

Calendars

This is the next tutorial in selenium-java series. Please go through the previous tutorials before you start this one. In the last tutorial, we learned about Java script executor. In this tutorial we will learn how to handle calendars using selenium webdriver!

What you will Learn:

1. Handle Calendars

Handle Calendars

Navigate to <https://www.goibibo.com/> > click 'Departure' field > a calendar pops up. Our task is to select any desired date from this calendar.

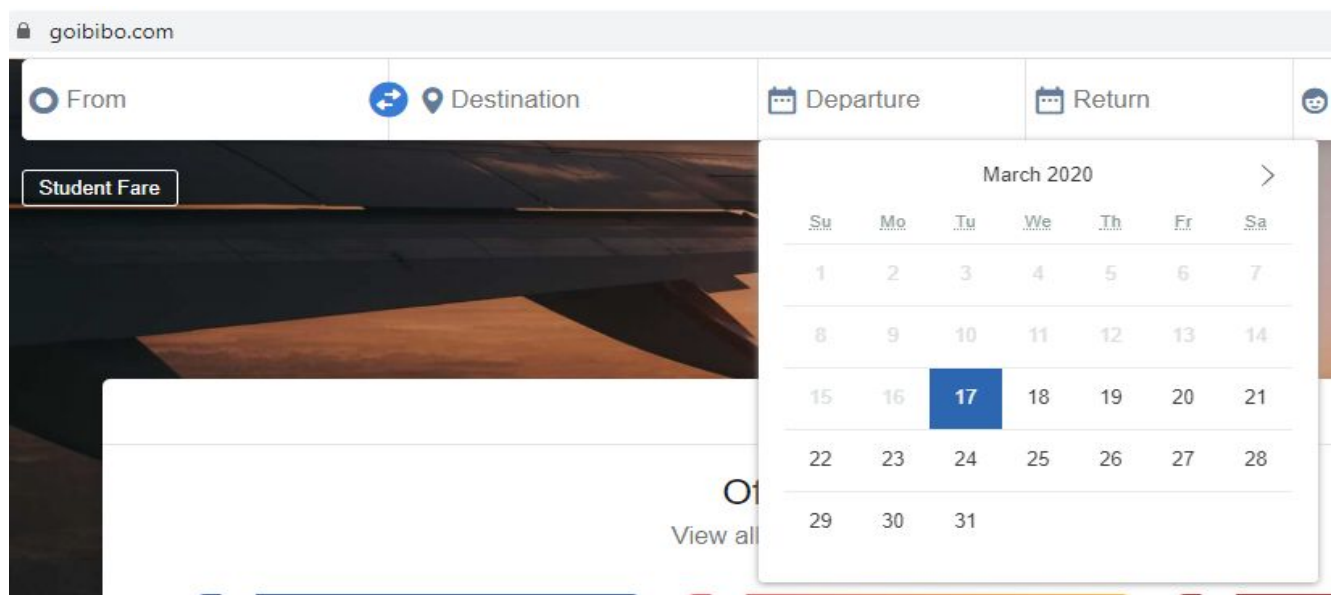


Figure 1

Inspect Departure field, we can use 'id' attribute to form an xpath

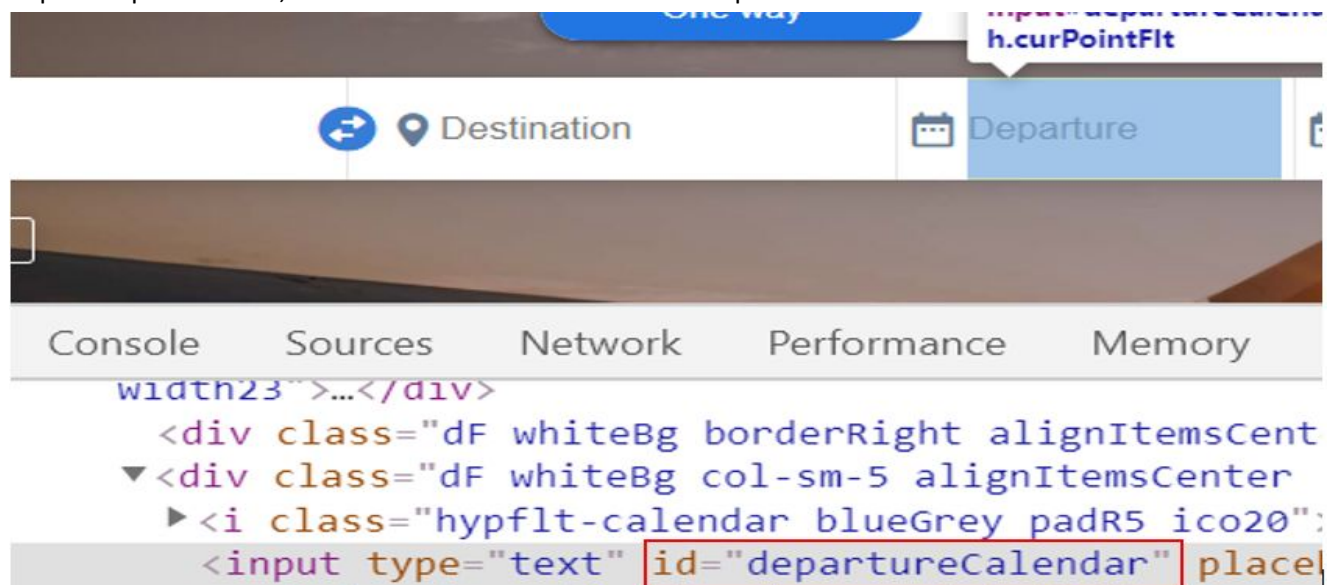


Figure 2

Create a new 'Calendar' class and add line#18, see below

```
16 driver.get("https://www.goibibo.com/");
17
18 driver.findElement(By.id("departureCalendar")).click();
```

Figure 3

Run the script, the calendar opens

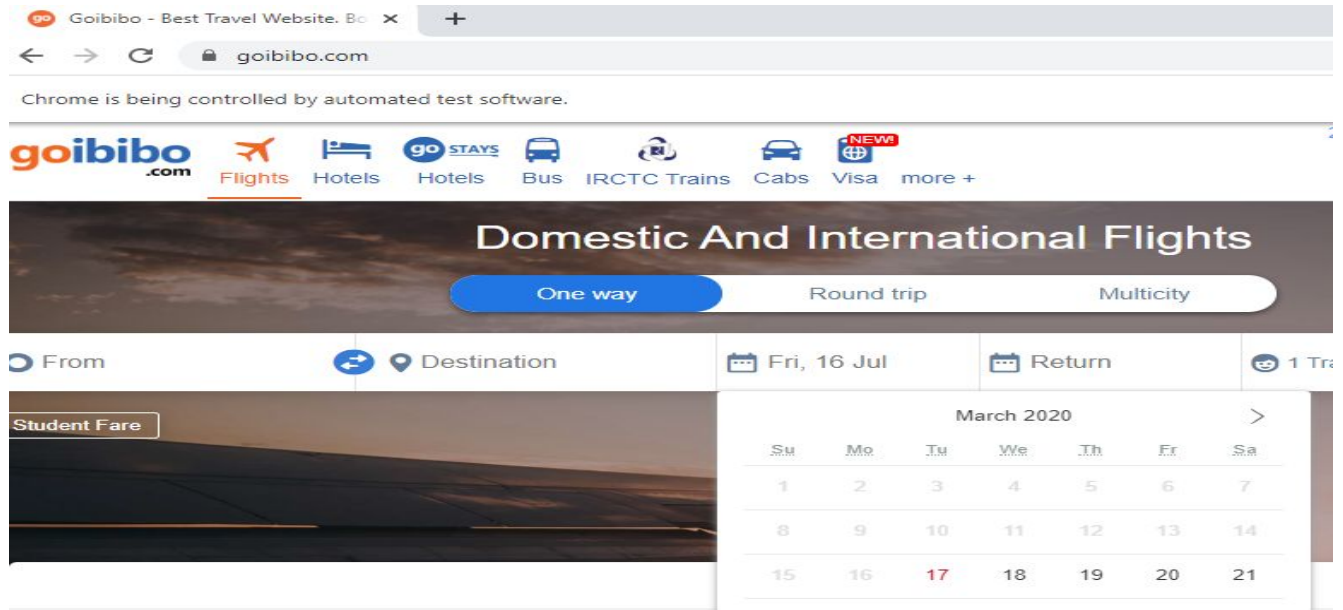


Figure 4

Let us add some wait time after the calendar opens

```
9 public class Calendar {
10
11     public static void main(String[] args) throws ParseException, InterruptedException {
12         System.setProperty("webdriver.chrome.driver", "C:\\Users\\DELL\\Desktop\\TRAINING
13         WebDriver driver = null;
14         driver = new ChromeDriver();
15
16         driver.get("https://www.goibibo.com/");
17
18         driver.findElement(By.id("departureCalendar")).click();
19
20         Thread.sleep(3000);
```

Figure 5

Comment below piece of code for time being

```

9 public class Calendar {
10
11     public static void main(String[] args) throws ParseException, InterruptedException
12         /*System.setProperty("webdriver.chrome.driver", "C:\\Users\\DELL\\Desktop\\TRA
13         WebDriver driver = null;
14         driver = new ChromeDriver();
15
16         driver.get("https://www.goibibo.com/");
17
18         driver.findElement(By.id("departureCalendar")).click();
19
20         Thread.sleep(3000);
21 */

```

Figure 6

Let us store a predefined date in variable 'd'. This date is in dd/mm/yyyy format

```

22 String d = "31/03/2021"; // dd/mm/yyyy format

```

Figure 7

We will now try to extract day, month and year from the date string. To do that, let us create an object of existing java class 'SimpleDateFormat'. We will pass the same dd/mm/yyyy format in the constructor that we have used in our predefined date above

```

//extract day, month, year
SimpleDateFormat df = new SimpleDateFormat("dd/MM/yyyy");

```

Figure 8

We then parse the predefined date 'd' that we had set in figure#7 with the help of parse method. This parse method returns an object of 'Date' class

```

//extract day, month, year
SimpleDateFormat df = new SimpleDateFormat("dd/MM/yyyy");
Date mydate = df.parse(d);

```

Figure 9

Add ParseException

```

9 public class Calendar {
10
11     public static void main(String[] args) throws ParseException, InterruptedException {

```

Figure 10

Next, we will create a calendar instance as mentioned below

```

//extract day, month, year
SimpleDateFormat df = new SimpleDateFormat("dd/MM/yyyy");
Date mydate = df.parse(d);
java.util.Calendar cal = java.util.Calendar.getInstance();

```

Figure 11

Next we will set date in calendar using setTime method

```
//extract day, month, year
SimpleDateFormat df = new SimpleDateFormat("dd/MM/yyyy");
Date mydate = df.parse(d);
java.util.Calendar cal = java.util.Calendar.getInstance();
cal.setTime(mydate);
```

Figure 12

So above, we have 'set' the date. We will now 'get' the day, month and year using 'get' method, see below

```
//extract day, month, year
SimpleDateFormat df = new SimpleDateFormat("dd/MM/yyyy");
Date mydate = df.parse(d);
java.util.Calendar cal = java.util.Calendar.getInstance();
cal.setTime(mydate);

int day = cal.get(java.util.Calendar.DAY_OF_MONTH);
int month = cal.get(java.util.Calendar.MONTH);
int year = cal.get(java.util.Calendar.YEAR);

System.out.println(day);
System.out.println(month);
System.out.println(year);
```

Figure 13

Before we run the script, recall that the day that we had set was 31/03/2021

```
--
22 String d = "31/03/2021"; // dd/mm/yyyy format
```

Figure 14

Run the script. The day is printed as 31, month is printed as numeric 2 (January is represented as 0, Feb as 1, Mar as 2 and so on), year gets printed as 2021

```
31
2
2021
```

Figure 15

Now in the website, the calendar shows the month as alphabetical viz 'March', April etc..

The month is getting printed as numeric (2) as of now. We don't want numeric month name, we want alphabetical, viz 'March' or April etc.

To handle this situation, we will create a month array & enter all the months inside it (line#23)


```

22 String d = "31/03/2021"; // dd/mm/yyyy format
23 String months[] = {"January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "De
24
25 //extract day, month, year
26 SimpleDateFormat df = new SimpleDateFormat("dd/MM/yyyy");
27 Date mydate = df.parse(d);
28 java.util.Calendar cal = java.util.Calendar.getInstance();
29 cal.setTime(mydate);
30
31 int day = cal.get(java.util.Calendar.DAY_OF_MONTH);
32 int month = cal.get(java.util.Calendar.MONTH);
33 int year = cal.get(java.util.Calendar.YEAR);
34
35 System.out.println(day);
36 //System.out.println(month);
37 System.out.println(months[month]);
38 System.out.println(year);

```

Figure 16

Comment line#36 and add line#37, as can be seen above.
Run the script, notice that month is now printed as 'March'

```

31
March
2021

```

Figure 17

When you click the calendar, by default, the current month would be shown. The date that we have set in our program is a future date viz 31st Mar 2021. Now we have to develop a logic that will keep clicking forward arrow > till we get March 2021 calendar

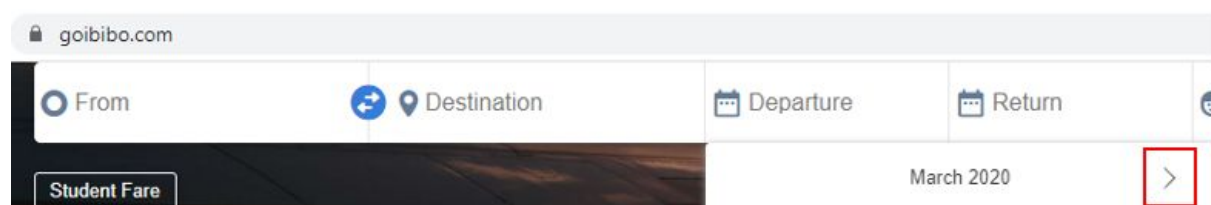


Figure 18

Inspect forward > button, it can be represented by
//span[@class='DayPicker-NavButton DayPicker-NavButton--next']

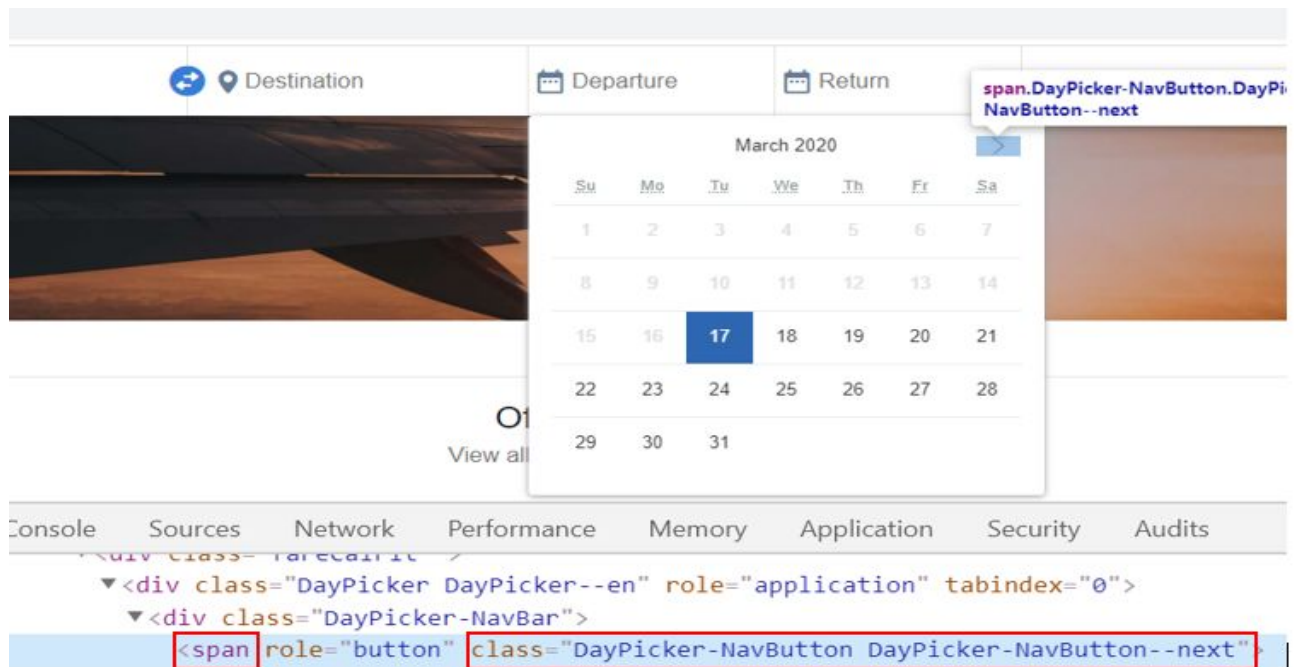


Figure 19

Let us put this xpath in a string, line#40, see below

```

23 String months[] = {"January", "February", "March", "April", "May", "June", "July", "August", "Se
24
25 //extract day, month, year
26 SimpleDateFormat df = new SimpleDateFormat("dd/MM/yyyy");
27 Date mydate = df.parse(d);
28 java.util.Calendar cal = java.util.Calendar.getInstance();
29 cal.setTime(mydate);
30
31 int day = cal.get(java.util.Calendar.DAY_OF_MONTH);
32 int month = cal.get(java.util.Calendar.MONTH);
33 int year = cal.get(java.util.Calendar.YEAR);
34
35 System.out.println(day);
36 //System.out.println(month);
37 System.out.println(months[month]);
38 System.out.println(year);
39
40 String forwardArrow = "//span[@class='DayPicker-NavButton DayPicker-NavButton--next']";

```

Figure 20

If you notice the website calendar, it shows month followed by space followed by year, example 'March 2020'

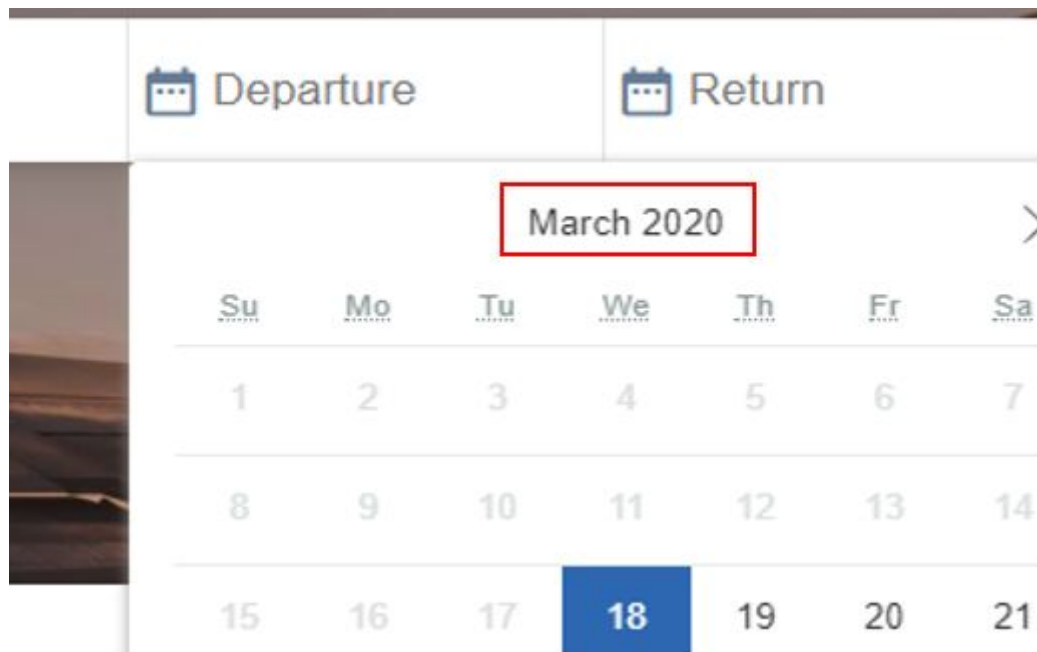


Figure 21

So we can create a string that is equal to month concatenated by white space concatenated by year, line#41

```
40 String forwardArrow = "//span[@class='DayPicker-NavButton DayPicker-NavButton--next']";
41 String travelMonth = months[month] + " "+year;
```

Figure 22

Let us try to print travelMonth to check if it prints desired format, line#42

```
40 String forwardArrow = "//span[@class='DayPicker-NavButton DayPicker-NavButton--next']";
41 String travelMonth = months[month] + " "+year;
42 System.out.println(travelMonth);
```

Figure 23

Run the script, it prints 'March 2021' (month space year), so correct format

March 2021

Figure 24

Now `//div[@class='fareCalFlt']/div/div[2]/div` is xpath of 'March 2020'. The same xpath will be valid for any month, example 'March 2021'

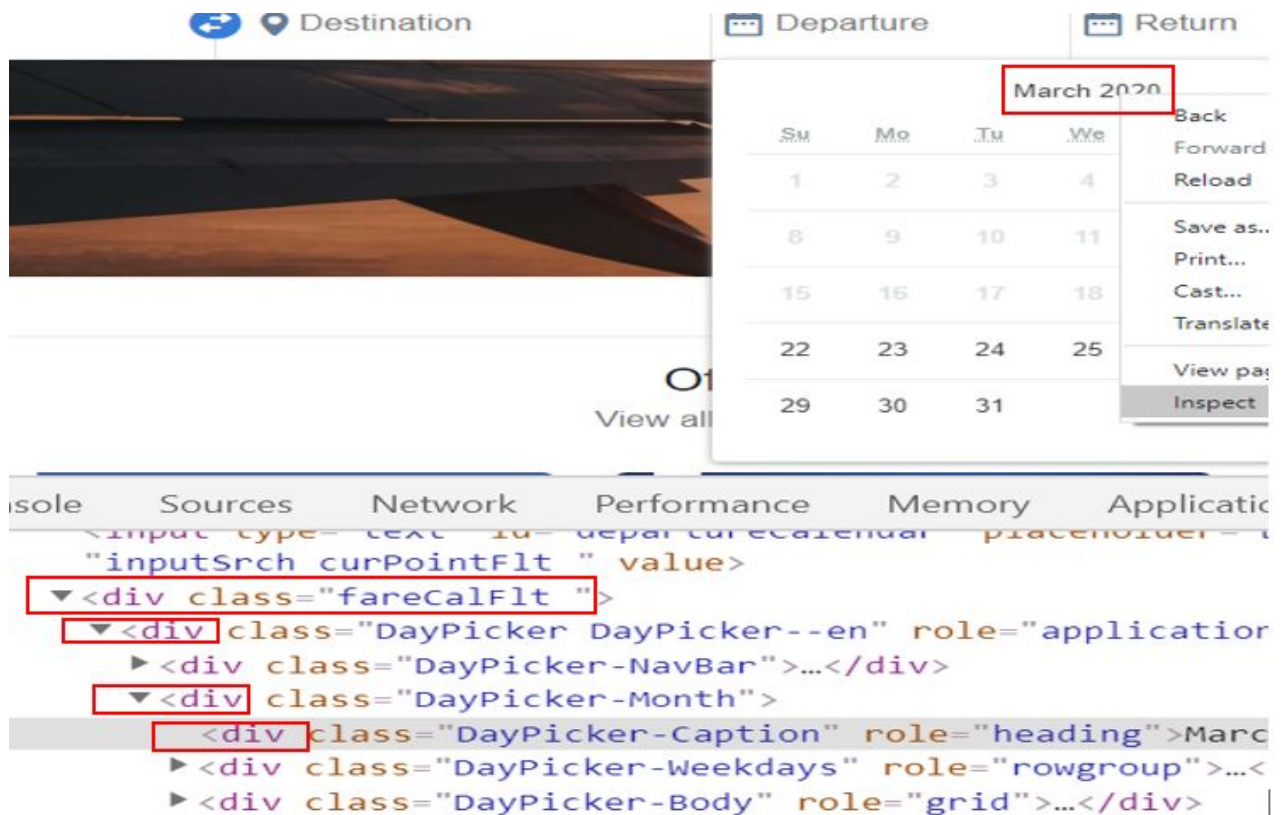


Figure 25

So we can put this xpath is a string, line#44

```

40 String forwardArrow = "//span[@class='DayPicker-NavButton DayPicker-NavButton--next']";
41 String travelMonth = months[month] + " " + year;
42 System.out.println(travelMonth);
43
44 String xpathMonthYearSection = "//div[@class='fareCalFlt ']/div/div[2]/div";

```

Figure 26

Now the logic we will be coding is: we will keep on clicking the forward arrow button till the time **xpathMonthYearSection** is NOT equal to **travelMonth**. A time will come when both of these would equal to 'March 2021'. At that point, we would stop clicking any further. This can be done using while loop. The exclamation mark ! in the 'while' loop is a negation viz till the time **xpathMonthYearSection** is NOT equal to **travelMonth**, the body of while loop keeps getting executed

```

40 String forwardArrow = "//span[@class='DayPicker-NavButton DayPicker-NavButton--next']";
41 String travelMonth = months[month] + " " + year;
42 System.out.println(travelMonth);
43
44 String xpathMonthYearSection = "//div[@class='fareCalFlt ']/div/div[2]/div";
45
46 while(!driver.findElement(By.xpath(xpathMonthYearSection)).getText().equals(travelMonth)) {
47     driver.findElement(By.xpath(forwardArrow)).click();
48 }

```

Figure 27

Run the script, notice that 'March 2021' calendar is shown & further click stops

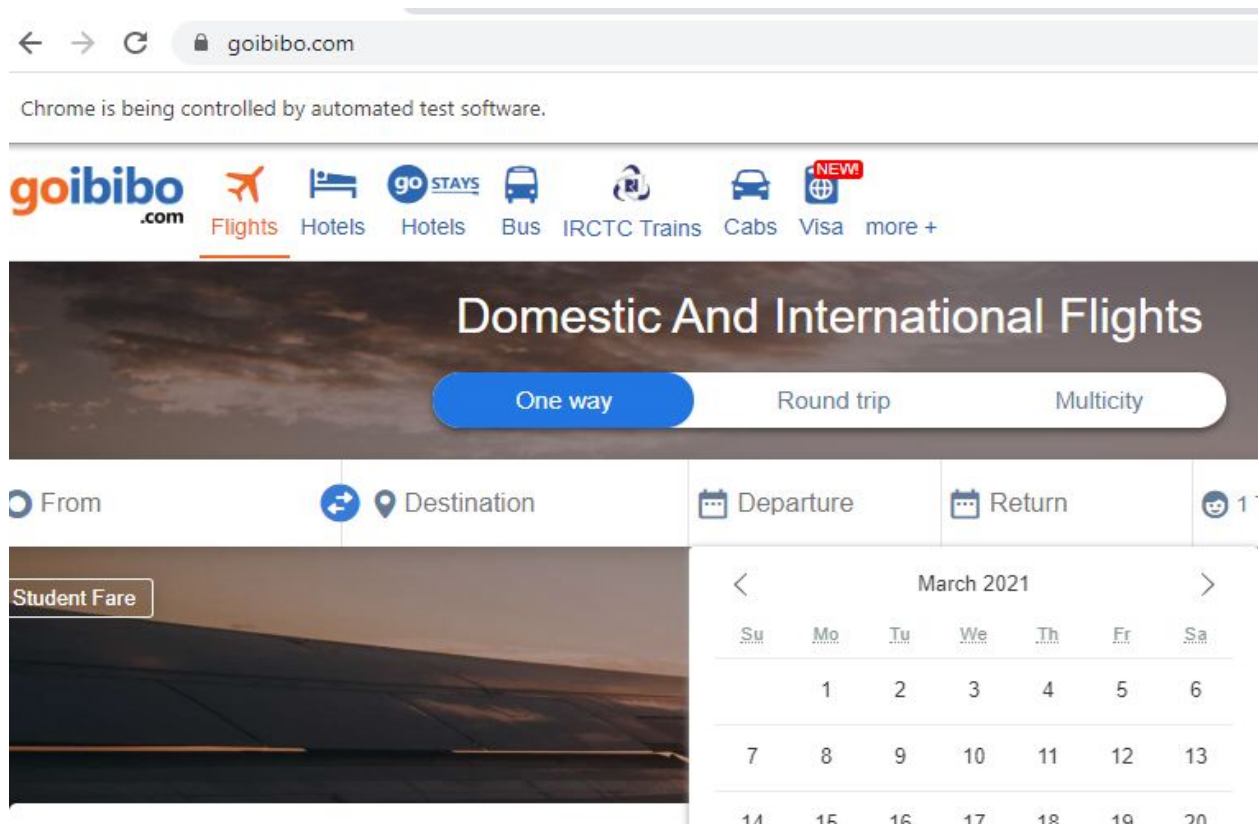


Figure 28

So far, we have been successful in selecting the desired month and year.

The next task is to select a desired date from the above calendar month.

To achieve this, we will create an xpath that would contain 2 strings:

The first string would be `"/div[text()='']"`

The second string is `""`

In between these 2 strings, we will concatenate 'day' using plus sign on either side, see line#50 below

```

40 String forwardArrow = "//span[@class='DayPicker-NavButton DayPicker-NavButton--next']";
41 String travelMonth = months[month] + " " + year;
42 System.out.println(travelMonth);
43
44 String xpathMonthYearSection = "//div[@class='fareCalFlt ']/div/div[2]/div";
45
46 while(!driver.findElement(By.xpath(xpathMonthYearSection)).getText().equals(travelMonth)) {
47     driver.findElement(By.xpath(forwardArrow)).click();
48 }
49
50 driver.findElement(By.xpath("/div[text()=''] + day + ''")).click(); //select desired date

```

Figure 29

Recall that the date we want to select is 31

```

22 String d = "31/03/2021"; // dd/mm/yyyy format

```

Figure 30

Run the script, notice that 31 Mar is selected

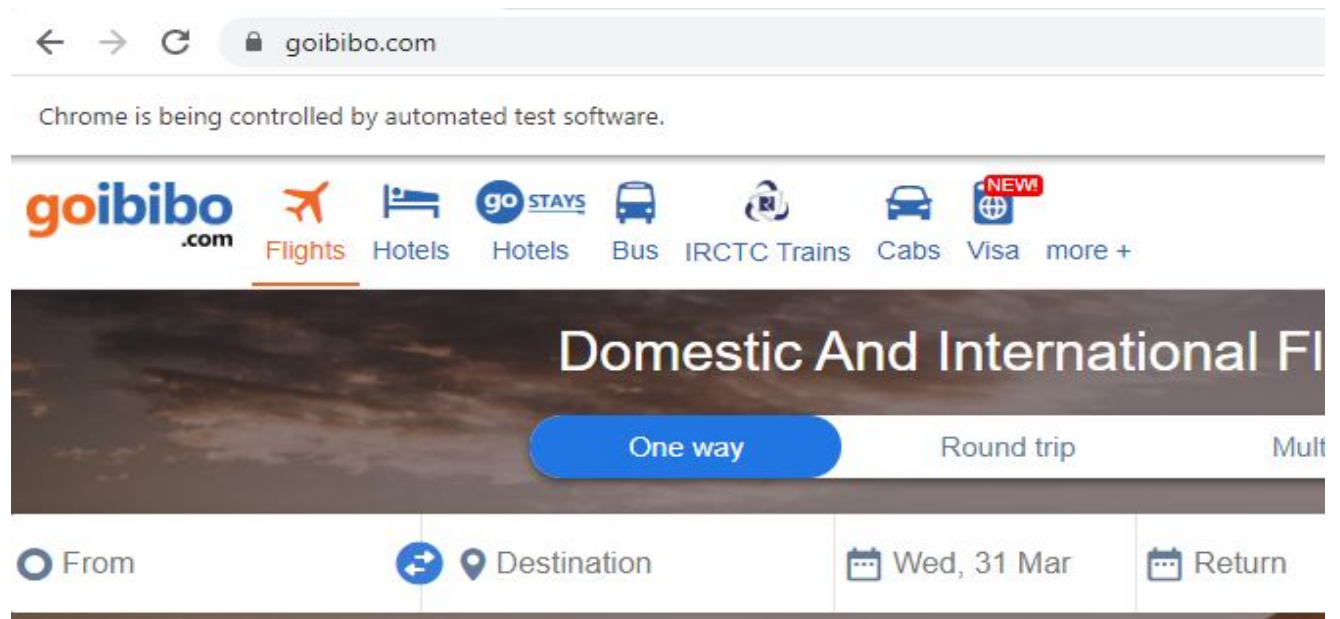


Figure 31

Let us change the date to 16th July 2021

```
22 String d = "16/07/2021";
```

Figure 32

Run the script, notice that 16 Jul is selected

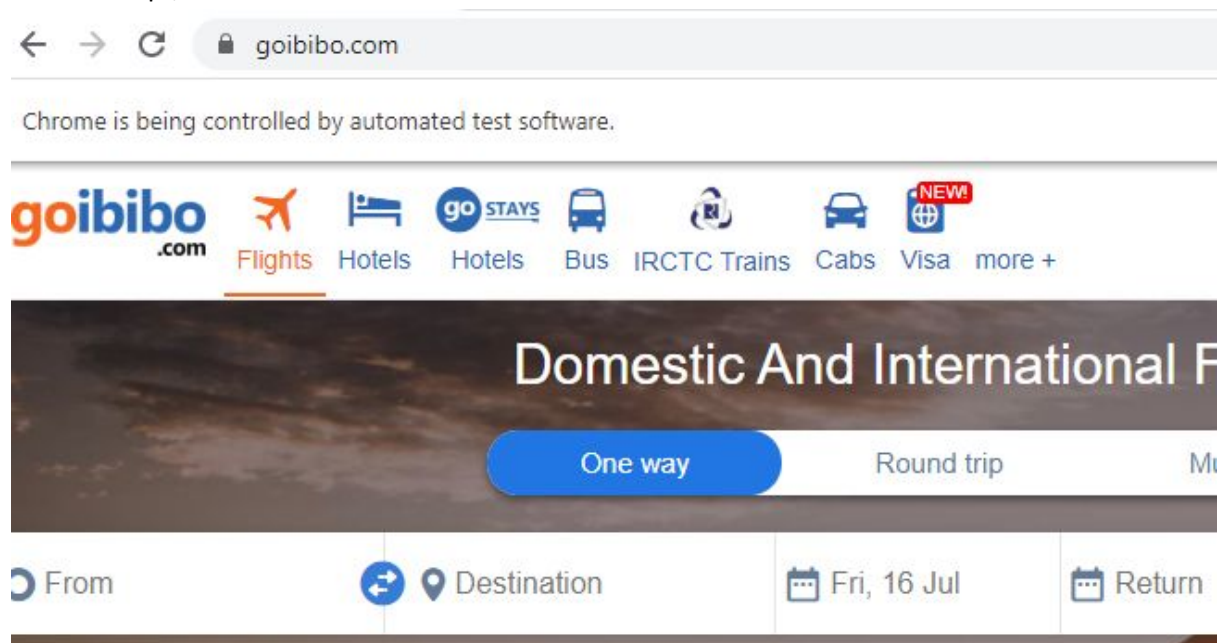


Figure 33

The complete code is as below:

```

Calendar.java
1 import java.text.ParseException;
2 import java.text.SimpleDateFormat;
3 import java.util.Date;
4 import org.openqa.selenium.By;
5 import org.openqa.selenium.WebDriver;
6 import org.openqa.selenium.chrome.ChromeDriver;
7
8 public class Calendar {
9
10     public static void main(String[] args) throws ParseException, InterruptedException {
11         System.setProperty("webdriver.chrome.driver", "C:\\Users\\DELL\\Desktop\\TRAINING\\Software\\chromedriver.exe");
12         WebDriver driver = null;
13         driver = new ChromeDriver();
14
15         driver.get("https://www.goibibo.com/");
16
17         driver.findElement(By.id("departureCalendar")).click();
18
19         Thread.sleep(3000);
20
21         String d = "16/07/2021"; // dd/mm/yyyy format
22         String months[] = {"January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December"};
23
24         //extract day, month, year
25         SimpleDateFormat df = new SimpleDateFormat("dd/MM/yyyy");
26         Date mydate = df.parse(d);
27
28         java.util.Calendar cal = java.util.Calendar.getInstance();
29         cal.setTime(mydate);
30
31         int day = cal.get(java.util.Calendar.DAY_OF_MONTH);
32         int month = cal.get(java.util.Calendar.MONTH);
33         int year = cal.get(java.util.Calendar.YEAR);
34
35         System.out.println(day);
36         //System.out.println(month);
37         System.out.println(months[month]);
38         System.out.println(year);
39
40         String forwardArrow = "//span[@class='DayPicker-NavButton DayPicker-NavButton--next']";
41         String travelMonth = months[month] + " " + year;
42         System.out.println(travelMonth);
43
44         String xpathMonthYearSection = "//div[@class='fareCalFlt ']/div/div[2]/div";
45
46         while(!driver.findElement(By.xpath(xpathMonthYearSection)).getText().equals(travelMonth)) {
47             driver.findElement(By.xpath(forwardArrow)).click();
48         }
49
50         driver.findElement(By.xpath("//div[text()=' " + day + "']")).click(); //select desired date
51     }

```

Figure 34

So this is how we can handle any calendar. Than you for reading!