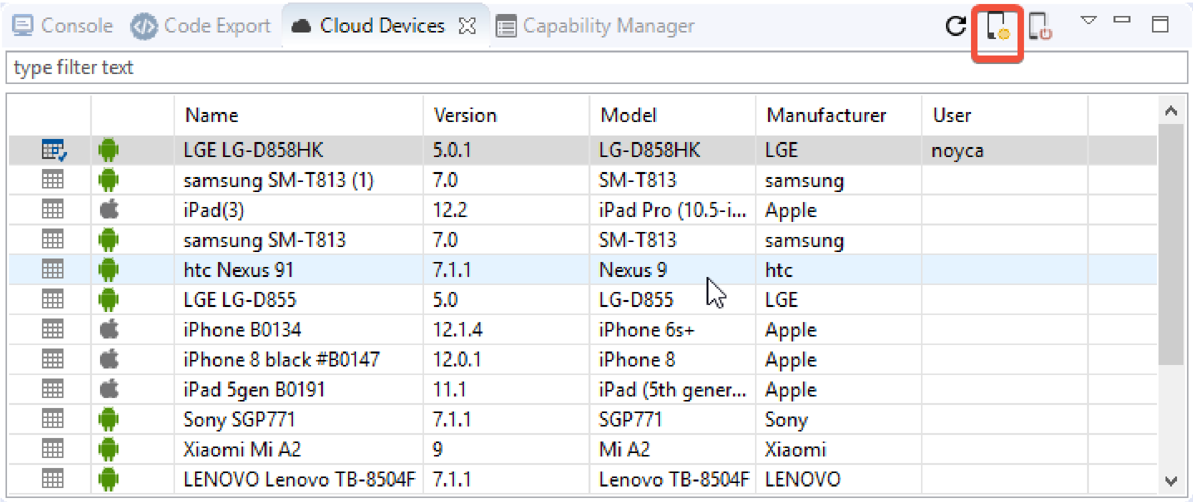
Appium Studio for Eclipse - Build your first test

The section below explains step by step procedure to develop an automated test case again Experitest Cloud Platform.

Open a device

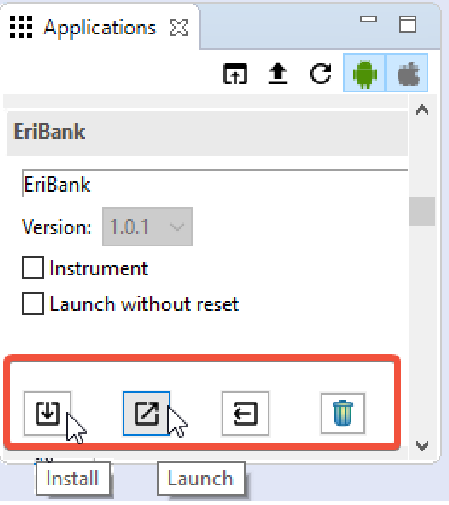
In the '**Cloud Devices**' view select an Android device and click on the '**Open**' button.



Name	Version	Model	Manufacturer	User
LGE LG-D858HK	5.0.1	LG-D858HK	LGE	noyca
samsung SM-T813 (1)	7.0	SM-T813	samsung	
iPad(3)	12.2	iPad Pro (10.5-i...	Apple	
samsung SM-T813	7.0	SM-T813	samsung	
htc Nexus 91	7.1.1	Nexus 9	htc	
LGE LG-D855	5.0	LG-D855	LGE	
iPhone B0134	12.1.4	iPhone 6s+	Apple	
iPhone 8 black #B0147	12.0.1	iPhone 8	Apple	
iPad 5gen B0191	11.1	iPad (5th gener...	Apple	
Sony SGP771	7.1.1	SGP771	Sony	
Xiaomi Mi A2	9	Mi A2	Xiaomi	
LENOVO Lenovo TB-8504F	7.1.1	Lenovo TB-8504F	LENOVO	

Install and launch your application

In the '**Applications**' view click on '**Install**'. Then click on '**Launch**'.

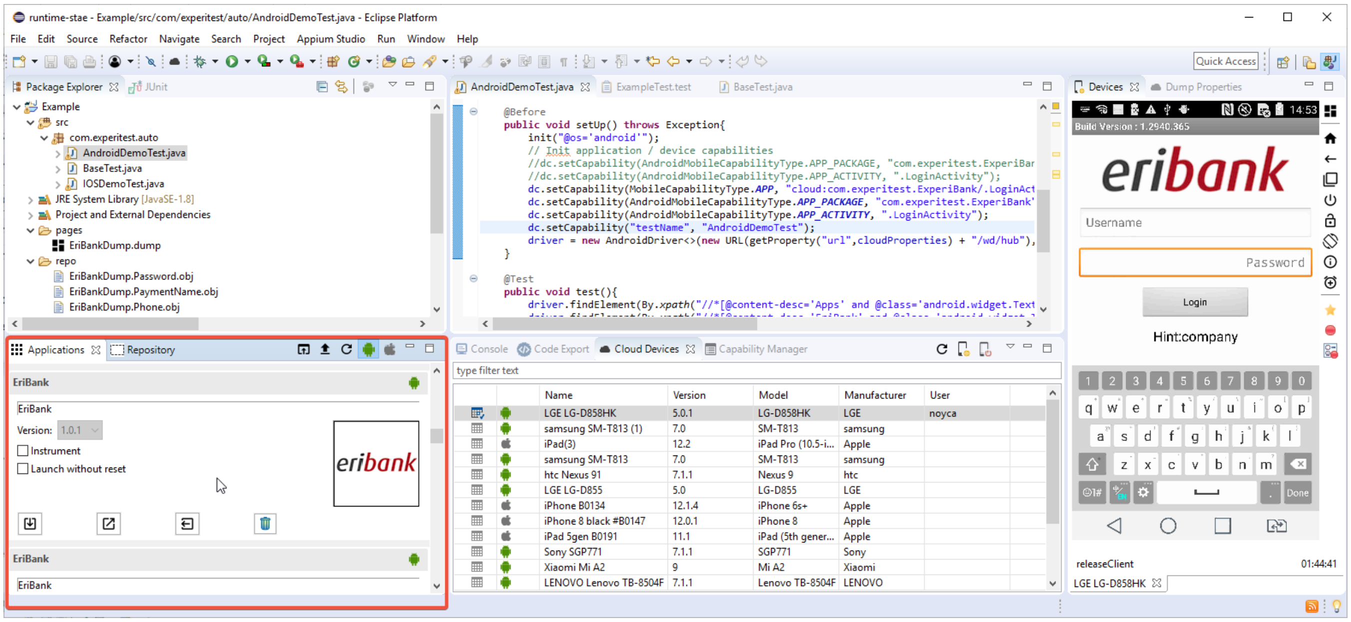


EriBank

Version: 1.0.1

□ Instrument

□ Launch without reset



```
@Before
public void setUp() throws Exception{
    init("@os='android'");
    // Init application / device capabilities
    //dc.setCapability(AndroidMobileCapabilityType.APP_PACKAGE, "com.experitest.ExperiBar
    //dc.setCapability(AndroidMobileCapabilityType.APP_ACTIVITY, ".LoginActivity");
    dc.setCapability(MobileCapabilityType.APP, "cloud:com.experitest.ExperiBank/.LoginAct
    dc.setCapability(AndroidMobileCapabilityType.APP_PACKAGE, "com.experitest.ExperiBank'
    dc.setCapability(AndroidMobileCapabilityType.APP_ACTIVITY, ".LoginActivity");
    dc.setCapability("testName", "AndroidDemoTest");
    driver = new AndroidDriver<>(new URL(getProperty("url",cloudProperties) + "/wd/hub"),
}

@Test
public void test(){
    driver.findElement(By.xpath("//*[@content-desc='Apps' and @class='android.widget.Text
```

EriBank

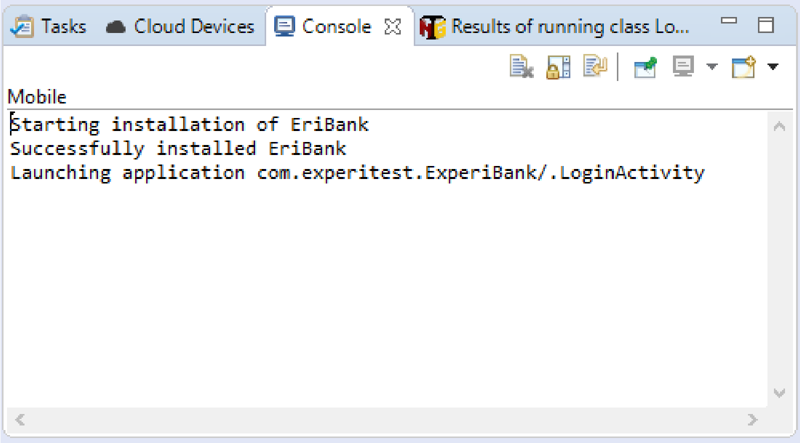
Version: 1.0.1

□ Instrument

□ Launch without reset

Name	Version	Model	Manufacturer	User
LGE LG-D858HK	5.0.1	LG-D858HK	LGE	noyca
samsung SM-T813 (1)	7.0	SM-T813	samsung	
iPad(3)	12.2	iPad Pro (10.5-i...	Apple	
samsung SM-T813	7.0	SM-T813	samsung	
htc Nexus 91	7.1.1	Nexus 9	htc	
LGE LG-D855	5.0	LG-D855	LGE	
iPhone B0134	12.1.4	iPhone 6s+	Apple	
iPhone 8 black #B0147	12.0.1	iPhone 8	Apple	
iPad 5gen B0191	11.1	iPad (5th gener...	Apple	
Sony SGP771	7.1.1	SGP771	Sony	
Xiaomi Mi A2	9	Mi A2	Xiaomi	
LENOVO Lenovo TB-8504F	7.1.1	Lenovo TB-8504F	LENOVO	

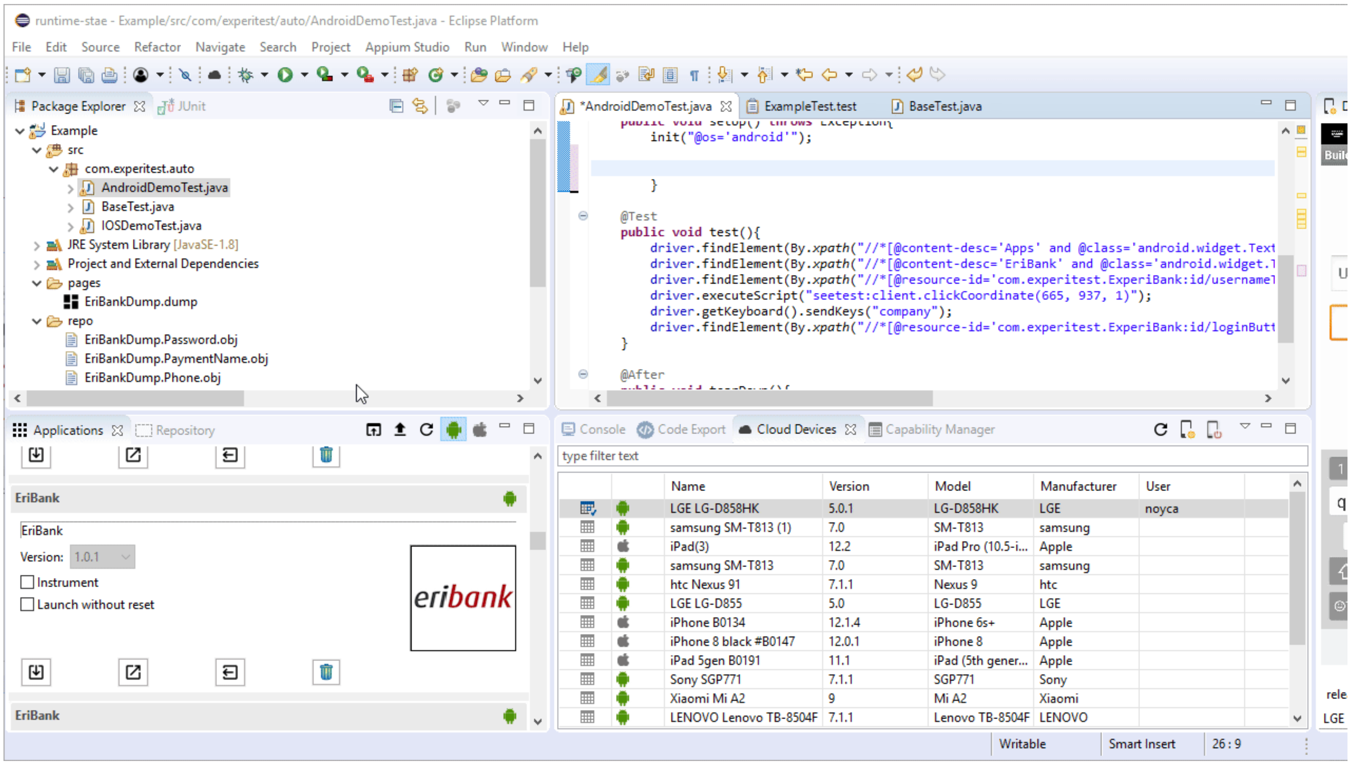
The '**Console**' view will show you a successful notification for both operations.



```
Mobile
Starting installation of EriBank
Successfully installed EriBank
Launching application com.experitest.ExperiBank/.LoginActivity
```

Application information to the Appium capabilities setup

The easiest way to add both the install and launch capabilities. Select the application section title in the 'Applications' view and drag it to the code area.



```java
        init("@os='android'");

    }

    @Test
    public void test(){
        driver.findElement(By.xpath("//*[@content-desc='Apps' and @class='android.widget.Text
        driver.findElement(By.xpath("//*[@content-desc='EriBank' and @class='android.widget.T
        driver.findElement(By.xpath("//*[@resource-id='com.experitest.ExperiBank:id/usernameT
        driver.executeScript("seetest:client.clickCoordinate(665, 937, 1)");
        driver.getKeyboard().sendKeys("company");
        driver.findElement(By.xpath("//*[@resource-id='com.experitest.ExperiBank:id/loginButt
    }

    @After
```

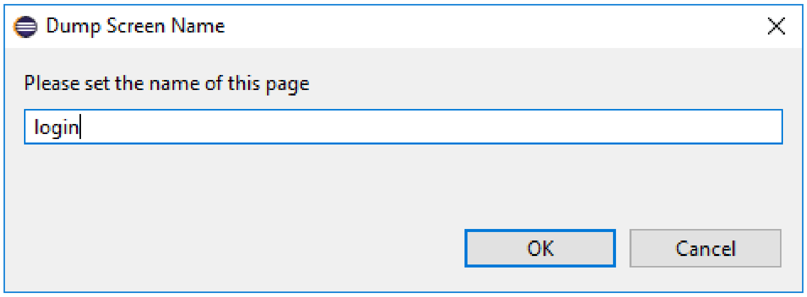
Name	Version	Model	Manufacturer	User
LGE LG-D858HK	5.0.1	LG-D858HK	LGE	noyca
samsung SM-T813 (1)	7.0	SM-T813	samsung	
iPad(3)	12.2	iPad Pro (10.5-i...	Apple	
samsung SM-T813	7.0	SM-T813	samsung	
htc Nexus 91	7.1.1	Nexus 9	htc	
LGE LG-D855	5.0	LG-D855	LGE	
iPhone B0134	12.1.4	iPhone 6s+	Apple	
iPhone 8 black #B0147	12.0.1	iPhone 8	Apple	
iPad 5gen B0191	11.1	iPad (5th gener...	Apple	
Sony SGP771	7.1.1	SGP771	Sony	
Xiaomi Mi A2	9	Mi A2	Xiaomi	
LENOVO Lenovo TB-8504F	7.1.1	Lenovo TB-8504F	LENOVO	

Write your test logic

Following section describes one of the many ways a test logic can be written.

Click on the '**Dump UI**' icon (▦) right to the **device reflection**.

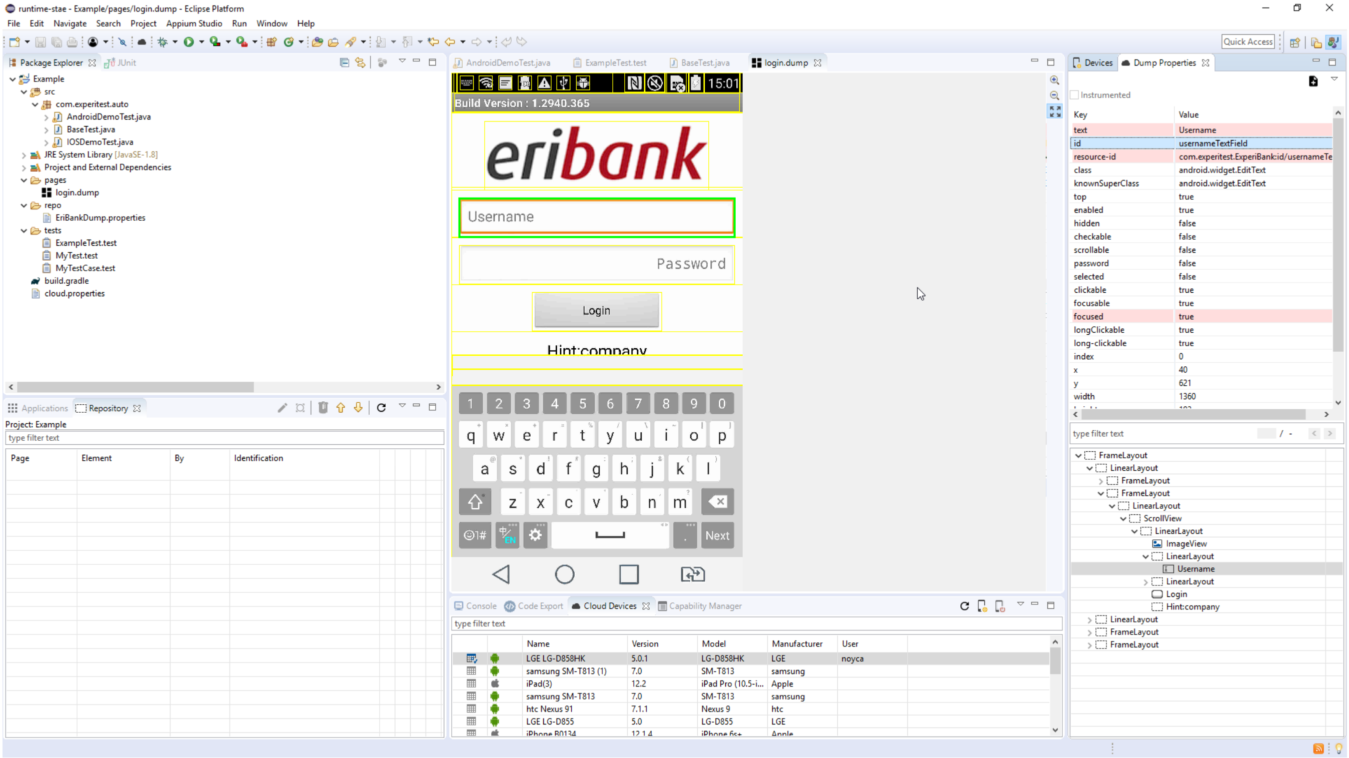
Provide the page name (in this case 'login') and click 'OK'.



Please set the name of this page

login

A '**dump**' file is created and is open by the editor.



Key	Value
text	Username
id	usernameTextField
resource-id	com.experitest.ExperiBank:id/usernameTe
class	android.widget.EditText
knownSuperClass	android.widget.EditText
top	true
enabled	true
hidden	false
checkable	false
scrollable	false
password	false
selected	false
clickable	true
focusable	true
focused	true
longClickable	true
long-clickable	true
index	0
x	40
y	621
width	1360

Name	Version	Model	Manufacturer	User
LGE LG-D858HK	5.0.1	LG-D858HK	LGE	noyca
samsung SM-T813 (1)	7.0	SM-T813	samsung	
iPad(3)	12.2	iPad Pro (10.5-i...	Apple	
samsung SM-T813	7.0	SM-T813	samsung	
htc Nexus 91	7.1.1	Nexus 9	htc	
LGE LG-D855	5.0	LG-D855	LGE	
iPhone B0134	12.1.4	iPhone 6s+	Apple	

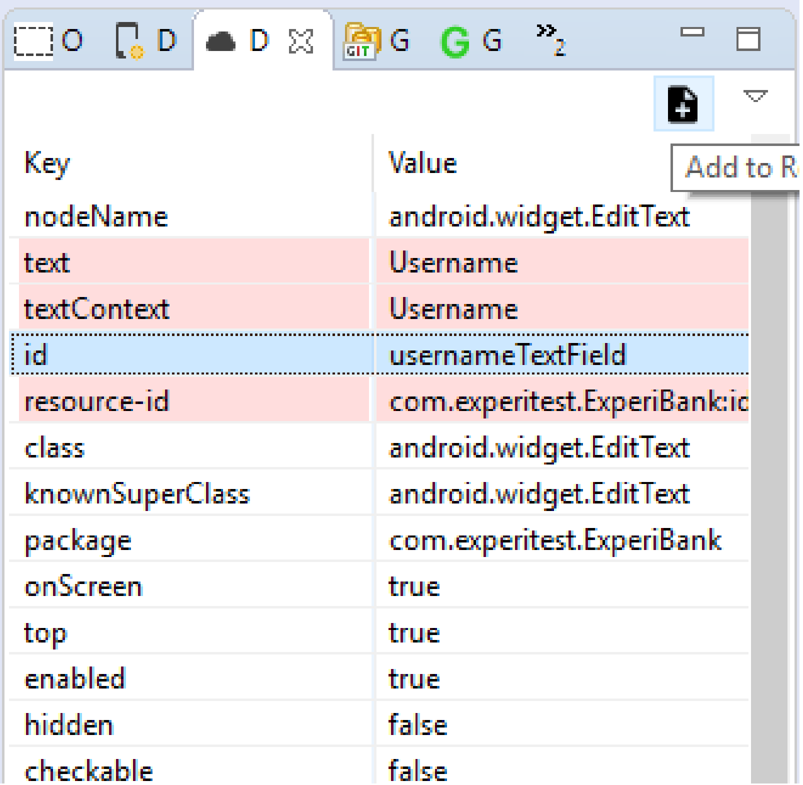
Select the element you would like to interact with (Example: 'Username' text field).

In the '**Dump Properties**' view, you will see all of the properties of the element.

✓ The lines that are colored in pink are properties that are unique to that page. Using those lines to identify the elements will assure unique identification.

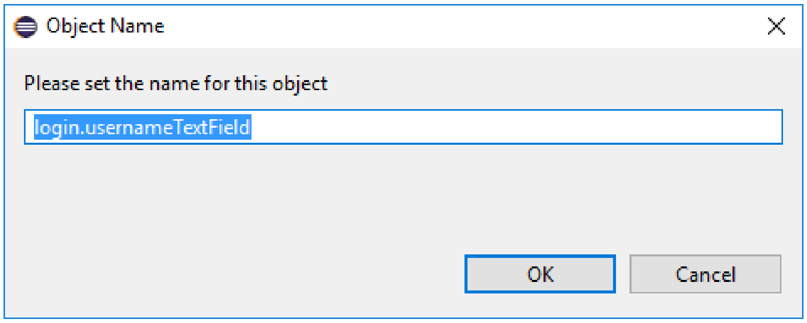
Select the property (or properties) you would like to use to **identify the element**.

Then click on '**Add to Repository**' icon (🗋).



Key	Value
nodeName	android.widget.EditText
text	Username
textContext	Username
id	usernameTextField
resource-id	com.experitest.ExperiBank:ic
class	android.widget.EditText
knownSuperClass	android.widget.EditText
package	com.experitest.ExperiBank
onScreen	true
top	true
enabled	true
hidden	false
checkable	false

Approve the name of elements in the repository. The element name is a compound of the page name and the element name separated by a dot ('.').

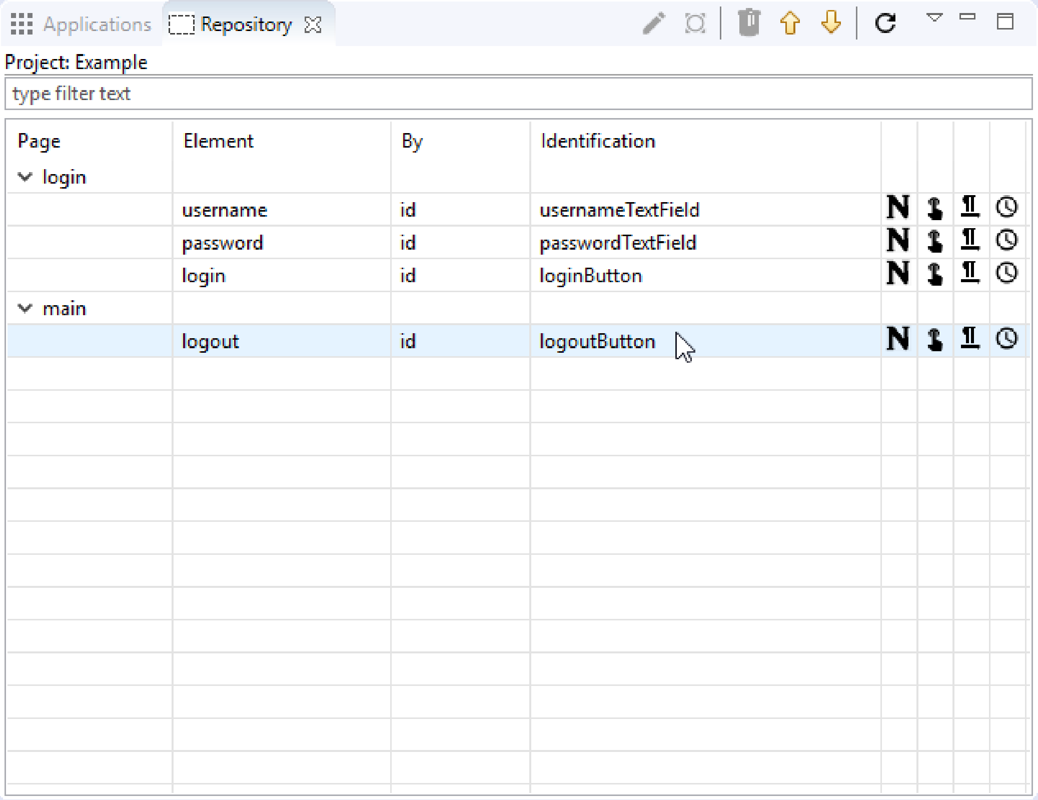


Please set the name for this object

login.usernameTextField

Once the element is approved it will be added to the repository (see the 'Repository' view).

Add all the 'interesting' elements on the page to the repository.



Project: Example

type filter text

Page	Element	By	Identification				
login							
	username	id	usernameTextField	N	👆	¶	🕓
	password	id	passwordTextField	N	👆	¶	🕓
	login	id	loginButton	N	👆	¶	🕓
main							
	logout	id	logoutButton	N	👆	¶	🕓

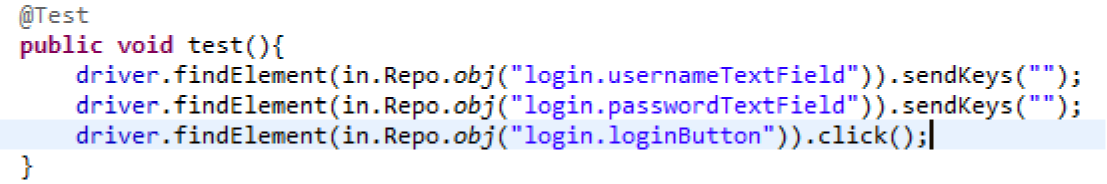
Build your test flow

Drag and drop the element into your test code.

👆 - dragging the 'Finger' icon will add a 'click' command.

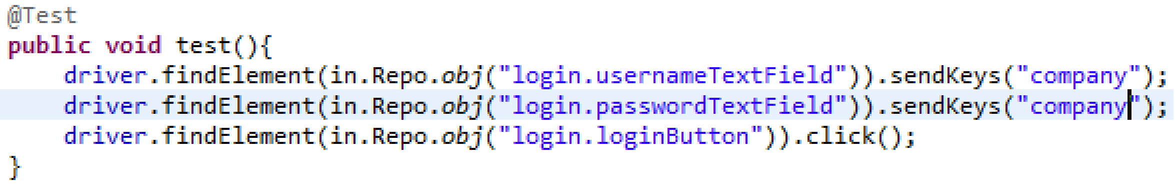
¶ - dragging the 'Text' icon will add a 'sendText' command.

In this case, I have used the 'Text' icon for both the username and the password elements, then I have used the 'Finger' icon for clicking on the login button.



```
@Test
public void test(){
    driver.findElement(in.Repo.obj("login.usernameTextField")).sendKeys("");
    driver.findElement(in.Repo.obj("login.passwordTextField")).sendKeys("");
    driver.findElement(in.Repo.obj("login.loginButton")).click();
}
```

To finalize the 'sendKeys' commands you need to provide the text need to be sent to the text field.



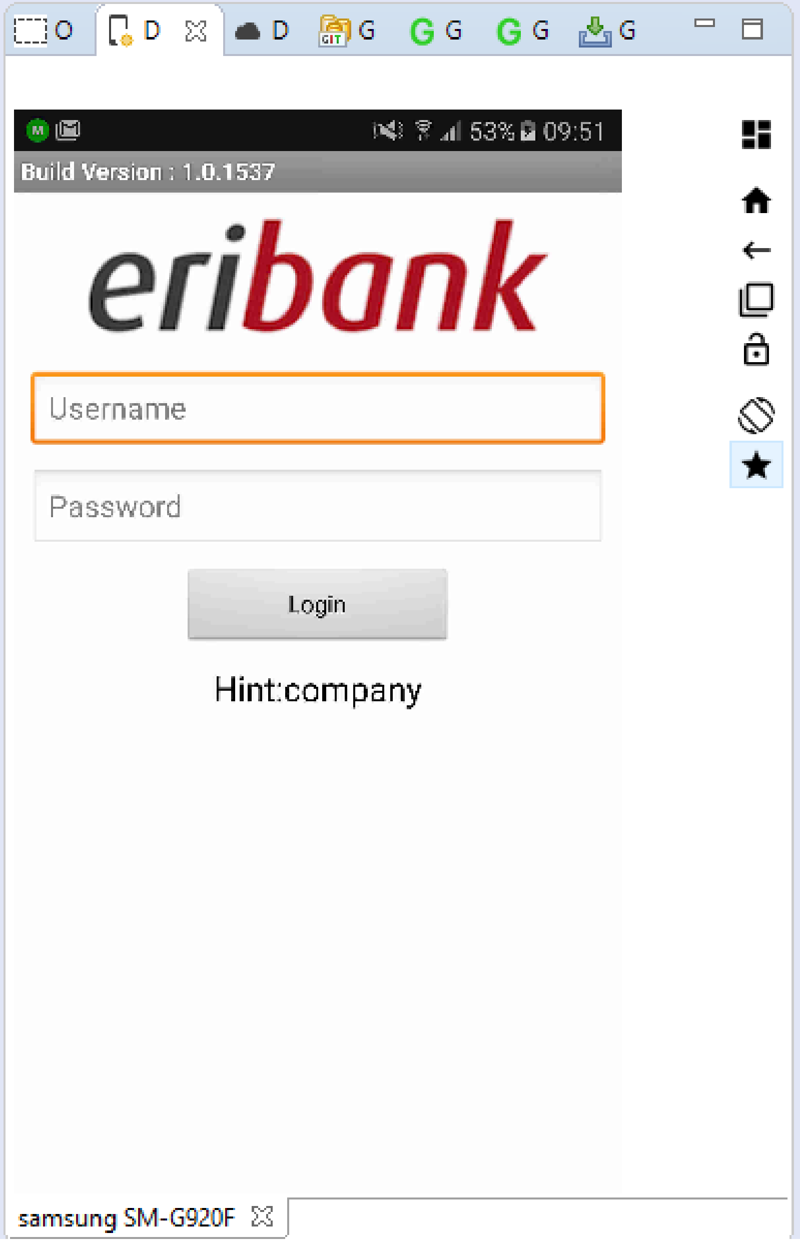
```
@Test
public void test(){
    driver.findElement(in.Repo.obj("login.usernameTextField")).sendKeys("company");
    driver.findElement(in.Repo.obj("login.passwordTextField")).sendKeys("company");
    driver.findElement(in.Repo.obj("login.loginButton")).click();
}
```

Run and verify your test code

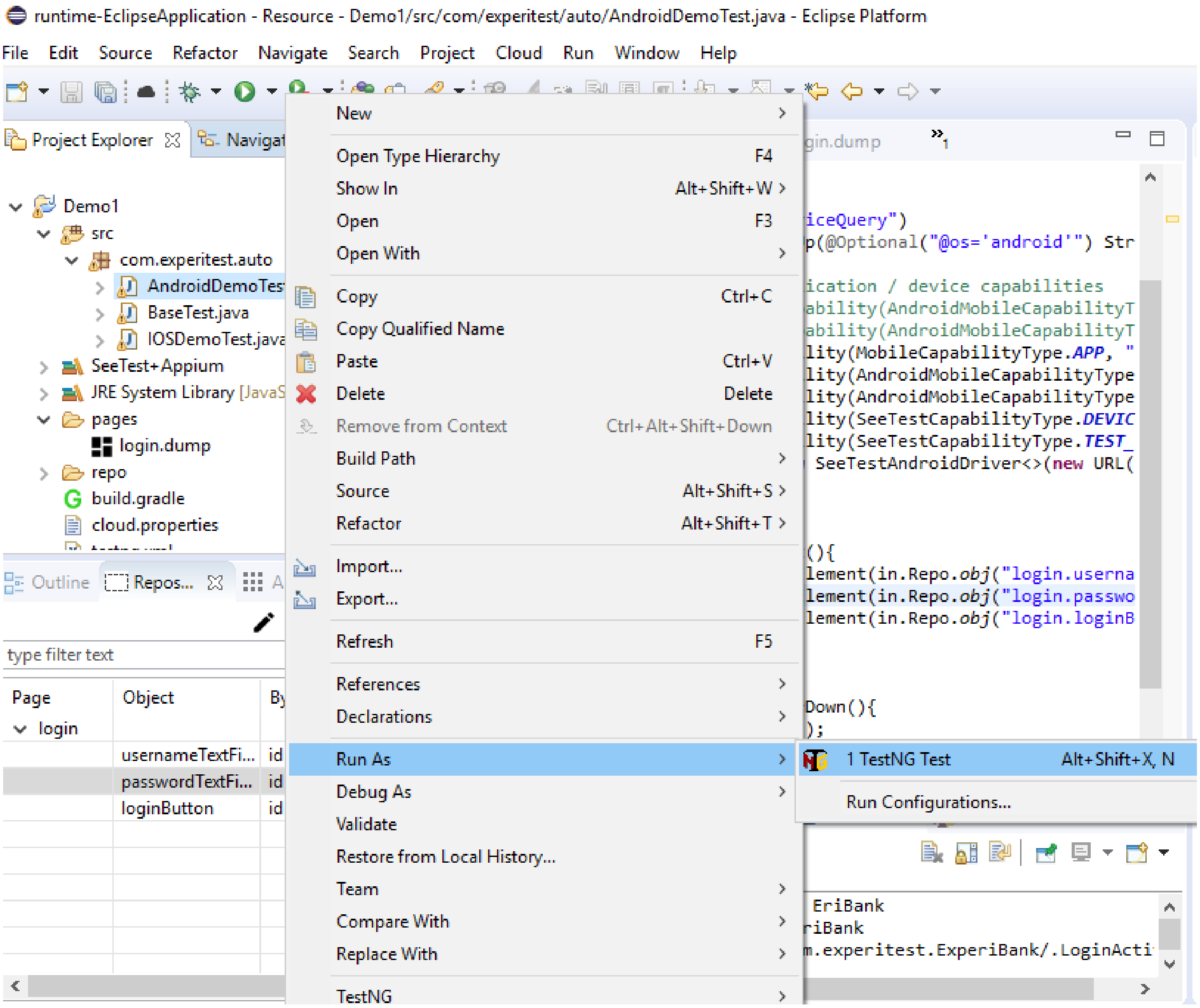
In the '**Devices**' view click on the '**Overwrite capabilities**' icon (★).

Clicking the '**Overwrite capabilities**' option will cause the execution of any test to be directed to this device (regardless of the device query capability), as well as ignoring the install and launch capability.

This way it is very easy to debug your test code without the need to reinstall the application every time.



Now run the test by right-clicking on the test class and selecting '**Run as**...'

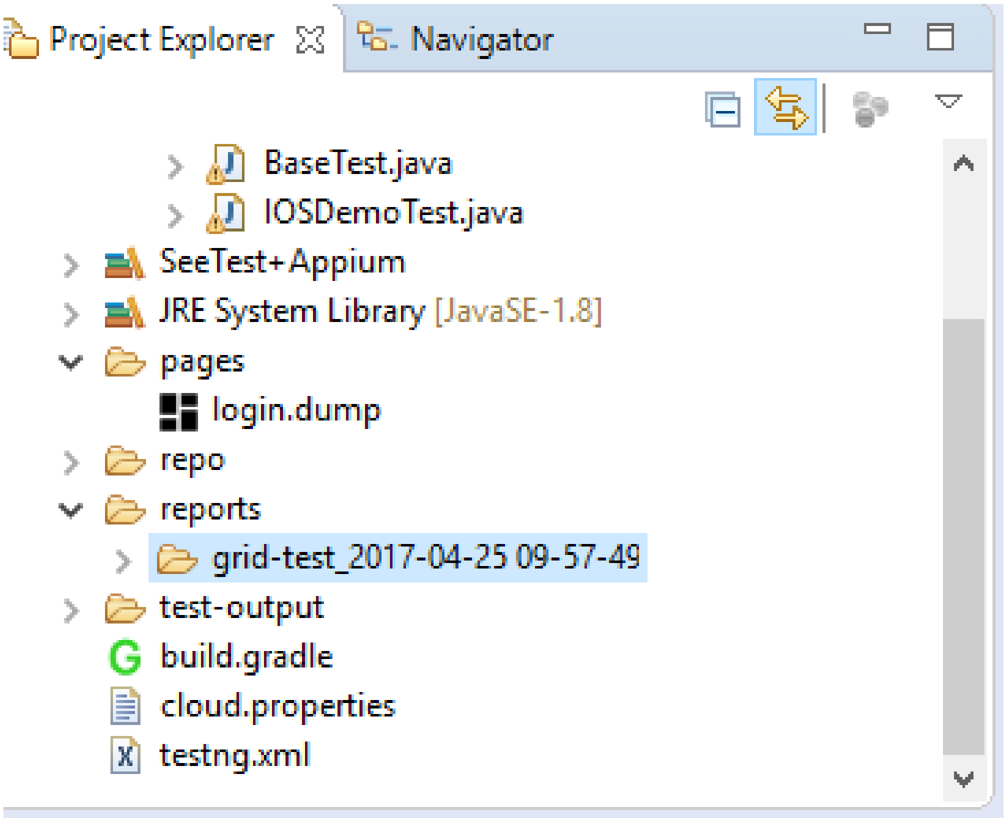


Page	Object	By
login		
	usernameTextFi...	id
	passwordTextFi...	id
	loginButton	id

```
iceQuery")
p(@Optional("@os='android'") Str

ication / device capabilities
ability(AndroidMobileCapabilityT
ability(AndroidMobileCapabilityT
lity(MobileCapabilityType.APP, "
lity(AndroidMobileCapabilityType
lity(AndroidMobileCapabilityType
lity(SeeTestCapabilityType.DEVIC
lity(SeeTestCapabilityType.TEST_
 SeeTestAndroidDriver<>(new URL(


(){
lement(in.Repo.obj("login.userna
lement(in.Repo.obj("login.passwo
lement(in.Repo.obj("login.loginB


Down(){
);
```

```
 EriBank
riBank
m.experitest.ExperiBank/.LoginActi
```

Once the execution finishes you can see the results in the '**reports**' folder under your project.



```
      BaseTest.java
      IOSDemoTest.java
  SeeTest+Appium
  JRE System Library [JavaSE-1.8]
  pages
     login.dump
  repo
  reports
     grid-test_2017-04-25 09-57-49
  test-output
  build.gradle
  cloud.properties
  testng.xml
```