

Run Appium script with AppliTools Eyes SDK

Learn how to run an Appium script with AppliTools Eyes SDK on Kobiton devices.

Before you start

- Enable [AppliTools Eyes integration](#).
- Prepare an Appium script that implements the [AppliTools Eyes SDK](#).
- Obtain your Kobiton username and [API key](#).
- Obtain a [runner API key](#) from AppliTools.

Add server URL and capabilities to your script

To run on Kobiton devices, add the server URL and capabilities to your Appium script. See [this guide](#) for more details.

The following example is based on an Android test script written in Java from the [AppliTools Eyes SDK tutorial page](#). Alter the script from the tutorial in 3 places:

- Capabilities:

```
1 // Set desired capabilities.
2 DesiredCapabilities capabilities = new DesiredCapabilities();
3
4 capabilities.setCapability("platformName", "Android");
5 // Use either deviceName or udid to select the device
6 // capabilities.setCapability("deviceName", "Pixel 7");
7 capabilities.setCapability("udid", "f40d06b2-daa4-4b62-896e-c0e675135ecd ");
8 capabilities.setCapability("platformVersion", "");
9 capabilities.setCapability("app", "https://appliToolsnmlresources.z19.web.core.windows.net/TestApps/app-deb");
10 capabilities.setCapability("browserName", "");
11 capabilities.setCapability("automationName", "UiAutomator2");
12 // Use the kobiton:runtime capability below if you want to launch a Basic Appium 2 session
13 // capabilities.setCapability("kobiton:runtime", "appium");
```

- Kobiton server URL:

```
1 // Open the app.
2 WebDriver driver = new AndroidDriver(new URL("https://[username]:[api_key]@api.kobiton.com/wd/hub"), capabilities);
```

- Replace `[username]` and `[api_key]` with your Kobiton account details.
- Replace `api.kobiton.com` with the custom domain of your organization (if applicable).

- Additionally, replace `YOUR_API_KEY` below with the runner API key from AppliTools.

```
1 // Initialize the eyes SDK and set your private API key.
2 Eyes eyes = new Eyes();
3 eyes.setApiKey("YOUR_API_KEY");
```

At the time of writing, the final test script would look like the below:

```
1 import org.openqa.selenium.WebDriver;
2 import org.openqa.selenium.remote.DesiredCapabilities;
3 import io.appium.java_client.android.AndroidDriver;
4 import java.net.URL;
5 import com.appliTools.eyes.appium.Eyes;
```

```

6
7
8
9 public class Appium_native_java {
10
11     public static void main(String[] args) throws Exception {
12
13         // Set desired capabilities.
14         DesiredCapabilities capabilities = new DesiredCapabilities();
15
16         capabilities.setCapability("platformName", "Android");
17         // Use either deviceName or udid to select the device
18         // capabilities.setCapability("deviceName", "Pixel 7");
19         capabilities.setCapability("udid", "f40d06b2-daa4-4b62-896e-c0e675135ecd ");
20         capabilities.setCapability("platformVersion", "");
21         capabilities.setCapability("app", "https://appliitoolsnmlresources.z19.web.core.windows.net/TestApps/ap
22         capabilities.setCapability("browserName", "");
23         capabilities.setCapability("automationName", "UiAutomator2");
24         // Use the kobiton:runtime capability below if you want to launch a Basic Appium 2 session
25         // capabilities.setCapability("kobiton:runtime", "appium");
26
27         // Open the app.
28         WebDriver driver = new AndroidDriver(new URL("https://johndoe:6aba1969-720f-5453-9234-a65435823638@api.k
29
30         // Initialize the eyes SDK and set your private API key.
31         Eyes eyes = new Eyes();
32         eyes.setApiKey("ak9FuNVsyEsDi62buhydjLi7HASD5435fggdorEHXY110");
33
34         eyes.setForceFullPageScreenshot(true);
35
36         try {
37
38             // Start the test.
39             eyes.open(driver, "Contacts!", "My first Appium native Java test!");
40
41             // Visual validation.
42             eyes.checkWindow("Contact list!");
43
44             // End the test.
45             eyes.close();
46
47         } finally {
48
49             // Close the app.
50             driver.quit();
51
52             // If the test was aborted before eyes.close was called, ends the test as aborted.
53             eyes.abortIfNotClosed();
54
55         }
56
57     }
58
59 }

```

Execute the script. After the script finishes and Kobiton has received the status from Applitools, an Applitools Eyes icon is displayed for the session in the Session List. And, a new status will be available in the Automation section of Session Overview allowing you to view the Applitools Eyes status and access a link to go directly to Applitools Eyes to review the details.

Limitations

- The Eyes status and link to AppliTools in Session Explorer are not available for sessions Selenium with JavaScript. Visit [AppliTools Eyes](#) directly to view the status of these sessions.