

Group 5

Hackathon: Find Doctors

Team members: Subrat Paikaray, Seetha Bethala, Karan Lunkad, Sirigiri Sashidhar

Problem Statement: GSD

Get the Doctors names,

1. Male doctors.
2. 40+ Patient Stories.
3. Available in next 7 days.
4. 10+ years of experience.
5. Get all the details from GSD of Different Locations.

(Suggested Site:practo.com however you are free to choose any other legitimate site)

Detailed Description: Hackathon Ideas

1. Click on Find Doctors.
2. For Bangalore city, pick all the top places & store in a List; Display the same.
3. In Ear-nose-throat (ent) specialist page for Bangalore city, identify Doctors who are male, has 40+ Patient Stories, 10+ years of experience, Available in next 7 days, Experience – High to Low: Display top 5 Doctors names.
4. Go to Corporate Wellness, fill invalid details, schedule & capture the warning message from the alert

(Suggested Site:practo.com however you are free to choose any other legitimate site)

Note: Take all necessary screenshots for all the testcases and print all the data in the console.

Key Automation Scope

- Handling alert, different browser windows, search option.
- Navigating back to home page.
- Extract multiple options items & store in collections.
- Filling form (in different objects in web page)
- Capture warning message.

Steps:

Here are the steps to create an automation project with Maven, TestNG, and Page Object Model (POM) in Eclipse IDE:

Step 1: Install Eclipse IDE: Download and install the latest version of Eclipse IDE from the official website.

Step 2: Create a New Maven Project:

Open Eclipse IDE and go to File -> New -> Maven Project.

Step 3: In the New Maven Project window, select Create a simple project (skip archetype selection) and click Next.

Enter the Group Id (usually the package name), Artifact Id (project name), and Version for your project. Click Finish.

Step 4: Add Dependencies in pom.xml:

Open the pom.xml file of your project.

Add dependencies for Selenium, TestNG, and any other libraries you need.

Save the pom.xml file. Maven should automatically start downloading the specified dependencies.

Step 5: Create TestNG Class:

Go to src/test/java in your project structure, right-click on the package and select New -> Class.

Give your class a name and click Finish.

Now you can start writing your TestNG tests in this class.

Step 6: Implement Page Object Model (POM):

Create a new package under src/main/java for your page objects.

For each page in your web application, create a corresponding class in this package.

In each page class, create methods for each action you can perform on that page (clicking buttons, entering text, etc.). Use the Selenium WebDriver API to interact with the web elements.

Step 7: Run Your Tests:

You can run your tests by right-clicking on your TestNG class and selecting Run As -> TestNG Test.

TestNG will execute all your tests and provide a report in the TestNG view.

Folder Structure:

```

v Hackathon_Practo_Build1
  > src/main/java
  > src/main/resources
  v src/test/java
    v pageObjectModel
      > FindAndBook.java
      > FormPage.java
      > Mydriver.java
      > PractoHome.java
      > ResultPage.java
    v test
      > FindAndBookTest.java
      > FormPageTest.java
      > PractoHomeTest.java
      > ResultPageTest.java
      > testng.xml
    v Utility
      > ExtendReport.java
```

CODE

pageObjectModel Package

FindAndBook.java

```

package pageObjectModel;

import java.time.Duration;
import java.util.List;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.PageFactory;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;

public class FindAndBook {
    WebDriver driver;
    WebDriverWait wait;

    public FindAndBook(WebDriver driver) {
        super();
        this.driver = driver;
        PageFactory.initElements(driver, this);
    }
}
```

```

    }

    @FindBy(xpath = "//input[@data-input-box-id='omni-searchbox-locality']")
    WebElement locationBoxElement;

    @FindBy(xpath = "//div[@class='c-omni-suggestion-item__content__title']")
    WebElement popularLocationDropDownElement;

    @FindBy(xpath = "//div[@class='c-omni-suggestion-item__content__title']")
    List<WebElement> popularlocationList;

    @FindBy(xpath = "//*[@data-qa-id='omni-suggestion-right'][text()='CITY']")
    WebElement suggestedCityElement;

    @FindBy(xpath = "//input[@data-qa-id='omni-searchbox-keyword']")
    WebElement specialistBoxElement;

    @FindBy(xpath = "//div[@data-qa-id='omni-suggestion-main']"[contains(normalize-space(), 'Ear-nose-
throat (ent) Specialist']")
    WebElement specialistElement;

    @FindBy(xpath = "//h1[@class='u-xx-large-font u-bold']")
    WebElement noOfDoctorElement;
    // WebElement noOfDoctorElement = driver.findElement(By.xpath("//h1[@class='u-xx-large-font u-
bold']"));
    // System.out.println(noOfDoctorElement.getText());

    public void extractingLocation(String city) {
        String userString = city;
        wait = new WebDriverWait(driver, Duration.ofSeconds(40));
        if (userString.equalsIgnoreCase("Bangalore")) {
            locationBoxElement.click();
            wait.until(ExpectedConditions.visibilityOf(popularLocationDropDownElement));

            //System.out.println(popularlocationList.size());
            for (WebElement x : popularlocationList) {
                System.out.println(x.getText());
            }
            Mydriver.screenShot(userString+" Places");
        } else {
            locationBoxElement.click();
            locationBoxElement.clear();
            locationBoxElement.sendKeys(userString);
            wait.until(ExpectedConditions.visibilityOf(suggestedCityElement));
            suggestedCityElement.click();
            locationBoxElement.click();
            wait.until(ExpectedConditions.visibilityOf(popularLocationDropDownElement));
            Mydriver.screenShot("location");
            //System.out.println(popularlocationList.size());
            for (WebElement x : popularlocationList) {
                System.out.println(x.getText());
            }

            Mydriver.screenShot(userString+" Places");
        }
    }

```

```

    }

    public void findingDoctor() {
        System.out.println(driver);
        specialistBoxElement.click();
        specialistBoxElement.sendKeys("ENT");
        wait.until(ExpectedConditions.visibilityOf(specialistElement));
        Mydriver.screenshot("Specialist");
        specialistElement.click();
        System.out.println(noOfDoctorElement.getText());
    }

    public void noOfDoctorPrint() {
        System.out.println(noOfDoctorElement.getText());
    }
}

```

MyDriver.java

FormPage.java

```

package pajeObjectModel;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.PageFactory;
import org.openqa.selenium.support.ui.Select;
import org.openqa.selenium.support.ui.WebDriverWait;

public class FormPage {
    WebDriver driver;
    WebDriverWait wait;

    public FormPage(WebDriver driver) {
        super();
        this.driver = driver;
        PageFactory.initElements(driver, this);
    }

    @FindBy(xpath = "//input[@id='name'][@name='name']")
    WebElement nameElement;

    @FindBy(xpath = "//input[@id='organizationName'][@name='organizationName']")
    WebElement organizationNameElement;

    @FindBy(xpath = "//input[@id='contactNumber'][@name='contactNumber']")
    WebElement contactElement;

    @FindBy(xpath = "//input[@id='officialEmailId']")
    WebElement officialEmailIdElement;

    @FindBy(xpath = "//select[@id='organizationSize']")

```

```

WebElement organizationSizElement;

@FindBy(xpath = "//select[@id='interestedIn']")
WebElement intrestedInElement;

@FindBy(xpath = "//header[@id='header']/button[contains(@type,'submit')][normalize-space()='Schedule
a demo']")
WebElement scheduleElement;

public Boolean formtester(String type, String name, String organizationName, String mobile, String email,
    String organizationSize, String intrest) {
    Mydriver.screenShot("Form page");
    Boolean statusBoolean = null;
    if (type.equalsIgnoreCase("positive")) {
        nameElement.sendKeys(name);
        organizationNameElement.sendKeys(organizationName);
        contactElement.sendKeys(mobile);
        officialEmailIdElement.sendKeys(email);

        Select organizationsizeselect = new Select(organizationSizElement);
        organizationsizeselect.selectByValue(organizationSize);
        Select intrestedSelect = new Select(intrestedInElement);
        intrestedSelect.selectByValue(intrest);

        statusBoolean = scheduleElement.isEnabled();

        // Assert.assertEquals(statusBoolean, true);
        // driver.navigate().to("https://www.practo.com/plus/corporate");
        String urlString = driver.getCurrentUrl();
        driver.navigate().to(urlString);

    } else if (type.equalsIgnoreCase("negative")) {
        nameElement.sendKeys(name);
        organizationNameElement.sendKeys(organizationName);
        contactElement.sendKeys(mobile);
        officialEmailIdElement.sendKeys(email);
        Select organizationsizeselect = new Select(organizationSizElement);
        organizationsizeselect.selectByValue(organizationSize);

        Select intrestedSelect = new Select(intrestedInElement);
        intrestedSelect.selectByValue(intrest);

        statusBoolean = scheduleElement.isEnabled();
        Assert.assertEquals(statusBoolean, false);
        String urlString = driver.getCurrentUrl();
        driver.navigate().to(urlString);
    }
    return statusBoolean;
}

public void testClose() {
    driver.quit();
}
}

```

MyDriver.java

```
package pajeObjectModel;

import java.io.File;

import java.text.SimpleDateFormat;
import java.time.Duration;
import java.util.Date;

import org.apache.commons.io.FileUtils;
import org.openqa.selenium.OutputType;
import org.openqa.selenium.TakesScreenshot;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.edge.EdgeDriver;

public class Mydriver {
    private static WebDriver driver;

    public static WebDriver getDriver() {
        return driver;
    }

    public static WebDriver getDriver(String browser1) {

        if (browser1.equalsIgnoreCase("chrome")) {
            if (driver == null) {
                driver = new ChromeDriver(); // or any other driver
                driver.manage().window().maximize();
                driver.get("https://www.practo.com/");
                driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(20));
            }
        }

        if (browser1.equalsIgnoreCase("edge")) {
            if (driver == null) {
                driver = new EdgeDriver(); // or any other driver
                driver.manage().window().maximize();
                driver.get("https://www.practo.com/");
                driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(20));
            }
        }

        return driver;
    }

    public static void updateDriver(WebDriver newdriver) {
        driver = newdriver;
    }

    public static void clearDriver() {
        driver=null;
    }
}
```

```

    public static void screenshot(String name){
        TakesScreenshot ts=(TakesScreenshot) driver;
        String timeStamp = new SimpleDateFormat("yyyy.MM.dd.HH.mm.ss").format(new Date());
        String imgName=name+timeStamp+".png";
        File systemFile=ts.getScreenshotAs(OutputType.FILE);
        File myFile=new File(System.getProperty("user.dir")+ "/ScreenShots/"+ imgName);
        try {
            FileUtils.copyFile(systemFile, myFile);
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}

```

PractoHome.java

```

package pageObjectModel;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.PageFactory;

public class PractoHome {
    WebDriver driver;

    public PractoHome(WebDriver driver) {
        super();
        this.driver = driver;
        PageFactory.initElements(driver, this);
    }

    @FindBy(xpath = "//div[@class='product-tab__title'][text()='Find Doctors']")
    WebElement findDoctElement;

    public void clickOnFindDoctor() {
        Mydriver.screenShot("Homepage");
        findDoctElement.click();
    }

    public WebDriver driver_update() {
        return driver;
    }
}

```


ResultPage.java

```
package pajeObjectModel;

import java.time.Duration;
import java.util.List;

import org.openqa.selenium.TakesScreenshot;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.support.FindBy;
import org.openqa.selenium.support.PageFactory;
import org.openqa.selenium.support.ui.ExpectedCondition;
import org.openqa.selenium.support.ui.ExpectedConditions;
import org.openqa.selenium.support.ui.WebDriverWait;

public class ResultPage {
    WebDriver driver;
    WebDriverWait wait;
    TakesScreenshot ts;
    String currentUrl;

    public ResultPage(WebDriver driver) {
        super();
        this.driver = driver;
        PageFactory.initElements(driver, this);
    }

    @FindBy(xpath = "//div[@data-qa-id=\"doctor_gender_section\"]")
    WebElement genderFilterBoxElement;

    @FindBy(xpath = "//li[@data-qa-id=\"male\"]")
    WebElement maleDoctorElement;

    @FindBy(xpath = "//*[ @data-qa-id=\"doctor_review_count_selected\"]")
    WebElement patientStoriesBoxElement;

    @FindBy(xpath = "//li[@aria-label='40+ Patient Stories']")
    WebElement fortypluspatientStoriesElement;

    @FindBy(xpath = "//li[@aria-label='20+ Patient Stories']")
    WebElement twentypluspatientStoriesElement;

    @FindBy(xpath = "//div[@data-qa-id=\"years_of_experience_section\"]")
    WebElement experiencBoxElement;

    @FindBy(xpath = "//li[@aria-label='10+ Years of experience']")
    WebElement tenYearExperienceElement;

    @FindBy(xpath = "//div[@class='u-d-inlineblock u-color--white u-c-pointer']")
    WebElement allFilterBoxElement;

    @FindBy(xpath = "//label[@for='Availability3']")
    WebElement availabiltyElement;

    @FindBy(xpath = "//span[@class='c-sort-dropdown__selected c-dropdown__selected']")
```

```

WebElement shortByBoxElement;

@FindBy(xpath = "//span[normalize-space()='Experience - High to Low']")
WebElement highExperienceElement;

@FindBy(xpath = "//div/h2[@data-qa-id=\"doctor_name\"]")
List<WebElement> doctorNameElement;

@FindBy(xpath = "//div[@class='para cushion-right-large display-inline nav-items nav-items--additional-link hover-dark u-d-trigger dropdown-toggle']")
WebElement forCorporateWebElement;

@FindBy(xpath = "//a[normalize-space()='Health & Wellness Plans']")
WebElement healthyAndWellnessElement;

public void waitForUrlChange(WebDriver driver) {
    currentUrl = driver.getCurrentUrl();
    WebDriverWait wait = new WebDriverWait(driver, Duration.ofSeconds(10));
    wait.until(new ExpectedCondition<Boolean>() {
        public Boolean apply(WebDriver driver) {
            return !driver.getCurrentUrl().equals(currentUrl);
        }
    });
}

public void clickOnmale() {
    Mydriver.screenShot("Initial Result");
    genderFilterBoxElement.click();
    maleDoctorElement.click();
    waitForUrlChange(driver);
}

public void clickonpatientstories() {
    patientStoriesBoxElement.click();
    fortypluspatientStoriesElement.click();
    waitForUrlChange(driver);
}

public void clickonexp() {
    experienceBoxElement.click();
    tenYearExperienceElement.click();
    waitForUrlChange(driver);
}

public void clickOnAllFilter() {
    allFilterBoxElement.click();
    availabilityElement.click();
    waitForUrlChange(driver);
}

public void clickOnSortBy() {
    wait=new WebDriverWait(driver, Duration.ofSeconds(30));
    shortByBoxElement.click();
}

```

```

        wait.until(ExpectedConditions.visibilityOf(highExperienceElement));
        highExperienceElement.click();

        System.out.println("After Applying Filter total number of doctor");
        FindAndBook findAndBookObj2=new FindAndBook(Mydriver.getDriver());
        findAndBookObj2.noOfDoctorPrint();

    }

    public void doctorDetails() {
        Mydriver.screenShot("After Filter result");
        for (WebElement x : doctorNamElement) {
            System.out.println(x.getText());
        }
    }

    public void corporatePageNavigator() {

        forCorporateWebElement.click();

        healthyAndWellnessElement.click();
    }

}

```

Tests Package

FindAndBookTest.java

```

package test;

import org.openqa.selenium.WebDriver;
import org.openqa.selenium.support.ui.WebDriverWait;
import org.testng.annotations.Parameters;
import org.testng.annotations.Test;

import pageObjectModel.FindAndBook;
import pageObjectModel.Mydriver;

public class FindAndBookTest {
    WebDriver driver;
    WebDriverWait wait;
    FindAndBook findAndBookObject;

    @Test(priority = 3)
    @Parameters({"city"})
    void popularPlace(String city) throws InterruptedException {
        findAndBookObject=new FindAndBook(Mydriver.getDriver());
        findAndBookObject.extractingLocation(city);
    }

    @Test(priority = 4)
    void speciality() {
        findAndBookObject.findingDoctor();
    }
}

```

```
}
```

FormPageTest.java

```
package test;
```

```
import org.openqa.selenium.WebDriver;  
import org.testng.Assert;  
import org.testng.annotations.BeforeClass;  
import org.testng.annotations.DataProvider;  
import org.testng.annotations.Test;
```

```
import pageObjectModel.FormPage;  
import pageObjectModel.Mydriver;
```

```
public class FormPageTest {  
    WebDriver driver;  
    FormPage formPageObject;  
    @BeforeClass  
    void setup() {  
        formPageObject=new FormPage(Mydriver.getDriver());  
    }  
  
    @Test(priority = 13, dataProvider = "formData")  
    void formtest(String type, String name, String organizationName, String mobile, String email, String  
organizationSize,  
                String intrest) throws InterruptedException {  
  
        Boolean actual=formPageObject.formtester(type, name, organizationName, mobile, email,  
organizationSize, intrest);  
        if(type.equalsIgnoreCase("positive")) {  
            Assert.assertEquals(actual, true);  
        } else {  
            Assert.assertEquals(actual, false);  
        }  
    }  
    @DataProvider(name = "formData")  
    String[][] formData() {  
        String data[][] = {  
  
                { "positive", "Subart", "NewCognizant", "9090904935", "Subrat@xyz.com",  
"1001-5000",  
                "Referring someone" },  
                { "negative", "Subart", "NewCognizant", "90909049", "Subrat@xyz.com",  
"1001-5000",  
                "Referring someone" },  
                { "negative", "Subart", "NewCognizant", "9090904935", "Subrat@xyz", "1001-  
5000",  
                "Referring someone" },  
        };  
  
        return data;  
    }  
}
```

```

        @Test(priority = 14)
        void tearDown() {
            Mydriver.clearDriver();
            formPageObject.testClose();
        }
    }
}

```

PractoHomeTest.java

```

package test;

import org.openqa.selenium.WebDriver;
import org.testng.annotations.Parameters;
import org.testng.annotations.Test;

import pageObjectModel.Mydriver;
import pageObjectModel.PractoHome;

public class PractoHomeTest {

    WebDriver driver;
    PractoHome practoHomeObject;

    @Test(priority = 1)
    @Parameters({"browser"})
    void setUp(String browser) {
        // driver = new ChromeDriver();
        // driver.manage().window().maximize();
        // driver.get("https://www.practo.com/");
        // driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(50));
        driver=Mydriver.getDriver(browser);
        practoHomeObject=new PractoHome(driver);
    }

    @Test(priority = 2)
    void searchDoctor() {
        practoHomeObject.clickOnFindDoctor();
        Mydriver.updateDriver(practoHomeObject.driver_update());
    }
}

```

ResultPageTest.java

```

package test;

import org.openqa.selenium.JavascriptExecutor;
import org.openqa.selenium.StaleElementReferenceException;
import org.openqa.selenium.WebDriver;
import org.testng.annotations.Test;

```

```

import pageObjectModel.Mydriver;
import pageObjectModel.ResultPage;

public class ResultPageTest {
    //WebDriver driver = new ChromeDriver();
    WebDriver driver;
    ResultPage resultPageObject;
    JavascriptExecutor jsExecutor;

    @Test(priority = 5)
    void genderFilter() throws StaleElementReferenceException {
        resultPageObject=new ResultPage(Mydriver.getDriver());
        resultPageObject.clickOnmale();
    }

    @Test(priority = 6)
    void patientStoriesFilter() {
        resultPageObject.clickonpatientstories();
    }

    @Test(priority = 7)
    void experienceFilter() {
        resultPageObject.clickonexp();
    }

    @Test(priority = 8)
    void allFilters() {
        resultPageObject.clickOnAllFilter();
    }

    @Test(priority = 9)
    void shortByFilter() {
        resultPageObject.clickOnSortBy();
    }

    @Test(priority = 11)
    void doctorDetails() {
        resultPageObject.doctorDetails();
    }

    @Test(priority = 12)
    void navigatingToCorporate() {
        resultPageObject.corporatePageNavigator();
    }
}

```

testing.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">
<suite name="Suite">

    <listeners>
        <listener class-name="Utility.ExtendReport" />
    
```

```

</listeners>
<test thread-count="5" name="Test" >
    <parameter name="browser" value="chrome" />
    <parameter name="city" value="Bangalore" />
    <classes>
        <class name="test.PractoHomeTest" />
        <class name="test.FindAndBookTest" />
        <class name="test.ResultPageTest" />
        <class name="test.FormPageTest" />
    </classes>
</test> <!-- Test -->

<test thread-count="5" name="Test2">
    <parameter name="browser" value="chrome" />
    <parameter name="city" value="Pune" />
    <classes>
        <class name="test.PractoHomeTest" />
        <class name="test.FindAndBookTest" />
        <class name="test.ResultPageTest" />
        <class name="test.FormPageTest" />
    </classes>
</test> <!-- Test -->
</suite> <!-- Suite -->

```

Utility Package

ExtendReport.java

```

package Utility;

import java.awt.Desktop;

import java.io.File;
import java.io.IOException;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.List;

import org.testng.ITestContext;
import org.testng.ITestListener;
import org.testng.ITestResult;

import com.aventstack.extentreports.ExtentReports;
import com.aventstack.extentreports.ExtentTest;
import com.aventstack.extentreports.Status;
import com.aventstack.extentreports.reporter.ExtentSparkReporter;
import com.aventstack.extentreports.reporter.configuration.Theme;

public class ExtendReport implements ITestListener {
    public ExtentSparkReporter sparkReporter;
    public ExtentReports extent;
    public ExtentTest test;
    String repName;

```

```

    public void onStart(ITestContext testContext) {
        String timeStamp = new SimpleDateFormat("yyyy.MM.dd.HH.mm.ss").format(new Date()); //
time stamp
        repName = "Test-Report-" + timeStamp + ".html";
        sparkReporter = new ExtentSparkReporter(System.getProperty("user.dir") + "/reports/" +
repName); // specify location of the report
        sparkReporter.config().setDocumentTitle("BeCognizant Automation Report"); // Title of report
        sparkReporter.config().setReportName("BeCognizant Functional Testing"); // name of the report
        sparkReporter.config().setTheme(Theme.STANDARD);
        extent = new ExtentReports();
        extent.attachReporter(sparkReporter);
        extent.setSystemInfo("Application", "GSD");
        extent.setSystemInfo("Module", "Admin");
        extent.setSystemInfo("Sub Module", "Customers");
        extent.setSystemInfo("User Name", System.getProperty("user.name"));
        extent.setSystemInfo("Environment", "QA");
        String os = testContext.getCurrentXmlTest().getParameter("os");
        extent.setSystemInfo("Operating System", os);
        String browser = testContext.getCurrentXmlTest().getParameter("browser");
        extent.setSystemInfo("Browser", browser);
        List<String> includedGroups = testContext.getCurrentXmlTest().getIncludedGroups();
        if(!includedGroups.isEmpty()) {
            extent.setSystemInfo("Groups", includedGroups.toString());
        }
    }

    public void onTestSuccess(ITestResult result) {
        test = extent.createTest(result.getName());
        test.assignCategory(result.getMethod().getGroups()); // to display groups in report
        test.log(Status.PASS, result.getName() + " got successfully executed");
    }

    public void onTestFailure(ITestResult result) {
        test = extent.createTest(result.getName());
        test.assignCategory(result.getMethod().getGroups());
        test.log(Status.FAIL, result.getName() + " got failed");
        test.log(Status.INFO, result.getThrowable().getMessage());
    }

    public void onTestSkipped(ITestResult result) {
        test = extent.createTest(result.getName());
        test.assignCategory(result.getMethod().getGroups());
        test.log(Status.SKIP, result.getName() + " got skipped");
        test.log(Status.INFO, result.getThrowable().getMessage());
    }

    public void onFinish(ITestContext testContext) {
        extent.flush();
        String pathOfExtentReport = System.getProperty("user.dir") + "\\reports\\" + repName;
        File extentReport = new File(pathOfExtentReport);
        try {
            Desktop.getDesktop().browse(extentReport.toURI());
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}

```


Pom.xml

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>Group-5</groupId>
  <artifactId>Hackathon_Practo_Build1</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <dependencies>
    <!-- https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-java -->
    <dependency>
      <groupId>org.seleniumhq.selenium</groupId>
      <artifactId>selenium-java</artifactId>
      <version>4.19.1</version>
    </dependency>
    <!-- https://mvnrepository.com/artifact/org.apache.poi/poi -->
    <dependency>
      <groupId>org.apache.poi</groupId>
      <artifactId>poi</artifactId>
      <version>5.2.5</version>
    </dependency>
    <!-- https://mvnrepository.com/artifact/org.apache.poi/poi-ooxml -->
    <dependency>
      <groupId>org.apache.poi</groupId>
      <artifactId>poi-ooxml</artifactId>
      <version>5.2.5</version>
    </dependency>
    <dependency>
      <groupId>org.slf4j</groupId>
      <artifactId>slf4j-api</artifactId>
      <version>2.0.7</version>
    </dependency>
    <!-- https://mvnrepository.com/artifact/org.slf4j/slf4j-nop -->
    <dependency>
      <groupId>org.slf4j</groupId>
      <artifactId>slf4j-nop</artifactId>
      <version>2.0.13</version>
      <scope>test</scope>
    </dependency>
    <dependency>
      <groupId>com.aventstack</groupId>
      <artifactId>extentreports</artifactId>
      <version>5.1.1</version>
    </dependency>
    <!-- https://mvnrepository.com/artifact/org.testng/testng -->

  </dependencies>
</project>
```

Screenshot taken by the automation script:

Gurugram Search doctors, clinics, hospitals, etc.



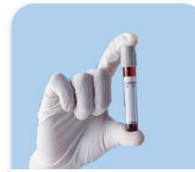
Instant Video Consultation
Connect within 60 secs



Find Doctors Near You
Confirmed appointments



Medicines
Essentials at your doorstep



Lab Tests
Sample pickup at your home



Surgeries
Safe and trusted surgery centers

Your home for health

Find and Book

Bangalore Search doctors, clinics, hospitals, etc.

Use my location

Search in entire Bangalore

Jp Nagar Bangalore

Hsr Layout Bangalore

Gynecologist/Obstetrician Others

practocare
Book appointment with an expert surgeon
Piles, Hernia & 50+ surgeries
KNOW MORE

Your home for health

Find and Book

Bangalore Search doctors, clinics, hospitals, etc.

Popular searches: Dermatologist Pediatrician

Popular Searches

Hysterectomy Normal Delivery

Common Specialities

Dentist

Gynecologist/obstetrician

practocare
Book appointment with an expert surgeon
Piles, Hernia & 50+ surgeries
KNOW MORE

practo

[Find Doctors](#) [Video Consult](#) [Medicines](#) [Lab Tests](#) [Surgeries](#) [NEW For Corporates](#) [For Providers](#) [Security & help](#) [Login / Signup](#)

Bangalore


ear-nose-throat (ent) specialist

Fed up of endless wait?
Look for clinic with Prime

Gender Patient Stories Experience All Filters Sort By Relevance

131 doctors available in Bangalore

Book appointments with minimum wait-time & verified doctor details



Dr. Ravi Sachidananda

Ear-Nose-Throat (ENT) Specialist

30 years experience overall

Yeshwanthpur,Bangalore

Practo Assured People Tree Hospitals

₹850 Consultation fee at clinic

Available Tomorrow

Provide current location to see ear-nose-throat (ent) specialist near you

You are seeing results from Bangalore. See results near you

Jp Nagar Hsr Layout

Bannerghatta Road Indiranagar

Rajajinagar

practo

[Find Doctors](#) [Video Consult](#) [Medicines](#) [Lab Tests](#) [Surgeries](#) [NEW For Corporates](#) [For Providers](#) [Security & help](#) [Login / Signup](#)

Bangalore


ear-nose-throat (ent) specialist

Fed up of endless wait?
Look for clinic with Prime

Male Doctor 40+ Patient Stories 10+ Years of experience All Filters Reset Filters Sort By Experience - High to Low

14 doctors available in Bangalore

Book appointments with minimum wait-time & verified doctor details



Dr. P.S Pradeep Kumar

Ear-Nose-Throat (ENT) Specialist

37 years experience overall

Banashankari 2nd Stage,Bangalore

Meenakshi ENT Speciality Hospital (MENTS) + 1 more

₹750 Consultation fee at clinic

Provide current location to see ear-nose-throat (ent) specialist near you

You are seeing results from Bangalore. See results near you

Jp Nagar Hsr Layout

Bannerghatta Road Indiranagar

Rajajinagar

practo

[Our Services](#) [Practo Ecosystem](#) [Product Capabilities](#) [Testimonials](#) [FAQs](#) [Group Insurance](#)

Your Workplace Health and Wellness Partner

Over 30cr users benefited by our holistic, customizable and accessible healthcare solutions

Schedule a Demo


Name

Organization Name

Contact Number

Official Email ID

Organization Size



Console Output

```
Console ×
<terminated> Hackathon_Practo_Build1_src.test.java.test.testng.xml [TestNG] C:\Users\2317399\p2\pool\plugins\org.eclipse.justj.openjdk.hotsp
|[RemoteTestNG] detected TestNG version 7.10.1
Jp Nagar
Hsr Layout
Whitefield
Indiranagar
Malleswaram
Rajajinagar
Bannerghatta Road
Vijayanagar
Sarjapur Road
Jayanagar
ChromeDriver: chrome on windows (7c6f012c339a2854a1d5209f6a53d97b)
131 doctors available in Bangalore
After Applying Filter total number of doctor
14 doctors available in Bangalore
Dr. P.S Pradeep Kumar
Dr. P Harihara Murthy
Dr. E.V. Raman
Dr. Satish Nair
Dr. Ravi Sachidananda
Dr. Sanjay Kumar
Dr. Gladson Guddappa Uchil
Dr. Vijay Rangachari
Dr. Sampath Chandra Prasad Rao
Dr. Srinivas Kunku
Pimpri-Chinchwad
Hadapsar
Kothrud
Wakad
Baner
Pimple Saudagar
Kharadi
Aundh
Viman Nagar
Shivajinagar
ChromeDriver: chrome on windows (c382b58a9b5fe678ea720a3fcfbf4f30)
27 doctors available in Pune
After Applying Filter total number of doctor
3 doctors available in Pune
Dr. Sumitpal Bhatti
Dr. Dushyant Khedikar
Dr. Manoj M.Bauskar

=====
Suite
Total tests run: 30, Passes: 30, Failures: 0, Skips: 0
=====
```

Test Report:

<div><div><div></div><div></div></div><div></div><div></div></div> <div>BeCognizant Functional Testing</div> <div>May 4, 2024 11:42:04 PM</div>		
<div><div></div><div></div><div></div></div>	Tests	<div>setUp</div> <div>05.04.2024 11:42:18 PM05.04.2024 11:42:18 PM00:00:00:011#test-id=1</div>
	setUp	11:42:18 PM / 00:00:00:011
	searchDoctor	11:42:21 PM / 00:00:00:000
	popularPlace	11:42:22 PM / 00:00:00:000
	speciality	11:42:25 PM / 00:00:00:000
	genderFilter	11:42:29 PM / 00:00:00:001
	patientStoriesFilter	11:42:32 PM / 00:00:00:000
	experienceFilter	11:42:35 PM / 00:00:00:000
	allFilters	11:42:38 PM / 00:00:00:000
<div>setUp</div> <div>STATUSTIMESTAMPDETAILS</div> <div>Pass11:42:18 PMsetUp got successfully executed</div>		