

How to capture a screen shot of pass and failed Test Case In TestNg using listener

Create One Base Case as below

```
public class ParentTest {

    WebDriver driver;
    @BeforeTest
    public void setUp()
    {
        WebDriverManager.chromedriver().setup();
        driver=new ChromeDriver();
        System.out.println("inside setUp");
    }
    @AfterTest
    public void tearDown()
    {
        driver.quit();
        System.out.println("inside tearDown");
    }
}
```

Write screen capture logic inside a separate class and extend base class in that as below

```
public class Utility extends ParentTest{

    public void takeScreenShot(String fileName,String extention)
    {
        String file=getDateAndTime();
        String finalFileName=fileName+file;
        System.out.println("inside takeScreenShot");
        TakesScreenshot ts=(TakesScreenshot)driver;
        byte [] b=ts.getScreenshotAs(OutputType.BYTES);
        try {
            FileOutputStream fos=new
FileOutputStream("./screenshot/"+finalFileName+"."+extention);
            fos.write(b);
            fos.close();
        }
        catch (Exception e) {
            System.out.println("takeScreenShot "+e.getMessage());
        }
    }

    private static String getDateAndTime()
    {
        Date date = new Date();
        SimpleDateFormat formatter = new
SimpleDateFormat("ddMyyyhhmmss");
        String strDate= formatter.format(date);
        System.out.println("strDate "+strDate);
        return strDate;
    }
}
```

Another logic to capture the screen shot as **BYTE** type

```
7 public class ScreenshotDemoTest {
8
9     @Test
10     Run | Debug
11     public void takeScreenshot() throws IOException
12     {
13         WebDriverManager.chromedriver().setup();
14         WebDriver driver=new ChromeDriver();
15         driver.get("https://www.google.com");
16
17         TakesScreenshot ts=(TakesScreenshot)driver;
18
19         byte [] b=ts.getScreenshotAs(OutputType.BYTES);
20         FileOutputStream fos=new FileOutputStream("./screenshot/image3.png");
21         fos.write(b);
22         fos.close();
23         driver.close();
24     }
25 }
26 }
```

Another logic to capture the screen shot as **FILE** type and **Base 64** Type

```
TakesScreenshot ts=(TakesScreenshot)driver;
File file=ts.getScreenshotAs(OutputType.FILE);
FileUtils.copyFile(file, new File("./screenshot/image1.png"));

String fileString=ts.getScreenshotAs(OutputType.BASE64);
byte [] b=Base64.decode(fileString);
FileOutputStream fos=new FileOutputStream("./screenshot/image2.png");
fos.write(b);
fos.close();
```

Implements Listener as below to log event either pass or fail

```
public class ITestListnerDemo implements ITestListener{

    @Override
    public void onTestFailure(ITestResult result) {
        System.out.println("inside onTestFailure");
        Utility utility=new Utility();
        //test.takeScreenShot(result.getTestName());
        utility.takeScreenShot(result.getMethod().getMethodName(), ".png");
    }
}
```

TestNg Test class for Test Case is below

```
public class FailTestScreenShotDemo extends ParentTest{

    @Test(testName="openGoole")
    public void openGoogle()
    {
        driver.get("https://www.google.com/");
        String title=driver.getTitle();
        assertEquals(title, "Google123");
        System.out.println("inside openGoogle");
    }

    @Test(testName="loginHrm")
    public void loginHrm()
    {
        System.out.println("inside loginHrm");
        driver.get("https://opensource-demo.orangehrmlive.com/");
        driver.findElement(By.name("txtUsername")).sendKeys("Admin");
        driver.findElement(By.name("txtPassword")).sendKeys("admin123");
        driver.findElement(By.name("Submit")).click();
        System.out.println("inside loginHrm");
    }
}
```