

Оглавление

Благодарности	20
Изучение Android	22
Подготовка	22
Что нового в четвертом издании?.....	23
Kotlin vs Java	24
Как работать с книгой	24
Структура книги	25
Упражнения	26
А вы любознательны?.....	26
Типографские соглашения.....	26
Версии Android	27
От издательства	27
Необходимые инструменты	28
Загрузка и установка Android Studio.....	28
Загрузка старых версий SDK	29
Аппаратное обеспечение	30
Глава 1. Первое Android-приложение	31
Основы разработки приложения.....	31
Создание проекта Android.....	32
Навигация в Android Studio	37
Создание макета пользовательского интерфейса	39
Иерархия представления.....	44
Атрибуты виджетов	45
Создание строковых ресурсов	46
Предварительный просмотр макета.....	46

От разметки XML к объектам View	48
Ресурсы и идентификаторы ресурсов	50
Разработка виджетов	53
Установка ссылок на виджеты	53
Назначение слушателей	55
Уведомления	56
Выполнение в эмуляторе	58
Для любознательных: процесс сборки Android-приложений.....	62
Инструменты сборки	62
Упражнения.....	64
Упражнение. Настройка уведомления.....	64
Глава 2. Android и модель MVC.....	65
Создание нового класса.....	66
Архитектура «Модель — Представление — Контроллер» и Android.....	67
Преимущества MVC	68
Обновление уровня представления	70
Обновление уровня контроллера.....	73
Добавление значка.....	77
Добавление ресурсов в проект	78
Ссылки на ресурсы в XML	80
Плотность пикселей.....	81
Запуск на устройстве.....	83
Упражнение. Добавление слушателя для TextView	85
Упражнение. Добавление кнопки возврата	85
Упражнение. От Button к ImageButton	86
Глава 3. Жизненный цикл activity	88
Поворот GeoQuiz	88
Обратные вызовы состояния и жизненного цикла activity	90
Регистрация событий жизненного цикла activity	92
Создание сообщений в журнале	92
Панель LogCat	94

Анализ жизненного цикла activity на примере.....	96
Временная остановка activity.....	96
Завершение activity.....	99
Поворот activity.....	100
Конфигурации устройств и жизненный цикл ресурса.....	101
Создание макета для альбомной ориентации.....	101
Для любознательных: обновление UI и мультиоконный режим.....	105
Для любознательных: уровни регистрации.....	106
Упражнение. Предотвращение ввода нескольких ответов.....	107
Упражнение. Вывод оценки.....	107
Глава 4. Сохранение состояния интерфейса.....	108
Добавление зависимостей ViewModel.....	108
Добавление ViewModel.....	110
Жизненный цикл ViewModel и ViewModelProvider.....	112
Добавление данных в ViewModel.....	116
Сохранение данных после завершения процесса.....	119
Переопределение onSaveInstanceState(Bundle).....	121
Сохраненное состояние экземпляра и записи activity.....	123
Сравнение ViewModel и сохраненного состояния экземпляра.....	125
Для любознательных: компоненты Jetpack, AndroidX и компоненты архитектуры ...	127
Для любознательных: как избежать костылей.....	128
Глава 5. Отладка Android-приложений.....	130
Исключения и трассировка стека.....	131
Диагностика ошибок поведения.....	133
Сохранение трассировки стека.....	134
Установка точек останова.....	135
Особенности отладки Android.....	139
Android Lint.....	140
Проблемы с классом R.....	143
Упражнение. Исследуем Layout Inspector.....	144
Упражнение. Profiler.....	145

Глава 6. Вторая activity.....	146
Подготовка к внедрению второй activity	146
Создание второй activity	147
Создание подкласса новой activity.....	150
Объявление activity в манифесте.....	151
Добавление кнопки CHEAT! в MainActivity	152
Запуск activity.....	154
Передача информации через интенды.....	154
Передача данных между activity	156
Дополнения интендов.....	157
Получение результата от дочерней activity.....	160
Ваши activity с точки зрения Android.....	165
Упражнение. Лазейка для читера	168
Упражнение. Отслеживание читов по вопросу.....	168
Глава 7. Версии Android SDK и совместимость	169
Версии Android SDK	169
Совместимость и программирование Android	171
Разумный минимум.....	171
Минимальная версия SDK.....	172
Целевая версия SDK.....	173
Версия SDK для компиляции	173
Безопасное добавление кода для более поздних версий API	174
Jetpack-библиотеки	177
Документация разработчика Android	178
Упражнение. Вывод версии Android на устройстве.....	181
Упражнение. Ограничение подсказок	182
Глава 8. UI-фрагменты и FragmentManager.....	183
Гибкость пользовательского интерфейса.....	183
Знакомство с фрагментами.....	185
Начало работы над CriminalIntent	186
Создание нового проекта	188
Создание класса Crime.....	189

Создание UI-фрагмента	191
Определение макета CrimeFragment	191
Создание класса CrimeFragment	193
Хостинг UI-фрагментов	201
Определение контейнерного представления	202
Добавление UI-фрагмента во FragmentManager	203
Архитектура приложений с фрагментами	209
Нужно ли использовать фрагменты	210
Глава 9. Вывод списков и RecyclerView	211
Добавление нового фрагмента и ViewModel	212
Жизненный цикл ViewModel с фрагментами	214
Добавление RecyclerView	215
Создание макета представления элемента	218
Реализация ViewHolder	220
Реализация адаптера для заполнения RecyclerView	222
Настройка адаптера RecyclerView	225
Переработка представлений	227
Очистка элементов связанного списка	228
Отклик на нажатия	229
Для любознательных: ListView и GridView	230
Упражнение. Типы View в RecyclerView	231
Глава 10. Создание интерфейсов с использованием макетов и виджетов	232
Знакомство с ConstraintLayout	233
Использование графического конструктора макетов	234
Использование ConstraintLayout	238
Освобождение пространства	239
Добавление виджетов	241
Внутренние механизмы ConstraintLayout	245
Редактирование свойств	246
Динамическое поведение элементов списка	251
Подробнее об атрибутах макетов	252
Стили, темы и атрибуты тем	252

Для любознательных: поля и отступы	254
Для любознательных: нововведения в ConstraintLayout	256
Упражнение. Форматирование даты	257
Глава 11. Базы данных и Room Library	258
Библиотека компонентов архитектуры Room	259
Создание базы данных	260
Определение сущностей.....	260
Создание класса базы данных	261
Определение объекта доступа к данным	264
Доступ к базе данных с помощью шаблона репозитория	265
Тестирование запросов	269
Загрузка тестовых данных.....	270
Потоки приложения.....	273
Фоновые потоки	274
Использование LiveData.....	276
Наблюдение за LiveData	277
Упражнение. Ошибка доступа к схеме	281
Для любознательных: синглтоны.....	282
Глава 12. Навигация по фрагментам	284
Одиночная activity: главный фрагмент.....	285
Интерфейсы обратного вызова фрагментов	286
Замена фрагмента.....	289
Аргументы фрагментов.....	291
Присоединение аргументов к фрагменту	292
Получение аргументов	293
Преобразования LiveData.....	294
Обновление базы данных	298
Использование исполнителя.....	300
Привязка записи в базу данных к жизненному циклу фрагмента	301
Для любознательных: зачем использовать аргументы фрагментов?	302
Для любознательных: библиотека компонентов архитектуры навигации	303
Упражнение. Эффективная перезагрузка RecyclerView	305

Глава 13. Диалоговые окна	306
Создание экземпляра DialogFragment	307
Отображение DialogFragment	309
Передача данных между фрагментами	311
Передача данных в DatePickerFragment	312
Возвращение данных CrimeFragment	314
Упражнение. Новые диалоговые окна.....	317
Глава 14. Панель приложения	318
Панель приложения AppCompat.....	319
Меню.....	320
Определение меню в XML.....	321
Создание меню.....	323
Реакция на выбор команд.....	325
Использование Android Asset Studio	328
Для любознательных: панель приложения/действий/инструментов.....	331
Для любознательных: доступ к AppCompat панели приложения	333
Упражнение. Пустое представление для списка.....	334
Глава 15. Неявные интенты	335
Добавление кнопок	336
Добавление подозреваемого в уровень модели	337
Форматные строки.....	339
Использование неявных интентов	341
Строение неявного интента.....	341
Отправка отчета.....	342
Запрос контакта у Android	346
Проверка реагирующих activity.....	352
Упражнение. Другой неявный интент	354
Глава 16. Интенты при работе с камерой	356
Размещение фотографий.....	356
Хранилище файлов.....	360
Использование FileProvider	361

Выбор локации фотографии	363
Использование интента камеры.....	364
Отправка интента.....	365
Масштабирование и отображение растровых изображений	368
Объявление функциональности.....	372
Упражнение. Вывод увеличенного изображения.....	374
Упражнение. Эффективная загрузка миниатюры	374
Глава 17. Локализация.....	375
Локализация ресурсов.....	376
Ресурсы по умолчанию	379
Проверка покрытия локализации в Translations Editor	380
Региональная локализация.....	381
Конфигурационные квалификаторы	384
Приоритеты альтернативных ресурсов	385
Множественные квалификаторы	387
Поиск наиболее подходящих ресурсов	388
Тестирование альтернативных ресурсов.....	390
Для любознательных: подробнее об определении размеров экрана устройства	390
Упражнение. Локализация дат.....	391
Глава 18. Специальные возможности	392
Приложение TalkBack.....	392
Функция Explore by Touch	395
Линейная навигация смахиванием	396
Чтение нетекстовых элементов	398
Добавление описаний к контенту	399
Фокусировка на виджетах	401
Создание сопоставимого опыта взаимодействия.....	402
Для любознательных: Accessibility Scanner.....	405
Упражнение. Улучшение списка	408
Упражнение. Предоставление контекста для ввода данных.....	408
Упражнение. Оповещения о событиях	409

Глава 19. Привязка данных и MVVM.....	410
Другие архитектуры: зачем они нужны?	411
Сравнение моделей представления MVVM и Jetpack-класса ViewModel.....	412
Создание приложения BeatBox	412
Простая привязка данных.....	413
Импорт активов.....	417
Доступ к активам.....	420
Подключение активов для использования	422
Установление связи с данными.....	425
Создание модели представления.....	427
Связывание с моделью представления	428
Отслеживаемые данные	430
Для любознательных: подробнее о привязке данных	433
Лямбда-выражения	433
Синтаксический сахар	433
BindingAdapter	434
Для любознательных: LiveData и привязка данных	435
Глава 20. Модульное тестирование и воспроизведение звуков	437
Создание объекта SoundPool.....	437
Доступ к активам.....	438
Загрузка звуков	439
Воспроизведение звуков.....	440
Зависимости при тестировании.....	441
Создание класса теста	442
Подготовка теста.....	444
Настройка тестируемых объектов.....	445
Написание тестов	446
Взаимодействия тестируемых объектов	447
Обратные вызовы привязки данных	453
Выгрузка звуков	454
Для любознательных: интеграционное тестирование	455

Для любознательных: фиктивные объекты и тестирование	457
Упражнение. Управление скоростью воспроизведения	458
Упражнение. Воспроизведение звуков после поворота	459
Глава 21. Стили и темы	460
Цветовые ресурсы	460
Стили	461
Наследование стилей	462
Темы	464
Изменение темы.....	465
Добавление цветов в тему.....	466
Переопределение атрибутов темы.....	468
Исследование тем	469
Изменение атрибутов кнопки	473
Для любознательных: подробнее о наследовании стилей.....	476
Для любознательных: доступ к атрибутам тем	478
Глава 22. Графические объекты	479
Унификация кнопок.....	480
Геометрические фигуры	481
Списки состояний	483
Списки слоев	485
Для любознательных: для чего нужны графические объекты XML.....	486
Для любознательных: Miramar	487
Для любознательных: девятизонные изображения.....	488
Упражнение. Темы кнопок.....	494
Глава 23. Подробнее об интентах и задачах	496
Создание приложения NerdLauncher.....	497
Обработка неявного интента	498
Создание явных интентов на стадии выполнения	503
Задачи и стек возврата.....	506
Переключение между задачами	507

Запуск новой задачи	508
Использование NerdLauncher в качестве главного экрана.....	511
Для любознательных: процессы и задачи.....	513
Для любознательных: параллельные документы	516
Упражнение. Значки.....	518
Глава 24. HTTP и фоновые задачи	519
Создание приложения PhotoGallery.....	520
Основы работы с сетью при помощи Retrofit.....	524
Определение интерфейса API	524
Сборка объекта Retrofit и создание экземпляра API.....	526
Выполнение веб-запроса.....	528
Запрос разрешения на работу с сетью.....	530
Переход к шаблону репозитория.....	531
Получение JSON-данных от Flickr.....	535
Десериализация JSON-текста в объекты модели.....	539
Работа с сетью после изменения конфигурации	545
Отображение результатов в RecyclerView	549
Для любознательных: альтернативные парсеры и форматы данных.....	550
Для любознательных: отмена запросов	551
Для любознательных: управление зависимостями.....	552
Упражнение. Добавление пользовательского десериализатора Gson.....	554
Упражнение. Навигация по страницам.....	555
Упражнение. Динамическая настройка количества столбцов.....	556
Глава 25. Классы Looper, Handler и HandlerThread	557
Подготовка RecyclerView к выводу изображений.....	557
Подготовка к загрузке через URL.....	560
Множественные загрузки.....	562
Создание фонового потока	563
Передача информации о жизненном цикле в поток.....	563
Запуск и остановка HandlerThread	567
Сообщения и обработчики сообщений.....	569

Структура сообщения	571
Структура обработчика	571
Использование обработчиков	573
Передача обработчиков	577
Прослушивание жизненного цикла представления	581
Сохраненные фрагменты	585
Поворот и сохранение фрагментов	586
Выполнять ли сохранение	588
Для любознательных: решение задачи загрузки изображений	589
Для любознательных: StrictMode	590
Упражнение. Наблюдение LiveData у LifecycleOwner	591
Упражнение. Больше информации о жизненном цикле ThumbnailDownloader	591
Упражнение. Предварительная загрузка и кэширование	592
Глава 26. Поиск: SearchView и SharedPreferences	593
Поиск во Flickr	593
Использование SearchView	598
Реакция SearchView на действия пользователя	600
Простое сохранение с помощью SharedPreferences	604
Последний штрих	608
Редактирование SharedPreferences с помощью Android KTX	609
Упражнение. Еще одно усовершенствование	610
Глава 27. Библиотека WorkManager	611
Создание класса Worker	611
Планирование работы	612
Проверка на наличие новых фотографий	616
Вывод уведомления пользователю	619
Управление опросом от пользователя	625
Глава 28. Широковещательные интенты	631
Обычные и широковещательные интенты	631
Отключение уведомлений при открытом приложении	632

Отправка широковещательных интентов	633
Создание и регистрация автономного широковещательного приемника	634
Ограничение широковещательной рассылки	635
Создание и регистрация динамического приемника	639
Передача и получение данных с упорядоченной широковещательной рассылкой	641
Приемники и продолжительные задачи	646
Для любознательных: локальные события	647
Использование EventBus	647
Использование RxJava	648
Для любознательных: ограничения широковещательных приемников	649
Для любознательных: проверка видимости фрагмента	650
Глава 29. Веб-серфинг и WebView	652
И еще один фрагмент данных Flickr	653
Простой способ: неявные интенты	654
Сложный способ: WebView	656
Класс WebChromeClient	661
Повороты в WebView	663
Опасности при обработке изменений конфигурации	665
Сравнение WebView и пользовательского интерфейса	665
Для любознательных: внедрение объектов JavaScript	666
Для любознательных: обновления WebView	667
Упражнение. Пользовательские вкладки Chrome (еще один простой способ)	668
Упражнение. Использование кнопки «Назад» для работы с историей просмотра	670
Глава 30. Пользовательские представления и события касания	671
Создание проекта DragAndDraw	671
Создание пользовательского представления	671
Создание класса BoxDrawingView	673
Обработка событий касания	675
Отслеживание событий перемещения	676
Рендеринг внутри onDraw(Canvas)	679
Для любознательных: GestureDetector	681

Упражнение. Сохранение состояния	682
Упражнение. Повороты прямоугольников	682
Упражнение. Поддержка специальных возможностей	683
Глава 31. Анимация свойств	684
Построение сцены	684
Простая анимация свойств	686
Свойства преобразований	690
Выбор интерполятора	690
Изменение цвета	692
Одновременное воспроизведение анимаций	694
Для любознательных: другие API анимации	696
Устаревшие средства анимации	696
Переходы	696
Упражнения	697
Послесловие	698
Последнее упражнение	698
Спасибо вам!	699

Анализ жизненного цикла activity на примере

Как видите, после запуска GeoQuiz и создания исходного экземпляра MainActivity были вызваны три функции жизненного цикла: onCreate(Bundle?), onStart() и onResume() (рис. 3.7). Экземпляр MainActivity после этого находится в состоянии выполнения (он находится в памяти, виден пользователю и находится на переднем плане).

По мере изучения этой книги вы будете переопределять различные функции жизненного цикла activity, чтобы решить те или иные задачи для вашего приложения. По ходу дела вы будете узнавать все больше об использовании каждой функции. А пока немного позабудемся и посмотрим, как ведет себя жизненный цикл в стандартных сценариях использования, поработаем с вашим приложением и посмотрим журналы на панели Logcat.

Временная остановка activity

Если приложение GeoQuiz не запущено на вашем устройстве, запустите его в программе Android Studio. Нажмите на устройстве кнопку «Главный экран», и MainActivity полностью исчезнет с экрана. В каком состоянии теперь MainActivity? Взгляните на панель LogCat. Activity приложения получила вызовы onPause() и onStop(), но не onDestroy() (рис. 3.8).

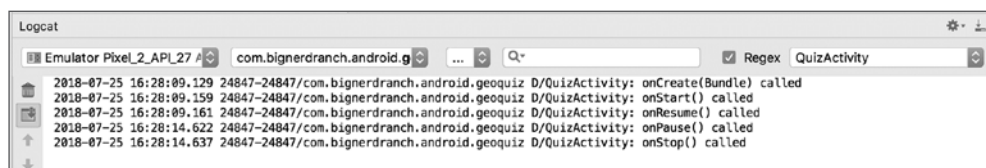


Рис. 3.8. Нажатие кнопки «Главный экран» приводит к остановке activity

Нажатие кнопки «Главный экран» сообщает Android: «Я сейчас займусь другим делом, но потом могу вернуться. Этот экран мне еще понадобится». Android приостанавливает, а в конечном итоге и останавливает activity.

Это означает, что после нажатия кнопки «Главный экран» ваш экземпляр MainActivity пребывает в состоянии остановки (находится в памяти, не виден, не активен). Android делает это для того, чтобы быстро и легко перезапустить MainActivity с того момента, на котором была прервана работа, если вы решите вернуться к GeoQuiz в будущем.

(Это не полное описание работы кнопки «Главный экран». Остановленные activity могут уничтожаться по усмотрению ОС — за более полным описанием обращайтесь к главе 4.)

Вернитесь к приложению GeoQuiz, выбрав его *сводный экран* в диспетчере задач. Для этого нажмите кнопку «Обзор приложений» рядом с кнопкой «Главный экран» (рис. 3.9).

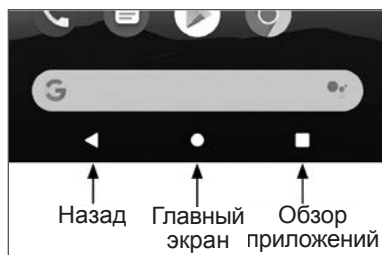


Рис. 3.9. Кнопки «Назад», «Главный экран» и «Обзор приложений»

Если на вашем устройстве нет кнопки «Обзор приложений», а лишь одна кнопка «Главный экран» (как показано на рис. 3.10), проведите пальцем вверх от нижней части экрана, чтобы открыть диспетчер задач. Если на устройстве не работает ни один из этих методов, обратитесь к руководству пользователя устройства.



Рис. 3.10. Одиночная кнопка «Главный экран»

Каждая миниатюра экрана в диспетчере задач представляет приложение, с которым пользователь взаимодействовал в прошлом (рис. 3.11). (В документации разработчика этот экран называется «сводным экраном», *overview screen*.)

Коснитесь сводного экрана приложения GeoQuiz; `MainActivity` заполнит экран.

Из информации на панели Logcat следует, что activity получила вызовы `onStart()` и `onResume()`. Обратите внимание: функция `onCreate(...)` при этом не вызывалась. Дело в том, что activity `MainActivity` находилась в состоянии остановки после нажатия пользователем кнопки «Главный экран». Так как экземпляр activity



Рис. 3.11. Диспетчер задач

все еще находится в памяти, создавать его не нужно. Вместо этого достаточно запустить activity (перевести ее в видимое приостановленное состояние), а затем продолжить ее выполнение (перевести в состояние выполнения с получением ввода от пользователя).

Ранее мы говорили, что activity может «зависнуть» в состоянии паузы, будучи либо частично видимой (например, при запуске новой activity с прозрачным фоном или не в полном экране), либо полностью видимой (в многооконном режиме). Остановимся подробнее на многооконном режиме.

Многооконный режим доступен только на Android 7.0 Nougat и выше, поэтому если ваше устройство работает на более ранней версии Android, для его тестирования потребуется эмулятор. Откройте диспетчер задач еще раз и коснитесь и удерживайте значок в верхней части GeoQuiz. Выберите режим Split screen (показано слева на рис. 3.12), и в нижней части экрана появится новое окно со списком приложений (в центре на рис. 3.12). Щелкните по одному из сводных экранов, чтобы запустить соответствующее приложение.

Запустится многооконный режим, при этом GeoQuiz окажется в верхнем окне, а второе приложение — в нижнем окне (показано справа на рис. 3.12).

Теперь щелкните по приложению в нижнем окне и посмотрите на сообщения журнала на панели Logcat. В MainActivity была вызвана функция onPause(), поэто-

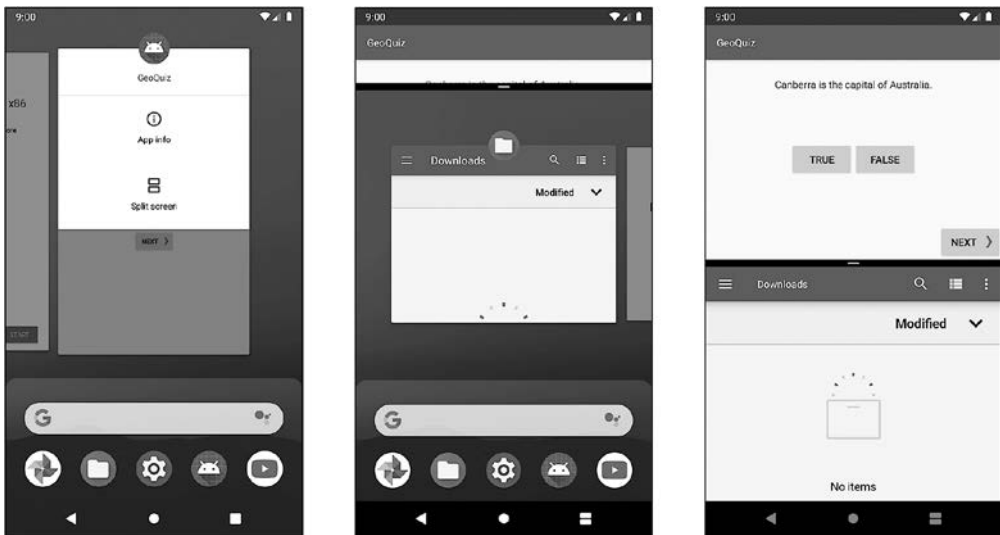


Рис. 3.12. Открытие двух приложений в многооконном режиме

му activity теперь находится в состоянии паузы. Нажмите на GeoQuiz в верхнем окне. В MainActivity была вызвана функция onResume(), поэтому activity теперь находится в возобновленном состоянии.

Чтобы выйти из многооконного режима, проведите разделителем окон в середине экрана вниз, чтобы закрыть нижнее окно. (Проведите разделителем вверх, чтобы закрыть верхнее окно.)

Завершение activity

Нажмите на устройстве кнопку «Назад», а затем проверьте вывод на панели LogCat. Activity приложения получила вызовы onPause(), onStop() и onDestroy() (рис. 3.13). Экземпляр MainActivity после этого уже не существует (он не находится в памяти, не виден и, конечно, не находится на переднем плане.)

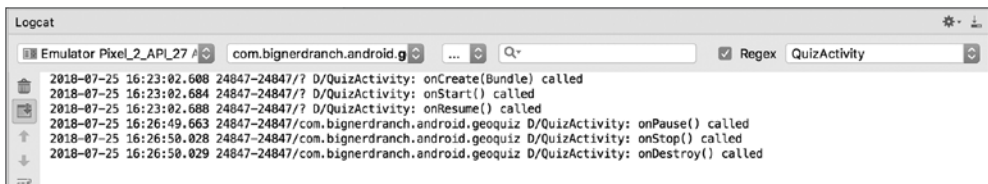


Рис. 3.13. Нажатие кнопки «Назад» приводит к уничтожению activity