

## SOURCE CODE:

### BaseTest.java

```
package base;

import io.cucumber.java.AfterAll;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;

public class BaseTest {
    public static WebDriver driver = new ChromeDriver();

    @AfterAll
    public void Close_Browser() throws InterruptedException {
        System.out.println("Browser closed");
        driver.close();
    }
}
```

### StarHealth.feature

**Feature:** Test the starHealth page on Chrome Browser

**Scenario Outline:** Validate the Star Health Buy Now flow

- Given** User launches the Star Health application with "**<URL>**"
- Then** User waits for the Welcome to Star Health pop-up and closes it
- Then** User validates the Star Health home page title using a JUnit assertion
- When** User clicks on the Buy Now button
- Then** User types Name as "**<FullName>**"
- Then** User types Phone as "**<PhNo>**"
- Then** User types the PIN as "**<PIN>**"
- Then** User clicks on I need health insurance from the drop-down menu
- When** User selects the option "**<optionPlan>**"
- Then** User sees the Plan for My Family page
- When** User validates that the mobile number is the same as the previously entered number using a JUnit assertion
- Then** User clicks on the Star Health logo
- Then** The Application should redirect to the home page
- When** User closes the child tab
- Then** User navigates back to the parent tab

**Examples:**

	URL	FullName	PhNo
PIN	optionPlan		
	https://www.starhealth.in/	Userdemo	9999999999
530009	My Family		

## HomePage.java

```
package pages;

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

import base.BaseTest;

public class HomePage {
    @FindBy(xpath = "//span[text()='Buy Now']")
    public static WebElement buyNow;

    public HomePage(WebDriver driver) {
        PageFactory.initElements(driver, this);
        // driver can now work with your elements and
        methods of this class
    }

    public String get_homepage_title() throws
    InterruptedException {
        String title = BaseTest.driver.getTitle();
        return title;
    }

    public void click_On_BuyNow() throws InterruptedException
    {
        Thread.sleep(2000);
        buyNow.click();
    }
}
```

## BuyNowPage.java

```
package pages;

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

public class BuyNowPage {
    @FindBy(xpath = "//input[@id='name']")
    public static WebElement name;

    @FindBy(xpath = "//input[@id='contact_no']")
    public static WebElement contactNo;

    @FindBy(xpath = "//input[@id='pinCode']")
    public static WebElement pinCode;

    @FindBy(xpath = "//div[@id='grouped-select']")
    public static WebElement dropDown;

    @FindBy(xpath = "//input[@id='contact_no']")
    public static WebElement contactNumberVeriry;

    @FindBy(xpath = "//img[@class='jss341']")
    public static WebElement logo;

    public BuyNowPage(WebDriver driver) {

        PageFactory.initElements(driver, this);

    }

    public void give_name(String fullname) throws
    InterruptedException {
        Thread.sleep(2000);
        name.sendKeys(fullname);
    }

    public void give_phno(String phno) throws
    InterruptedException {
        Thread.sleep(2000);
        contactNo.sendKeys(phno);
    }

    public void give_pin(String pin) throws
    InterruptedException {
        Thread.sleep(2000);
        pinCode.sendKeys(pin);
    }
}
```

```

        public void click_dropdown() throws InterruptedException
        {
            Thread.sleep(2000);
            dropDown.click();
        }

        public String ContactNo_verify() throws
InterruptedException {
            Thread.sleep(2000);
            String C_no =
contactNumberVeriry.getAttribute("value");
            return C_no;
        }

        public void click_logo() throws InterruptedException {
            Thread.sleep(2000);
            logo.click();
        }
    }
}
=====

```

#### StarHealthTestRunner.java

```

package runner;

import org.junit.runner.RunWith;

import io.cucumber.junit.Cucumber;
import io.cucumber.junit.CucumberOptions;

@RunWith(Cucumber.class)
@CucumberOptions(features = "C:\\\\Users\\DELL\\eclipse-
workspace\\CucumberProject\\EndPhaseProject\\src\\test\\java\\
features\\StarHealth.feature", glue = {
    "steps" }, plugin = {
    "html:target/Cucumberreport.html", "pretty",

    "com.aventstack.extentreports.cucumber.adapter.ExtentCucu
mberAdapter:",
        "timeline:test-output-thread/" })

public class StarHealthTestRunner {

}
=====

```

## Step Defination.java

```
package steps;

import org.junit.Assert;
import org.openqa.selenium.By;

import base.BaseTest;
import io.cucumber.java.en.Given;
import io.cucumber.java.en.Then;
import io.cucumber.java.en.When;
import pages.BuyNowPage;
import pages.HomePage;

public class StepDefination {
    public static String ph_no;
    HomePage homePage = new HomePage(BaseTest.driver);
    BuyNowPage buyNowPage = new BuyNowPage(BaseTest.driver);

    @Given("User launches the Star Health application with {string}")
    public void user_launches_the_star_health_application_with(String URL) {

        BaseTest.driver.manage().window().maximize();
        BaseTest.driver.get(URL);

    }

    @Then("User waits for the Welcome to Star Health pop-up and closes it")
    public void user_waits_for_the_welcome_to_star_health_pop_up_and_closes_it() {

    }

    @Then("User validates the Star Health home page title using a JUnit assertion")
    public void user_validates_the_star_health_home_page_title_using_a_j_unit_assertion() throws InterruptedException {

        Thread.sleep(2000);
        String title = homePage.get_homepage_title();
        boolean isEqual = title.equals("Star Health Insurance: Medical, Accident and Travel insurance policies");
        Assert.assertTrue(isEqual);
    }
}
```

```

        // System.out.println("Star Health Home Page Title:
" + title);

    }

    @When("User clicks on the Buy Now button")
    public void user_clicks_on_the_buy_now_button() throws
InterruptedException {
        Thread.sleep(2000);
        homePage.click_On_BuyNow();
    }

    @Then("User types Name as {string}")
    public void user_types_name_as(String FullName) throws
InterruptedException {
        Thread.sleep(2000);
        buyNowPage.give_name(FullName);
    }

    @Then("User types Phone as {string}")
    public void user_types_phone_as(String PhNo) throws
InterruptedException {
        ph_no = PhNo;
        buyNowPage.give_phno(PhNo);
    }

    @Then("User types the PIN as {string}")
    public void user_types_the_pin_as(String PIN) throws
InterruptedException {
        buyNowPage.give_pin(PIN);
    }

    @Then("User clicks on I need health insurance from the
drop-down menu")
    public void
user_clicks_on_i_need_health_insurance_from_the_drop_down_menu
() throws InterruptedException {
        buyNowPage.click_dropdown();
        Thread.sleep(1000);
    }

    @When("User selects the option {string}")
    public void user_selects_the_option(String OptionPlan)
throws InterruptedException {
        BaseTest.driver.findElement(By.xpath("//li[text()="
+ "'" + OptionPlan + "'" + "]"")).click();
        Thread.sleep(3000);
    }

    @Then("User sees the Plan for My Family page")
    public void user_sees_the_plan_for_my_family_page() {

```

```

    }

    @When("User validates that the mobile number is the same
as the previously entered number using a JUnit assertion")
    public void
user_validates_that_the_mobile_number_is_the_same_as_the_previ
ously_entered_number_using_a_j_unit_assertion()
        throws InterruptedException {
        Thread.sleep(3000);
        String phoneNumber = buyNowPage.ContactNo_verify();
        Assert.assertEquals("Phone Number Matched ", ph_no,
phoneNumber);
    }

    @Then("User clicks on the Star Health logo")
    public void user_clicks_on_the_star_health_logo() throws
InterruptedException {
        buyNowPage.click_logo();
    }

    @Then("The Application should redirect to the home page")
    public void
the_application_should_redirect_to_the_home_page() {

    }

    @When("User closes the child tab")
    public void user_closes_the_child_tab() {

    }

    @Then("User navigates back to the parent tab")
    public void user_navigates_back_to_the_parent_tab() {

    }

}

```

=====

## Extent.properties

```
extent.reporter.spark.start=true
extent.reporter.spark.out=test-output/SparkReport/Spark.html
extent.reporter.spark.config=src/test/resources/extent-
config.xml

extent.reporter.spark.out=test-output/SparkReport/

screenshot.dir=test-output/
screenshot.rel.path=../
extent.reporter.pdf.start=true
extent.reporter.pdf.out=test output/PdfReport/ExtentPdf.pdf
#basefolder.name=reports
#basefolder.datetimepattern=d-MMM-YY HH-mm-ss
extent.reporter.spark.vieworder=dashboard, test, category, except
ion, author, device, log
systeminfo.os=Windows
systeminfo.user=Ashmitha
systeminfo.build=1.1
systeminfo.AppName=AutomationPractice
```

=====

## extent-config.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<extentreports>
  <configuration>
    <!-- report theme -->

    <!-- standard, dark -->

    <theme>dark</theme>
    <!-- document encoding -->

    <!-- defaults to UTF-8 -->

    <encoding>UTF-8</encoding>
    <!-- protocol for script and stylesheets -->

    <!-- defaults to https -->

    <protocol>http</protocol>
    <!-- title of the document -->
    <documentTitle>Extent</documentTitle>
    <!-- report name - displayed at top-nav -->
```



```
<reportName>Grasshopper Report</reportName>
<!-- location of charts in the test view -->

<!-- top, bottom -->

<testViewChartLocation>bottom</testViewChartLocation>
<!-- custom javascript -->

<scripts>

<![CDATA[
$(document).ready(function() {
});
]]>
    </scripts>
    <!-- custom styles -->
    <styles>
<![CDATA[
]]>
    </styles>
</configuration>
</extentreports>
```