push Unique.ttf /system/fonts/DroidSans.ttf
761 KB/s (50700 bytes in 0.065s)

F:\Mobile and PDA\Android\dev kits\android-sdk_r07-windows\android-sdk-windows\t
ools>adb reboot

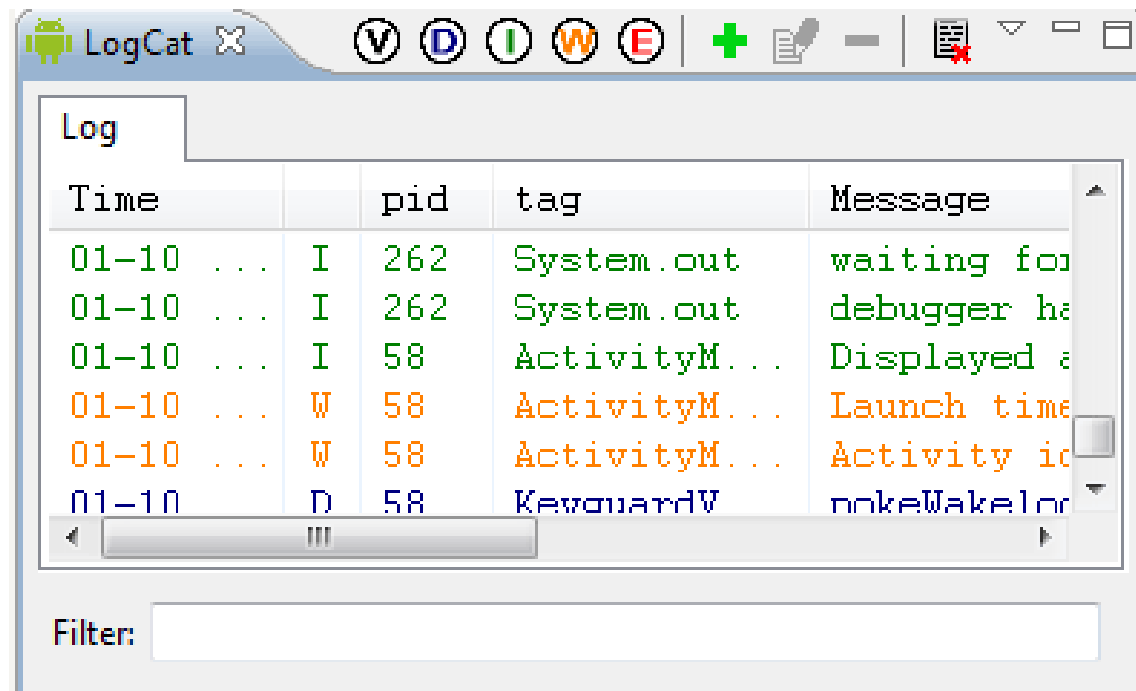
F:\Mobile and PDA\Android\dev kits\android-sdk_r07-windows\android-sdk-windows\t
ools>_
```



Отладка

- ▶ *LogCat* - Система ведения лога в Android, которая обеспечивает механизм для сбора и просмотра системных отладочных сообщений. Логи собираются в серии кольцевых буферов, которые можно просматривать и фильтровать посредством команды `logcat`.
- ▶ Формат использования: `[adb] logcat [<option>] ... [<filter-spec>] ...`
- ▶ Также предусмотрена фильтрация вывода лога. Каждое сообщение лога в Android имеет *Тэг* и *приоритет*. Выражения фильтра имеют следующий формат `<тэг>:<приоритет>`.

Logcat



САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ



Благодарю за внимание!

Вопросы?