1. Please write complete test cases for these Search hotel scenario:
   a. Check-in date should be not less than today and max 30 days from today

HotelMenu_ShouldBeAbleToSearch_WhenCheckInDateIsToday
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set checkIn Date is today at Calender Box* - Select Next Button
Assert Equal (Expected) “Search Hotel should be able to select the date”

HotelMenu_ShouldBeAbleToSearch_WhenCheckInDateIsThirthDaysFromToday
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set checkIn Date is 30 days from today at Calender Box* - Select Next Button
Assert Equal (Expected) “Search Hotel should be able to select the date”

HotelMenu_ShouldNotBeAbleToSearch _WhenCheckInDateIsLessThanToday
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set checkIn Date is a day before current date at Calender Box*
Assert Equal (Expected) “Should be return response : check In Date is invalid”

HotelMenu_ShouldNotBeAbleToSearch_WhenCheckInDateIsMoreThan30DaysFromToday
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set checkIn Date is 31 days from today at Calender Box*
Assert Equal (Expected) “Should be return response : check In Date is invalid, maximal is 30 days from today”

HotelMenu _ShouldNotBeAbleToSearch _WhenCheckInDateIsEmpty
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set checkIn Date is empty*
Assert Equal (Expected) “Should be return response to fill the checkIn Date first”

   b. Duration of stay, max 5 nights

HotelMenu_ShouldBeAbleToSearch_WhenCheckInDateIsTodayAndStayDurationIsMaxDuration
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set checkIn Date is today at Calender Box*

- *Set checkout Date is 5 nights from today*
Assert Equal (Expected) “Should be able to apply the stay duration”

HotelMenu_ ShouldBeAbleToSearch _WhenCheckInDateIsTodayAndStayDurationIsLessThanMaximumDuration
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set checkIn Date is today at Calender Box* - *Set checkout Date is 4 nights from checkIn Date*
Assert Equal (Expected) “Should not be able to apply the stay duration”

HotelMenu_ShouldBeAbleToSearch_WhenCheckInDateIsThirtyDaysFromTodayAndStayDurationIsEqualToMaximumDuration
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set checkIn Date is 30 days from today at Calender Box* - *Set checkout Date is 5 nights from checkIn Date*
Assert Equal (Expected) “Should be able to apply the stay duration”

HotelMenu_ShouldNotBeAbleToSearch_WhenCheckInDateIsTodayAndStayDurationIsMoreThanMaximumDuration
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set checkIn Date is today at Calender Box* - *Set checkout Date is 6 nights from checkIn Date*
Assert Equal (Expected) “Should not be able to apply the stay duration and show response to set the valid check out Date which is max 5 days”

HotelMenu_ShouldNotBeAbleToSearch_WhenCheckInDateIsThirthDaysFromTodayAndStayDurationIsMoreThanMaximumDuration
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set checkIn Date is 30 days from today at Calender Box* - *Set checkout Date is 6 nights from checkIn Date*
Assert Equal (Expected) “Should not be able to apply the stay duration”

HotelMenu_ShouldNotBeAbleToSearch_WhenCheckInDateIsTodayAndStayCheckOutDateIsEmpty
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set checkIn Date is today Calender Box* - *Set checkout Date is empty*

Assert Equal (Expected) “Should not be able to apply the stay duration and show response to fill the check out date first”

HotelMenu_ShouldNotBeAbleToSearch_WhenCheckInDateIsThirthDaysFromTodayAndCheckOutDateIsEmpty
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set checkIn Date is 30 days from today Calender Box* - *Set checkout Date is empty*
Assert Equal (Expected) “Should not be able to apply the stay duration and show response to fill the check out date first”

c. 1 Room must be at least 1 adult (max 5 adults in 1 room)

HotelMenu_ShouldBeAbleToSearch_WhenOneRoomForMinimumAdultCapacityGuess
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set valid stay duration* - *Set number of room is 1* - *Set number of adult guess is 1* - *Set number of child guess is 0* - *Set Apply Button*
Assert Equal (Expected) “Should be able to search the hotel”

HotelMenu_ShouldBeAbleToSearch_WhenOneRoomForLessThanMaximumAdultCapacityGuess
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set valid stay duration* - *Set number of room is 1* - *Set number of adult guess is 4* - *Set number of child guess is 0* - *Set Apply Button*
Assert Equal (Expected) “Should be able to search the hotel”

HotelMenu_ShouldBeAbleToSearch_WhenOneRoomForMaximumAdultCapacityGuess
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set valid stay duration* - *Set number of room is 1* - *Set number of adult guess is 5* - *Set number of child guess is 0* - *Set Apply Button*
Assert Equal (Expected) “Should be able to search the hotel”

HotelMenu_ShouldBeAbleToSearch_WhenNumberOfGuestIsEquealToMultipleAdultCapacityGuessPerRoom
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set valid stay duration* - *Set number of room is 2* - *Set number of adult guess is 10* - *Set number of child guess is 0* - *Set Apply Button*
Assert Equal (Expected) "Should not be able to search the hotel "

HotelMenu_ShouldNotBeAbleToSearch_WhenOneRoomForMoreThanAdultCapacityGuess
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set valid stay duration* - *Set number of room is 1* - *Set number of adult guess is 6* - *Set number of child guess is 0* - *Set Apply Button*
Assert Equal (Expected) "Should not be able to search the hotel and show response the rule which is 1 Room must be at least 1 adult (max 5 adults in 1 room)"

HotelMenu_ShouldNotBeAbleToSearch_WhenNumberOfGuestsIsNotEquealToAdultMultipleCapacityGuessPerRoom
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set valid stay duration* - *Set number of room is 2* - *Set number of adult guess is 11* - *Set number of child guess is 0* - *Set Apply Button*
Assert Equal (Expected) "Should not be able to search the hotel and show response the rule which is 1 Room must be at least 1 adult (max 5 adults in 1 room)"

HotelMenu_ShouldNotBeAbleToSearch_WhenNumberOfRoomIsMoreThanNumberOfGuess
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set valid stay duration* - *Set number of room is 2* - *Set number of adult guess is 1* - *Set number of child guess is 0* - *Set Apply Button*

Assert Equal (Expected) “Should not be able to search the hotel and show the response which is number of room is not able more than number of guess“

d. and children (max 5 years old and max 2 children in 1 room) must be accompanied by min 1 adult

HotelMenu_ ShouldBeAbleToSearch _WhenOneRoomForMinimumAdultGuessAndLessThanMaximumChildrenGuessCapacity
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set valid stay duration* - *Set number of room is 1* - *Set number of adult guess is 1* - *Set number of child guess is 1* - *Set Apply Button*
Assert Equal (Expected) “Should not be able to search the hotel “

HotelMenu_ ShouldBeAbleToSearch _WhenOneRoomForMinimumAdultGuessAnd MaximumChildrenGuessCapacity
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set valid stay duration* - *Set number of room is 1* - *Set number of adult guess is 1* - *Set number of child guess is 2* - *Set Apply Button*
Assert Equal (Expected) “Should not be able to search the hotel “

HotelMenu_ ShouldBeAbleToSearch _WhenOneRoomForMaximumAdultGuessAnd MaximumChildrenGuessCapacity
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set valid stay duration* - *Set number of room is 1* - *Set number of adult guess is 5* - *Set number of child guess is 2* - *Set Apply Button*
Assert Equal (Expected) “Should not be able to search the hotel “

HotelMenu_ ShouldNotBeAbleToSearch _WhenOneRoomForMinimumAdultGuessAnd MoreThanMaximumChildrenGuessCapacity
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set valid stay duration* - *Set number of room is 1*

- *Set number of adult guess is 1* - *Set number of child guess is 3* - *Set Apply Button*
Assert Equal (Expected) "Should not be able to search the hotel "

HotelMenu_ ShouldNotBeAbleToSearch _WhenOneRoomForMaximumAdultGuessAndMoreThanMaximumChildrenGuessCapacity
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set valid stay duration* - *Set number of room is 1* - *Set number of adult guess is 5* - *Set number of child guess is 3* - *Set Apply Button*
Assert Equal (Expected) "Should not be able to search the hotel "

HotelMenu_ShouldNotBeAbleToSearch_WhenRoomOnlyForChildren
- Navigate to Search Hotel Form Page - Set Hotel Location - *Set valid stay duration* - *Set number of room is 1* - *Set number of adult guess is 0* - *Set number of child guess is 1* - *Set Apply Button*
Assert Equal (Expected) "Should not be able to search the hotel and show response the direction which is 1 Room must be at least 1 adult (max 5 adults in 1 room) "

2. Analyze anomaly on screenshot below and elaborate whether it's a FE or BE bug.
   a. Elaborate your analysis
      - I found that hotel search feature is not give the expected result for what user looking for by the filter inputted in form.
        - User looking for The Hotel in ***Jakarta*** at Saturday, 30 Apr 2022 for 1 night stay duration, 1 room, 1 adult and no children guess.
        - But the result is, Traveloka response by the same filter of date, number of room, number of guess except the Location filter (***Bali***).

        But if we looking carefully , the user have some history of hotel searching before, which the last history is looking for the hotels in Bali.
      - In my opinion it's a **FE Bug.**

        It seems like it related to cache handling (FE Scope) : about history of hotel location searching. The location in cache still exist and used as a location parameter in new request to server.

b. Create the bug report.

Category	Label	Value
Bug Id	ID Number	#1
	Name	Accomodation – Mismatched Search Results for Hotel Location
	Reporter	Lily Tarigan
	Submit Date	February 12, 2024
Bug Overview	Summary	When I filter Hotel Location in Jakarta, the results give me Hotel in Bali but all the other filter is already appropriate.
	URL	Dashboard > Hotels > Search Form > Location > Search
	Screenshot	
Env	Platform	iOS
	OS	-
	App	Traveloka version __
Bug Details	Steps to Reproduce	First Step - Search hotels in Bali at Today - Set for 1 night - Set for 1 room - Set for 1 adults guess only (no children) - Search Second Step - Seach hotel again in Jakarta for tomorrow - Set for 1 night - Set for 1 room - Set for 1 adults guess on;y (no children) - Search
	Expected	The hotel should be filter by Jakarta Location
	Actual	The hotel filtered by Bali Location
	Desc	-
Bug Tracking	Severity	Major
	Assignted to	
	Priority	High
Notes	Notes	-

3. Write the automation script for this test case

Language: Any

○ Select Cars Product

○ Select tab Without Driver

○ Select Pick-up Location (e.g.: Jakarta)

○ Select Pick-up Date & Time (e.g.: today+2d 09:00)

○ Select Drop-off Date & Time (e.g.: today+5d 11:00)

○ Click button Search Car

○ Select Car

- Select Car Provider
- Click button Continue in Product Detail
- Select Pick-up Location in “Rental Office”
- Select Drop-off Location in “Other Location”
- Input Pick-up/Drop-off notes is optional
- Click button Book Now
- Fill Contact Details
- Fill Driver Details
- Click Continue
- Add a special request is optional
- Check all rental requirements
- Click Continue
- Select payment method and proceed payment

Analyze : this test case is to running automatically process of car rental reservation without driver from select the car untill to the payment proceed.

I choose to use C# Language

```
Using openQA.Selenium.Chrome;

Class Program
{
  Static void Main()
  {
    IWebDriver driver = new ChromeDriver();
    driver.Navigate().goToUrl(“url_traveloka_rental_mobil”);

    selectCatProduct(driver);
    selectWithoutDriver(driver);
    selectPickupLocation(driver, “Jakarta”);
    selectPickupDateTime(driver, dateTime.Today.AddDays(0), “09:00”);
    selectDropOffDateTime(driver, dateTime.Today.AddDays(5), “11:00”);
    clickStartSearchCarButton(driver);
    selectCar(driver);
    selectCarProvider(driver);
    clickProductDetailButton(driver);
    selectPickupLocationInRentalOffice(driver);
    selectDropOffLocationInOtheLocation(driver);
    inputPickUpDropOfNotes(driver, “note”);
    clickbuttonBook(driver);
    fillContactDetails(driver);
    fillDriverDetails(driver);
    clickContinueButton(driver);
    addSpecialRequestOptional(driver, “note”);
    checkAllRentalRequirements(driver);
```

```
clickContinue(driver);
selectPaymentMethod(driver);
processPayment(driver);

 driver.Quit();

}


// Method

selectCatProduct(IWebDriver driver);
{
  // click cars product
  Driver.FindElement(By.Id(“cars_product”)).Click();
}

selectWithoutDriver(IWebDriver  driver);
{
  // click without driver tab
  Driver.FindElement(By.Id(“without_driver”)).Click();
}

selectPickupLocation(IWebDriver driver,  string location)
{
  // select pickup Location
  IWebElement pickupLocationInput = driver.FindElemet(By.Id(“pickup_location”));
  pickupLocationInput.Clear();
  pickupLocationInput.SendKeys(location);
}

selectPickupDateTime(IWebDriver driver,  string time, DateTime date)
{
  // select pickup date
  IWebElement pickupDateInput = driver.FindElemet(By.Id(“pickup_date”));
  pickupDateInput.Clear();
  pickupDateInput.SendKeys(date.ToString(“YYYY-MM-DD”);

  // select pickup time
  IWebElement pickupTimeInput = driver.FindElemet(By.Id(“pickup_time”));
  pickupTimeInput.Clear();
  pickupTimeInput.SendKeys(time);
}

selectDropOffDateTime(IWebDriver driver,  string time, DateTime date)
{
  // select dropoff date
  IWebElement dropoffDateInput = driver.FindElemet(By.Id(“dropoff_date”));
  dropoffDateInput.Clear();
  dropoffDateInput.SendKeys(date.ToString(“YYYY-MM-DD”);

  // select dropoff time
  IWebElement dropoffTimeInput = driver.FindElemet(By.Id(“dropoff_time”));
  dropoffTimeInput.Clear();
  dropoffTimeInput.SendKeys(time);
}

clickStartSearchCarButton(IWebDriver  driver);
{
  // click cars product
  driver.FindElement(By.Id(“search_car_button”)).Click();
}

selectCar(IWebDriver driver);
{
  // click cars product
  IWebElmenet carListItem = driver.FindElement(“.car_list_item”);
  carListItem.Click();
}

selectCarProvider(IWebDriver driver);
{
  // click cars product
  IWebElmenet carListItem = driver.FindElement(“.car_provider_item”);
  carListItem.Click();
}

clickContinueButtonProductDetailButton(IWebDriver driver);
{
  // click cars product
  driver.FindElement(By.Id(“continue_button”)).Click();
}

selectPickupLocationInRentalOffice(IWebDriver driver);
{
  // click cars product
  driver.FindElement(By.Id(“pick_location_at_rental_office”)).Click();
}

selectDropOffLocationInOtherLocation(IWebDriver driver);
{
  // click cars product
  driver.FindElement(By.Id(“dropoff_location_at_other_location”)).Click();
}

inputPickUpDropOffNotes(IWebDriver driver, string notes)
```

```
{
  // click cars product
  IWebElmenet notesInput = driver.FindElementBy.Id(“pickup_dropoff_notes”));
  notesInput.Clear();
  notesInput.SendKeys(notes);
}

clickbuttonBook(IWebDriver driver);
{
  // click cars product
  driver.FindElement(By.Id(“book_button”)).Click();
}

fillContactDetails(IWebDriver driver);
{
  // click cars product
  IWebElmenet nameInput = driver.FindElementBy.Id(“name_input”));
  notesInput.SendKeys(“Lily”);

  IWebElmenet emailInput = driver.FindElementBy.Id(“email_input”));
  emailInput.SendKeys(“lilyursula6@gmail.com”);

  IWebElmenet phoneInput = driver.FindElementBy.Id(“phone_input”));
  notesInput.SendKeys(“+6282282775880”);

}

fillDriverDetails(IWebDriver driver);
{
  // click cars product
  IWebElmenet driverNameInput = driver.FindElementBy.Id(“driver_name_input”));
  driverNameInput.SendKeys(“Lily”);

  IWebElmenet emailDriverInput = driver.FindElementBy.Id(“driver_email_input”));
  emailInput.SendKeys(“lilyursula6@gmail.com”);

  IWebElmenet driverLicense = driver.FindElementBy.Id(“driver_license_input”));
  driverLicense.SendKeys(“+6282282775880”);

}

clickContinueButton(IWebDriver driver);
{
  // click cars product
  driver.FindElement(By.Id(“continue_button”)).Click();
}

addSpecialRequestOptional(IWebDriver driver, string notes)
```

```
    {
      // click cars product
      IWebElmenet specialRequest = driver.FindElementBy.Id("special_request"));
      specialRequest.SendKeys(notes);
    }

    checkAllRentalRequirements(IWebDriver driver);
    {
      // click cars product
      driver.FindElement(By.Id("rental_requirements_ckeckbox")).Click();
    }

    clickContinue(IWebDriver driver);
    {
      // click cars product
      driver.FindElement(By.Id("continue_button")).Click();
    }

    selectPaymentMethod(IWebDriver driver);
    {
      // click cars product
      IWebElmenet paymentMethod = driver.FindElement(".list_payment_method");
      paymentMethod.Click();
    }

    processPayment(IWebDriver driver);
    {
      // click cars product
      driver.FindElement(By.Id("process_payment_button")).Click();
    }
}
```

4. Open this link 1to50 :
    a. How to find elements / selectors from the web user interface ?
    In my opinion, to find elements / selectors of this game is by initialize 50 id for each boxes.

    b. Please create a pseudocode or logic step to solve the following game using an automated test.

    Scenario :
    There are 5x5 boxes containing random values from 1-25. We will select the number sequensce from the boxes start from 1 to 25. After one number removed, number 26 until 50 will appear one by one to fill the position was just selected (but their apprarance order is randomly). Its possible that number 31 appers first, followed by 26 and so on. If the selected number is out of order, then the box cant be clicked.

After the sequence of 1-25 is completed, all numbers from 26-50 will remain in the boxes. Starting from the sequence that beginning with 26, whet it selected.. the position will not be filled from sequence 51 and so on anymore. It will stop in this session and the box will disappear one by one untul 50 is successfully completed we selected. Eventually, all boxes disappear and obtain score will remain.

```
BEGIN

Function GridInitialize();
  Create a 5x5 grid
  Fill the grid with random numbers from 1 to 25

Function playGame()
  Initialize grid
  Initialize currentNumber to 1
  Inisialize score to 0

  Loop until all numbers from 1 to 50 are selected;
     if currentNumber <=25:
        Display grid
        Allow player to click on a box
          if clickedBox value == curentNumber:
            Set clickedBox value to 0
            Increment score
            Increment currentNumber by1
          else:
            Box cant be click
     else:
        Display grid
        Allow player to click on a box
          if clickedBox == currentNumber:
            Set clickedBox to 0
            clickedBox hide
           else:
            Box cant be click

Display final score

END
```