# Лабораторная работа №8 Настройка Qt Creator для разработки приложений под android

Время десктопных приложений подходит к концу, всё больше популярности набирает мобильная и web разработка. В данной лабораторной работе рассмотрен один из вариантов настройки Qt 5.14 и IDE Qt Creator для разработки приложений под android ОС.

**Цель:**

1. Настроить IDE Qt Creator для разработки приложений под android ОС;
2. Изучить базовые понятия связанные с мобильной разработкой;
3. Научиться создавать мобильные приложения при помощи фреймворка Qt 5.14.

## Инструменты

1. IDE Qt Creator с набором предварительно собранных компонент фреймворка Qt 5.14 для android (560 Мб);
2. Java SE Development Kit 8 (JDK) или OpenJDK 8 (100 Мб);
3. Android Studio (749 Mб);
4. Компоненты для сборки приложений (3 Гб);
5. Genymotion (250 Мб + (образ) 350 Мб).

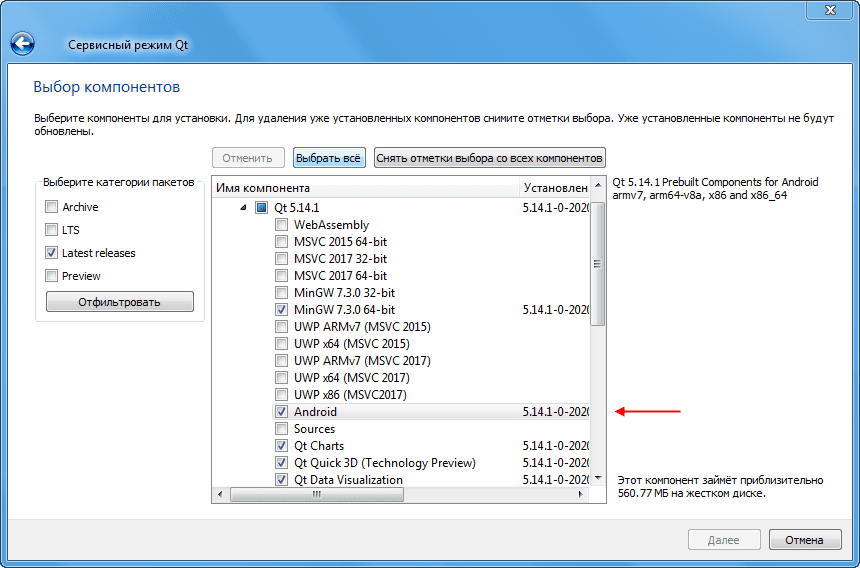
Все компоненты из вышеуказанного списка выбирайте в соответствии с вашей операционной системой и её разрядностью (битностью). Дальнейшее описание будет приводиться на примере Windows 7 x64.

### **Установка предварительно собранных компонент фреймворка Qt 5.14 для android**

Предполагается, что на данный момент у вас на ПК уже установлен Qt, если это не так, то все нижеперечисленные действия можно проделать в момент установки Qt через web installer который можно загрузить с официального сайта qt framework https://www.qt.io/.

Первым делом запустим **Qt Maintenance Tool**. Для этого напишите **Qt Maintenance Tool** в строке поиска меню Пуск. С помощью данного инструмента можно устанавливать/обновлять/удалять компоненты qt framework.

После запуска нажмите Далее и введите данные от своей учётной записи. На следующем экране выберите **Добавление или удаление компонентов** и снова нажмите Далее. После небольшой подгрузки перед вами откроется окно выбора компонентов. Установите галочку рядом с компонентом **Android** и нажмите Далее.

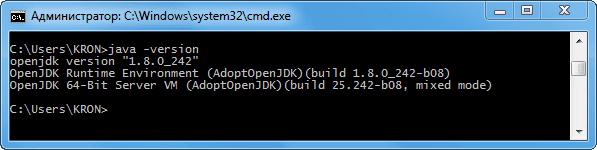
  
Рис. 1 Компоненты для android

После окончания установки закройте окно **Qt Maintenance Tool**.

### **Установка JDK 8/OpenJDK 8**

Если этот компонент у вас уже установлен, данный раздел можно пропустить.

**Внимание!** Нужна именно восьмая версия. Для того, чтобы проверить какая версия JDK установлена на вашем ПК, и есть ли она вообще, напишите в командной строке: java -version. Открыть командную строку можно комбинацией клавиш Win + R.

  
Рис. 2 Проверка версии JDK

Обратите внимание, в выводе должно быть написано именно **JDK**, а не **JRE**. Если вы видите сообщение об ошибке (значит Java не установлена) или версия отличается от 1.8, то переходите к процедуре установки.

На этом шаге есть два основных варианта:

* Установить Oracle JDK 8 с [официального сайта](https://www.oracle.com/java/technologies/javase-jdk8-downloads.html). В этом случае вам придется создать учётную запись заполнив довольно объёмную форму регистрации с последующим подтверждением через электронную почту;
* Установить одну из сборок OpenJDK 8.

Для лабораторной работы рекомендуется использовать [AdoptOpenJDK](https://adoptopenjdk.net/releases.html" \t "_blank) как менее затратный по времени вариант. Из списка выберите подходящий для вашей системы вариант **JDK** и нажмите скачать.

* **msi** версия произведёт автоматическую установку и настройку ОС для работы с JDK включая добавление пути к папке с JDK в переменную среды **Path** и **JAVA\_HOME** . Выбирайте этот вариант, если в вашей системе ещё не установлена Java вообще или установленная версия ниже восьмой.
* **zip** версия содержит папку с JDK, которую можно распаковать в удобное для вас место. Это вариант стоит выбирать, тем, у кого установленная версия Java выше 1.8.

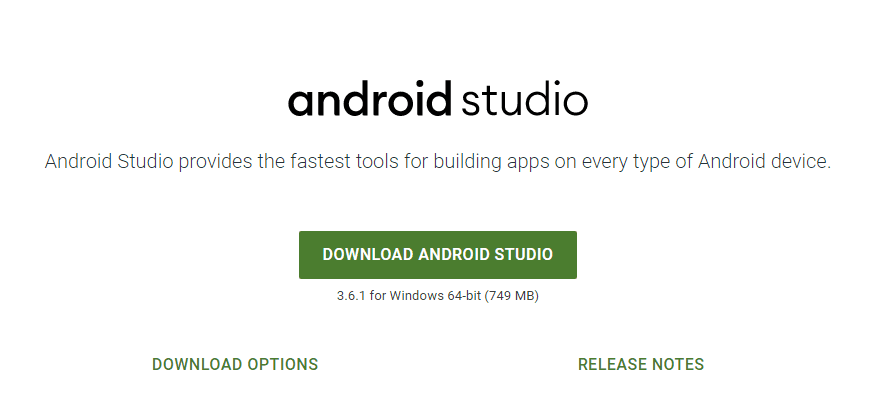
Если вы выбрали установку через **msi**, проверьте корректность установки выполнив команду: java -version в командной строке. Если всё установилось правильно вы увидите вывод как на рис. 2.

### **Установка Android Studio**

В связи с санкциями против Крыма прежде чем перейти непосредственно к установке выполним некоторые дополнительные действия, а именно настроим VPN. Возможные варианты:

* Вы уже пользуетесь VPN сервисами (не плагин для браузера), используйте их;
* У вас VPN на смартфоне + много мобильного интернета, раздайте Wi-Fi или подключите телефон к компьютеру в режиме USB-модема;
* Используйте способ указанный [здесь](https://neroid.ru/nastrojka-vpn-podkljuchenija-bez-ustanovki-dopolnitelnyh-prilozhenij/).

Подключите VPN-соединение и перейдите на [сайт](https://developer.android.com/studio?hl=ru). Нажмите на кнопку **Download Android Studio** предварительно убедившись, что ваша операционная система была определена верно (надпись под кнопкой). В противном случае воспользуйтесь ссылкой **Download Options**, чтобы перейти к списку доступных для загрузки версий **Android Studio**.

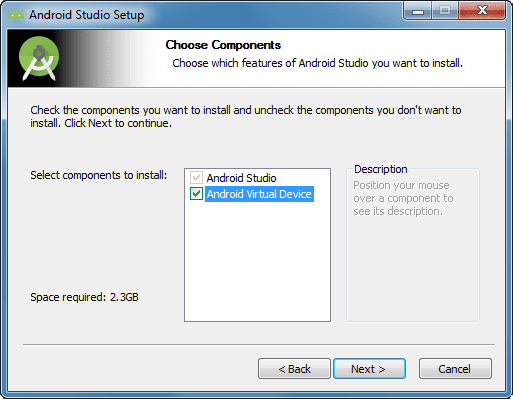
  
Рис.3 Загрузка Android Studio

В зависимости от загруженности VPN-канала загрузка может занять от 40 мин и более.

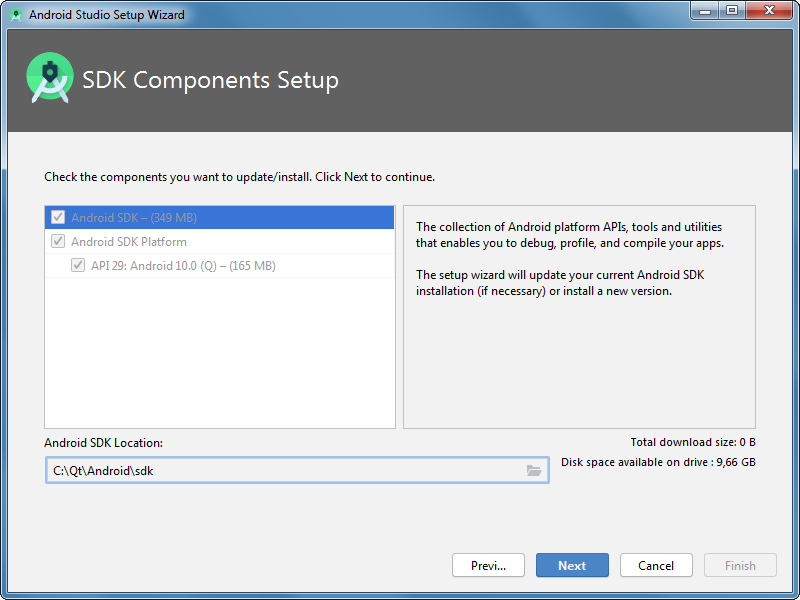
Если вы предпочитаете работать с командной строкой/терминалом, то можно обойтись без загрузки **Android Studio**, а воспользоваться инструментом [sdkmanager](https://developer.android.com/studio?hl=ru" \l "command-tools" \t "_blank). Плюсом консольной версии **sdkmanager** является размер, он почти в 10 раз меньше **Android Studio** и составляет 77 Мб. Но для начинающих разработчиков данный вариант выбирать не рекомендуется, т.к. версия SDK Manager с графическим интерфейсом встроенная в **Android Studio** существенно удобнее в использовании.

После окончания загрузки установите **Android Studio** в удобное для вас место. В целом процедура установки стандартная, единственный вопрос может возникнуть на вкладке выбора компонентов.

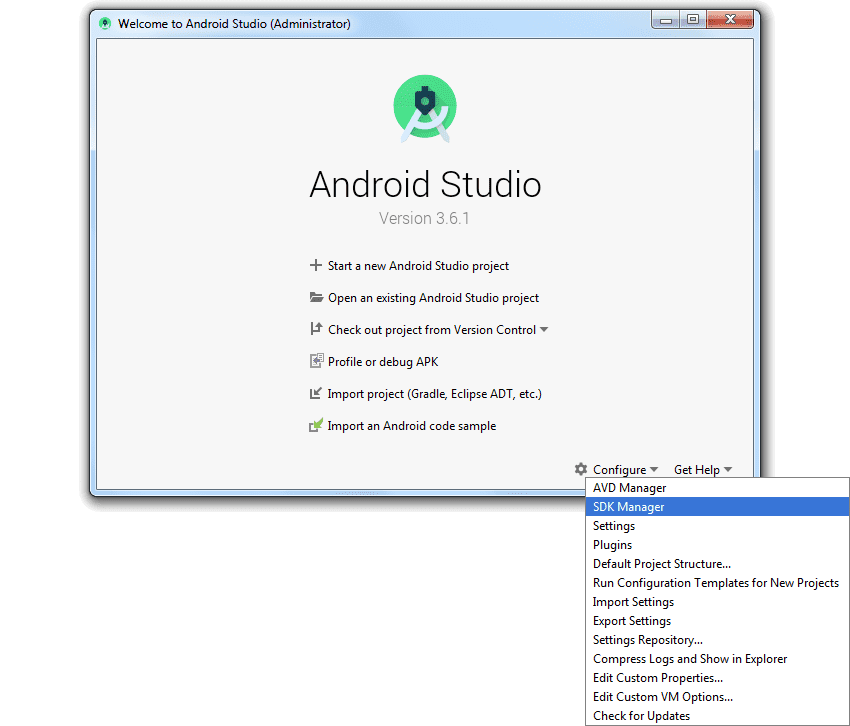
Компонент **Android Virtual Device** нужен в том случае если вы планируете использовать стандартный эмулятор андроид устройств. Но кроме **Android Virtual Device** потребуется установить ряд других компонентов. Т.к. в данной лабораторной работе предполагается использование эмулятора от сторонних разработчиков — **Genymotion**, поэтому, установка доп. компонентов, требуемых для работы стандартного эмулятора, описана не будет, а значит и установка **Android Virtual Device** не имеет смысла, то есть галочку можно убрать.

  
Рис.4 Выбор компонентов Android Studio

Во время запуска **Android Studio**попробует найти папку с загруженными SDK. Если ему это не удастся, то запустится мастер который предложит скачать последнюю версию SDK. Путь для размещения SDK можно указать любой, кроме той папки где установлена **Android Studio**. Если у вас уже есть папка с SDK, можно просто указать путь к ней.

  
Рис. 5 Android Studio Setup Wizard

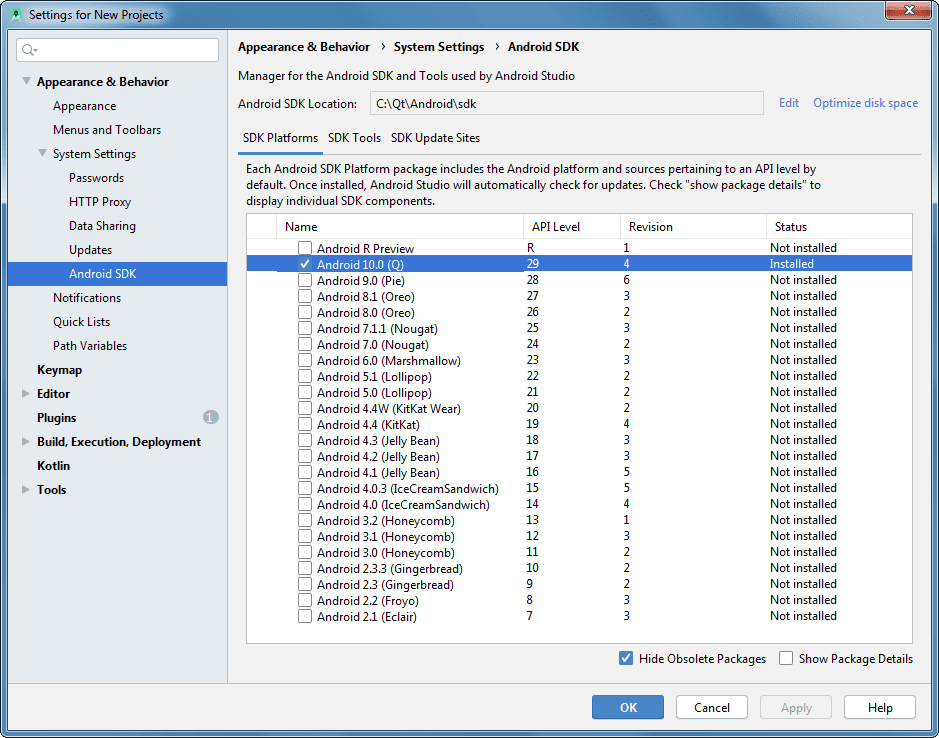
После окончания загрузки SDK перезапустите **Android Studio** и, в открывшемся окне, щёлкните по кнопке **Configure**. Выпадающем списке выберите **SDK Manager**.

  
Рис. 6 Запуск SDK Manager

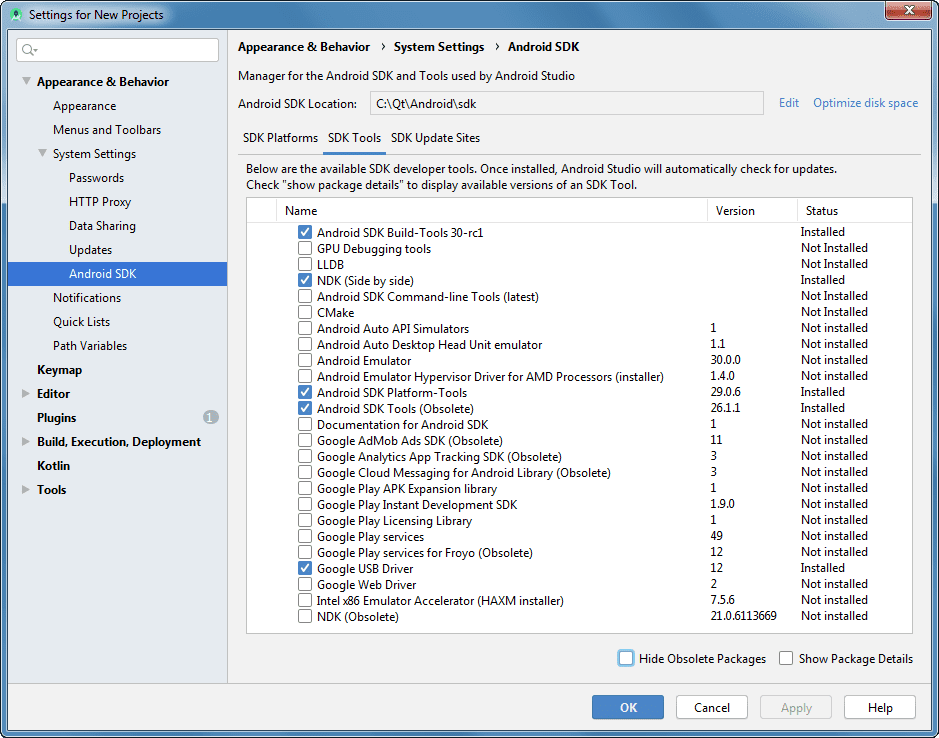
В открывшемся окне нас будут интересовать две вкладки: **SDK Platforms** и **SDK Tools**.

На вкладке **SDK Platforms** вы увидите список доступных для загрузки **SDK**. Версия SDK совпадает с уровнем **API** (второй столбец). В первом столбце указана версия android начиная с которой данный уровень API был введён.

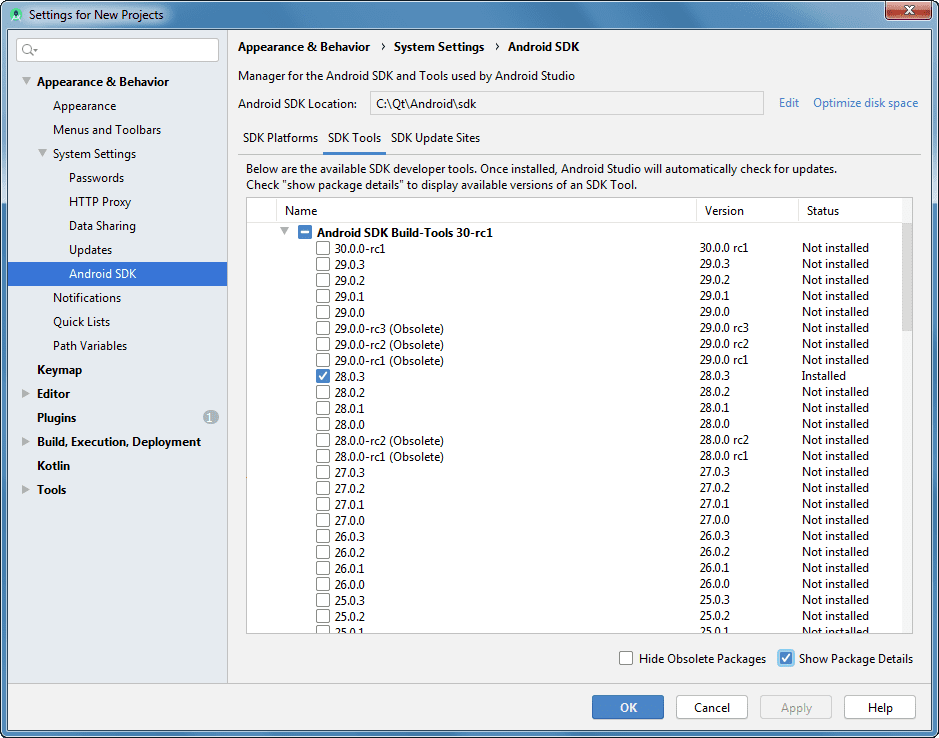
Разные версии Qt framework поддерживают разные уровни API. Qt версии 5.14 поддерживает API начиная с 21 уровня и выше (информация с официального сайта, но на практике лучше использовать SDK начиная с 24 версии). Это значит, что для устройств работающих на версии Android ниже 5 (Lollipop) ваша программа собранная с помощью Qt 5.14, скорее всего, не заработает, а значит нет смысла скачивать SDK ниже 21 версии. В следствии обратной совместимости разных уровней API нам нет необходимости для каждой версии Android скачивать свой SDK. Для сборки приложений мы можем использовать SDK для самого последнего уровня API, в этом случае наши приложения будут работать на всех android устройствах. Разумеется, если мы хотим обеспечить работу приложения на Android 5.0 (API 21) мы не должны использовать функции которые появились только начиная c API 22 уровня и т.д.

  
Рис. 7 Выбор комплекта SKD

Переключитесь на вкладку **SDK Tools**. Снимите галочку **Hide Obsolete Packages** (внизу окна). И выберите все компоненты указанные на скриншоте.

  
Рис. 8 Выбор SDK Tools

Поставьте галочку **Show Package Details**. В разделе **Android SDK Build-Tools 30-rc1** установите галочку напротив версии **28.0.3**, а в разделе **NDK (Side by side)** проверьте, что выбрана самая последняя версия (на данный момент это 21.0.6113669).

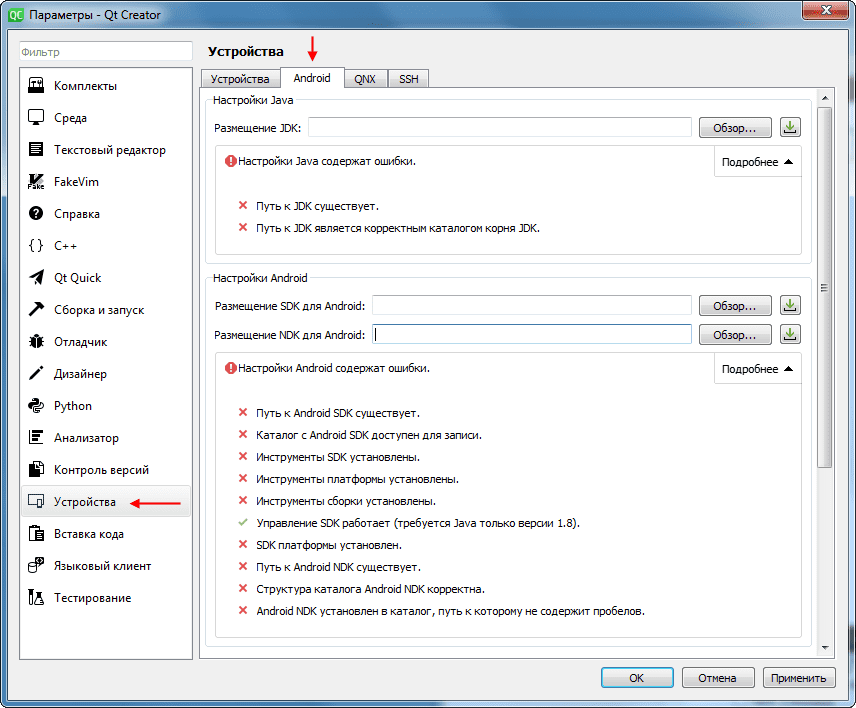
  
Рис. 9 Выбор Android SDK Build-Tools

Пояснения по поводу Android SDK Build-Tools. В целом, обычно рекомендуют всегда выбрать самую последнюю версию инструментов сборки. Здесь рекомендуется выбирать версию 28.0.3, по причине того, что в скрипте создаваемом для инструмента сборки android приложений **Gradle** по умолчанию прописана именно версия 28.0.3. Это значение можно изменить вручную отредактировав скрипт, но это придётся делать каждый раз.  
**Внимание!** Возможно версия Android SDK Build-Tools используемая Gradle по умолчанию будет изменена, в таком случае вы увидите сообщение об ошибке с указанием требуемой версии. Кроме того, Qt умеет скачивать требуемую версию сам, но для этого нужно будет включить VPN, для обхода санкций.

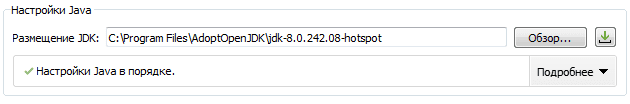
Загрузка выбранных компонентов может занять продолжительное время (более часа).

## Настройка Qt Creator

Запустите Qt Creator и выберите пункт меню **Инструменты → Параметры…** В открывшемся окне выберите раздел **Устройства** вкладку **Android**.

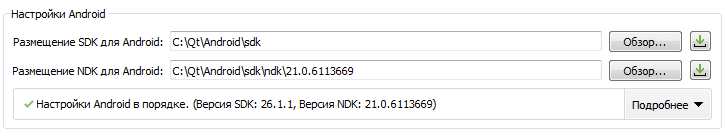
  
Рис. 10 Окно Параметры

В разделе **Настройки Java** в поле **Размещение JDK** укажите путь к **корневой** папке в которой расположена JDK 8 версии.

  
Рис. 11 Настройка Java

Если путь прописан корректно вы увидите сообщение как на рис. 11.

В разделе **Настройки Android** в поле **Размещение SDK для Android** укажите путь к **корневой** папке в которую вы скачивали SDK (можно узнать через SDK Manager в поле Android SDK Location). В поле **Размещение NDK для Android** укажите путь к корневой папке NDK. Он располагается в: [путь к корню sdk]\ndk\[номер версии ndk].

  
Рис. 12 Настройки Android

На этом настройка Qt Creator завершена.

## Варианты запуска/отладки приложения

Существует два основных варианта отладки приложения:

* На реальном устройстве. О том как настраивать устройство и проект в Qt Creator можно прочитать [тут](https://neroid.ru/otladka-android-prilozhenij-sobrany-v-qt-creator-na-realnyh-ustrojstvah/);
* На эмуляторе. О настройке эмулятора и проекта в Qt Creator можно прочить [тут](https://neroid.ru/ispolzovanie-genymotion-android-emulator-vmeste-s-qt/).

**Внимание!** Перед сборкой/запуском/отладкой **каждого нового**android-приложения рекомендуется включить VPN, т.к. Qt Creator и система автоматической сборки Gradle могут попробовать **докачать недостающие компоненты и зависимости**, в том числе, и с **подсанкционных** гугловских серверов. В случае неудачи сборка завершится с ошибкой. Скаченные зависимости остаются в системе и не требуют повторного скачивания.

## Что должно быть в отчёте

1. Скриншот окна настроек **Qt Creator** после настройки для сборки android приложений (везде должны быть зелёные галочки);
2. Выберите произвольный проект, из списка примеров, для сборки под android и добавьте в отчёт его название и скриншот.
3. Соберите проект и вставьте в отчёт скриншот экрана с запущенным приложением либо на эмуляторе либо на устройстве.

## Источники

1. [Qt for Android | Qt 5.14 (официальный сайт)](https://doc.qt.io/qt-5/android.html);
2. [Какую версию JDK выбрать](https://habr.com/ru/company/haulmont/blog/434790/);
3. [Уровень Android API, обратная и прямая совместимость](https://habr.com/ru/company/otus/blog/466367/);
4. [Термины, жаргон, определения Android](https://habr.com/ru/post/179095/);
5. [Gradle](http://developer.alexanderklimov.ru/android/theory/gradle.php).