

Лабораторная работа: утилиты Android SDK

Цель

Научится работать с инструментами, предоставляемыми SDK.

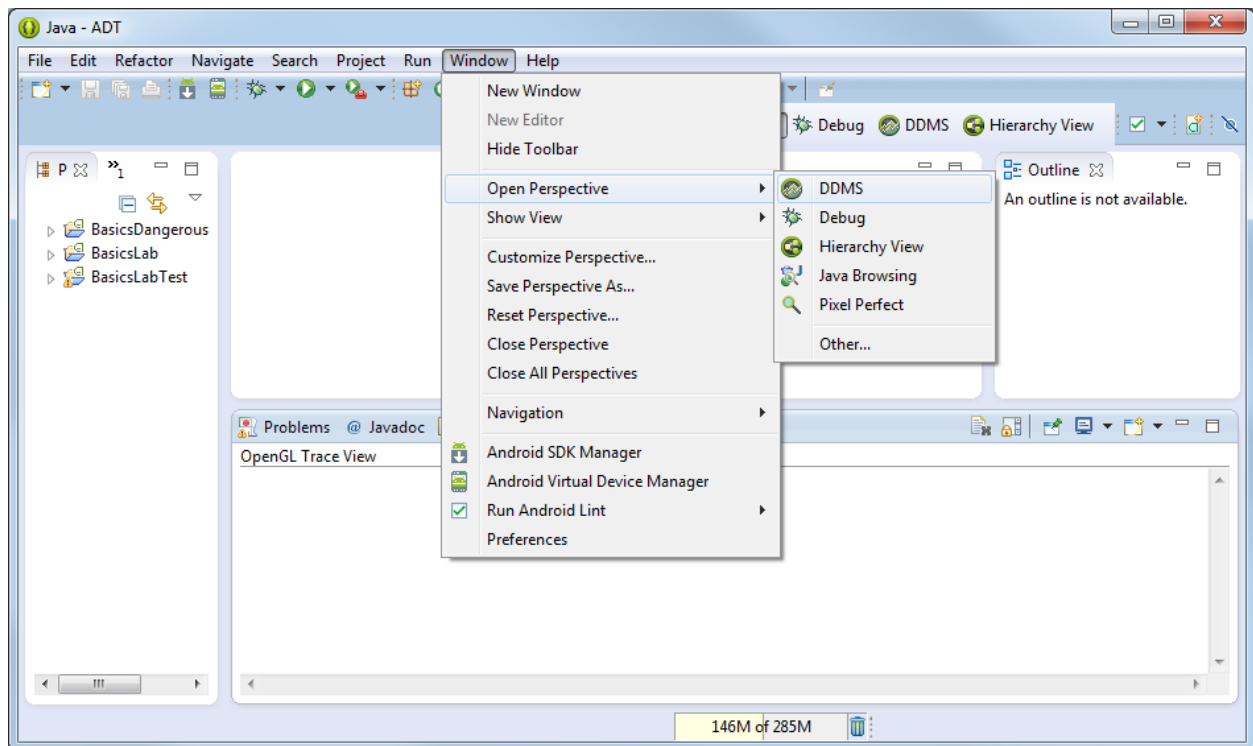
Предварительная подготовка

Подготовленный к работе ADT.

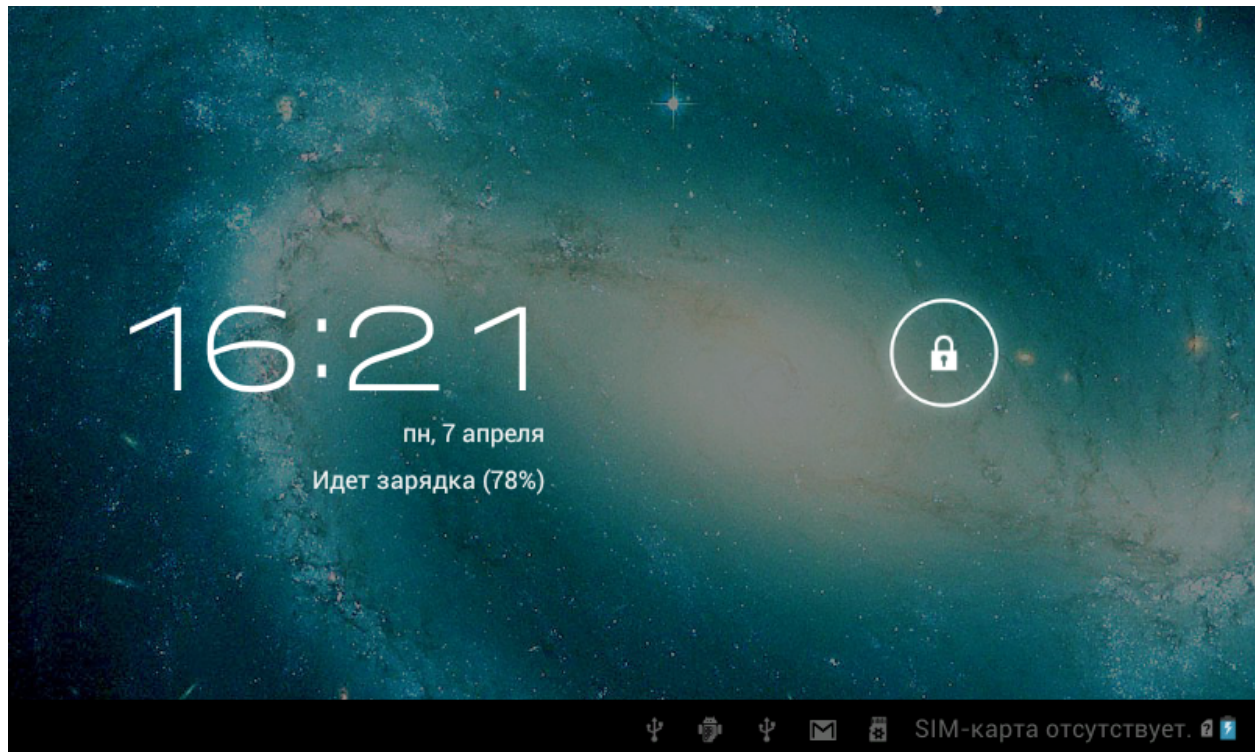
Программа работы

1. Исследование IDE

Для начала научимся создавать скриншоты с экрана нашего устройства. Либо запустим эмулятор с android, либо подключим по USB android устройство. После этого перейдем на перспективу DDMS.

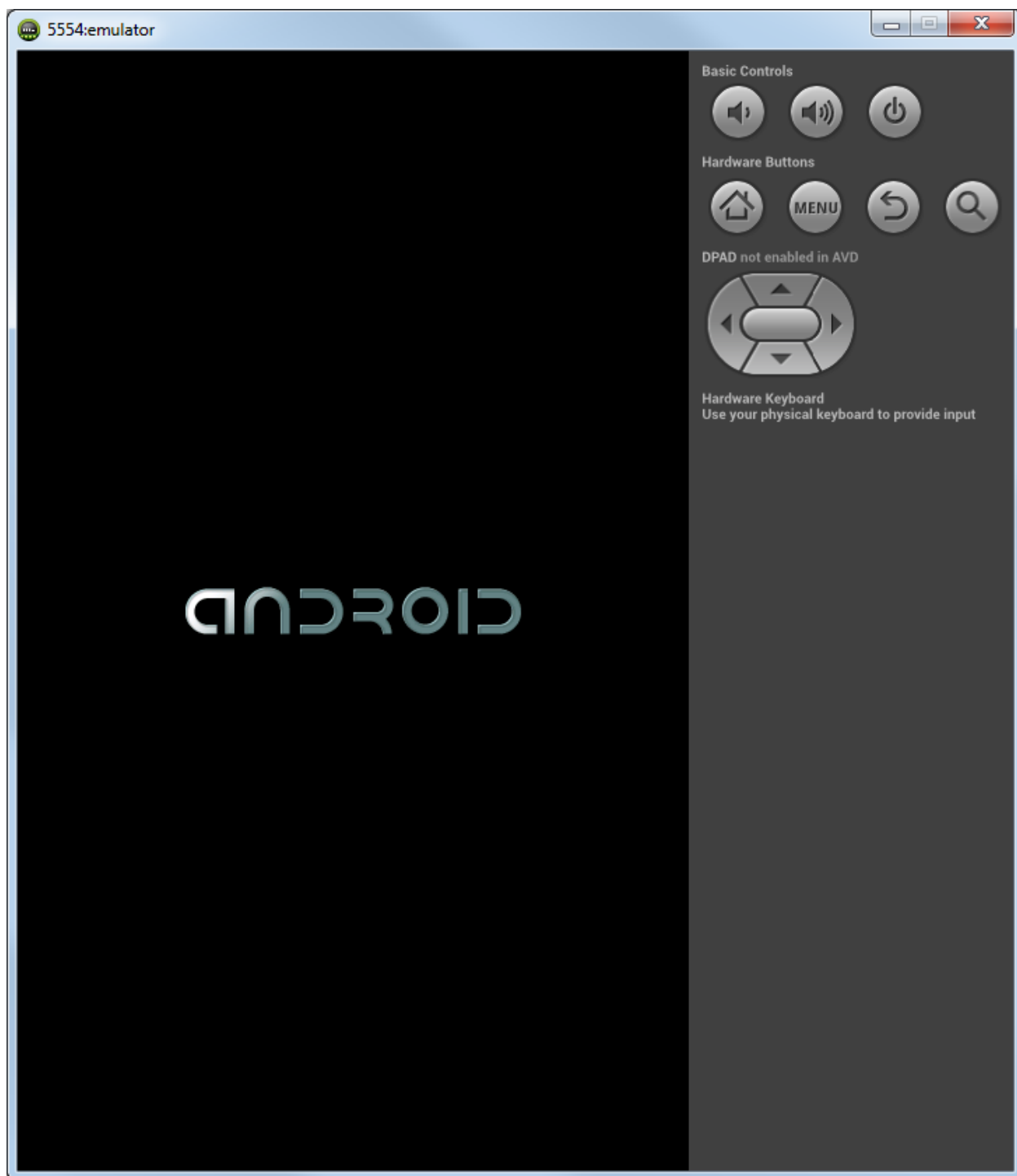


Нажмем на кнопку Screen capture



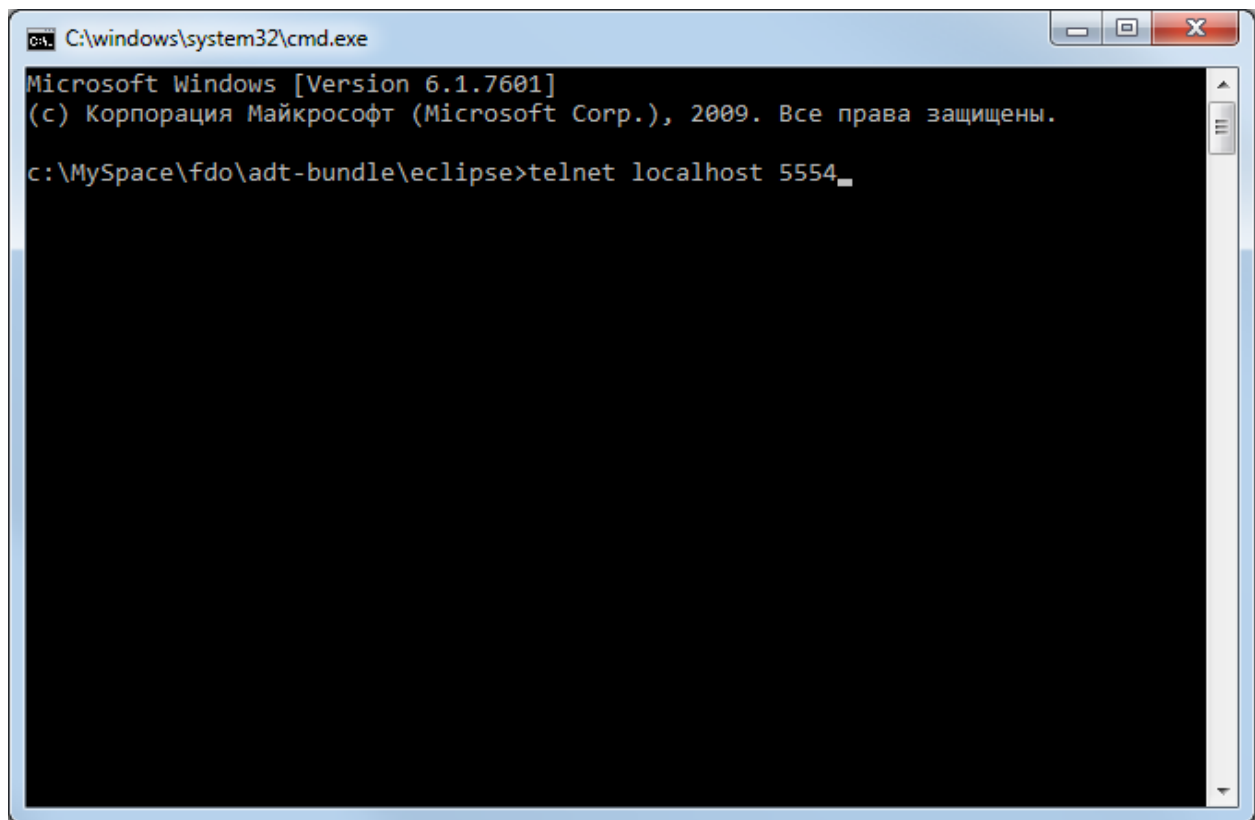
2. Работа с telnet

Для следующих действий нам потребуется запущенный эмулятор android. Запустите эмулируемое устройство на компьютере.



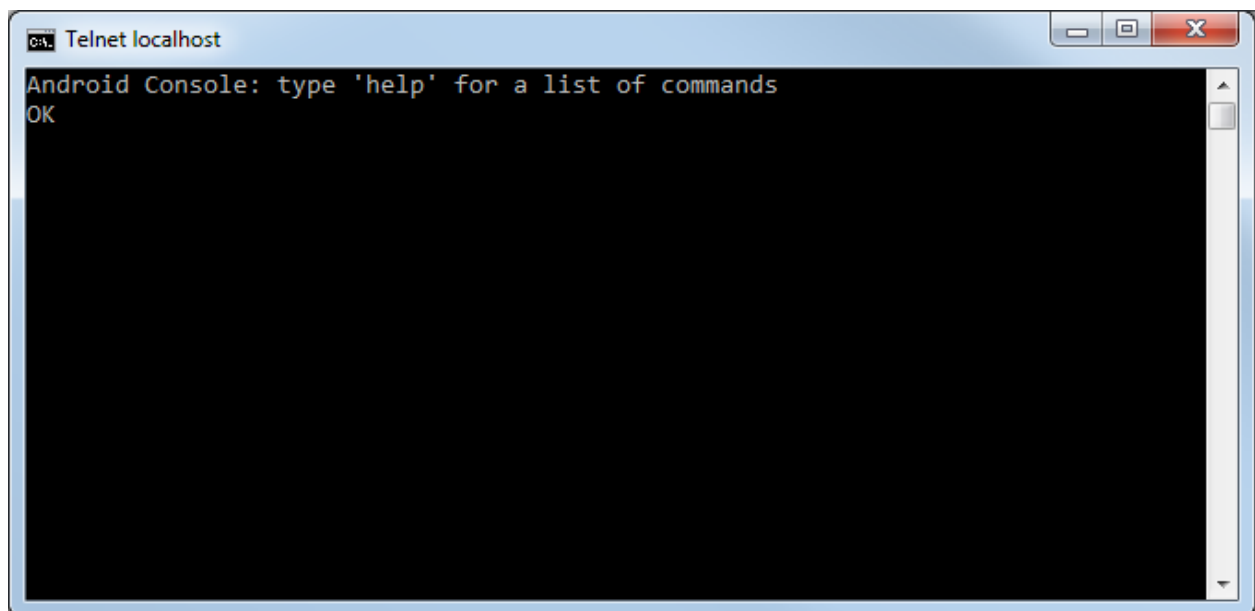
В верхнем левом углу указан порт, по которому можно управлять эмулируемым устройством. В нашем случае это порт 5554.

Введите в командной строке: `telnet localhost 5554`
И нажмите enter, вы подключитесь к консоли устройства.



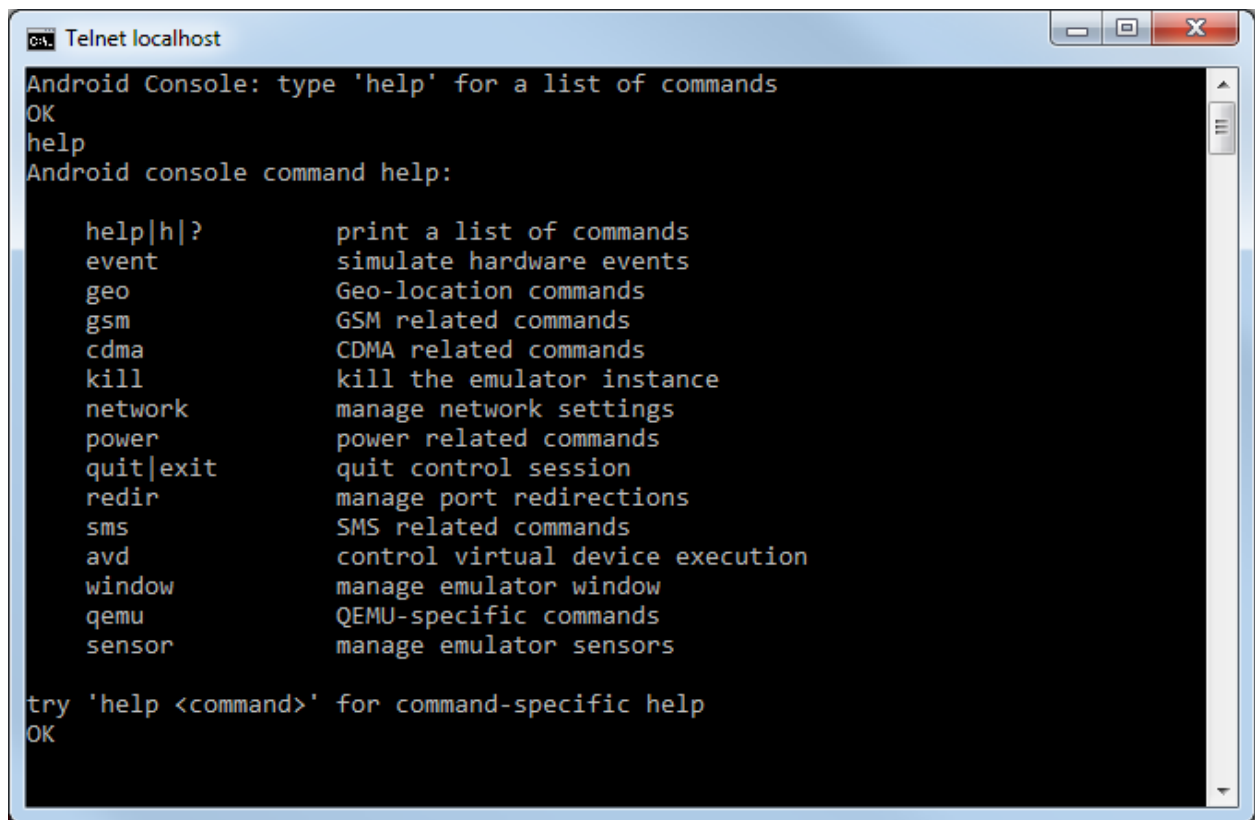
```
C:\windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
(c) Корпорация Майкрософт (Microsoft Corp.), 2009. Все права защищены.

c:\MySpace\fdo\adt-bundle\eclipse>telnet localhost 5554_
```



```
Telnet localhost
Android Console: type 'help' for a list of commands
OK
```

Можете посмотреть список команд, доступных в android console, набрав команду help

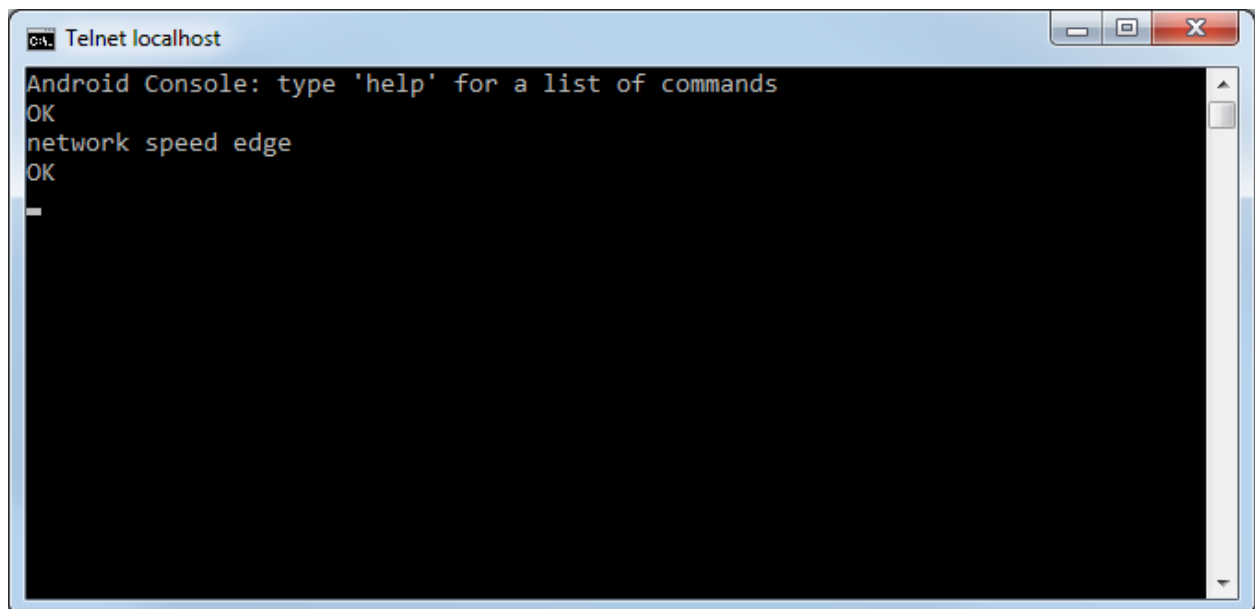


```
Android Console: type 'help' for a list of commands
OK
help
Android console command help:

    help|h|?      print a list of commands
    event         simulate hardware events
    geo           Geo-location commands
    gsm           GSM related commands
    cdma          CDMA related commands
    kill          kill the emulator instance
    network       manage network settings
    power         power related commands
    quit|exit     quit control session
    redir         manage port redirections
    sms           SMS related commands
    avd           control virtual device execution
    window        manage emulator window
    qemu          QEMU-specific commands
    sensor        manage emulator sensors

try 'help <command>' for command-specific help
OK
```

Для того, чтобы задать скорость интернет соединения, введите `network speed edge`. В ответ получите OK.



```
Android Console: type 'help' for a list of commands
OK
network speed edge
OK
_
```

Вернитесь в эмулятор и откройте браузер. Замерьте сколько времени требуется для того, чтобы загрузить страницу google.com. Измерения не должны быть абсолютно точными. После каждой загрузки страницы для чистоты эксперимента очищайте кэш браузера. Settings > Privacy&Security > Clear cache

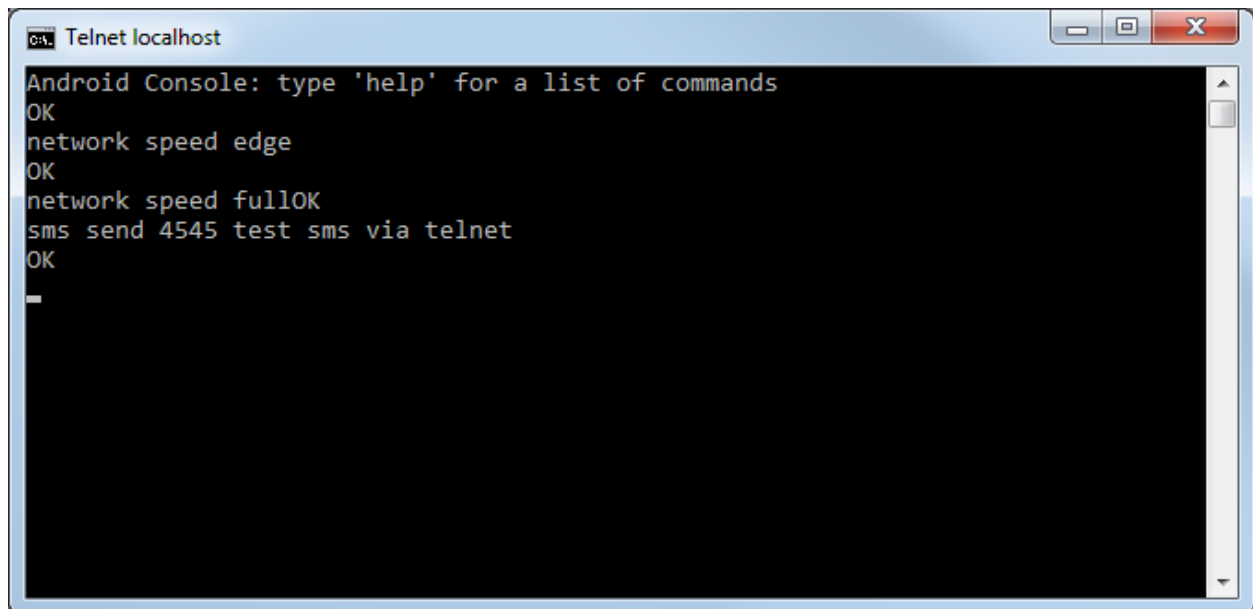
Возвратитесь в командную строку и измените скорость интернета на подключение без ограничений. Введите network speed full.

Снова возвратитесь в эмулятор и замерьте скорость загрузки страницы google.com, на этот раз на полной скорости интернета, доступной вам.

Действительно ли эмулятор изменяет скорость загрузки страниц при различных настройках?

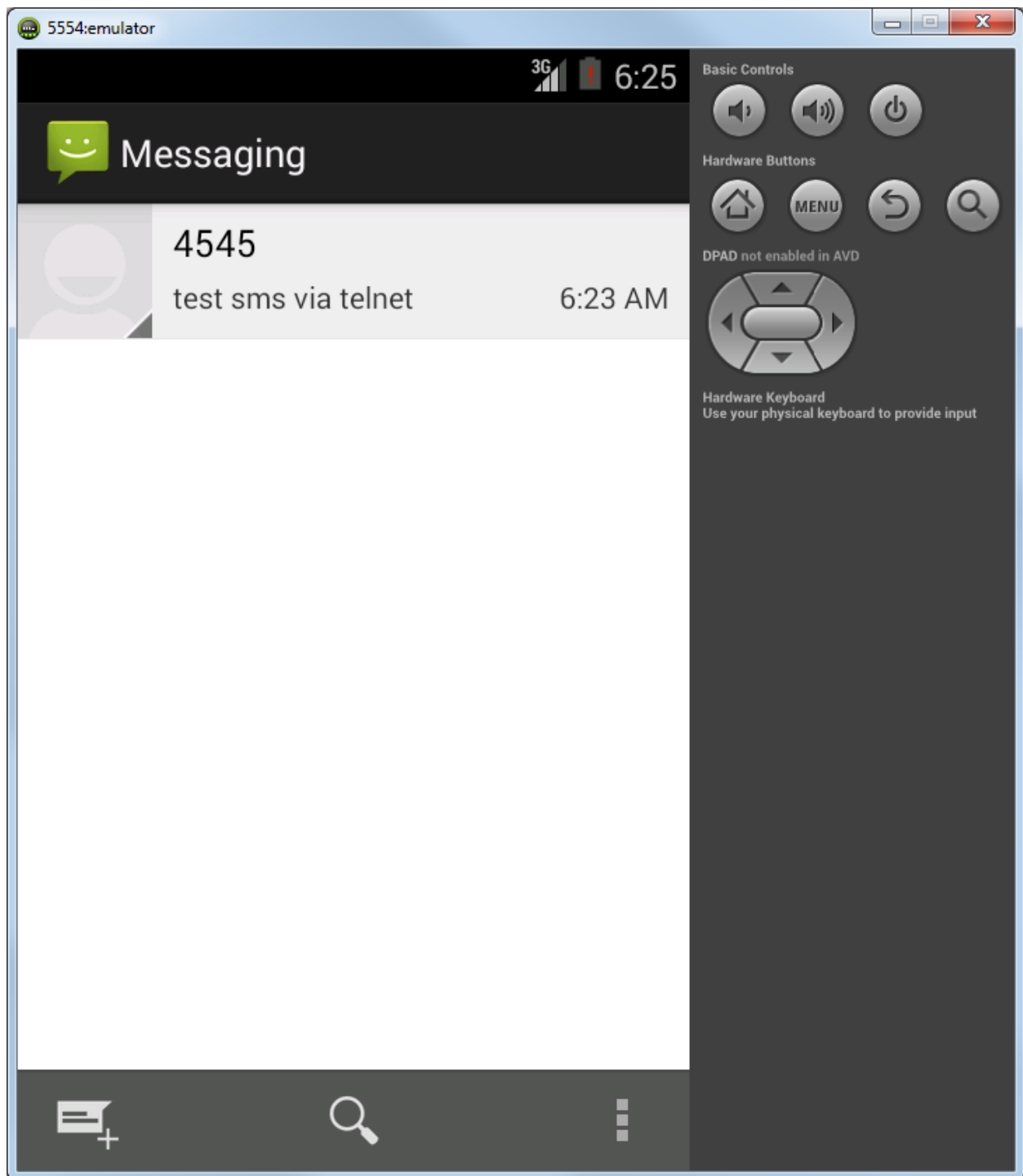
3. Посылка SMS

В этой части практики вы попробуете послать SMS вашему эмулятору с помощью командной строки. Возвратитесь к командной строке и введите: "sms send <sender's phone number> <message>"

A screenshot of a Telnet window titled 'Telnet localhost'. The window has a black background with white text. The text shows a sequence of commands and responses: 'Android Console: type \'help\' for a list of commands', 'OK', 'network speed edge', 'OK', 'network speed full', 'OK', and 'sms send 4545 test sms via telnet', 'OK'. There is a small cursor at the bottom left.

```
Telnet localhost
Android Console: type 'help' for a list of commands
OK
network speed edge
OK
network speed full
OK
sms send 4545 test sms via telnet
OK
_
```

В эмуляторе придет СМС от номера, который вы ввели с текстом, который вы ввели.

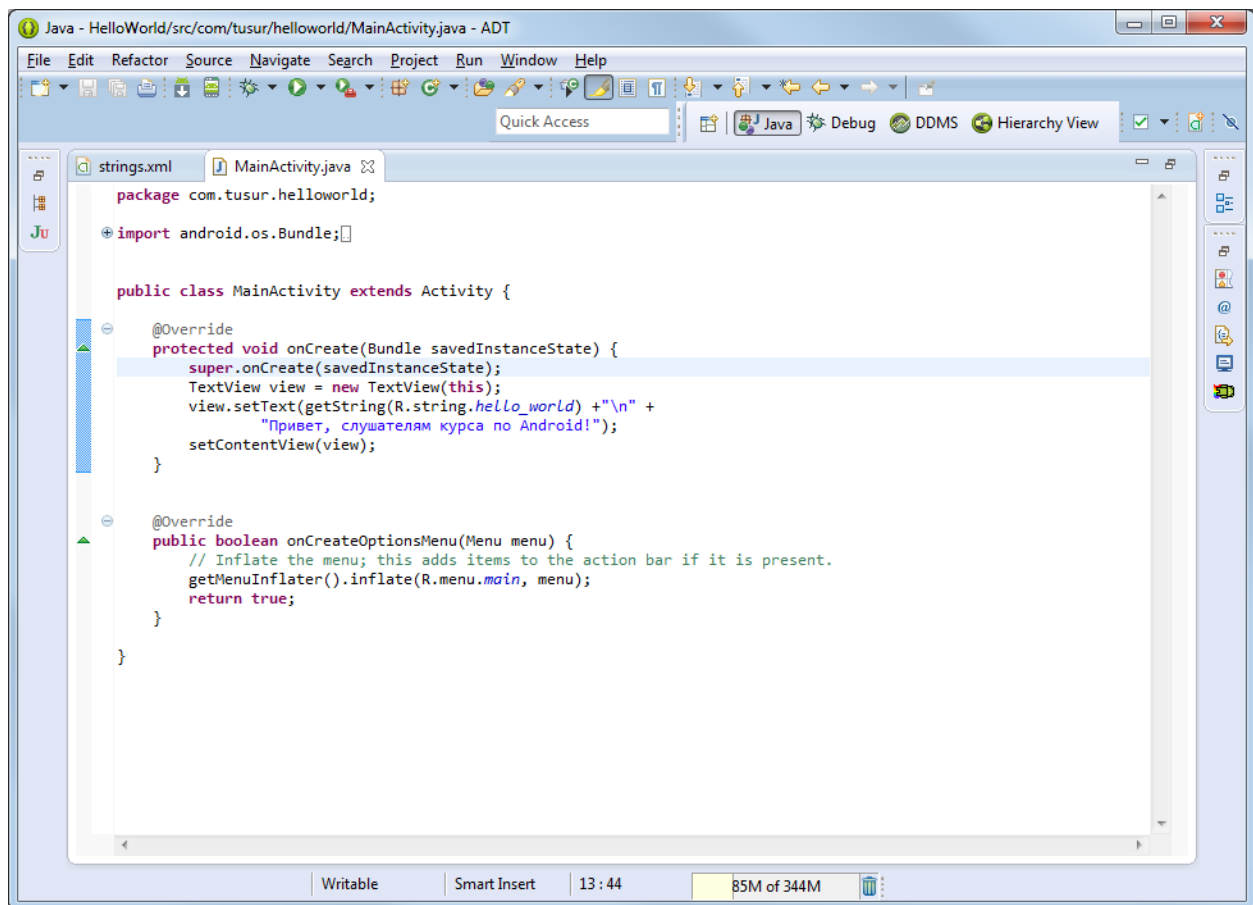


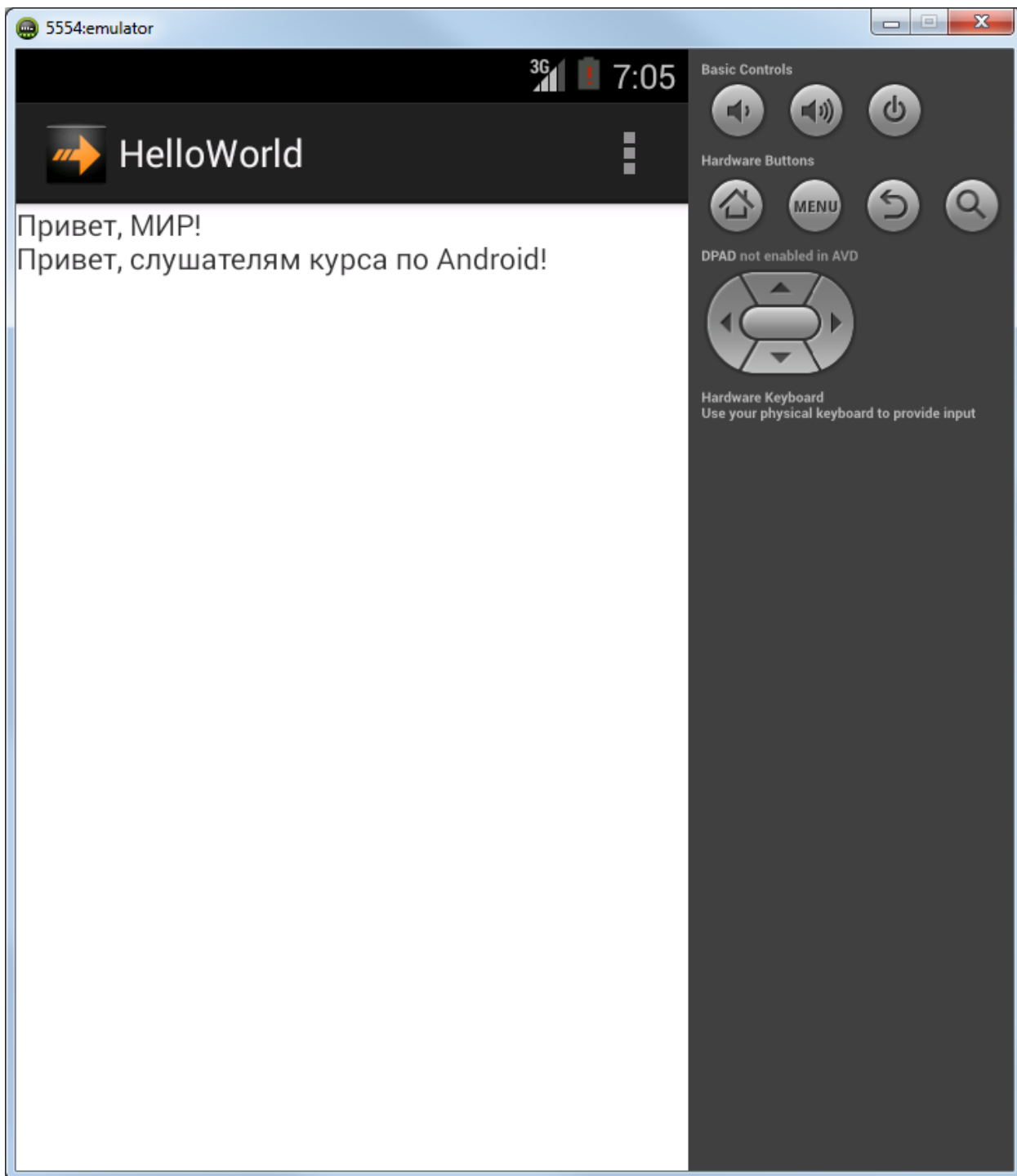
4. Поддержка мультиязычности

Откроем наше первое приложение HelloWorld. Найдем файл ресурсов со строками. Добавим поддержку русского языка. Для этого создадим новый файл ресурсов для русского языка и создадим наше приветственное сообщение. Например «Привет, МИР!» В каталоге res/ создадим дополнительную папку values-ru/ — каталог для русской локализованной версии. В этот каталог мы поместим локализованную версию файла строковых ресурсов с нашим переводом.

Не забудем изменить языковые настройки эмулятора. И немного изменим код проекта. Добавим вот такую строку:

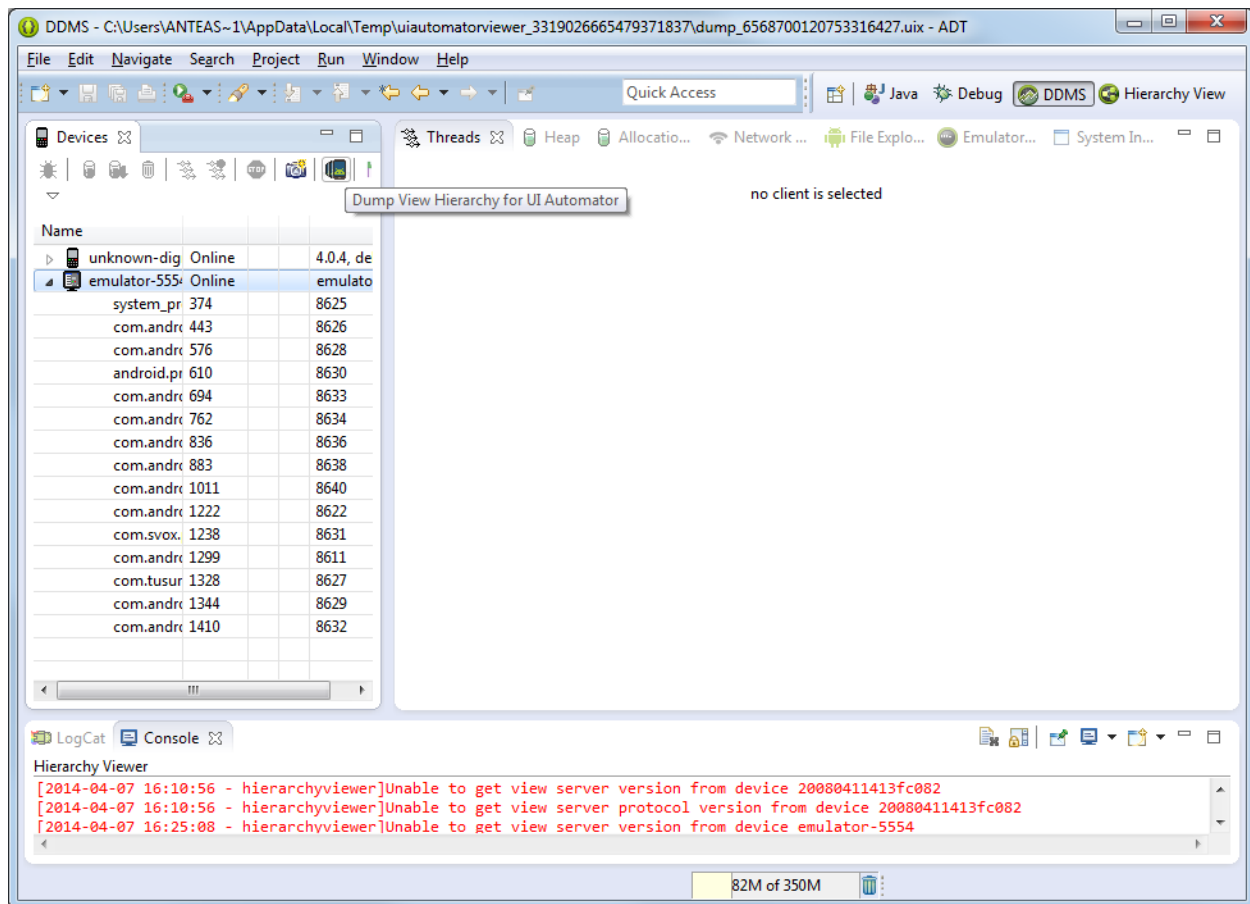
```
view.setText(getString(R.string.hello_world) + "\n" +  
    "Привет, слушателям курса по Android!");
```

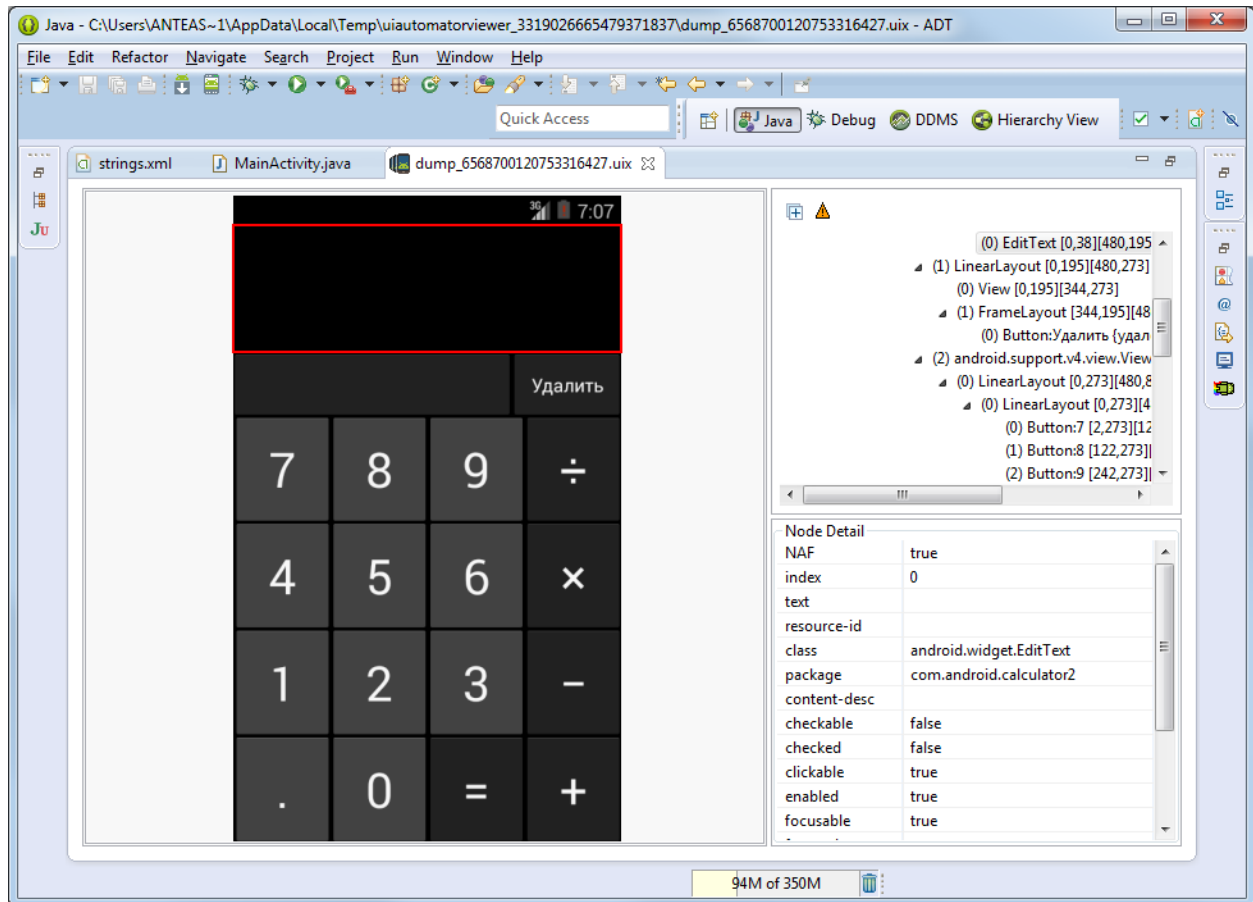




5. HierarchyView tool

Воспользуйтесь HierarchyView tool или Dump View Hierarchy в DDMS для отображения и проверки пользовательского интерфейса стандартного приложения калькулятора вашего эмулятора. Для того, чтобы сделать это, запустите приложение стандартного калькулятора. Далее нажмите кнопку дампа.



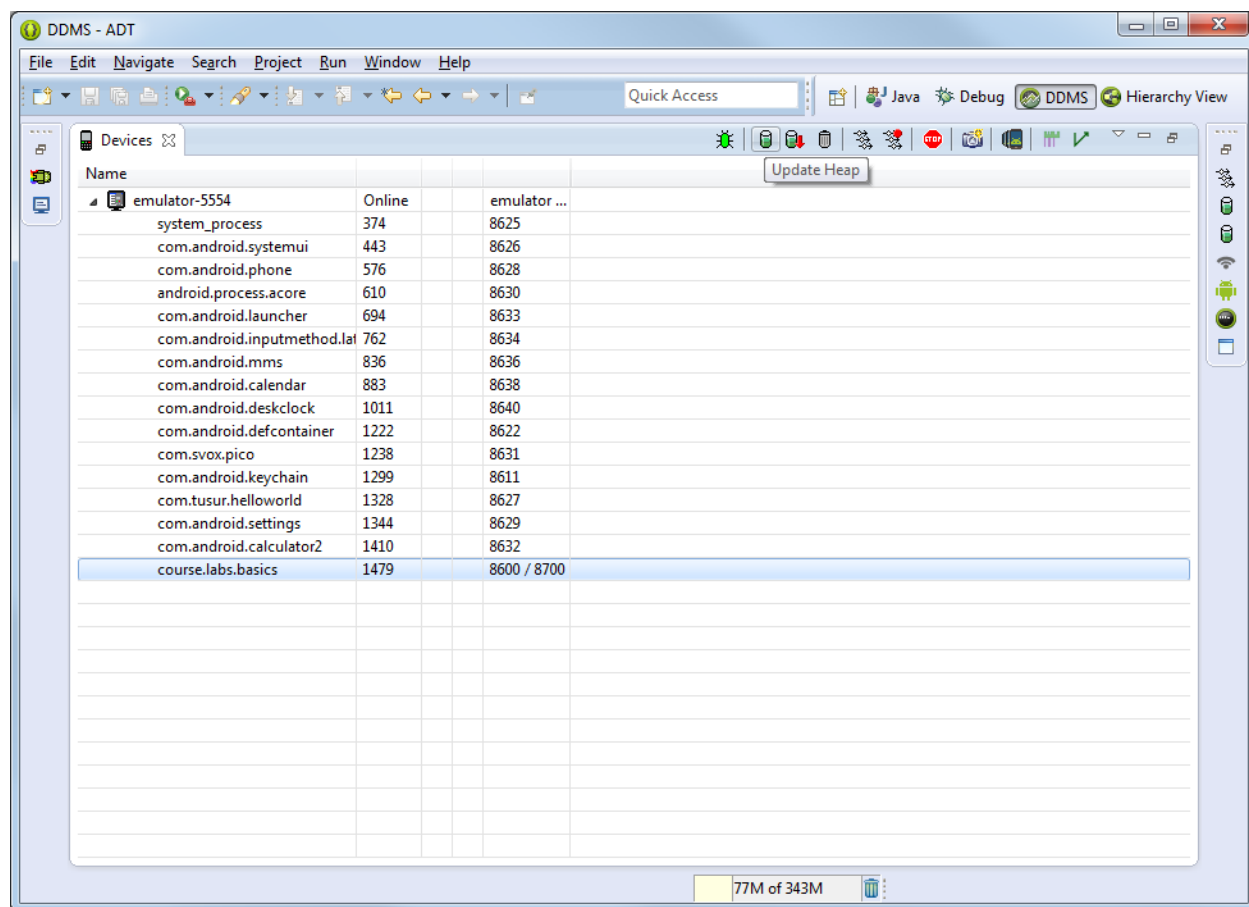


Попробуйте ответить на следующие вопросы с помощью полученного дампа:

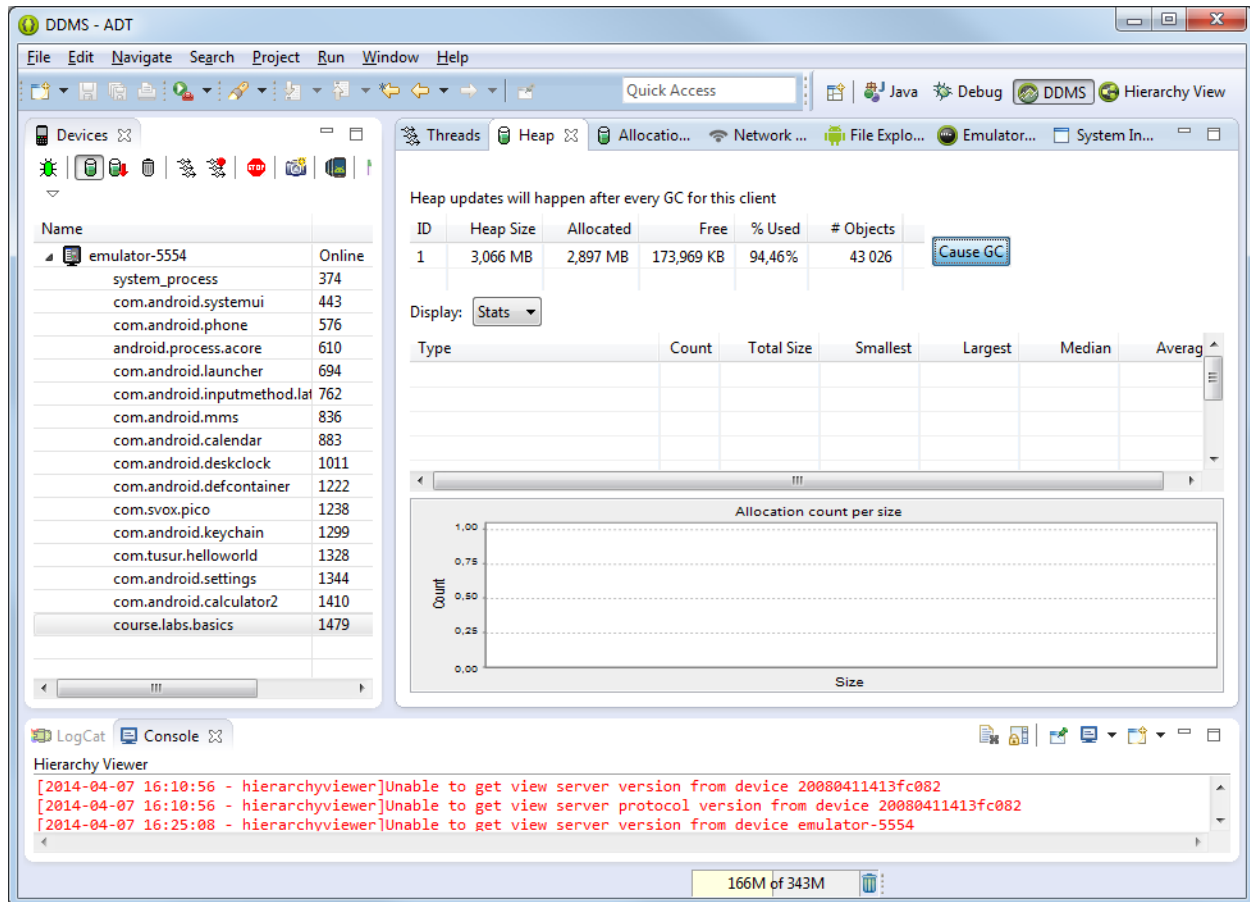
- Какая высота дерева View (сколько вложенных уровней в иерархии дерева)
- Проверьте View, показывающий кнопку «+». Какого типа View его родитель? Сколько потомков в родительском View? Какой относительный индекс кнопки «+» внутри родительского View.

6. Heap View

Воспользуемся еще раз нашим приложением BasicsLab. Для того, чтобы посмотреть использование памяти в куче, перейдем на перспективу DDMS. Выберем интересующее нас приложение, загруженное в память. И нажмем «update heap».



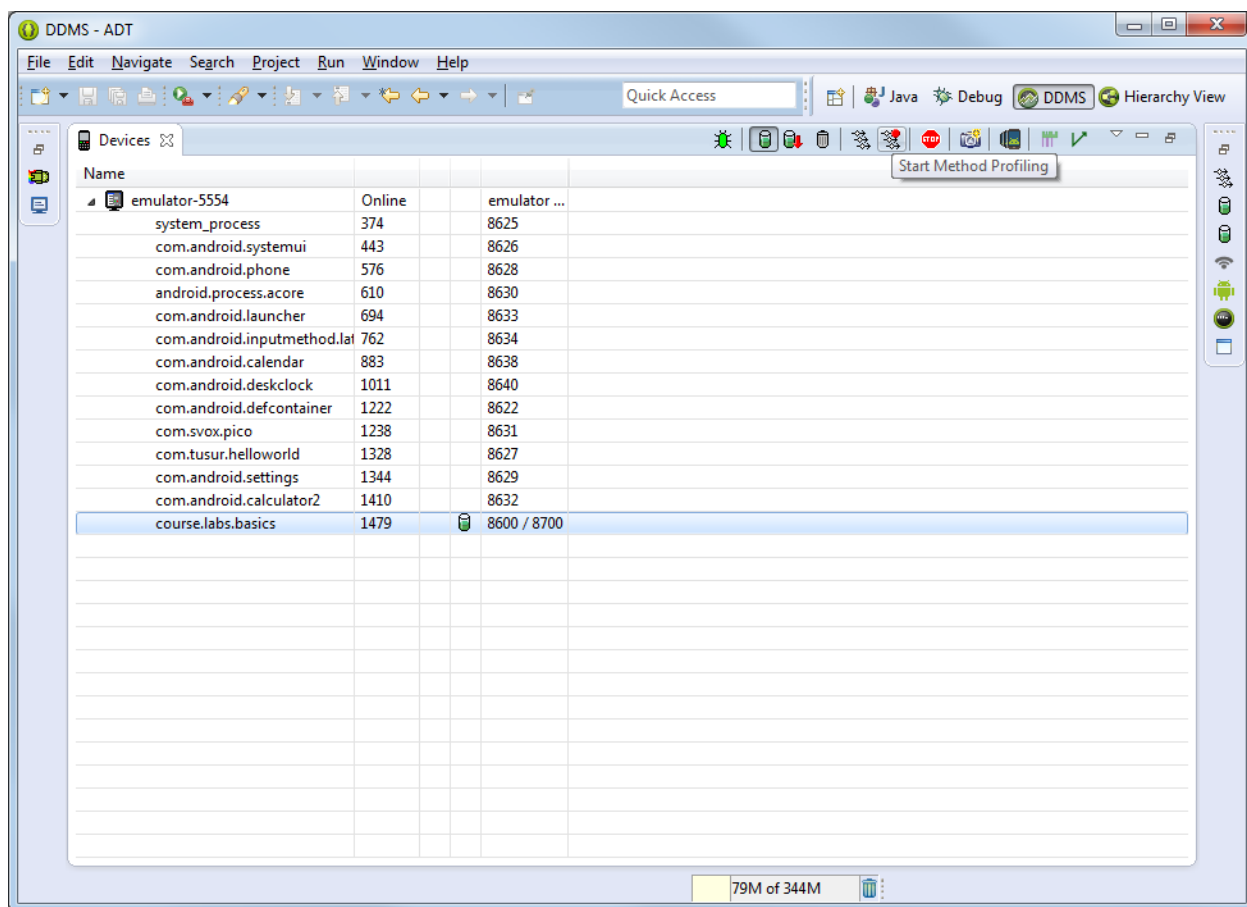
На вкладке «Heap» нажмем кнопку «Cause GC» для запуска сборщика мусора.



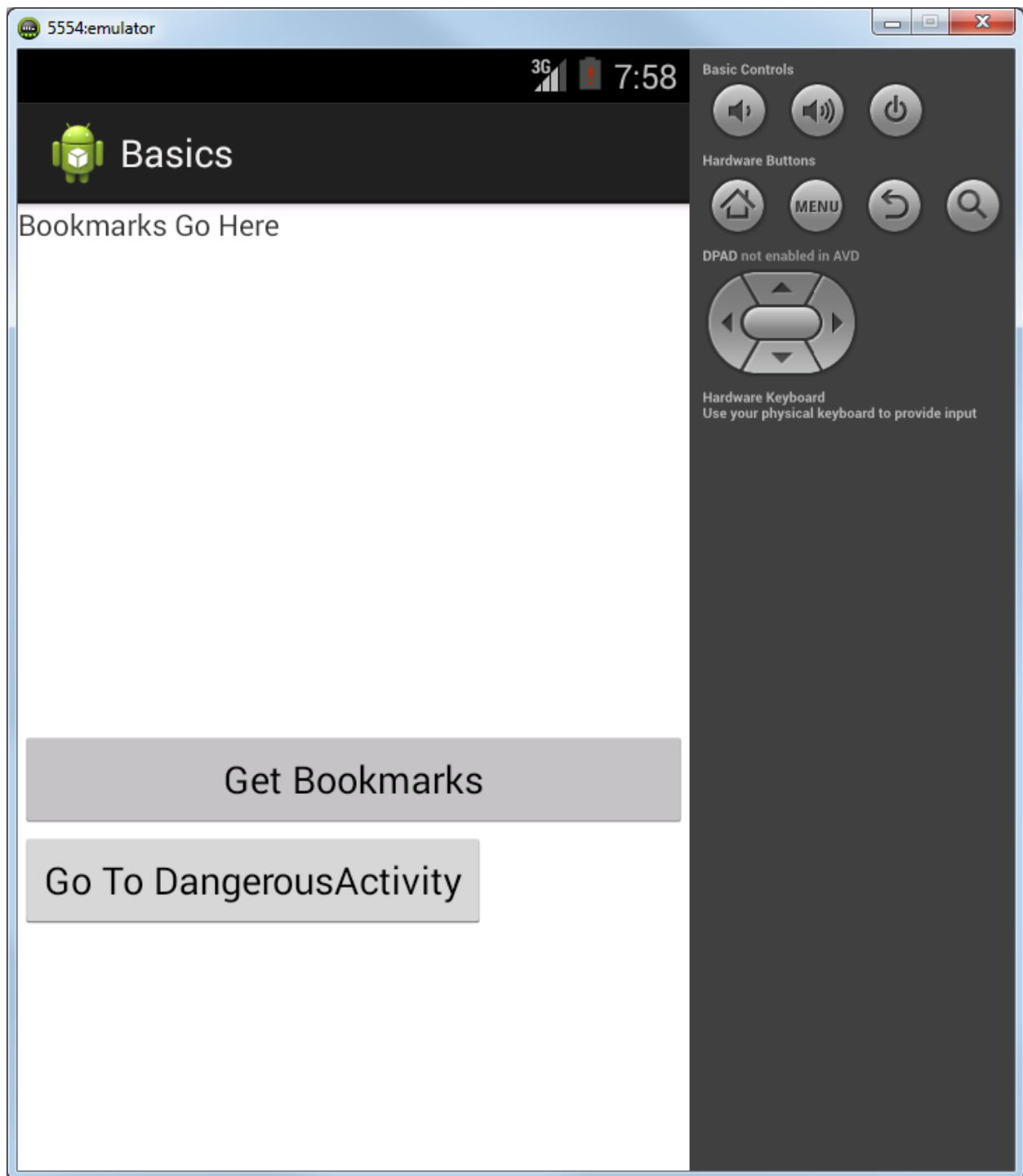
- Как много объектов было создано?
- Какой тип объектов подвергся удалению больше?
- Сколько памяти было освобождено?

7. Профилирование методов

Воспользуемся DDMS для просмотра последовательности вызова методов для BasicsLab. Для этого на вкладке «device» выберем нужный нам процесс и нажмем «Start Method Profiling»



По взаимодействию с нашим приложением.



Нажимаем «Stop Method profiling». Вы увидите что-то на подобии следующего экрана:

DDMS - ADT

File Edit Navigate Search Project Run Window Help

Quick Access

Java Debug DDMS Hierarchy View

Devices

Name	Online	emulator ...
emulator-5554	Online	emulator ...
system_process	374	8625
com.android.systemui	443	8626
com.android.phone	576	8628
android.process.acore	610	8630
com.android.launcher	694	8633
com.android.inputmethod.la	762	8634
com.android.calendar	883	8638
com.android.deskclock	1011	8640
com.android.defcontainer	1222	8622
com.svox.pico	1238	8631
com.tusur.helloworld	1328	8627
com.android.settings	1344	8629
com.android.calculator2	1410	8632
course.labs.basics	1479	8600 / 8700
com.android.browser	1535	8601

Stop Method Profiling

78M of 343M

DDMS - C:\Users\ANTEAS-I\AppData\Local\Temp\ddms4519965718471647853.trace - ADT

File Edit Navigate Search Project Run Window Help

Quick Access

Java Debug DDMS Hierarchy View

Devices

Name	Online	emulator ...
emulator-5554	Online	emulator ...
system_process	374	8625
com.android.systemui	443	8626
com.android.phone	576	8628
android.process.acore	610	8630
com.android.launcher	694	8633
com.android.inputmethod.la	762	8634
com.android.calendar	883	8638
com.android.deskclock	1011	8640
com.android.defcontainer	1222	8622
com.svox.pico	1238	8631
com.tusur.helloworld	1328	8627
com.android.settings	1344	8629
com.android.calculator2	1410	8632
course.labs.basics	1479	8600 / 8700
com.android.browser	1535	8601

ms:0 max msec: 250,000 (real time, dual clock)

[1] main

[9] Binder_1

[10] Binder_2

[11] Binder_3

[4] JDWP

Name	Incl Cpu Time %	Incl Cpu Time	Excl Cpu %
0 (toplevel)	100.0%	2020,000	
1 android.os.Handler.dispatchMessage (Landroid/os/Message;	84.2%	1700,000	
2 android.os.Handler.handleCallback (Landroid/os/Message;V	42.1%	870,000	
3 android.app.ActivityThread\$H.handleMessage (Landroid/os/I	35.6%	720,000	
4 android.view.Choreographer\$FrameDisplayEventReceiver.run	35.6%	720,000	
5 android.view.Choreographer.doFrame (J)V	35.6%	720,000	
6 android.view.Choreographer\$CallbackRecord.run (J)V	35.1%	710,000	
7 android.view.Choreographer.doCallbacks (J)V	35.1%	710,000	
8 android.app.ActivityThread.access\$3700 (Landroid/app/Activit	34.2%	690,000	
9 android.app.ActivityThread.handleLaunchActivity (Landroid/I	34.2%	690,000	
10 android.view.ViewRootImpl\$1TraverseRunnable.run (J)	32.2%	650,000	
11 android.view.ViewRootImpl.doTraverse (J)	32.2%	650,000	
12 android.view.ViewRootImpl.performTraverse (J)	32.2%	650,000	
13 android.app.ActivityThread.performLaunchActivity (Landroi	31.2%	630,000	
14 android.app.Activity.performCreate (Landroid/os/Bundle;V	30.2%	610,000	

Start Tracking Get Allocations Filter: Inc. trace

Alloc Order Allocated Class T... Allo... Allo...

LogCat Console

Hierarchy Viewer

[2014-04-07 16:10:56 - hierarchyviewer]Unable to get view server version from device 200808411413fc082

[2014-04-07 16:10:56 - hierarchyviewer]Unable to get view server protocol version from device 200808411413fc082

[2014-04-07 16:25:08 - hierarchyviewer]Unable to get view server protocol version from device emulator-5554

[2014-04-07 16:25:08 - hierarchyviewer]Unable to get view server protocol version from device emulator-5554

[2014-04-07 16:25:09 - ViewServerDevice]Unable to debug device: emulator-5554

[2014-04-07 16:25:09 - hierarchyviewer]Missing forwarded port for emulator-5554

[2014-04-07 16:25:09 - hierarchyviewer]Unable to get the focused window from device emulator-5554

150M of 343M

- a. Найдите метод из вашего приложения на панели профиля.
- b. Какого его наименование?
- c. Есть ли исходники для этого метода?
- d. Какого время потраченное на выполнение метода?