*Purpose*

This tutorial covers how to configure a brand new Raspberry Pi with the Raspbian OS. It also covers how to configure a Raspberry Pi as a development platform for the JavaFX platform.

#### Time to Complete

Approximately 1 hour

#### Introduction

The Java platform is designed to run on different machines regardless of operating system. Computers comes in different shapes and capabilities, and Java still runs on most of them. The Raspberry Pi is a small, affordable, and extensible computer that is based on an ARM 11 processor, which is capable of running a Linux distribution. This new, compact computer also runs Java with the new Java 8 ARM distribution, providing you with a small and portable device on which to run your applications.

The Raspberry Pi has USB, Ethernet, audio, High-Definition Media Interface (HDML), and RCA composite video output, making it capable of running applications that connect to the network, are displayed in a connected screen, and are controlled by using any USB keyboard or mouse.

In this tutorial, you install the Java Development Kit (JDK) for Java Platform, Standard Edition Release 8 (Java SE 8) on the Raspberry Pi, and then you deploy, run, and stop JavaFX applications directly and remotely in the Pi.

#### Software Requirements

Download and install the following software on your PC:

* [NetBeans 8 Developer version](https://netbeans.org/community/releases/80/)
* [Oracle Java 8 JDK](https://jdk8.java.net/download.html) (Windows, Linux, or Mac, depending on your system)
* [SD Association Formatter tool](https://www.sdcard.org/downloads/formatter_4/eula_windows/)
* [Win32DiskImager](http://sourceforge.net/projects/win32diskimager/) (Windows)
* [SHA-1 checksum verifier tool](http://lists.gnupg.org/pipermail/gnupg-announce/2004q4/000184.html) (Windows)
* [PuTTY Telnet and Secure Shell (SSH) Client](http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html)
* [PSFTP client](http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html)

Download the following software:

* [Raspbian Wheezy image (2013-09-16-wheezy-raspbian or later)](http://www.raspberrypi.org/downloads)
* [Oracle Java SE 8 JDK for Linux ARMV6/7](https://jdk8.java.net/download.html)

**Note:** File versions and file names might differ.

#### Hardware Requirements

* Raspberry Pi Model B Revision 2
* 4 GB (or larger) Secure Digital (SD) memory card
* Micro-USB power supply (5v, 1A)
* USB keyboard and USB mouse
* HDMI cable
* HDMI-compatible monitor (or TV screen)
* Ethernet cable
* (Optional) [Wireless USB module](http://www.adafruit.com/products/814)

#### Prerequisites

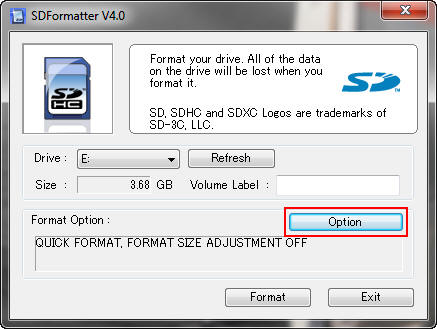
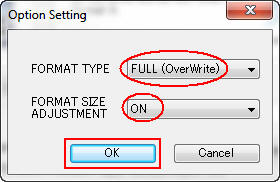
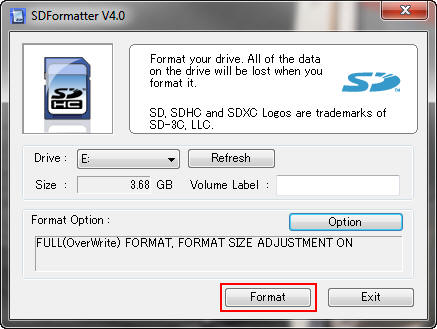
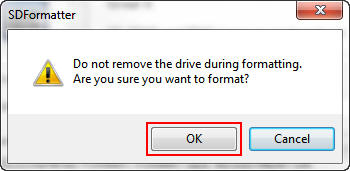
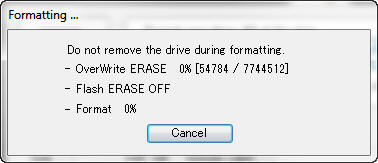
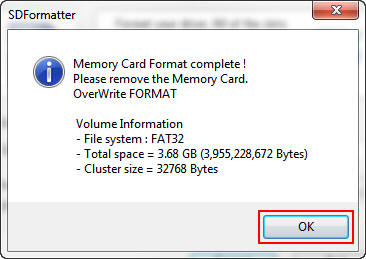
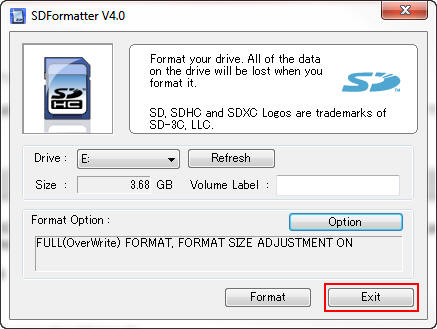
Before starting this tutorial, you should have:

* Read the [Raspberry Pi Quick Start Guide](http://www.raspberrypi.org/quick-start-guide) for instructions on how to connect your Raspberry Pi to a power source, a keyboard, a mouse, and a monitor.
* (Optional) Completed the [Configuring the Raspberry Pi as an Oracle Java Embedded Development Platform OBE](http://apex.oracle.com/pls/apex/f?p=44785:24:0::NO:24:P24_CONTENT_ID,P24_PREV_PAGE:8489,29). If you did, you can skip sections 1 and 2 of this tutorial.

## Creating a Bootable Image for the Raspberry Pi

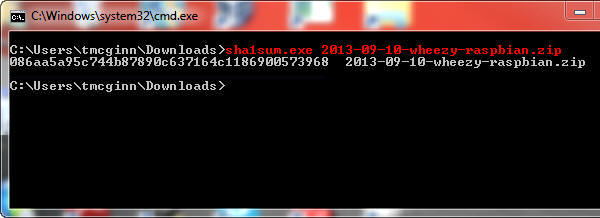
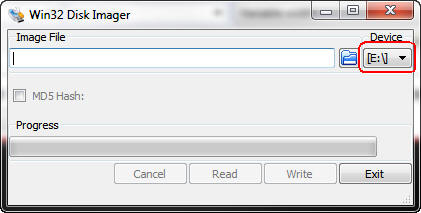
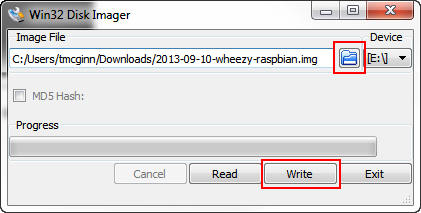
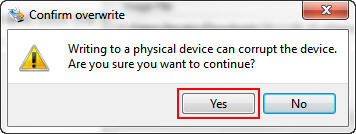
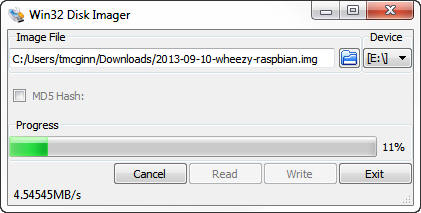
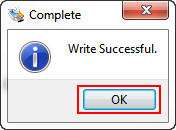
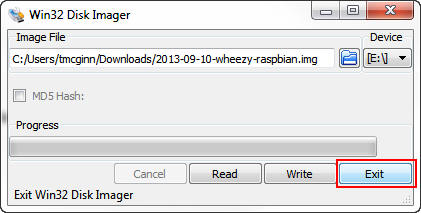
In order to boot, the Raspberry Pi requires a bootable Linux image on an SD memory card. There is no hard drive for the computer. Instead, the 4 GB card stores the image that the computer runs when it is powered on. This SD memory card also acts as the storage for other applications that are loaded onto the card.

### Formatting the SD Memory Card with the SD Formatter Tool

1. Insert the SD card into your computer, or connect it to your computer by using an SD card peripheral.
2. Start the SD Formatter tool and click **Option**.
3. 
4. Sizes might differ, depending on the SD size.
5. In the Option Setting dialog box, select **FULL (Overwrite)** from Format Type and **ON** from Format Size Adjustment, and then click **OK**.
6. 
7. Click **Format** to start the SD Formatter tool.
8. 
9. Click **OK** to format the SD memory card.
10. 
11. A dialog box displays the progress of the format.
12. 
13. When the format is completed, click **OK** to close the dialog box.
14. 
15. Click **Exit** to close the SD Formatter tool.
16. 

You are now ready to install the Raspbian Wheezy image in the SD memory card. Leave the card in the computer.

### Downloading and Confirming the Raspbian Wheezy Image

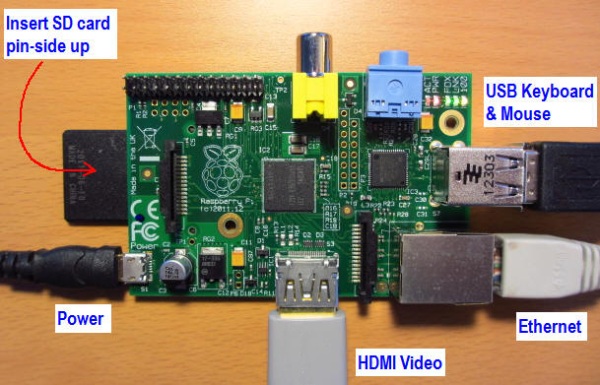
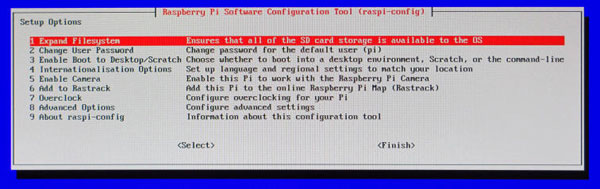
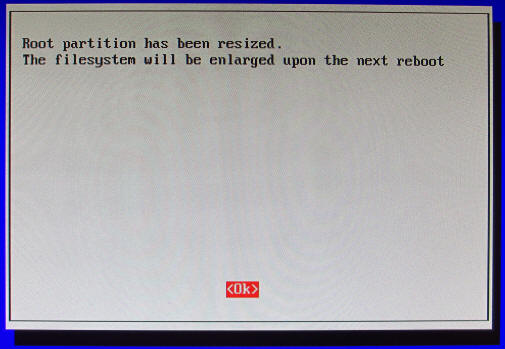
1. Make a note of the SHA-1 checksum for the zip file that you downloaded.
2. 
3. **Note:** The SHA-1 checksum may also be in a file with a .sha1 extension.
4. Open a command window in the directory where you downloaded the zip file and enter **sha1sum.exe 2013-09-10-wheezy-raspbian.zip**.
5. 
6. **Note:** If the SHA-1 checksum does not match the checksum listed on the download page, try downloading the zip file again.
7. Unzip **2013-09-10-wheezy-raspbian.zip** with your favorite unzip tool.
8. Start the Win32DiskImager application. Make sure that the device address is the same as the mount point for the SD memory card.
9. 
10. Click the folder button, navigate to the location of the image file, and then click **Write**.
11. 
12. Click **Yes** to confirm that you want to overwrite the contents of the SD memory card.
13. 
14. A progress meter is displayed.
15. 
16. When the Write Successful message is displayed, click **OK**.
17. 
18. Click **Exit** to close Win32 Disk Imager.
19. 
20. Remove the SD memory card from the computer.

You are now ready to boot the Raspberry Pi and install the JDK 8 distribution for Pi.

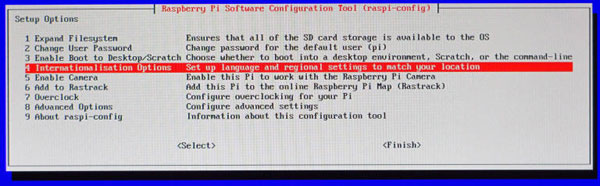
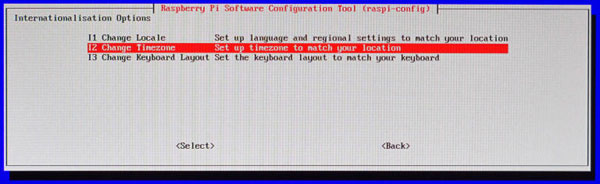
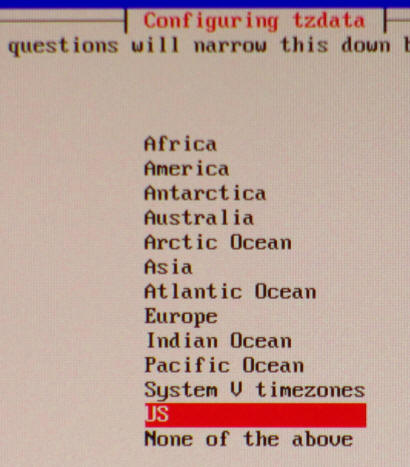
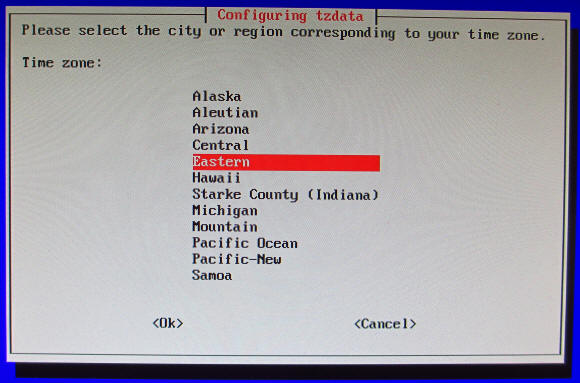
## Setting Up the Raspberry Pi for Remote Access

To set up the Raspberry Pi for remote access, you configure Raspbian to allow SSH connections. To ensure that the Raspberry Pi always has the same IP address, you configure the Raspberry Pi to use a static IP address.

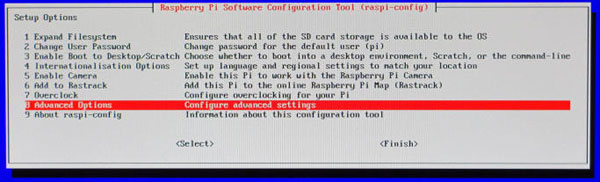
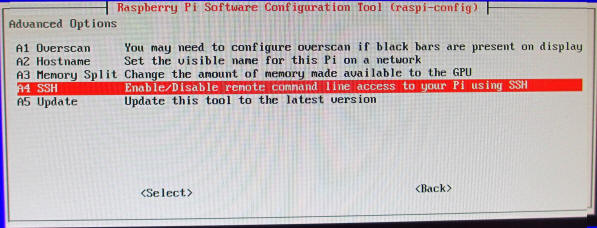
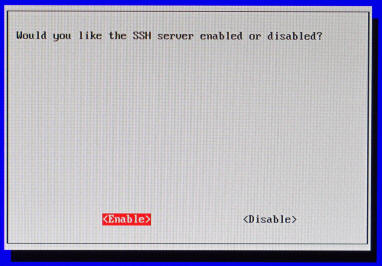
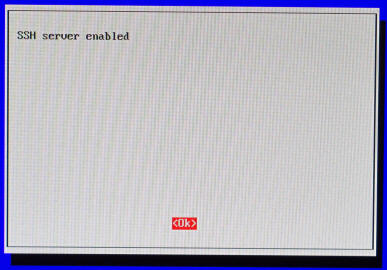
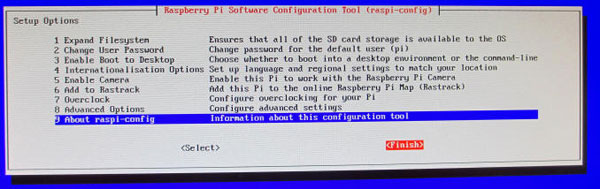
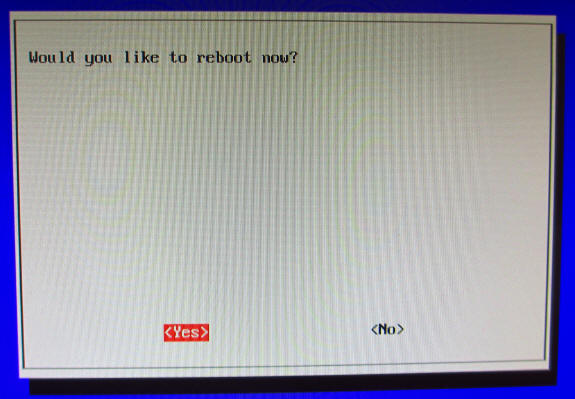
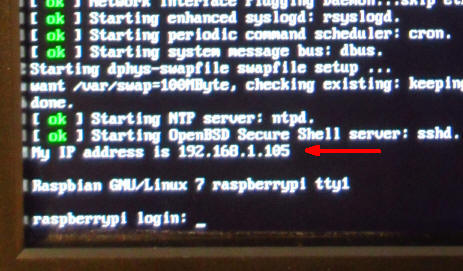
### Booting the Raspberry Pi and Configuring Raspbian Wheezy Linux

1. Perform the following steps:
   1. Insert the SD memory card into the Raspberry Pi.
   2. Connect the USB keyboard and USB mouse.
   3. Connect the HDMI display.
   4. Connect the Ethernet cable.
   5. Plug in the power cable.
2. 
3. After a few moments, the monitor displays text, and the Raspberry Pi raspi-config configuration screen appears.
4. Select setup option 1, **Expand Filesystem**, and press **Enter** to expand the file system and use all of the SD memory card storage.
5. 
6. Press **Enter** to close the confirmation screen.
7. 

### Configuring the Time Zone

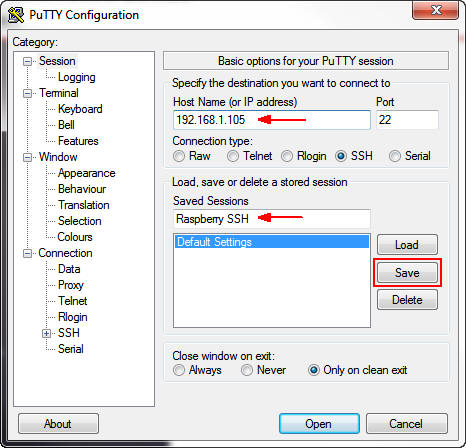
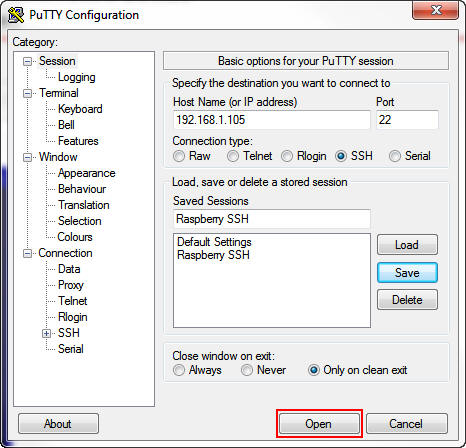
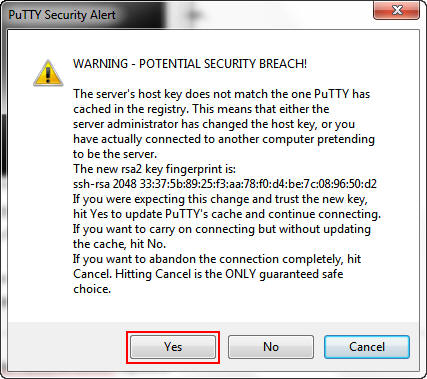
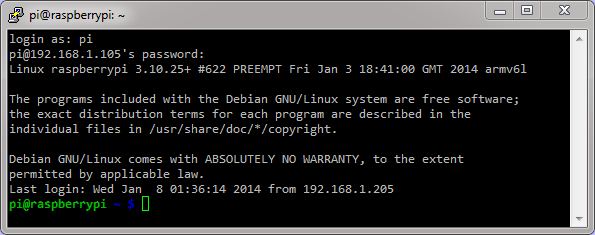
1. Press the up or down arrow key to select option 4, **Internationalisation Options**, and press **Enter**.
2. 
3. Press the up or down arrow key to select option I2, **Change Timezone**, and press **Enter**.
4. 
5. Press the up or down arrow key to select your country and press **Enter**.
6. 
7. Press the up or down arrow key to select the time zone in your country (if applicable) and press **Enter**.
8. 

### Configuring the Raspbian Wheezy OS to Enable SSH Connections

1. Press the up or down arrow key to select option 8, **Advanced Options**, and press **Enter**.
2. 
3. Press the up or down arrow key to select option A4, **SSH**, and press **Enter**.
4. 
5. Press the left or right arrow key to select **Enable** and press **Enter**.
6. 
7. Press **Enter**.
8. 
9. Press the left or right arrow key to select **Finish** and press **Enter**.
10. 
11. Press the left or right arrow key to select **Yes** and then press **Enter** to reboot the Raspberry Pi.
12. 
13. The Raspberry Pi reboots.
14. 
15. Notice the line that indicates the IP address of the Raspberry Pi.

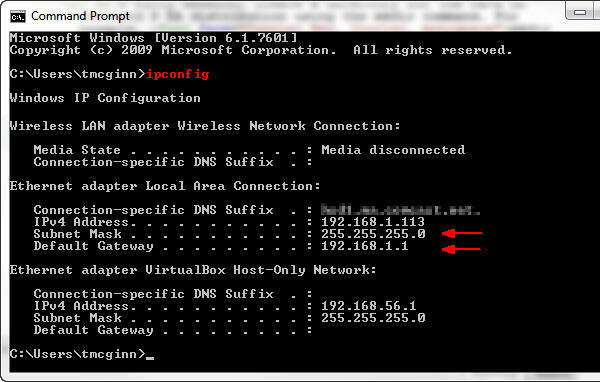
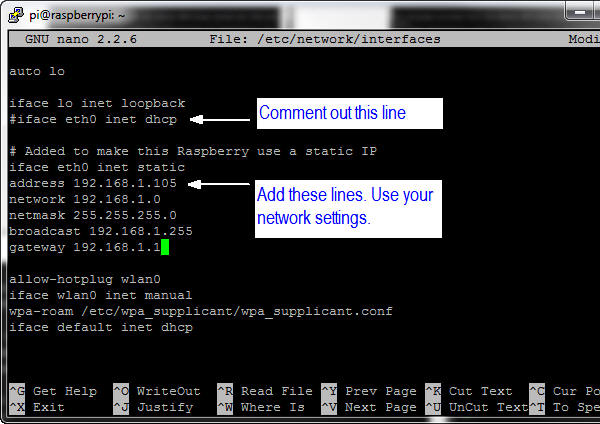
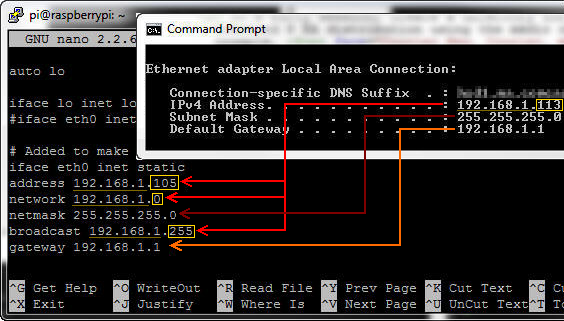
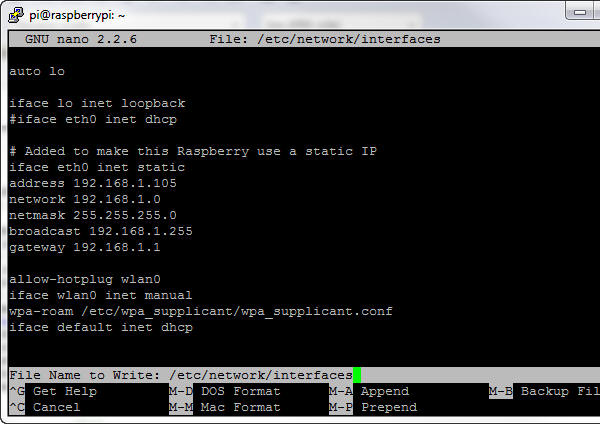
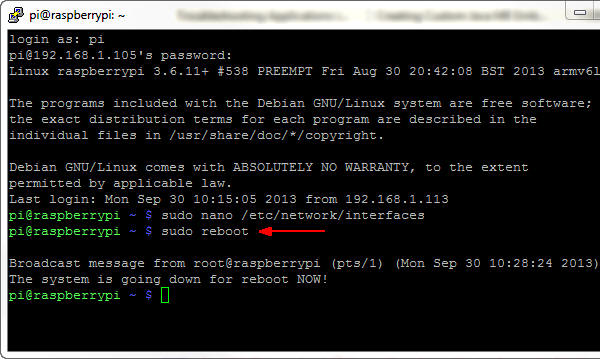
At this point, you can disconnect the HMDI monitor, the USB mouse, and the USB keyboard.

### Connecting to the Raspberry Pi with the Secure Shell

1. Start **putty.exe** and perform the following steps:
   1. On the boot screen, enter the IP address in the IP address field.
   2. Enter **Raspberry SSH** in the Saved Sessions field.
   3. Click **Save**.
2. 
3. Click **Open**.
4. 
5. Click **Yes** to accept the security alert.
6. 
7. Enter **pi** as the login name and **raspberry** as the password.
8. 

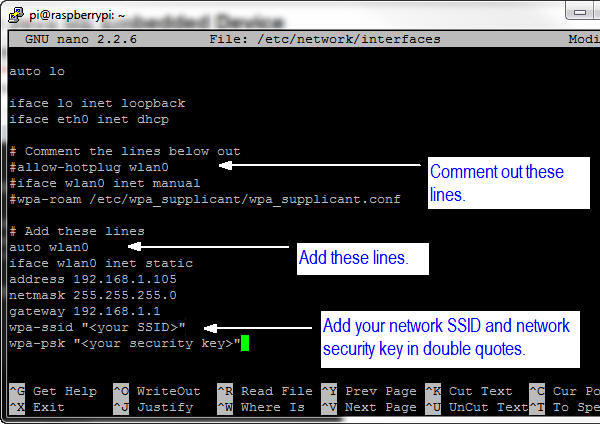
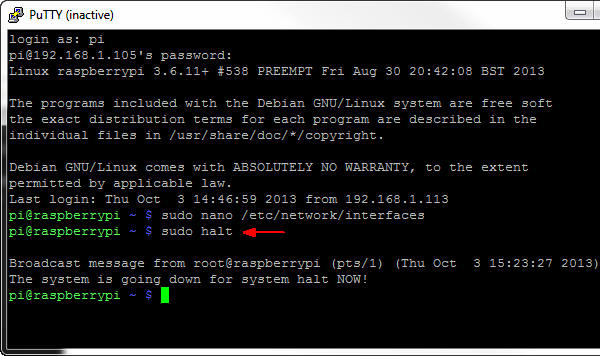
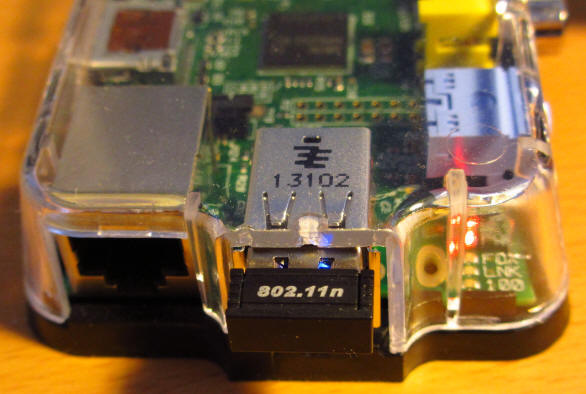
### Setting Up the Raspberry Pi to Use a Fixed Wired IP Address

If you plan to use your Raspberry Pi on a wireless network, skip this section.

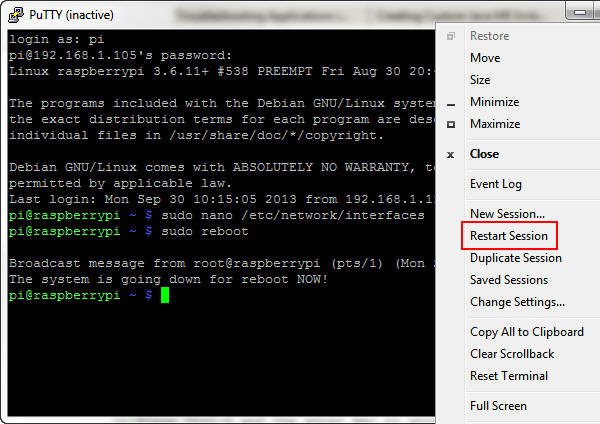
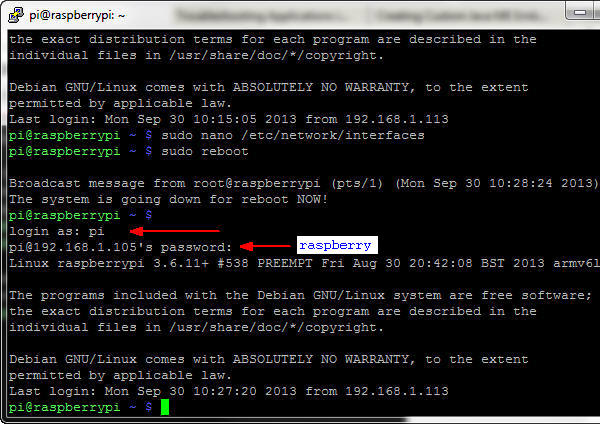
1. In a separate command-prompt window, enter **ipconfig** to get the netmask and default gateway for your network.
2. 
3. **Note:** If you are using a wireless connection, the information appears under the Wireless LAN adapter heading.
4. In a PuTTY window, enter **sudo nano /etc/network/interfaces** and then perform the following steps:
   1. Comment out the **iface eth0 inet dhcp** line.
   2. Below that, add the **iface eth0 inet static** line.
   3. Use your network settings to add the IP, network, netmask, broadcast, and gateway addresses. For example:  
      address 192.168.1.105 should be similar to the IPv4 address.  
      network 192.168.1.0 should be the network prefix and zeroes.  
      netmask 255.255.255.0 should match the subnet mask.  
      broadcast 192.168.1.255 should be the network broadcast address.  
      gateway 192.168.1.1 should match the default gateway.
5. 
6. **Note:** You have to set the network properties to match the network configuration that you are using on your PC.
7. 
8. Press **Ctrl + O** and then press **Enter** to write the file.
9. 
10. Press **Ctrl + X** and then press **Enter** to close nano and return to the prompt.
11. Enter **sudo reboot** to reboot the Raspberry Pi and make the changes.
12. 

### Optional: Setting Up the Raspberry Pi to Use a Wireless Network with a Fixed IP Address

If you plan to use a wireless USB module with your Raspberry Pi, you need to edit the network interfaces file.

1. In a PuTTY window, enter **sudo nano /etc/network/interfaces** and then perform the following steps:
   1. Comment out the following lines: **allow-hotplug wlan0**  
      **iface wlan0 inet manual**  
      **wpa-roam /etc/wpa\_spplicant/wpa\_supplicant.conf**
   2. Below that, add the following lines: **auto wlan0**  
      **iface wlan0 inet static**
   3. Use your network settings to add the IP address, network mask, and gateway address. For example:  
      address 192.168.1.105  
      netmask 255.255.255.0  
      gateway 192.168.1.1
   4. Add the network SSID and passkey for your network in double quotation marks. For example:   
      wpa-ssid "mynetwork"  
      wpa-psk "cants@y"
2. 
3. Press **Ctrl + O** and then press **Enter** to write the file. Press **Ctrl + X** to exit nano.
4. Enter **sudo halt** to halt the Raspberry Pi.
5. 
6. Unplug the power cable to the Raspberry Pi.
7. The Raspberry Pi supports only two USB connections. To use the USB Wifi module, unplug the USB mouse, the USB keyboard, and the Ethernet cable, and then plug in the USB Wifi module.
8. To use JavaFX applications, you need a mouse and preferably a keyboard. You can use a USB hub to connect everything to the Raspberry Pi.
9. 
10. Plug the power cable into the Raspberry Pi.

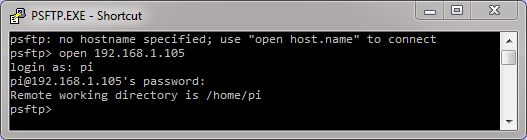
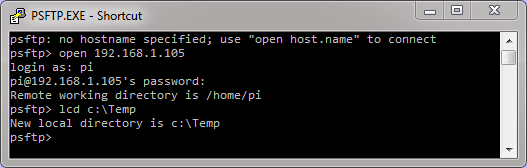
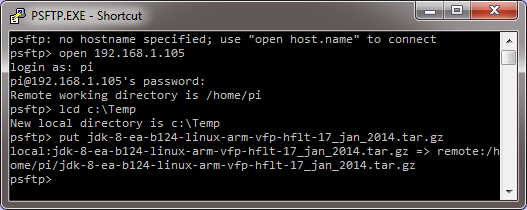
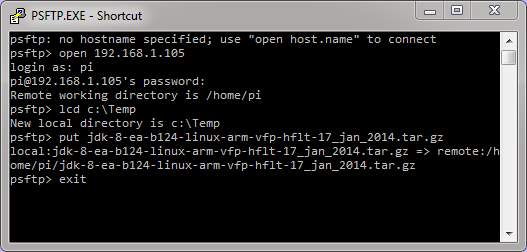
### Restarting the PuTTY Connection

1. To reconnect to the Raspberry Pi, right-click in the header of the PuTTY window and select **Restart Session**.
2. 
3. Enter **pi** as the login name and **raspberry** as the password.
4. 

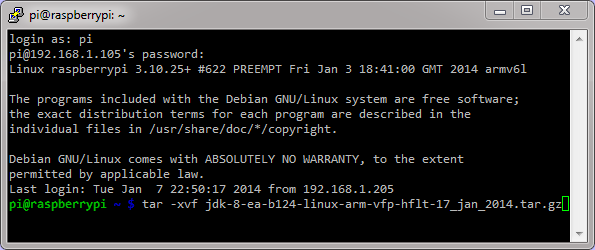
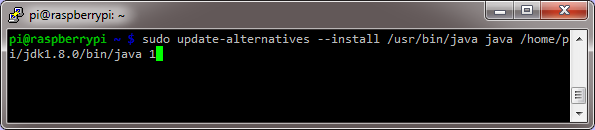
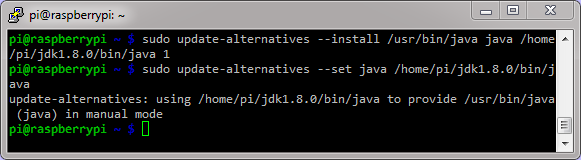
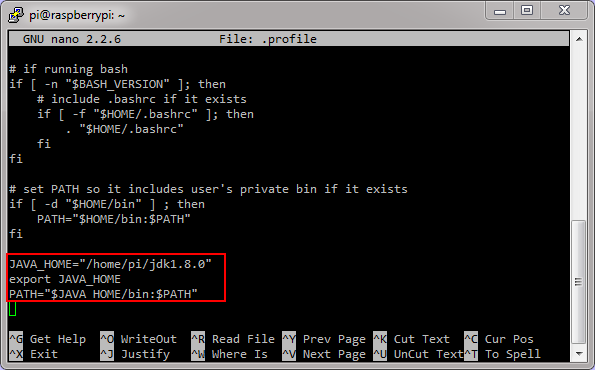
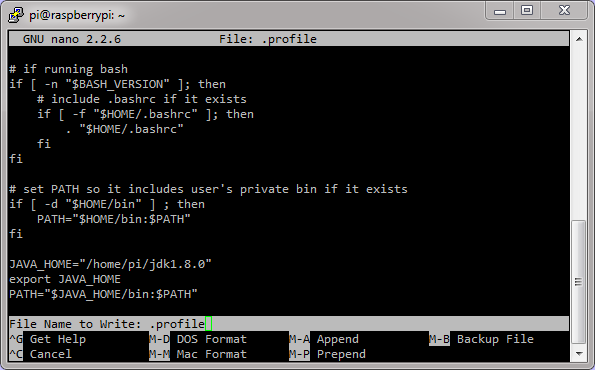
## Installing JDK 8 in the Raspberry Pi

Raspbian comes with an installed version of Java. To install a new version of Java with the full JDK, you transfer the ARMV6/7VFP gzip distribution to the Pi by using PSFTP. After the gzip is on the Pi, you unzip it and set the required environment variables.

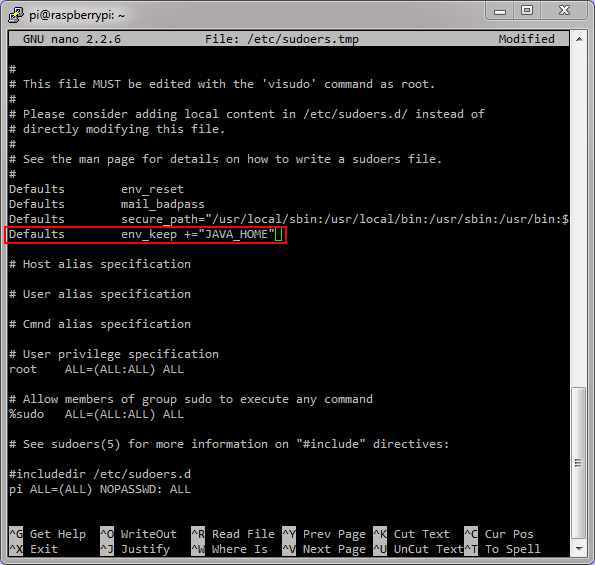
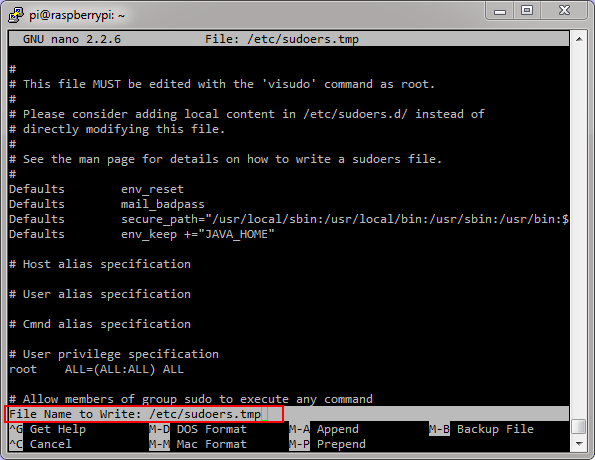
### Transferring JDK 8 to the Raspberry Pi by Using PSFTP

1. Launch the PSFTP tool and then perform the following steps:
   1. Enter **open 192.168.1.105**.
   2. Enter **pi** as the login name and **raspberry** as the password.
2. 
3. If a dialog box about a cached key is displayed, press **Y** to update the key.
4. Enter **lcd C:\Temp** to change the local directory to the location of JDK 8 for Linux ARMv6/7.
5. 
6. Enter **put jdk-8-ea-b124-linux-arm-vfp-hflt-17\_jan\_2014.tar.gz**.
7. 
8. This process may take several minutes because it uploads JDK to the Raspberry Pi.
9. Type **exit** to quit the PSFTP tool.
10. 

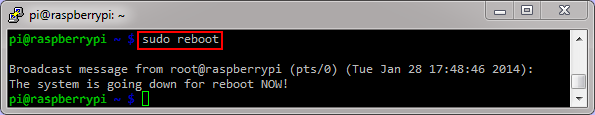
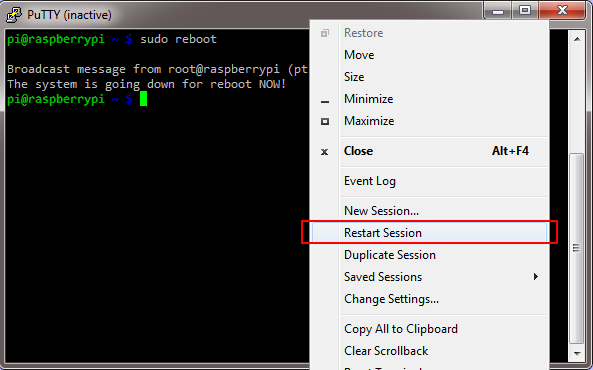
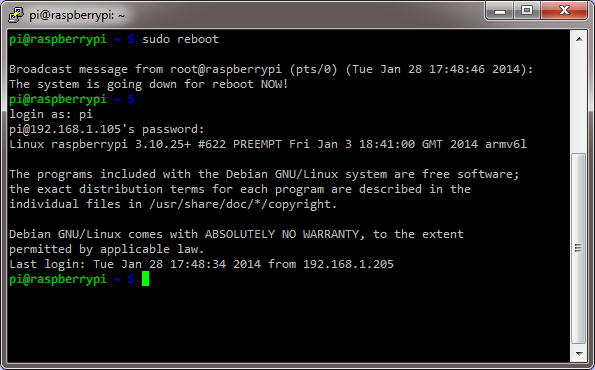
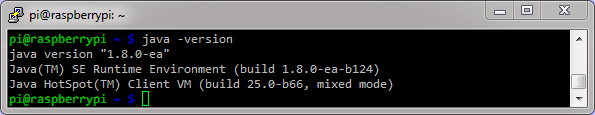
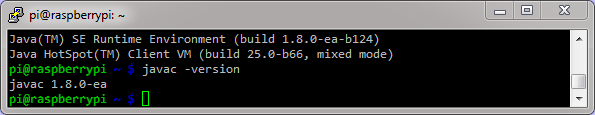
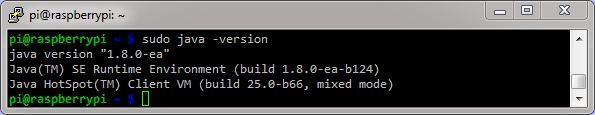
### Installing the Raspberry Pi JDK 8

1. In the PuTTY window, enter **tar -xvf jdk-8-ea-b124-linux-arm-vfp-hflt-17\_jan\_2014.tar.gz** to extract JDK 8.
2. 
3. Files are extracted inside the jdk1.8.0 folder. (The process may take several minutes.)
4. If a message indicates that some files have a timestamp, perform the following steps to correct the error:
5. [**Solving Timestamp Error**](http://www.oracle.com/webfolder/technetwork/tutorials/obe/java/RaspberryPiFX/raspberryfx.html#)
6. Enter **sudo update-alternatives --install /usr/bin/java java /home/pi/jdk1.8.0/bin/java 1**.
7. 
8. Enter **sudo update-alternatives --set java /home/pi/jdk1.8.0/bin/java**.
9. 
10. Enter **nano .profile**.
11. 
12. At the bottom of the file, add the following:
13. JAVA\_HOME="/home/pi/jdk1.8.0" export JAVA\_HOME PATH="$JAVA\_HOME/bin:$PATH"
14. 
15. Press **Ctrl + O** and then press **Enter** to save the changes. Press **Ctrl + X** to exit nano.
16. 

### Setting Up JAVA\_HOME for sudo Commands

1. In the PuTTY window, enter **sudo visudo**.
2. 
3. Enter **Defaults env\_keep +="JAVA\_HOME"** after all the Defaults declarations.
4. 
5. Press **Ctrl + O** and then press **Enter** to save the changes. Press **Ctrl + X** to exit nano.
6. 
7. Using visudo, you edit the sudoers file safely. After the editor is closed, the file is validated. If any validation errors are in the file, the sudoers file is not modified.

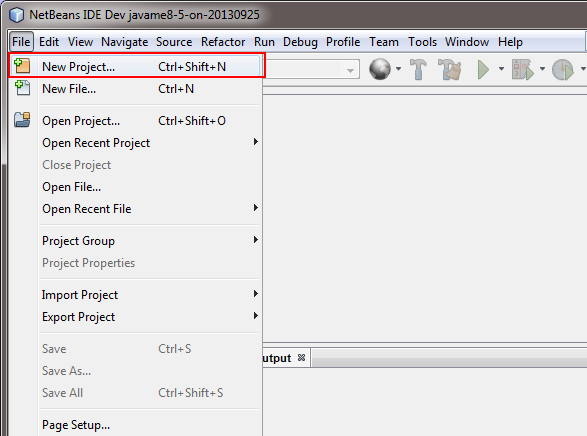
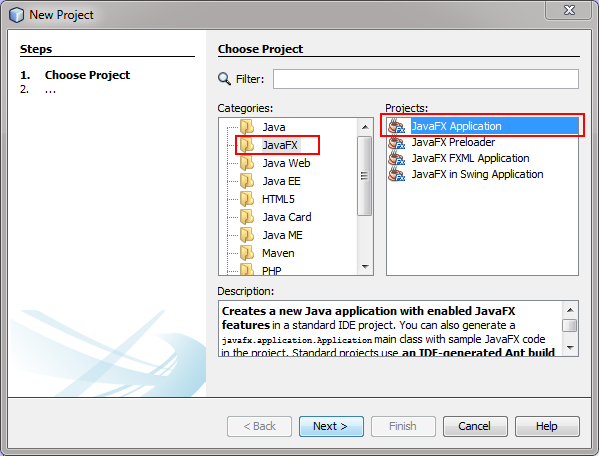
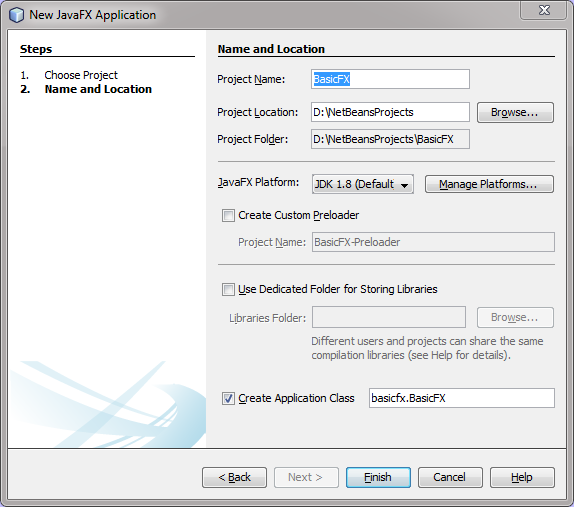
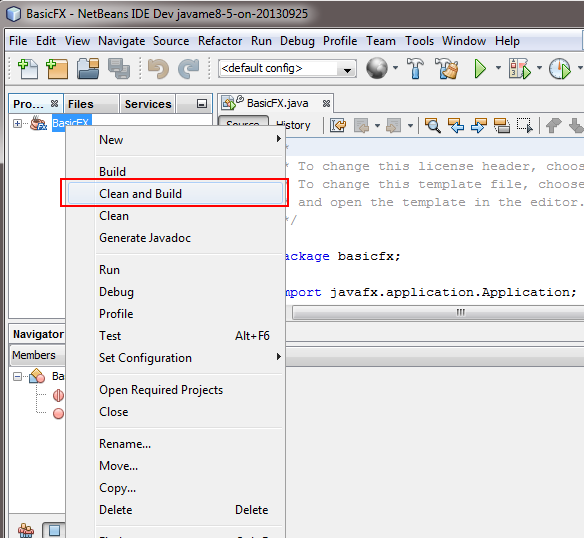
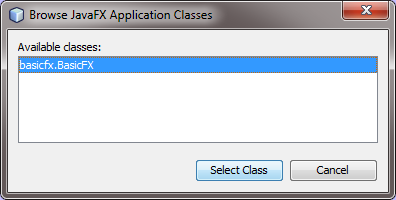
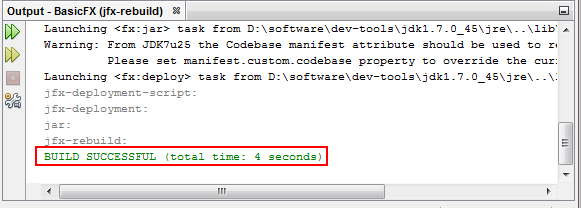
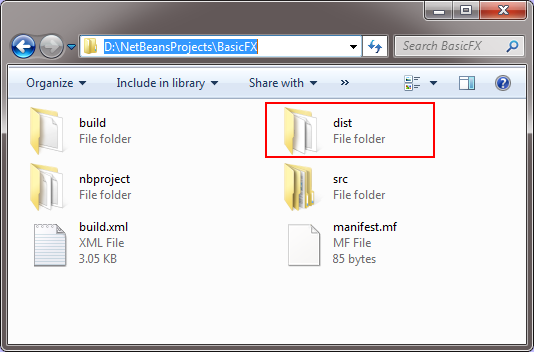
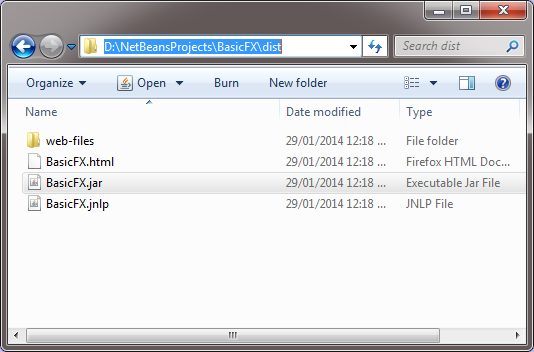
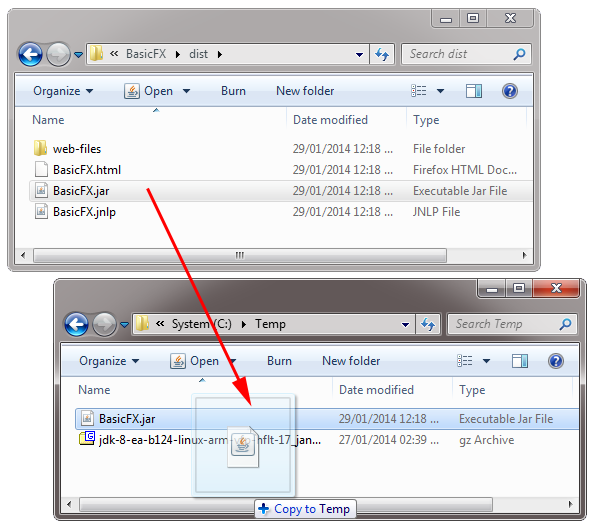
### Verifying the JDK 8 Installation

1. Enter **sudo reboot** to restart the Raspberry Pi. Wait a couple of seconds to let the Raspberry Pi reboot.
2. 
3. Right-click in the header of the PuTTY window and select **Restart Session** to reconnect to the Raspberry Pi.
4. 
5. Enter **pi** as the user name and **raspberry** as the password.
6. 
7. Enter **java -version** to see the current Java version that is being used.
8. 
9. **Note:** Your Java version should be at least 1.8.
10. Enter **javac -version** to see the current Java compiler version that is being used.
11. 
12. **Note:** Your Java version should be at least 1.8.
13. Enter **sudo java -version** to see the Java version that is being used by the root user.
14. 
15. **Note:** Your Java version should be at least 1.8.

## Building JavaFX Applications

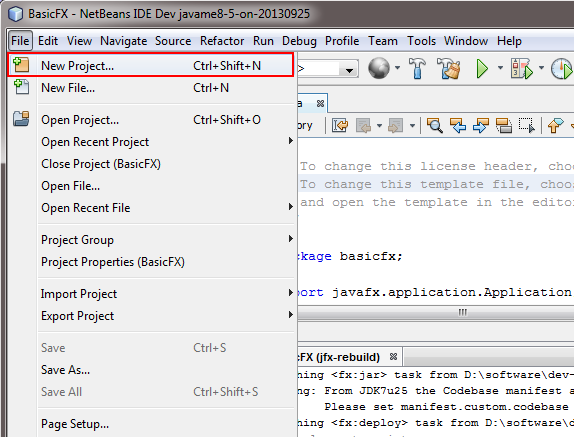
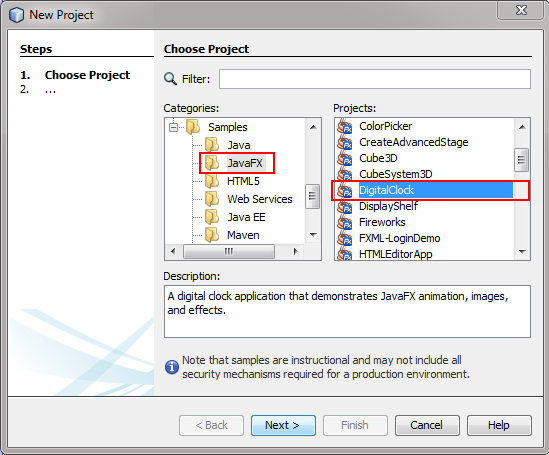
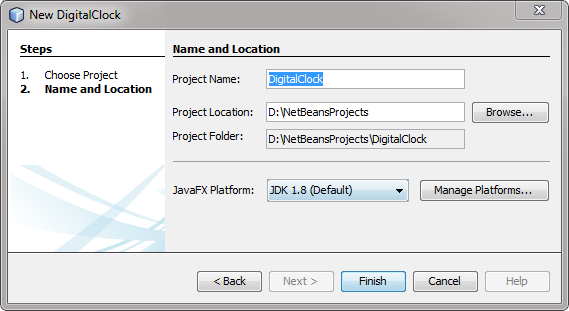
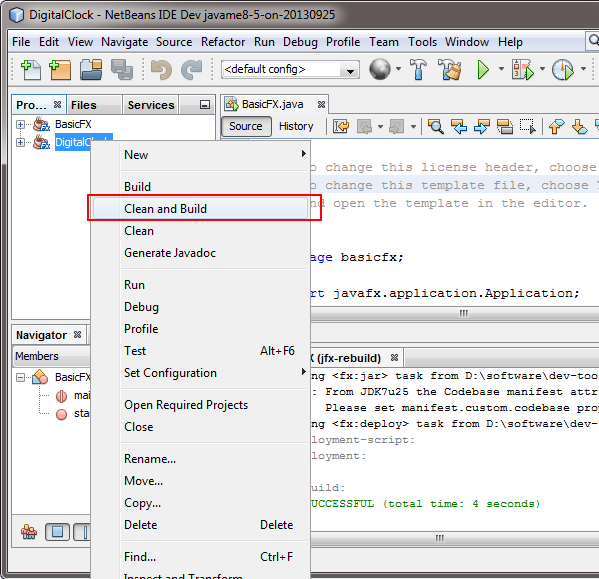
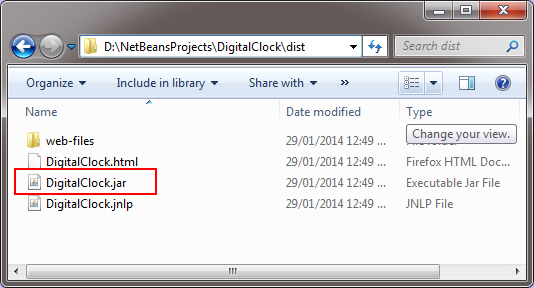
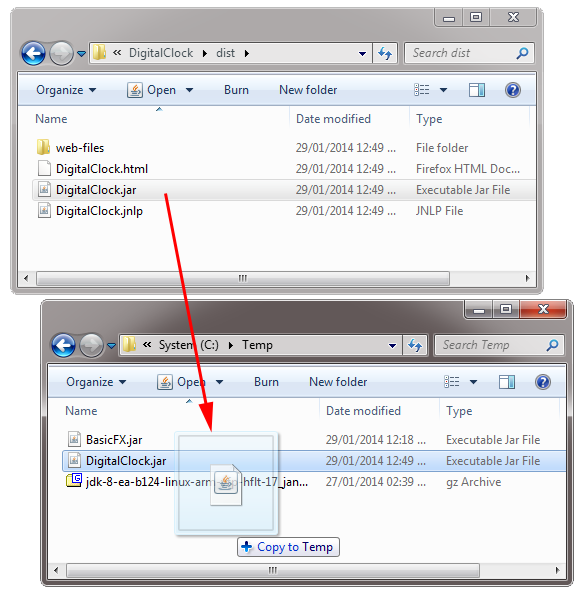
To develop JavaFX applications faster, you use your desktop computer with the NetBeans IDE. The JavaFX projects generate JAR files that can be deployed to any device that can run JavaFX. In this section, you create two applications with NetBeans and move the resulting JAR files to upload them later to the Raspberry Pi.

### Building a Basic JavaFX Application

1. In NetBeans, select **File > New Project**.
2. 
3. Select **JavaFX** from Categories and **JavaFX Application** from Projects and click **Next**.
4. 
5. Enter **BasicFX** as the project name, select a location for the project, and click **Finish**.
6. 
7. Your project is created and loaded into the IDE, and you can explore the contents of the project.
8. Right-click the project and select **Clean and Build** to generate a distributable JAR file.
9. 
10. Click **Select Class** in the Browse JavaFX Application Classes if the dialog box is displayed.
11. The Output tab shows the progress and indicates when the process is completed.
12. Open a File Explorer window and locate the project folder.
13. 
14. Open the **dist** folder and locate the **BasicFX.jar** file
15. 
16. Copy the **BasicFX.jar** file to **c:\Temp** to make it easier to upload to the Raspberry Pi.
17. 

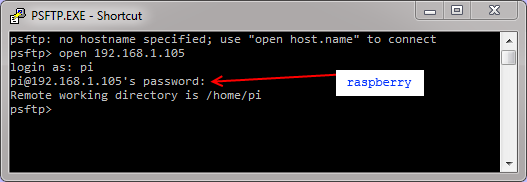
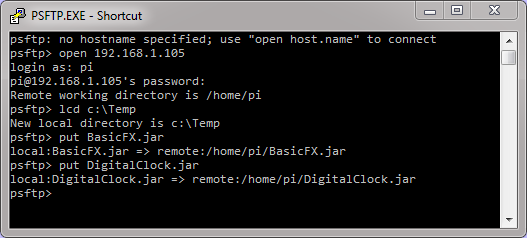
### Building the Clock Sample JavaFX Application

The NetBeans IDE includes sample applications that you can create and modify to showcase Java technologies. In this section, you create a JavaFX clock application by using one of these templates.

1. In NetBeans, select **File > New Project**.
2. 
3. Select **Samples > JavaFX** from Categories and **DigitalClock** from Projects and click **Next**.
4. 
5. Set a location for the project and click **Finish**.
6. 
7. The project is created with the code required to display a digital clock using JavaFX.
8. Right-click the project, select **Clean and Build**, and wait until the project is built.
9. 
10. Open a File Explorer window, browse to the **DigitalClock** project folder, and open the **dist** folder.
11. 
12. Copy the **DigitalClock.jar** file to the **c:\Temp** folder to make it easier to deploy it on the Raspberry Pi.
13. 

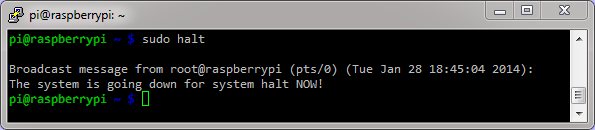
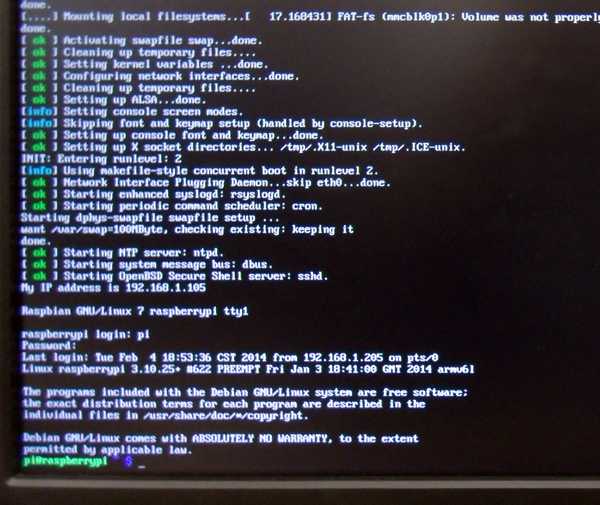
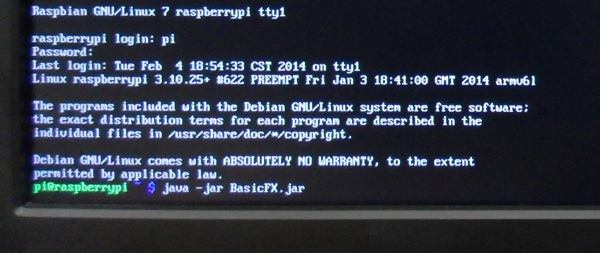
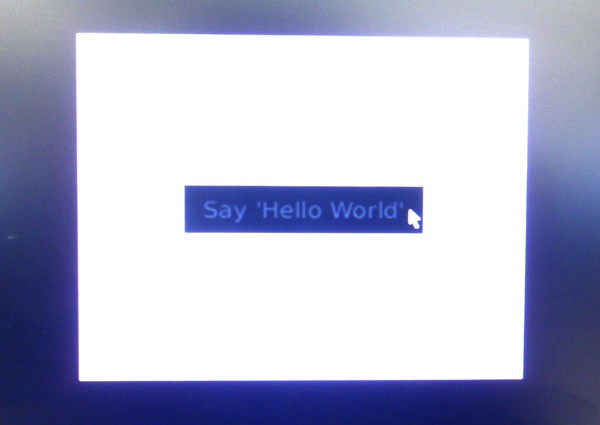
## Deploying, Running, and Stopping JavaFX Applications

### Transferring the JavaFX Application JARs to the Raspberry Pi by Using PSFTP

1. Launch the PSFTP tool and then perform the following steps:
   1. Enter **open 192.168.1.105**.
   2. Enter **pi** as the login name and **raspberry** as the password.
2. 
3. Perform the following steps to transfer the JavaFX application JAR files:
   1. Enter **lcd C:\Temp** to change the local directory to the location of the JDK 8 for Linux ARMv6/7.
   2. Enter **put BasicFX.jar**.
   3. Enter **put DigitalClock.jar**.
4. 
5. Type **exit** to quit the PSFTP tool.

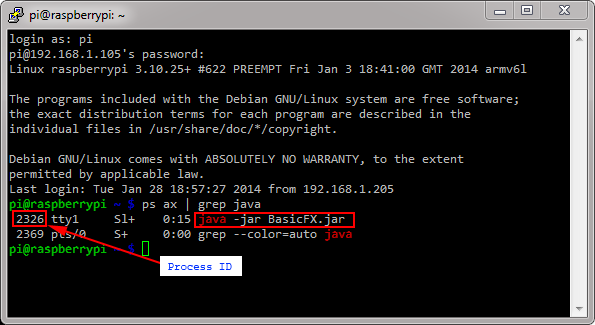
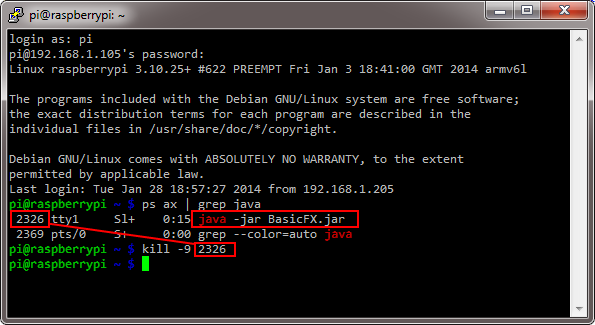
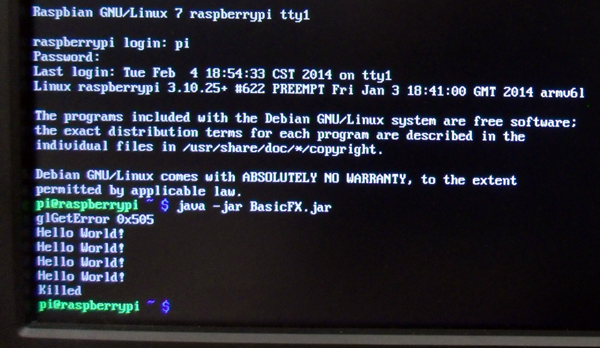
The applications are deployed on the Raspberry Pi.

### Running a JavaFX Application JAR in the Raspberry Pi

1. In a PuTTY window, enter **sudo halt** and wait for the LEDs to stop blinking.
2. 
3. When only the PWR LED is on, unplug the power cable from the Pi.
4. 
5. Connect the video cable, mouse, and keyboard to the Raspberry Pi, and then connect the power cable.
6. 
7. In the Pi, log in as the user **pi** with the password **raspberry**.
8. 
9. Enter **java -jar BasicFX.jar**.
10. 
11. The application is displayed on the Raspberry Pi screen.
12. 

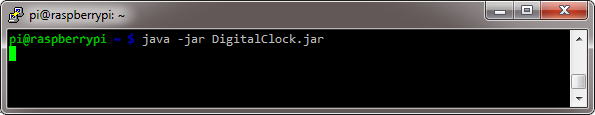
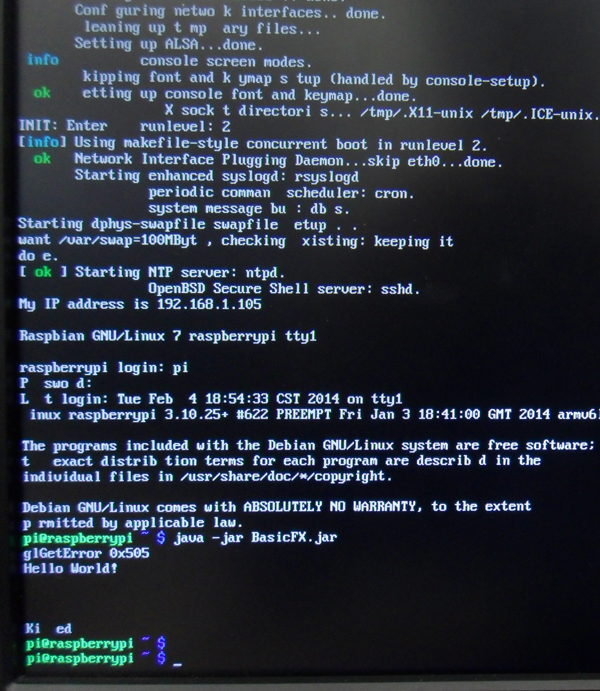
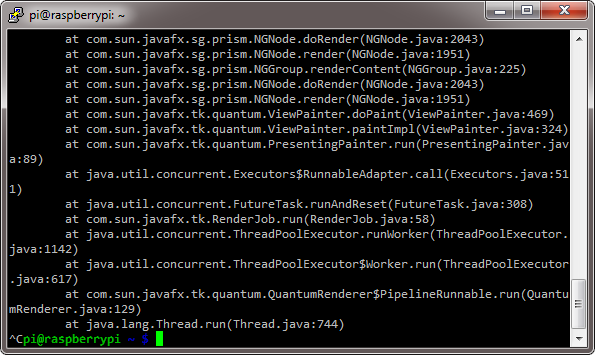
### Stopping a JavaFX Application

JavaFX applications are displayed on the main screen of the Raspberry Pi, but you can start and stop them from a remote console, which is useful when debugging or when the application cannot be stopped. In some cases, you can stop applications by pressing **Ctrl + C**. If the application doesn't stop, use the method outlined in this section.

1. On your Desktop PC, open a PuTTY window and log in to the Raspberry as the user **pi** with the password **raspberry**.
2. Enter **ps ax | grep java** and locate the process ID of the BasicFX.jar application.
3. 
4. Enter **sudo kill -9 2326** and replace 2326 with the process ID from the previous step.
5. 
6. Notice that the JavaFX application stops.
7. 

You can use this method whenever you cannot shut down an application normally and you need to forcefully stop it.

### Running a JavaFX Application Remotely

1. In the PuTTY window, enter **java -jar DigitalClock.jar**.
2. 
3. The Digital Clock application starts and runs in full screen mode.
4. 
5. In the PuTTY window, press **Ctrl + C** to stop the application.
6. 
7. If the screen is blank, press **Enter** to display the shell console.

## Summary

In this tutorial, you learned to:

* Create a bootable image for the Raspberry Pi
* Set up the Raspberry Pi for remote access
* Install JDK 8 on the Raspberry Pi
* Deploy and run JavaFX applications on the Raspberry Pi
* Run and stop JavaFX applications remotely

#### Resources

* [Raspberry Pi Foundation](http://www.raspberrypi.org/)
* [JDK 8 Project page](https://jdk8.java.net/)
* To learn more about Java 8 and the Raspberry Pi, refer to additional OBEs in the [Oracle Learning Library](http://www.oracle.com/goto/oll)

#### Credits

* Lead Curriculum Developer: Eduardo Moranchel
* Other Contributors: Tom McGinn