34. Handling Date Picker or Calender Element

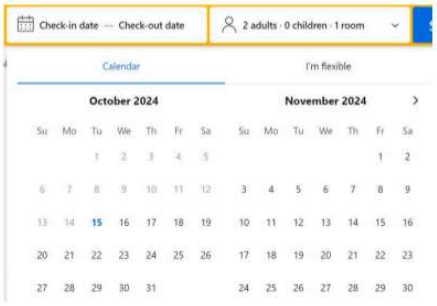
Date Pickers

1. Date pickers are specialized web elements that can be customized.
2. They come in various designs and structures like dropdowns,all dates will be arranged in a table inside these we can see some arrow marks buttons,etc
3. Typically, applications maintain a consistent date picker design across all pages, making it useful to create a user-defined method to handle them efficiently.
4. 90% applications will allow date through the send_keys method.
5. DatePicker will be different from one application to another application.we need to develop different logics accordingly.

Focus on 3 things

1. Year selection (It can be a label or drop down)
2. Month Selection (It can be label or drop down)
3. Date Selection (Most of the times It is a table)

Date Range Picker (Check-in & Check-out)



DatePickerDemo1.py

```
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys
def select_date(driver, target_year, target_month, target_date, is_future):
  while True:
    current_month = driver.find_element(By.XPATH, "//span[@class='ui-datepicker-month']").text
    current_year = driver.find_element(By.XPATH, "//span[@class='ui-datepicker-year']").text
    if target_month == current_month and target_year == current_year:
      break  # Desired month and year reached
```

Navigate to the next or previous month

```
    if is_future:
      driver.find_element(By.XPATH, "//span[@class='ui-icon ui-icon-circle-triangle-e']").click()
    else:
```

```
        driver.find_element(By.XPATH, "//span[@class='ui-icon ui-icon-circle-triangle-w']").click()
                                        Select date
    all_dates = driver.find_elements(By.XPATH, "//table[@class='ui-datepicker-calendar']//td/a")
    for current_date in all_dates:
        if current_date.text == target_date:
            current_date.click()
            break
def main():
    opt = webdriver.ChromeOptions()
    opt.add_experimental_option("detach", True)
    driver = webdriver.Chrome(options=opt)
    driver.get("https://jqueryui.com/datepicker/")
    driver.implicitly_wait(10)
    driver.maximize_window()
                                Switch to the frame containing the date picker
    driver.switch_to.frame(0)
                                Locate the date picker input field
    input_date_picker = driver.find_element(By.ID, "datepicker")
                                        using send_keys()
    input_date_picker.send_keys("12/14/2024"); #mm / dd / yyyy
    input_date_picker.send_keys(Keys.ENTER)
                                        DatePicker Method
                                        Open the date picker
    input_date_picker.click()
                                        Target date to be selected
    target_year = "2024"
    target_month = "April"
    target_date = "20"
                    Select the target date (set `True` for future dates, `False` for past dates)
    select_date(driver, target_year, target_month, target_date, False)
    driver.quit()
if __name__ == "__main__":
    main()
```

Note - In Travel applications we select only **current and future dates** not previous dates.In some applications if we want to enter date of birth DatePicker allows only **previous year** dates.so accordingly we have to write the logic.

DatePickerDemo2.py

```
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.support.select import Select
options = webdriver.ChromeOptions()
```

```
options.add_experimental_option("detach",True)
driver = webdriver.Chrome(options=options)
driver.get("https://www.dummyticket.com/dummy-ticket-for-visa-application/")
driver.maximize_window()
driver.implicitly_wait(10)
```

Departure Date

```
driver.find_element(By.XPATH,"//input[@id='departon']").click() # opens date picker
datepicker_month = Select(driver.find_element(By.XPATH,"//select[@data-handler='selectMonth']"))
datepicker_month.select_by_visible_text("Dec")  # month
datepicker_year = Select(driver.find_element(By.XPATH,"//select[@data-handler='selectYear']"))
datepicker_year.select_by_visible_text("2024")
alldates = driver.find_elements(By.XPATH,"//div[@id='ui-datepicker-div']//table/tbody/tr/td/a")
for date in alldates:
  if date.text=="25":
     date.click()
     break
```

DatePickerDemo3.py

```
import time
from selenium import webdriver
from selenium.webdriver.common.by import By
opt = webdriver.ChromeOptions()
opt.add_experimental_option("detach", True)
driver = webdriver.Chrome(options=opt)
driver.implicitly_wait(10)
driver.get("https://www.booking.com/")
driver.maximize_window()
```

To avoid signup window

```
time.sleep(3)
driver.refresh()
time.sleep(3)
```

Open the date picker

```
driver.find_element(By.XPATH, "//button[@data-testid='date-display-field-start']").click()
```

Check-in date selection

```
target_year = "2024"
target_month = "December"
target_date = "10"
while True:
  month_year =
driver.find_element(By.XPATH, "//*[@id='calendar-searchboxdatepicker']/div/div[1]/div/div[1]/h3").text
  current_month, current_year = month_year.split(" ")
  if target_year == current_year and target_month == current_month:
```

```
        break  # Desired month and year reached
    else:
        driver.find_element(By.XPATH, "//button[@aria-label='Next month']").click()
all_dates =
driver.find_elements(By.XPATH,"//*[@id='calendar-searchboxdatepicker']/div/div[1]/div/div[1]/table//td/
/span")
for current_date in all_dates:
    if current_date.text == target_date:
        current_date.click()
        break
```

Check-out date selection

```
target_year = "2025"
target_month = "March"
target_date = "15"
while True:
    month_year =
driver.find_element(By.XPATH, "//*[@id='calendar-searchboxdatepicker']/div/div[1]/div/div[2]/h3").text
    current_month, current_year = month_year.split(" ")
    if target_year == current_year and target_month == current_month:
        break  # Desired month and year reached
    else:
        driver.find_element(By.XPATH, "//button[@aria-label='Next month']").click()
all_dates =
driver.find_elements(By.XPATH,"//*[@id='calendar-searchboxdatepicker']/div/div[1]/div/div[2]/table//tb
ody//td/span")
for current_date in all_dates:
    if current_date.text == target_date:
        current_date.click()
        break
driver.quit()
```

comparedates.py

```
import time
d1='02/11/2021'   # DD/MM/YYYY
d2='05/12/2021'  #DD/MM/YYYY
dep_date = time.strptime(d1, "%d/%m/%Y")
return_date = time.strptime(d2, "%d/%m/%Y")
print(return_date>dep_date)  # returns true/false
```

Note

The **time.strptime function** in Python is used to parse a string representing a date and time and return a corresponding struct_time object. The function takes two arguments:**string, format**

- **time.strptime(string, format)**