



GUJARAT TECHNOLOGICAL UNIVERSITY

PYTHON & AUTOMATION

A PROJECT REPORT

Submitted by

Dhruvi Vimalbhai Virani

190320111070

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

ELECTRONICS & COMMUNICATION ENGINEERING

L. J. INSTITUTE OF ENGINEERING AND TECHNOLOGY, AHMEDABAD

Gujarat Technological University, Ahmedabad

[MAY , 2023]



L. J.INSTITUTE OF ENGINEERING AND TECHNOLOGY, AHMEDABAD

LJ Campus, LJ College Rd, near Sanand - Sarkhej Road, Makarba, Sarkhej-Okaf, Gujarat 382210

CERTIFICATE

This is to certify that the project report submitted along with the project entitled **PYTHON & AUTOMATION** has been carried out by **Dhruvi Vimalbhai Virani** under my guidance in partial fulfillment for the degree of Bachelor of Engineering in Electronics and Communication , 8th Semester of Gujarat Technological University, Ahmadabad during the academic year 2021-22.

Mosam Pandya

Internal Guide

Mosam Pandya

Head of the Department

EINFOCHIPS OFFER LETTER

e-Infochips Private Limited

Ground Floor to Fourth Floor, Block-E, Ratna,
3rd Eye Voice, IT/ITES SEZ, Calica Construction and Impex Pvt. Ltd.
Village: Ognaj, Taluka; Daskroi, Ahmedabad - 380060, Gujarat
(Developed by Calica Construction and Impex Private Limited)
Phone Number: 02718681000



Dated: 4th May 2023

To,

L J Institute of Engineering and Technology

To Whom so ever it may Concern

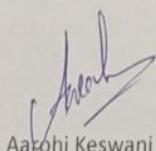
This is to certify that Dhruvi Virani has successfully completed her project on "Python & Automation" using Python under the guidance of Jimmy Patel during her ongoing internship.

Her internship project activities include Python, Robot Framework, and Selenium technologies, and learned different software like Pytest, and Appium.

This certificate is intended for project completion only.

From,

eInfochips Private Limited



Aarohi Keswani

(Authorized Signatory)

e-Infochips Private Limited

Registered Office: Building No. 2, Aaryan Corporate Park,
Near Shilaj Railway Crossing, Thaltej-Shilaj Road,
Ahmedabad - 380054, Gujarat

Phone Number: 079-67128500/8700 Website: www.einfochips.com
CIN: U72200GJ1988PTC010830

ACKNOWLEDGEMENT

The internship opportunity I am having with Einfochips-An Arrow Company was a great chance for learning and professional development. Therefore, I consider myself as a very lucky individual as I was provided with an opportunity to be a part of it.

Bearing in mind previous I am using this opportunity to express my deepest gratitude and special thanks to the JIMMY PATEL, Technical lead at the Company who in spite of being extraordinarily busy with her duties, took time out to hear, guide, and keep me on the correct path and allowing me During My training at Einfochips-An Arrow Company.

I express my deepest thanks to Miss. Shruti Pethani , Head of customer support for taking part in useful decisions& giving necessary advice and guidance and arranging all facilities to make life easier. I choose this moment to acknowledge his/her contribution gratefully.

ABSTRACT

Industrial training is an important phase of a student's life. A well-planned, properly executed, and evaluated industrial training helps a lot in developing a professional attitude. It develops an awareness of the industrial approach to problem-solving, based on a broad understanding of the process and mode of operation of an organization.

The aim and motivation of this industrial training are to receive discipline, skills, teamwork, and technical knowledge through a proper training environment, which will help me, as a student in the field of IT, to develop responsiveness to the self-disciplinary nature of problems.

During 12 weeks of training at Einfochips-An Arrow Company, I was assigned as a technical supporter to help the company in the field of IT. Throughout this industrial training, I have learned new software development languages used in automation and software such as Python, Python in Selenium, Robot Framework.

*I am also able to implement what I have learned over the past year as an **Electronics and Communication** student at **L.J. INSTITUTE OF ENGINEERING & TECHNOLOGY**, Ahmedabad.*

Table of Contents

Acknowledgement.....	i
Abstract.....	ii
List of Figures.....	iii
Chapter 1: Company Overview.....	
1.1 History	
1.2 Services Provided by Einfochips	
1.2.1 Digital Engineering	
1.2.2 Device Engineering	
1.2.3 Quality engineering	
1.2.4 Silicon Engineering	
1.3 Capacity of Plant	
1.4 Chart of Organization	
1.5 Overview Of Different Department Of The Organization	
Chapter 2: My Role in my ongoing Internship at the Company.....	
Chapter 3: Introduction to Python & Automation.....	
3.1 Introduction	
3.2 Advantages of Python & Automation	
3.3 Disadvantages of Python & Automation	
Chapter 4: Python (Project).....	
4.1 Movie Guessing Game	
4.2 Rock, Paper, scissor Game	
4.3 Flames Game	

4.4 Tic Tac Toe Game

4.5 Snake and Ladder game

4.6 Auction

4.7 Game Zone

Chapter 5: Python with Selenium.....

Chapter 6: Python Plagiarism checking from text files.....

Chapter 7: Robot Framework Automation.....

Chapter 8: GitHub.....

Chapter 9: Full Automation of Websites Using Robot Framework.....

9.1 Automation on PhpTravels Website

9.2 Automation on Make My Trip Website

Chapter 1: Company Overview

1.1 History

--> EInfochips, an Arrow company, is a leading global provider of product engineering and semiconductor design services. With over 500+ products developed and 40M deployments in 140 countries, eInfochips continues to fuel technological innovations in multiple verticals. The company's service offerings include digital transformation and connected IoT solutions across various cloud platforms, including AWS and Azure.

--> Vision of Einfochips:

"We want to make eInfochips a leading global innovative technology company that will have a transformational impact on the society by creating leaders and generating stakeholder value"

– Sumit Sethi, COO.

Founded in 1994, our work culture is built over years of experience in providing innovative solutions to our clients and our indomitable spirit to excel in all aspects of our engagement. We are geographically spread across India, Japan, and the USA with 10 design centers and offices.

The fuel to our growth is driven by our passion and strong commitment to our core values:

- Customer First
- Disciplined Execution
- Embrace Impossible Challenges
- Continuous Learning
- Serving Society





1.2 Services provided by Einfochips:

--Digital Engineering

- ~Cloud Services
- ~DevOps Services
- ~IoT Solutions & Services
- ~Mobility Solutions & Services
- ~Machine Learning Services
- ~Data Analytics Services
- ~Remote Device Management

--Device Engineering

- ~Hardware Design Services
- ~Design to Manufacturing
- ~Embedded Systems & Software Development
- ~Multimedia & Digital Solutions
- ~Image Tuning Service
- ~Product Sustenance Engineering

--Quality Engineering

- ~Product Testing Services
- ~IoT Testing Services
- ~Device-to-Cloud QA Automation Services
- ~Cognitive QA Services
- ~Mobile and Web Testing Services
- ~Quality Process Consulting ~Wireless
Testing Services

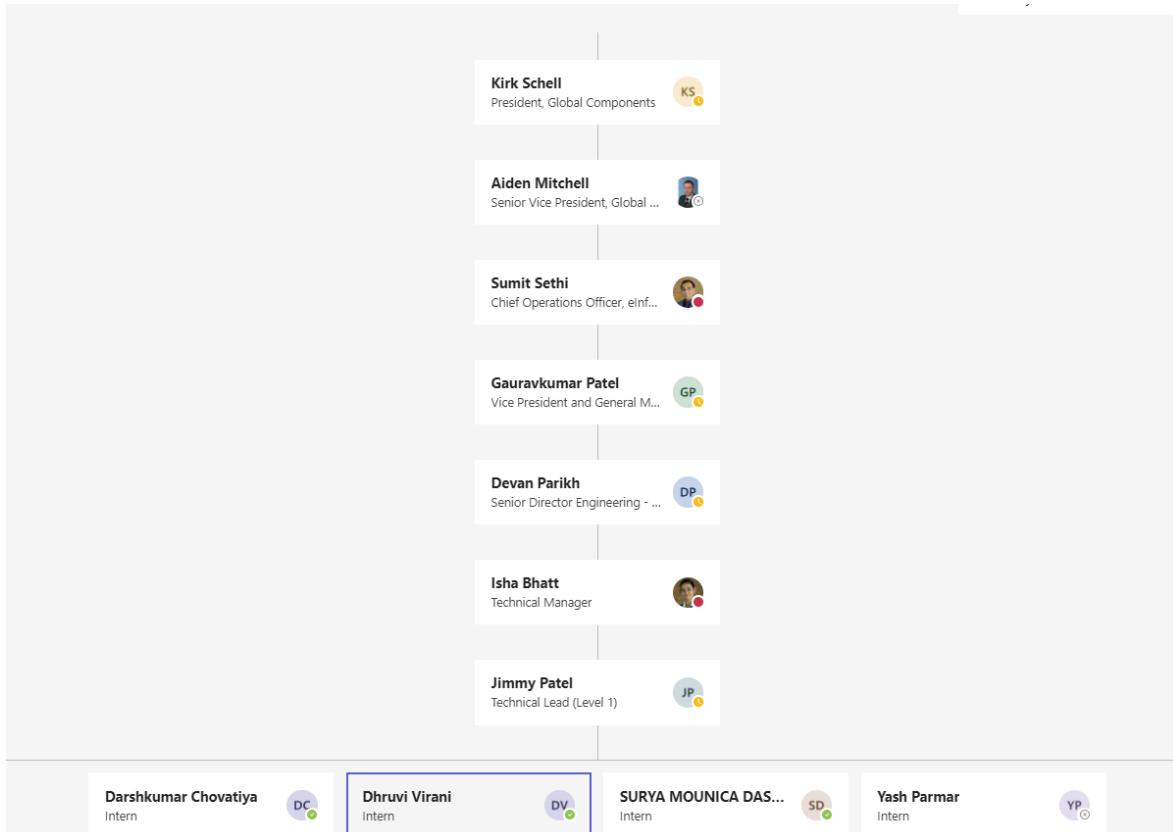
--Silicon Engineering

- ~ASIC/FPGA Design & Development
- ~Design Verification & Validation
- ~Physical Design and DFT

1.3 Capacity of plant

Size: 1001 to 5000 Employees

1.4 Chart of Organization

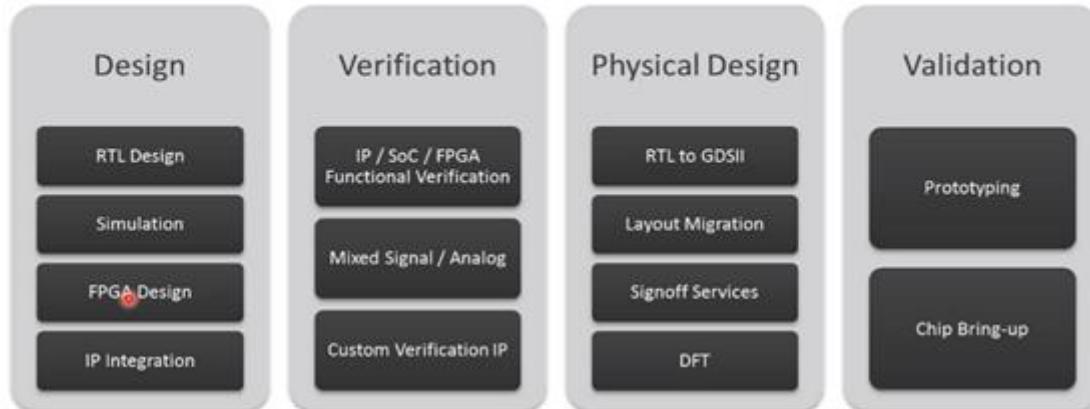


1.5 OVERVIEW OF DIFFERENT DEPARTMENT OF THE ORGANIZATION

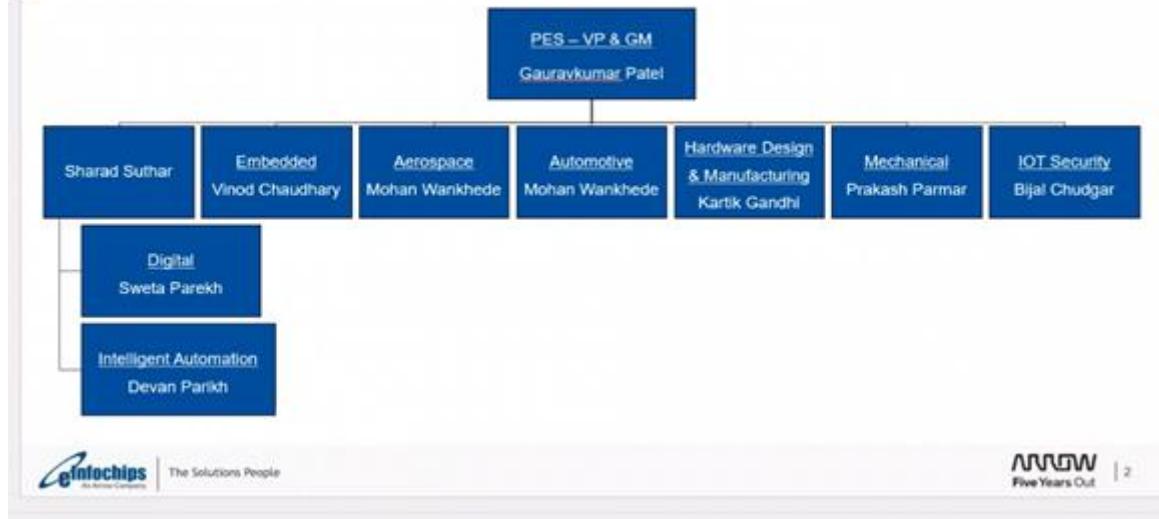
There are mainly 2 departments at E-Infochips.

- ASIC
- PES
- QEG

ASIC Services



PES Team Structure



Chapter 2: My Role in my ongoing Internship at the Company

- My role in Einfochips- An Arrow Company is an engineer in Quality assurance(QA).
- I have been placed in the department of Intelligence Automation(IA).
- Under the mentorship of Jimmy Patel who is the Technical Guide- Level 1 of the IA department, I have been assigned in automation field in robot framework with python which is one of the most important framework in the field of automation.
- I have been given a project of Python to make certain games using different functions and classes at first.
- Python with selenium was also taught to us to understand the working of how automation can be done.
- Further, using robot framework, I have been given several tasks to automate different websites.

Chapter 3: Introduction to Python & Automation

3.1 Introduction:

Python:

Python was designed for readability, and has some similarities to the English language with influence from mathematics. Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses. Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Python was designed for readability, and has some similarities to the English language with influence from mathematics. Python was designed for readability, and it uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses. Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Python was designed for readability, and has some similarities to the English language with influence from mathematics.

It was created by Guido van Rossum in 1991 and further developed by the Python Software Foundation. It was designed with an emphasis on code readability, and its syntax allows programmers to express their concepts in fewer lines of code. Python is a programming language that lets you work quickly and integrate systems more efficiently.

Python can be used on a server to create web applications.

Python can be used alongside software to create workflows.

Python can connect to database systems. It can also read and modify files.

Python can be used to handle big data and perform complex mathematics.

Python can be used for rapid prototyping, or for production-ready software development

Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).

Python has a simple syntax similar to the English language.

Python has syntax that allows developers to write programs with fewer lines than some other programming languages.

Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.

Python can be treated in a procedural way, an object-oriented way or a functional way.

Python was designed for readability, and has some similarities to the English language with influence from mathematics.

Python uses new lines to complete a command, as opposed to other programming languages which often use semicolons or parentheses.

Python relies on indentation, using whitespace, to define scope; such as the scope of loops, functions and classes. Other programming languages often use curly-brackets for this purpose.

Automation:

Automation Testing is a software testing technique that performs using special automated testing software tools to execute a test case suite. On the contrary, Manual Testing is performed by a human sitting in front of a computer carefully executing the test steps.

The automation testing software can also enter test data into the System Under Test, compare expected and actual results and generate detailed test reports. Software Test Automation demands considerable investments of money and resources.

Successive development cycles will require execution of same test suite repeatedly. Using a test automation tool, it's possible to record this test suite and re-play it as required. Once the test suite is automated, no human intervention is required. This improved ROI of Test Automation. The goal of Automation is to reduce the number of test cases to be run manually and not to eliminate Manual Testing altogether.

Test Automation is the best way to increase the effectiveness, test coverage, and execution speed in software testing. Automated software testing is important due to the following reasons:

- ~ Manual Testing of all workflows, all fields, all negative scenarios is time and money consuming
- ~It is difficult to test for multilingual sites manually
- ~Test Automation in software testing does not require Human intervention. You can run automated test
- ~Test Automation in software testing does not require Human intervention. You can run automated test unattended.
- ~Test Automation increases the speed of test execution
- ~Automation helps increase Test Coverage
- ~Manual Testing can become boring and hence error-prone.

3.2 Advantages of Python & Automation:

The knowledge of a programming language can help a QA specialist in different ways. Besides providing access to test automation tools, the ability to understand code also advances manual testing. One way or another, it enhances a person's competencies and makes a software testing company better prepared for the market challenges. The question is what programming language to learn and use for writing tests. Usually, tech specialists face a "Python vs Java" dilemma. Let's focus on the former in today's post.

Python test automation is simply using Python for test automation.

Python is one of the testing teams' favorite programming languages. It has multiple features that make it suitable for automated testing. For example, you can benefit from Python's dynamic typing but also make use of checkers like Pyre for static typing. Python is easy to learn and can also be used for complex problems. Therefore, users of different skill levels can use Python.

Python is well-known for having a library available for almost everything, so you don't need to waste time writing code from scratch. And if you find something that's not exactly what you wanted, you can also make changes to the libraries and use them as per your use case. And in addition to all this, Python has an amazing community to help you when you're stuck.

PyTest is a native Python test library with a superset of PyUnit's features. Rather than modeling JUnit's architecture, it has a distinctly Python flavor. It makes heavy use of Python decorators and assertions.

PyTest also supports parameterized testing (without the aid of plugins like Nose) that improves code reuse and simplifies code coverage.

You can create a program using Python for your web automation tasks like gathering data from websites and putting the data into an Excel spreadsheet. Most repeated workflows involving actions on the web with defined steps can be automated by writing a Python program. Once you have your scripts written, it's simple to reuse them to make your routine tasks easier.

For an all-around better web automation solution, look to robotic process automation (RPA). With RPA, website automation is made easy for even the most basic business users. Plus, with a great RPA solution, you can easily integrate any existing Python scripts while moving forward with automation for other departments in your enterprise-wide workflows so you don't have to start completely from scratch. The best RPA software should be easy to use with simple drag-and-drop features. That way, anyone on staff can become an automation expert without having to become a programmer.

3.2 Disadvantages of Python & Automation:

The downside of using Python for web automation is the programming. You're going to need some pretty extensive programming skills, and if you're not a trained programmer—or don't have one on staff—setting up automated tasks with Python can be a daunting task. If you don't have a Python expert, you shouldn't rely on Python for automating your workflows. And if you happen to have legacy Python programs already implemented that break, you'll need that expert around to put the pieces back together.

Chapter 4: Python

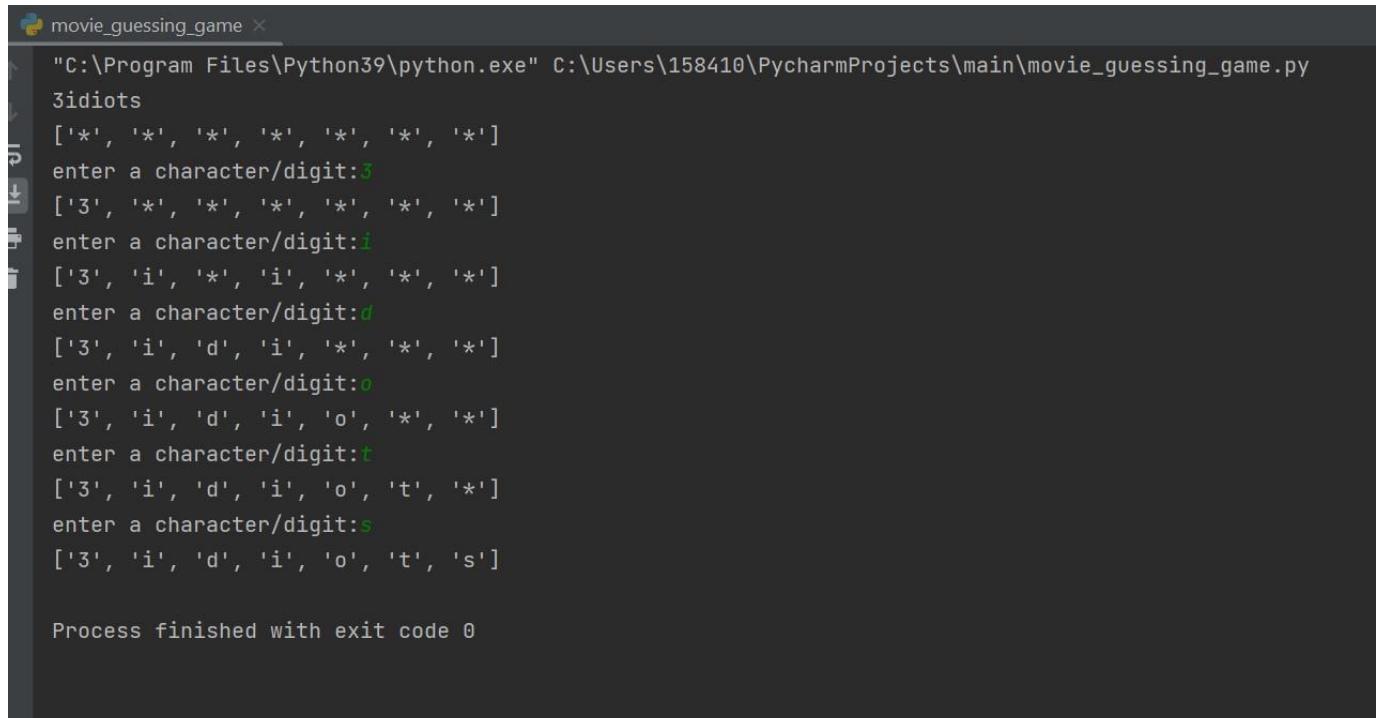
4.1 Movie Guessing Game:

My first code of the project was to make a movie guessing game which is a one user game in which a random movie will be selected from the list of movie and there will be 5 chances to guess the correct character of the movie name.

```
import random

def A():
    #game = 30
    movie = ["3idiots", "Shershah", "Pathaan", "Yeh jawani Hai deewani"]
    movie = random.choice(movie)
    movie = movie.replace(" ", "")
    movie = movie.lower()
    print(movie)
    movie = list(movie)
    movie1 = ["*"] * len(movie)
    print(movie1)
    chance = 5
    while (movie != movie1 and chance > 0):
        c = input("enter a character/digit:")
        if c in movie:
            for i in range(len(movie)):
                if movie[i] == c:
                    movie1[i] = c
        else:
            chance = chance - 1
            print("wrong answer you have", chance, "chances left")
        print(movie1)
A()
```

Output:



```
movie_guessing_game x
C:\Program Files\Python39\python.exe" C:\Users\158410\PycharmProjects\main\movie_guessing_game.py
3idiots
['*', '*', '*', '*', '*', '*', '*']
enter a character/digit:3
['3', '*', '*', '*', '*', '*', '*', '*']
enter a character/digit:i
['3', 'i', '*', 'i', '*', '*', '*']
enter a character/digit:d
['3', 'i', 'd', 'i', '*', '*', '*']
enter a character/digit:o
['3', 'i', 'd', 'i', 'o', '*', '*']
enter a character/digit:t
['3', 'i', 'd', 'i', 'o', 't', '*']
enter a character/digit:s
['3', 'i', 'd', 'i', 'o', 't', 's']

Process finished with exit code 0
```

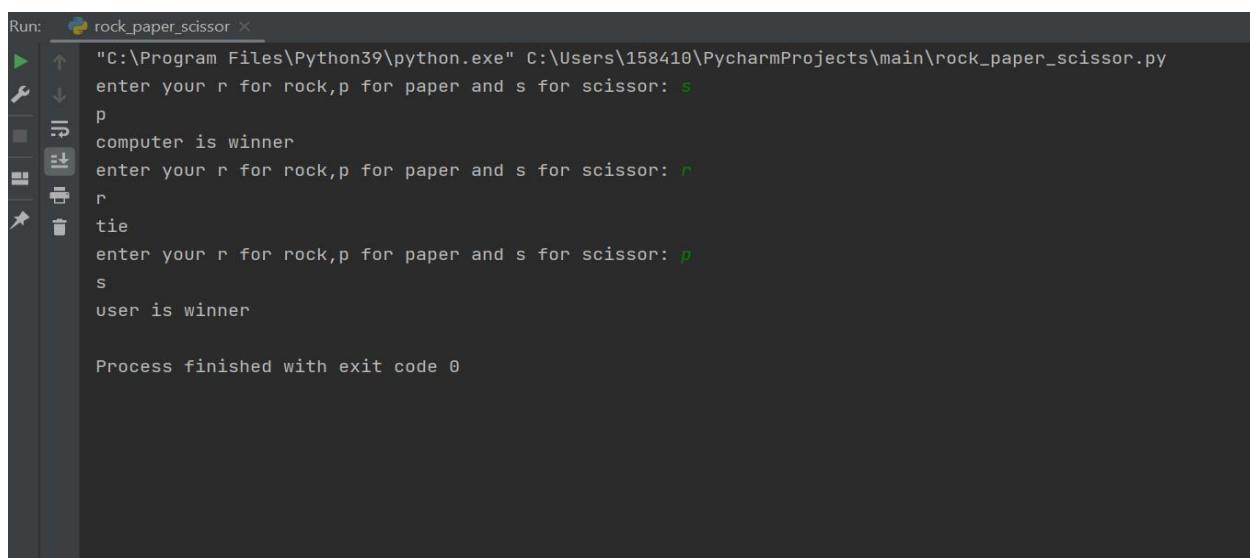
4.2 Rock ,Paper, scissor game:

My second game of the project was to make a rock, paper scissor game which is a two player game in which first user will enter r(rock) or p(paper) or s(scissor) and computer will choose random input.

```
import random

def B():
    # rock paper scissor game
    chance = 0
    while chance < 3:
        turn = ["r", "p", "s"]
        turn = random.choice(turn)
        user = input("enter your r for rock,p for paper and s for scissor: ")
        print(turn)
        # turn=list(turn)
        b = turn
        if user == "r" and b == "p":
            print("computer is winner")
        elif user == "s" and b == "r":
            print("computer is winner")
        elif user == "s" and b == "p":
            print("computer is winner")
        elif user == b:
            print("tie")
        else:
            print("user is winner")
        chance += 1
B()
```

Output:



The screenshot shows the PyCharm Run console window. The title bar says "Run: rock_paper_scissor x". The main area displays the following text output:

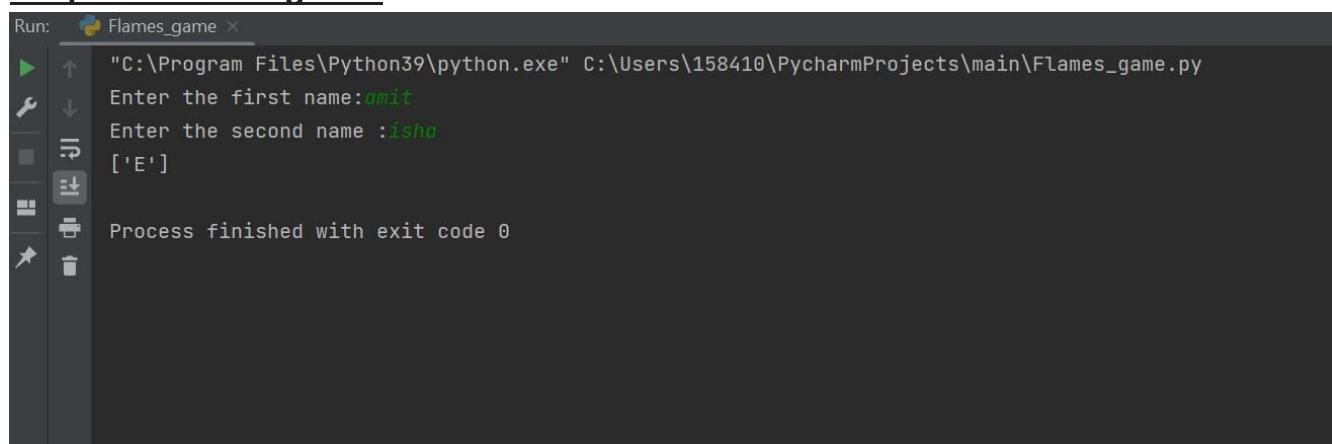
```
Run: rock_paper_scissor x
"C:\Program Files\Python39\python.exe" C:/Users/158410/PycharmProjects/main/rock_paper_scissor.py
enter your r for rock,p for paper and s for scissor: s
p
computer is winner
enter your r for rock,p for paper and s for scissor: r
r
tie
enter your r for rock,p for paper and s for scissor: p
s
user is winner

Process finished with exit code 0
```

4.3 Flames Game: My third game was to make Flames game in which user will enter two inputs of different names and the relation between that two names will be shown in the form of flames ('F' for Friend, 'L' for Love, 'A' for Affection, 'M' for Marriage, 'E' for Enemy, 'S' for Sibling)

```
def C():
    # FLAMES GAME
    string1 = "FLAMES"
    lst1 = list(string1)
    name1 = input("Enter the first name:").lower()
    name2 = input("Enter the second name :").lower()
    lsname1 = list(name1)
    lsname2 = list(name2)
    for x in lsname1[:]:
        if x in lsname2:
            lsname1.remove(x)
            lsname2.remove(x)
    total = len(lsname1) + len(lsname2)
    len_lst1 = len(lst1)
    count = 0
    # To define the range how many times the game need to be played
    while ((len(lst1)) != 1): # While length is =1 then the game is not possible
        for x in range(len(lst1)):
            count += 1
            if (count == total):
                count = 0
                lst1.remove(lst1[x])
    # print("The list size ",len_lst1)
    # len_lst1-=1
    print(lst1)
C()
```

Output of Flames game:



The screenshot shows the PyCharm IDE's Run tab. The title bar says "Run: Flames_game". The run configuration dropdown shows "C:\Program Files\Python39\python.exe" C:\Users\158410\PycharmProjects\main\Flames_game.py. The main pane displays the program's output:
Enter the first name:amit
Enter the second name :isha
['E']
Process finished with exit code 0

4.4 Tic Tac Toe Game: My fourth game of the project was to make a tic tac toe game which is a two player game.

```
import numpy as np
```

```
def G():
```

```
mat = np.array([["_", "_", "_"], ["_", "_", "_"], ["_", "_", "_"]])
def print_board(mat_copy):
    global mat
    print(mat_copy)
def check_winner():
    global cwin
    for i in range(0, 3):
        if mat[i][0] == mat[i][1] and mat[i][0] == mat[i][2]:
            if mat[i][0] == "x":
                print("X won the match")
                cwin = "win"
                break
            else:
                cwin = "win"
                break
        else:
            cwin = "loss"
    for i in range(0, 3):
        if mat[0][i] == mat[1][i] and mat[0][i] == mat[2][i]:
            if mat[0][i] == "x":
                print("X won the match")
                cwin = "win"
                break
            else:
                cwin = "win"
                break
        else:
            cwin = "loss"
turn = 1
cwin = "loss"
print_board(mat)
# 1 for x and 0 for O
while (cwin != "win"):
    if (turn == 1):
        print("x turn")
        row = int(input("please enter a value of row: "))
        column = int(input("please enter a value of column: "))
        mat[row][column] = "x"
        # print(x)
        print_board(mat)
        print(cwin)
        cwin = check_winner()
        print(cwin)
    turn = 0
```

```

else:
    print("y turn")
    row = int(input("please enter a value of row: "))
    column = int(input("please enter a value of column: "))
    mat[row][column] = "O"
    # print(x)
    print_board(mat)
    cwin = check_winner()
    turn = 1
if (cwin == "win"):
    print("match over")

```

4.5 Snake and Ladder game: My Fifth game of my project was to create a snake and ladder game which is a two player game.

```

import random
ladder={1:38,4:14,8:30,21:42,28:76,50:67,71:92,80:99}
snake={97:78,95:56,88:24,62:18,48:26,36:6,32:10} i=0
position=0 b="y" while position<100 and b=="y":
    c=0
    b=input("Do you want to play: ")
    Dice=random.randint(1,6)
    print(Dice)  for i,j in
    ladder.items():  if i==position
    and c==0:
        position=j
        print("position is: ",position)
    position=position+Dice  print("position is:
    ",position)  c=1  if position>100:
        position=position-Dice  print("position
    is: ",position)  for k,n in snake.items():  if
    k==position and c==0:
        position=n
        print("position is: ",position)
    position=position+Dice  print("position is:
    ",position)  c=1  if position>100:
        position=position-Dice  if c==0:
            position=position+Dice  if
    position>100:
        position=position-Dice  if
    position==100:  print("you won")

```

4.6 Auction: My Sixth task of the project was to create an auction between buyers and sellers.

```

S1_vm = {"S1":7,"S2":6,"S3":5,"S4":9}
S1_bid = {"S1":20,"S2":23,"S3":19,"S4":30}

```

```

U1_vm={"B1":4 , "B2":3,"B3":2,"B4":5 }
U1_bid={"B1":25 , "B2":22,"B3":17,"B4":21 }

sorted_s1 = dict(sorted(S1_bid.items(), key=lambda kv: kv[1]))
sorted_u1 = dict(sorted(U1_bid.items(), key=lambda kv: kv[1], reverse=True)) print(sorted_s1)
print(sorted_u1) dict1={} for i in sorted_u1.keys(): for j in sorted_s1.keys(): if
S1_vm[j]>=U1_vm[i]:
    dict1[i]=j
    S1_vm[j]=S1_vm[j]-U1_vm[i]
    #del U1_vm[i]
break print("Allocation: ",dict1)
lst_seller=[] lst_user=[] for i,j in
dict1.items():
    lst_seller.append(S1_bid[j]) lst_user.append(U1_bid[i])
print("Price to be paid by each buyer will be: ",max(lst_seller)) print("Payment
received to all sellers will be: ",min(lst_user)) a1={} a2={} for i,j in
dict1.items():
    p=max(lst_seller)-S1_bid[j]
    q=U1_bid[i]-min(lst_user) a1[j]=p
    # print(a1) a2[i]=q
    # print(a2)
print("profit of each seller is ",a1) print("profit of each user is
",a2)
sum1=0 for i, j in a1.items(): for k,
m in dict1.items(): if (i==m):
    sum1=sum1+(j*S1_vm[k]) print(sum1)
print("the average of seller after selling the machines: ",sum1/len(a1))
sum2=0 for i,j in a2.items(): for
k,m in U1_vm.items():
    if(i==k):
        sum2=sum2 + (j*m) print(sum2)
print("the average profit of seller after selling the machines: ",sum2/len(a2))
b1=len(a1)
b2=len(a2)
print("the average profit of buyer per machine is ",(sum(a1.values())/b1)) print("the average profit of user
per machine is ",(sum(a2.values())/b2))

```

Explanation:

STEP1:

S1_vm = {"S1":7,"S2":6,"S3":5,"S4":9}

S1_bid = {"S1":20,"S2":23,"S3":19,"S4":30}

U1_vm={"B1":4 , "B2":3,"B3":2,"B4":5 }

U1_bid={"B1":25 , "B2":22,"B3":17,"B4":21 }

~ S1_vm is the dictionary for available VMs, S1_bid is the dictionary for cost of each VM.

~ U1_vm is the dictionary for required VMs of buyer, U1_bid is the dictionary for bid of each VM.

~ We have 4 sellers(CSP=CLOUD SERVICE PROVIDER) and 4 buyers/users here.

~ We are taking availability of virtual machines(VMs) with each seller and the cost per VM.

~ Similarly we are taking requirement of each buyer of VM and the bid that they are keeping.

STEP2:

```
sorted_s1 = dict(sorted(S1_bid.items(), key=lambda kv: kv[1]))
sorted_u1 = dict(sorted(U1_bid.items(), key=lambda kv: kv[1], reverse=True))
print(sorted_s1)
print(sorted_u1)
```

~ Now we are sorting the number of sellers in ascending order of their bid

~ Also we are sorting the number of buyers in descending order of their bid.

STEP3:

```
dict1={ } for i in
sorted_u1.keys():    for
j in sorted_s1.keys():
if
S1_vm[j]>=U1_vm[i]:
    dict1[i]=j
    S1_vm[j]=S1_vm[j]-U1_vm[i]
    #del U1_vm[i]
```

break

```
print("Allocation: ",dict1)
```

~ In next step, we are allocating the buyer to the seller if seller has sufficient resources then allocated. ~ We have stored the allocation in an empty dictionary named dict1.

STEP4:

```
lst_seller=[]
lst_user=[] for i,j
in dict1.items():
    lst_seller.append(S1_bid[j])
    lst_user.append(U1_bid[i])
print("Price to be paid by each buyer will be: ",max(lst_user))
print("Payment received to all sellers will be: ",min(lst_seller))
```

~ In this step, we are creating two empty list to store maximum bid of seller and minimum bid of buyer/user respectively.

STEP5: a1={} a2={} for i,j in

```
dict1.items():
    p=max(lst_seller)-
S1_bid[j]    q=U1_bid[i]-
min(lst_user)    a1[j]=p
# print(a1)    a2[i]=q    #
print(a2)
print("profit of each seller is ",a1)
print("profit of each user is ",a2)
```

- ~ In this step, we are creating another two empty lists to store profit of each seller and each buyer in those empty lists respectively.
- ~ For calculating profit of each seller, we are considering maximum value of cost and subtracting remaining values from that maximum

Here in my code, it is `max(lst_seller)-S1_bid[j]`

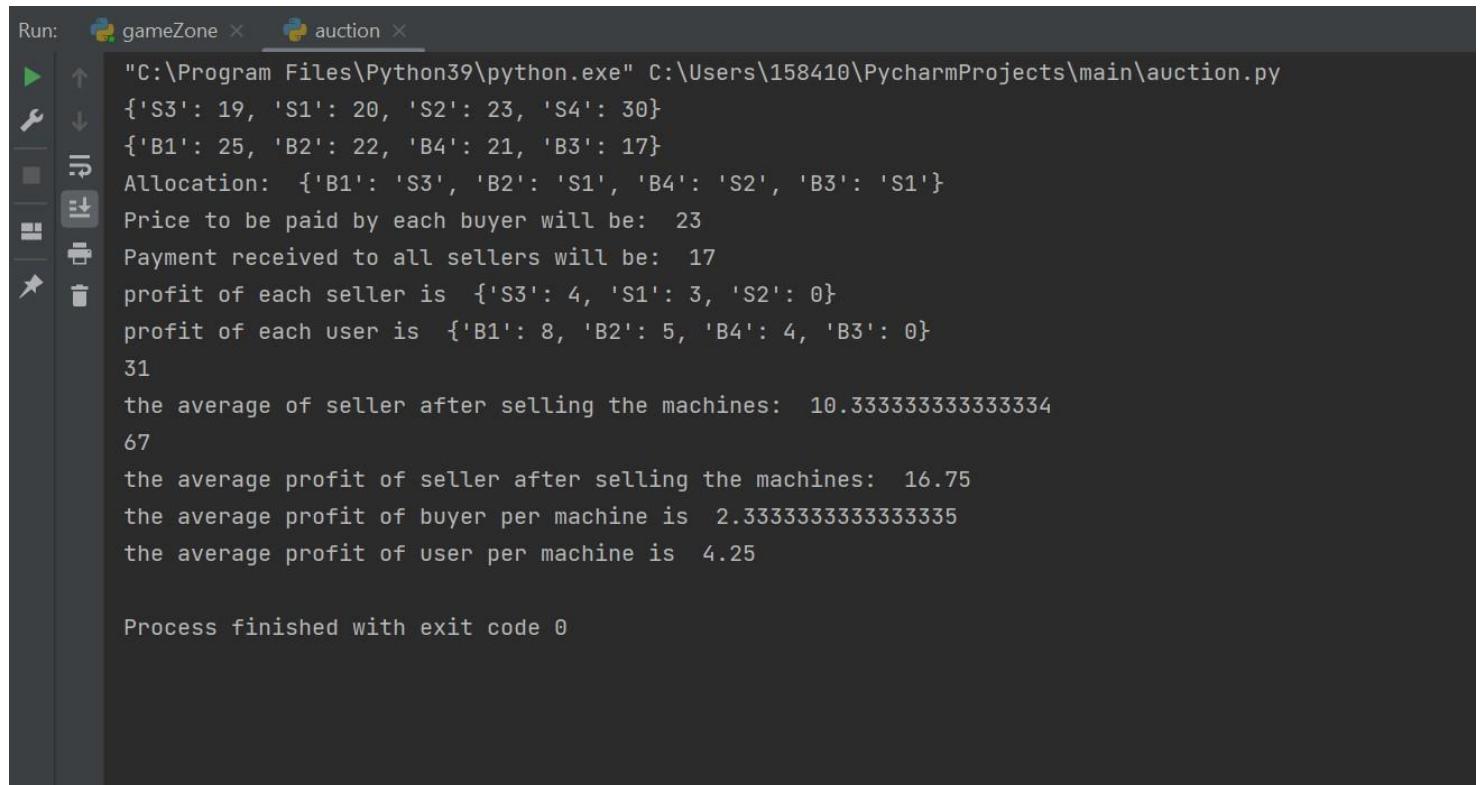
- ~ Similarly, for calculating profit of each buyer/user, we are considering minimum value of bid and subtracting remaining values to that minimum value. Here in my code, it is `U1_bid[i]-min(lst_user)`

STEP6:

```
b1=len(a1)
b2=len(a2)
print("the average profit of seller is ",(sum(a1)/b1))
print("the average profit of user is ",(sum(a2)/b2))
```

~ In this step, we are calculating the average of profit of seller and buyer/user.

~ So, here `sum(a1)` is addition of profits stored in `a1` and `sum(a2)` is addition of profits stored in `a2`.



```
Run: gameZone × auction ×
C:\Program Files\Python39\python.exe" C:\Users\158410\PycharmProjects\main\auction.py
{'S3': 19, 'S1': 20, 'S2': 23, 'S4': 30}
{'B1': 25, 'B2': 22, 'B4': 21, 'B3': 17}
Allocation: {'B1': 'S3', 'B2': 'S1', 'B4': 'S2', 'B3': 'S1'}
Price to be paid by each buyer will be: 23
Payment received to all sellers will be: 17
profit of each seller is {'S3': 4, 'S1': 3, 'S2': 0}
profit of each user is {'B1': 8, 'B2': 5, 'B4': 4, 'B3': 0}
31
the average of seller after selling the machines: 10.33333333333334
67
the average profit of seller after selling the machines: 16.75
the average profit of buyer per machine is 2.33333333333335
the average profit of user per machine is 4.25

Process finished with exit code 0
```


4.7 Game Zone: My final task of the project was to merge all the games into a gamezone in which user will have a bonus card with a limit of 500 points and each game will cost 30 points. Also there will be cafe from which user can buy food from the available balance.

```
import random
print("Welcome to GameZone\n"\u0001f600)
games={"A":"movie_guessed game", "B":"rock_paper_scissor", "C":"Flames", "D":"cafe", "G":"TicTacToe", "F":s
print(games)
for i, j in games.items():
    print("Enter", i, "to play", j)
userValue = input("Enter Your choice: ").lower()
balance = 500 #total balance is 500 when entering into gamezone and per game costs it 30 rupees
Flag=0
def A():
    #game = 30
    global balance
    balance = balance - 30
    print(balance)
    movie = ["3idiots", "Shershah", "Pathaan", "Yeh jawani Hai deewani"]
    movie = random.choice(movie)
    movie = movie.replace(" ", "")
    movie = movie.lower()
    print(movie)
    movie = list(movie)
    movie1 = ["*"] * len(movie)
    print(movie1)
    chance = 5
    while (movie != movie1 and chance > 0):
        c = input("enter a character/digit:")
        if c in movie:
            for i in range(len(movie)):
                if movie[i] == c:
                    movie1[i] = c
        else:
            chance = chance - 1
        print("wrong answer you have", chance, "chances left")
```

```

    print(movie1)

if chance > 0:
    balance = balance + 20
    print("Your current balance is: ", balance)

def B():
    # rock paper scissor game
    global balance
    balance = balance - 30
    print(balance)
    chance = 0
    while chance < 3:
        turn = ["r", "p", "s"]
        turn = random.choice(turn)
        user = input("enter your r for rock,p for paper and s for scissor: ")

```

```

    print(turn)
    # turn=list(turn)
    b = turn
    if user == "r" and b == "p":
        print("computer is winner")
    elif user == "s" and b == "r":
        print("computer is winner")
    elif user == "s" and b == "p":
        print("computer is winner")
    elif user == b:
        print("tie")
    else:
        balance = balance + 20
        print("Your current balance is: ", balance)
        print("user is winner")

```

```

        chance += 1
def C():
    # FLAMES GAME
    print("This game is just for fun so no bonus points will be given here")
    global balance
    balance = balance - 30
    print(balance)
    string1 = "FLAMES"
    lst1 = list(string1)
    name1 = input("Enter the first name:").lower()
    name2 = input("Enter the second name :").lower()
    lsname1 = list(name1)
    lsname2 = list(name2)
    for x in lsname1[:]:
        if x in lsname2:

```

```

lsname1.remove(x)
lsname2.remove(x)

total = len(lsname1) + len(lsname2)
len_lst1 = len(lst1)
count = 0

# To define the range how many times the game need to be played
while ((len(lst1)) != 1): # While length is =1 then the game is not possible
    for x in range(len(lst1)):
        count += 1
        if (count == total):
            count = 0
            lst1.remove(lst1[x])
    # print("The list size ",len_lst1)
    # len_lst1-=1
print(lst1)

def D():
    # exercise of restaurant
    global balance
    # balance = balance - 100
    # print(balance)
    menu = {"pizza": 90, "burger": 50, "sandwich": 60, "pasta": 75, "buttermilk": 30}
    print("menu list\n")
    a = "".lower() # a is the key
    sum = 0
    lst_order = {}
    for i, j in menu.items():
        print(i, "price", j)
    while (a != "no"):

        order = input("\nenter your order: ").lower()
        Quantity = input("enter the quantity: ")

        lst_order[order] = Quantity # the order and quantity is getting stored in lst_order dictionary
        print("\n'press ok to order more and no to stop'\n")
        a = input("do you want to order more?").lower()
    print("\norder confirmation: ", order)
    for i, j in lst_order.items():
        for k, m in menu.items():
            if (i == k):
                sum = sum + int(j) * int(m) # here j is the quantity which is stored in lst_order and m is the price
    print("your total bill is: ", sum)
    add = input("do you want to add items in your order?: ").lower()
    if add == "yes":
        order = input("please enter your order: ").lower()
        quantity = input("please enter the quantity: ")
        lst_order[order] = quantity

```

```

sum = 0
for i, j in lst_order.items():
    for k, m in menu.items():
        if (i == k):
            sum = sum + int(j) * int(m)
print("your total bill is: ", sum)
r = input("do you want to remove any item?: ").lower()
if r == "yes":
    remove_item = input("enter the item you want to remove: ").lower()
    del lst_order[remove_item]
sum = 0
for i, j in lst_order.items():
    for k, m in menu.items():
        if (i == k):
            sum = sum + int(j) * int(m)
print("your total bill is: ", sum)

print("Thank you for ordering")
balance=balance-sum
print(balance)

def F():
    global balance
    balance = balance - 30
    print(balance)
    ladder = {1: 38, 4: 14, 8: 30, 21: 42, 28: 76, 50: 67, 71: 92, 80: 99}
    snake = {97: 78, 95: 56, 88: 24, 62: 18, 48: 26, 36: 6, 32: 10}
    i = 0
    position = 0
    b = "y"
    while position < 100 and b == "y":
        c = 0
        b = input("Do you want to play: ")

        Dice = random.randint(1, 6)
        print("position is: ", position)
        print(Dice)
        # for ladder part
        # j is the position of user after dice
        for i, j in ladder.items():
            if i == position and c == 0:
                position = j
                print("position is: ", position)
                position = position + Dice
                print("current position is: ", position)
                c = 1
            if position > 100:
                position = position - Dice
                print("position is: ", position)

```

```

for k, n in snake.items():
    if k == position and c == 0:
        position = n
        print("position is: ", position)
        position = position + Dice
        print("position is: ", position)
        c = 1
        if position > 100:
            position = position - Dice
    if c == 0:
        position = position + Dice
        if position > 100:
            position = position - Dice
    if position == 100:
        print("you won")
def G():
    global balance
    balance = balance - 30
    print(balance)
    import numpy as np
    mat = np.array([["_", "_", "_"], ["_", "_", "_"], ["_", "_", "_"]])

    def print_board(mat_copy):
        global mat
        print(mat_copy)

    def check_winner():
        global cwin

        for i in range(0, 3):
            if mat[i][0] == mat[i][1] and mat[i][0] == mat[i][2]:
                if mat[i][0] == "x":
                    print("X won the match")
                    cwin = "win"
                    break
                else:
                    cwin = "win"
                    break
            else:
                cwin = "loss"
        for i in range(0, 3):
            if mat[0][i] == mat[1][i] and mat[0][i] == mat[2][i]:
                if mat[i][0] == "x":
                    print("X won the match")

```

```

cwin = "win"
break
else:
    cwin = "win"
    break
else:
    cwin = "loss"

turn = 1
cwin = "loss"
print_board(mat)
# 1 for x and 0 for 0
while (cwin != "win"):
    if (turn == 1):
        print("x turn")

        row = int(input("please enter a value of row: "))
        column = int(input("please enter a value of column: "))
        mat[row][column] = "x"
        # print(x)
        print_board(mat)
    |   print(cwin)
    |   cwin = check_winner()
    |   print(cwin)
    |   turn = 0
    else:
        print("y turn")
        row = int(input("please enter a value of row: "))
        column = int(input("please enter a value of column: "))
        mat[row][column] = "0"
        # print(x)

        print_board(mat)
        cwin = check_winner()
        turn = 1

    if (cwin == "win"):
        print("match over")

print(userValue)
while(userValue != 'e'):
    if(userValue == 'a'):
        A()
    elif(userValue == 'b'):
        B()
    elif(userValue == 'c'):
        C()

```

```

elif(userValue == 'd'):
    D()
elif(userValue=='f'):
    F()
elif(userValue=='g'):
    G()
elif (userValue == 'e'):
    break
# elif(userValue==
# 'G'):
#     G()
else:
    print("Invalid game option")
print(games)
userValue = input("Enter Your choice: ").lower()

print("Your current balance is ", balance)

```

Output:

```

Run: gameZone X
C:\Program Files\Python39\python.exe C:/Users/158410/PycharmProjects/main/gameZone.py
Welcome to GameZone
{:A: 'movie_guessing game', 'B': 'rock_paper_scissor', 'C': 'Flames', 'D': 'cafe', 'G': 'TicTacToe', 'F': 'snake_ladder', 'E': 'Exit Zone'}
Enter A to play movie_guessing game
Enter B to play rock_paper_scissor
Enter C to play Flames
Enter D to play cafe
Enter G to play TicTacToe
Enter F to play snake_ladder
Enter E to play Exit Zone
Enter Your choice: c
c
This game is just for fun so no bonus points will be given here
470
Enter the first name:kajol
Enter the second name :ajay
['F']
{:A: 'movie_guessing game', 'B': 'rock_paper_scissor', 'C': 'Flames', 'D': 'cafe', 'G': 'TicTacToe', 'F': 'snake_ladder', 'E': 'Exit Zone'}
Enter Your choice:

```

Chapter 5: Python With Selenium

→ Selenium Python bindings provide APIs using which you can write functional tests using the Selenium WebDriver. Like other Selenium language bindings, Selenium Python APIs can be leveraged to develop highly efficient tests that let you interact with the WebElements on the AUT (Application Under Test).

The bindings support local browser drivers (e.g., Chrome, Firefox, Internet Explorer, etc.) and provide a remote WebDriver that lets you connect to a remote cloud Selenium Grid.

→ Different websites such as kubernetes, webucator, docker, pizza hut has been automated by me.

→ Using different actions such as .click(), .is_displayed(), .send_keys(), .scroll_location_into_view(), .is_selected(), driver.find_elements(), action chains, etc, we can automate any type of website.

→ To automate a website, we need to inspect the particular element by Xpath, css locators, id, class,etc.

→ Several libraries and keys need to be install and import at the beginning of the python file. such as

- from selenium import webdriver
- from selenium.webdriver.chrome.service import Service as Service
- from webdriver_manager.chrome import ChromeDriverManager
- from selenium.webdriver.common.by import By
- from Reusable import common
- from selenium.webdriver.support.select import Select
- from selenium.webdriver.common.action_chains import ActionChains
- from selenium.webdriver.common.alert import Alert your text

Code of automation to validate sign in of webucator website:

```
def Sign_in(self):  
    driver.get("https://www.webucator.com/account/login/")  
    driver.implicitly_wait(10)  
    time.sleep(2)  
    driver.find_element(By.NAME, "login").click()  
    driver.find_element(By.NAME,  
    "login").send_keys(common.resdXMLAsPerNode("email"))  
    driver.find_element(By.NAME, "password").click()  
    driver.find_element(By.NAME,  
    "password").send_keys(common.resdXMLAsPerNode("passc"))  
    time.sleep(3)
```

```

driver.find_element(By.XPATH, "(//button[text()='Sign In'])).click()
# if driver.find_element(By.XPATH, "(//div[@class='text-danger']/strong)").get_attribute('value')!="" :
if driver.find_element(By.XPATH,
    "(//div[@class='text-danger']/strong)").text==common.resdXMLAsPerNode("passc"):
    print(common.resdXMLAsPerNode("error"))
time.sleep(3)

```

Code of automation to validate the search box and scrolling of flipkart website:

```

def practice_website2(self):
global driver

    driver = webdriver.Chrome()
driver.get("https://www.flipkart.com/")
driver.implicitly_wait(10)
driver.maximize_window()

    driver.find_element(By.XPATH,"//button[@class='_2KpZ6l _2doB4z']").click()
time.sleep(2)

    driver.find_element(By.XPATH,
        "//form[@class='_2M8cLY header-form-search']").is_displayed()
time.sleep(2)

    driver.find_element(By.XPATH,
        "//input[@class='_3704LK']").send_keys("i phone 12 128bg")
driver.find_element(By.XPATH,
    "//button[@class='L0Z3Pu']").click()
time.sleep(2)

    ele = driver.find_element(By.XPATH,
        "//div[text()='APPLE iPhone 12 Pro (Gold, 128 GB)']")
actionChains = ActionChains(driver)
actionChains.move_to_element(ele).perform()

    #driver.find_element(By.XPATH,
    "//div[text()='APPLE iPhone 12 Pro (Gold, 128 GB)']").location_once_scrolled_into_view
    #driver.find_element(By.XPATH,
    "//div[text()='APPLE iPhone 12 Pro (Gold, 128 GB)']").is_displayed()
time.sleep(2)

    driver.find_element(By.XPATH,
        "//div[text()='APPLE iPhone 12 Pro (Gold, 128 GB)']").click()
time.sleep(3)

```

Code of automation to drag and drop:

```

def test_DemoWebsite(self):
    global driver      driver =
webdriver.Chrome()

    driver.get("http://www.dhtmlgoodies.com/scripts/drag-drop-custom/demo-drag-drop-
3.html")      driver.implicitly_wait(5)      driver.maximize_window()

    src=driver.find_element(By.XPATH,"//div[@id='box5']")
dest=driver.find_element(By.XPATH,"//div[@id='box105']")

```

```
actionChains=ActionChains(driver)
actionChains.drag_and_drop(src,dest).perform()
```

Code of automation to validate the alert and confirm box:

```
class test_selenium: def
handle_Web_Alert(self):
    driver.get("https://the-internet.herokuapp.com/javascript_alerts")
    driver.implicitly_wait(10) time.sleep(2)

    driver.find_element(By.XPATH, "//button[contains(.,'Click for JS Alert')]").click()
    time.sleep(2) alert = Alert(driver) if alert.text == "I am a JS Alert":
        alert.accept()
    time.sleep(1)

    driver.find_element(By.XPATH,
        "//p[text()='You successfully clicked an alert']").is_displayed()
    print("Click for Js Alert Pass.") else:
        print("Click for Js Alert Fail.")

    # Click for Js Confirmation.....
    driver.find_element(By.XPATH, "//button[contains(.,'Click for JS Confirm')]").click()
    time.sleep(2) alert = Alert(driver) if alert.text == "I am a JS Confirm":
        alert.dismiss()
    time.sleep(2)

    driver.find_element(By.XPATH, "//p[text()='You clicked: Cancel']").is_displayed()
    print("Click for Js confim pass.") else:
        print("Click for Js confim fail.")

    # Click for Js Confirmation.....
    driver.find_element(By.XPATH, "//button[contains(.,'Click for JS Prompt')]").click()
    time.sleep(2) alert = Alert(driver) if alert.text == "I am a JS prompt":
        alert.send_keys("dhruvi")
    alert.accept() time.sleep(2)

    driver.find_element(By.XPATH, "//p[text()='You entered: dhruvi']").is_displayed()
    print("Click for Js Prompt With name pass.") else:
        print("Click for Js Prompt With namefail.")
obj=test_selenium obj.handle_Web_Alert()
```

Chapter 7: Python Plagiarism checking from text files

To check the plagiarism of two different codes using python code

Jigar Dankhara [158171]
Dhruvi Virani [158410]

Plagiarism checking from text files

- To be done in team of 2
- Write an essay of 10 line on the same topic as text file (using file functions only))
- Read both text files
- Find frequency of common words between them (ignore case sensitivity)
- % of plagiarism= total frequency of common words/ max(len(file))

Mother1.txt

my mother
my mother is everything for me.
My mother is an ordinary woman she is my superhero.
In every step of my, she supported and encouraged me.
Whether day or night she was always there for me no matter what the condition is.
Furthermore, her every work, persistence, devotion, dedication, conduct is an inspiration for me.
In this essay on my mother, I am going to talk about my mother and why she is so special to me.
She is an essence of truthfulness, love, and sincerity. Another reason is that she showers her family with her blessing and love.
Furthermore, she gives us everything but never demand anything in return.
The way she cares for everyone in the family inspires me to the same in my future.
Also, her love is not just for the family she treats every stranger and animals the same way she did to me.
Due to, this she is very kind and sensible towards the environment and animals.

Mother2.txt

MY MOTHER
My mother is my god.
She takes care of me a lot.
She cooks very good food.
She always buy me new things whenever i want.
She always comes with me for shopping.
She is very sweet kind and caring.
My mom is a housewife.
Also she is very fast at her household work.
She is my everything.
I love my mom a lot.

```

#creating file1 with its content
file1=open("mother1.txt","w")
line=[ "my mother\n", "my mother is everything for me.\n", "My mother is an ordinary woman she is my superhero.\n", "In every step of my, she supported and encouraged me.\n", "Whether day or night she was always there for me no matter what the condition is.\n", "Furthermore, her every work, persistence, devotion, dedication, conduct is an inspiration for me. \n",
"In this essay on my mother, I am going to talk about my mother and why she is so special to me.\n", "She is an essence of truthfulness, love, and sincerity. Another reason is that she showers her family with her blessing and love.\n",
"Furthermore, she gives us everything but never demand anything in return.\n", "The way she cares for everyone in the family inspires me to the same in my future.\n",
"Also, her love is not just for the family she treats every stranger and animals the same way she did to me.\n", "Due to, this she is very kind and sensible towards the environment and animals."]
file1.writelines(line)
file1.close()

#creating file2 with its content
file2=open("mother2.txt","w")
line=[ "MY MOTHER\n",
"My mother is my god.\n",
"She takes care of me a lot.\n",
"She cooks very good food.\n",
"She always buys me new things whenever I want.\n",
"She always comes with me for shopping.\n",
"She is very sweet, kind and caring.\n",
"My mom is a housewife.\n",
"Also she is very fast at her household work.\n",
"She is my everything.\n",
"I love my mom a lot."]
file2.writelines(line)
file2.close()

#creating dictionary m1 for file1
file1=open("mother1.txt","r")
m1={}
con=0
for i in file1.read().lower().split():
    con=0
    file2=open("mother1.txt","r")
    for jc in file2.read().lower().split():
        if(i==jc):

```

```

        con=con+1
        m1[i]=con
        file2.close()

#creating dictionary m2 for file2
file1=open("mother2.txt","r")
m2={}
con=0
for i in file1.read().lower().split():
    con=0
    file2=open("mother2.txt","r")
    for jc in file2.read().lower().split():
        if(i==jc):
            con=con+1
    m2[i]=con
    file2.close()

#find total frequency of common words
sum=0
for i,j in m1.items():
    for p,k in m2.items():
        if(i==p):
            if(j>k):
                sum=sum+k

#find maximum length of the file
lenm1=len(m1)
lenm2=len(m2)
pmax=0
if(lenm1>lenm2):
    pmax=lenm1
else:
    pmax=lenm2
#find plagiarism of both the file.
plg=(sum/pmax)*100
print("your document have ","{:.2f}%".format(plg)," plagiarism")

output:-

your document have 25.26 % plagiarism

```

Chapter 8: Robot Framework Automation

Robot Framework is a generic, Python-based, open-source automation framework. It can be used for test automation and robotic process automation (RPA).

Robot Framework is supported by Robot Framework Foundation. Many industry-leading companies use the tool in their software development.

Robocorp provides tools to write, execute and orchestrate software robots that are powered by Robot Framework to be used in RPA, so understanding the basics is fundamental for any Software Robot Developer.

Robot Framework code aims to be easily readable so that even an untrained eye can understand what the code does.

Keywords are operations that your robot can execute to accomplish various tasks, and are the foundation of any robot script. Keywords can be reused, and they can be composed, which means that you can create keywords by calling other keywords.

Robot Framework itself is not built to interact with a specific system and does not have a strictly defined list of functionalities. Instead, it is built to be easily extended to work with any possible target system. These extensions to the framework are called libraries.

The framework itself ships with standard libraries, used to handle common cases. There are many libraries available to cover a variety of situations and systems.

Robocorp maintains a set of libraries that are built specifically to support RPA cases. These libraries form the RPA Framework.

Keywords are the building blocks of robots, and libraries will get you more keywords, but a robot does nothing if you don't assign it a Task.

Automation Code to perform the validation of holiday list given in the intranet website:

TestCase File:

```
*** Settings ***
Library SeleniumLibrary
Library Collections
Resource ../../Common/testCase_resources.robot
Set Selenium Timeout 10 seconds
Suite Setup Open the browser
Suite TearDown Close Browser Session #Author:Dhruvi
Virani
```

```
*** Test Cases ***
Validate the Intranet Holiday List
  Go To Home And Then Holiday List
```

Fetch the Holidays list 2023 Pune

Fetch The Holidays List 2023 Ahmedabad

#Compare And Find The Difference Between Holiday List Of Both The Years

#\${diff_list}= List Difference \${HolidayList} \${HolidayList1}

Keywords File:

*** Settings ***

Library SeleniumLibrary

*** Keywords ***

Open the browser

Open Browser \${url} Chrome

Maximize Browser Window

Set Selenium Timeout 30 seconds

Wait Until Element Is Visible \${ElementVisible}

Go To Home And Then Holiday List

Wait Until Element Is Visible \${MouseHoverToHome}

Mouse Over \${MouseHoverToHome}

Wait Until Element Is Visible \${ClickOnHolidayList}

Click Element \${ClickOnHolidayList}

Fetch The Holidays List

[Arguments] \${year} \${location}

Click Element \${SelectHolidayYearDropdown}

Wait Until Element Is Visible \${SelectHolidayYearDropdown}

Select From List By Label \${SelectHolidayYearDropdown} \${year}

Click Element \${SelectMonths}

Click Button \${SelectSubmitButton}

@{ExpectedHolidayList} = Create List \${DescriptionOfHolidayList}

\${Elements} = Get WebElements \${DescriptionOfHolidayList}

Set Global Variable \${Elements}

@{HolidayList} = Create List

Set Global Variable @{HolidayList}

FOR \${Element} IN @{Elements}

Log \${Element.text}

Append To List \${HolidayList} \${Element.text}

END

\${Count}= Click Element \${ClickPagination}

\${LengthOfPagination}= Get Length \${ClickPagination}

```

FOR ${Count} IN RANGE ${LengthOfPagination} 1 -1
    Click Element [${Count}+1]
    #Continue For Loop If '${Count}' == '${LengthOfPagination}'
    Log ${Count}
END

```

```

FOR ${Element} IN @{Elements}
    Log ${Element.text} Append To List ${HolidayList}
${Element.text}
END
Log ${HolidayList}
[return] ${HolidayList}

```

Compare And Find The Difference Between Holiday List Of Both The Years

```

${list1}= Get Length ${Elements}
Log ${list1}
${list2}= Get Length ${Elements1}
Log ${list2}

```

List Difference

```

[Arguments] ${arg1} ${arg2}
@{New_list}= Create List @{arg1} @{arg2}
FOR ${item} IN @{arg1}
    Remove Values From List ${New_list} ${item}
END
Log ${New_list}
Get Length ${New_list}

```

Close Browser Session

Close Browser

Variables File:

```

*** Variables ***
${url} https://intranet.einfochips.com/
${ElementVisible} //a[text()='Dhruvi Virani']
${MouseHoverToHome} css=div[class='collapse navbar-collapse navbar-ex1-collapse']>
ul>li>a[class='dropdown-toggle nav-link'] ${ClickOnHolidayList} //a[text()='Holiday List']
${SelectHolidayYearDropdown} css=select[id='holiday_year']
${ClickYear} css=option[value='2022']
${SelectMonths} css=select[id='month_id']>option
${SelectSubmitButton} css=button[class='btn btn-success']
${DescriptionOfHolidayList} css=table[id='mt'] tbody td:nth-child(4)
${ClickPagination} //ul[@class='pagination pagination-lg']
${SelectYear} css=option[value='2023']

```

Resources File:

```
*** Settings ***
Resource  testCase_variable.robot
Resource  testCase_labels.robot
Resource  testCase_common_keywords.robot
Library   ExtendedLibrary.py
```

Automation Code to perform the validation of Practice Automation

Website:

TestCase File:

```
*** Settings ***
Resource      ../../Common/Testcase_resources.robot

*** Test Cases ***
Validate Practice Automation website
    Open the browser
    Validate Page Title
    Validate and click Home
    Click on the radio button
    Type the suggestion of a country name to get the options
    Select dropdown
    Click on the checkbox
    Validate Switch window example
    Validate Switch Tab example
    Validate Switch to alert example
    Scroll down and validate hide and show button
    Scroll down and validate mouse hover
    Close Browser Session
```

Keywords File:

```
*** Settings ***
Library      SeleniumLibrary

*** Keywords ***
Open the browser
    Open Browser  ${url}  Chrome
    Maximize Browser Window  sleep  2

Validate Page Title
    ${title}=  Get Title

Validate and click Home
    Click Element  ${validate_home}
    Sleep  5
    Go Back

Click on the radio button
    Click Element  ${click_radio_button}
    Sleep  2
```

Type the suggestion of a country name to get the options

```
Click Element ${suggestion_textbox}
Input Text ${suggestion_textbox} ind
Sleep 2
Click Element ${select_dropdown}
Sleep 1
```

Select dropdown

```
Click Element ${dropdown_example}
Sleep 2
Click Element ${select_option}
Sleep 1
```

Click on the checkbox

```
Click Element ${select_checkbox}
Sleep 2
```

Validate Switch window example

```
Click Element ${switch_window}
Sleep 7
Switch Window MAIN
Sleep 3
```

Validate Switch Tab example

```
Click Element ${switch_tab}
Sleep 12
Switch Window MAIN
Sleep 5
```

Validate Switch to alert example

```
Click Element ${switch_to_alert}
Input Text ${switch_to_alert} dhruvi
Sleep 2
Click Element ${alert_button}
Sleep 5
```

Scroll down and validate hide and show button

```
Scroll Element Into View ${scroll_to_element}
Click Element ${hide_button}
Sleep 1
Click Element ${show_button}
Sleep 2
```

Scroll down and validate mouse hover

```
Scroll Element Into View ${scroll_down}
Sleep 2
Mouse Over ${mouse_hover}
```

```
Sleep 2
Click Element ${hover_to_top}
Sleep 2
```

```
Close Browser Session
Close Browser
```

Variables File:

```
*** Variables ***
${Url}    https://rahulshettyacademy.com/AutomationPractice/
${validate_home}  //button[text()='Home']
${click_radio_button}  name:radioButton
${suggestion_textbox}  id:autocomplete
${select_dropdown}  //li/div[text()='India']
${dropdown_example}  id:dropdown-class-example
${select_option}  css:option[value='option3']
${select_checkbox}  name:checkBoxOption1
${switch_window}  id:openwindow
${switch_tab}  id:opentab
${switch_to_alert}  css:input[placeholder='Enter Your Name']
${alert_button}  id:alertbtn
${scroll_to_element}  css:div[class='totalAmount']
${hide_button}  id:hide-textbox
${show_button}  id:show-textbox
${scroll_down}  id:courses-iframe
${mouse_hover}  id:mousehover
${hover_to_top}  css:a[href='#top']
```

Resources File:

```
*** Settings ***
Library  SeleniumLibrary
Resource  Testcase_common_keywords.robot
Resource  Testcase_labels.robot
Resource  Testcase_variables.robot
Library  ExtendedLibrary.py
```

Chapter 9: GitHub

Assignment3: Create or delete a new branch in GIT, add files and commit and push your changes in branch.

1. Clone your GIT repository
2. Create a new branch in your repository
3. Add some files in it and commit and push your changes
4. Edit files and commit in same branch and push your changes
5. Delete your newly created branch

```
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (main)
$ git init
Initialized empty Git repository in C:/Users/158410/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment/.git/
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (main)
$ git status
On branch main
Changes to be staged for commit:
  (use "git add/rm <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    deleted:   test2.txt

no changes added to commit (use "git add" and/or "git commit -a")

158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (main)
$ git add .

158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (main)
$ git commit -m "commit3"
[main 361d018] commit3
1 file changed, 0 insertions(+), 0 deletions(-)
delete mode 100644 test2.txt

158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (main)
$ git remote -v
origin https://github.com/Dhrushi-EI/GitAssignment.git (fetch)
origin https://github.com/Dhrushi-EI/GitAssignment.git (push)
```

```
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (test)
$ git remote -vvv
dhrushi https://github.com/Dhrushi-EI/GitAssignment.git (fetch)
dhrushi https://github.com/Dhrushi-EI/GitAssignment.git (push)
origin https://github.com/Dhrushi-EI/GitAssignment.git (fetch)
origin https://github.com/Dhrushi-EI/GitAssignment.git (push)

158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (test)
$ git remote set-url origin git@github.com:Dhrushi-EI/GitAssignment.git

158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (test)
$ git remote update
Fetching origin
The authenticity of host 'github.com (20.207.73.82)' can't be established.
ED25519 key fingerprint is SHA256:+DiY3wvvV6TuJJhbpZisF/ZLDA0zPMsvHdkr4UvC0qU.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'github.com' (ED25519) to the list of known hosts.
Fetching dhrushi

158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (test)
$ git remote set-url dhrushi git@github.com:Dhrushi-EI/GitAssignment.git

158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (test)
$ git remote update
Fetching origin
Fetching dhrushi

158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (test)
$ git push dhrushi test
Enumerating objects: 13, done.
Counting objects: 100% (13/13), done.
Delta compression using up to 8 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (13/13), 1010 bytes | 1010.00 KiB/s, done.
Total 13 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), done.
To github.com:Dhrushi-EI/GitAssignment.git
 * [new branch]      test -> test
```

```

15841099NGYOV3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (test)
$ git status
On branch test
nothing to commit, working tree clean

15841099NGYOV3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (test)
$ git add test2.txt
[test 8738cb] commits
 1 file changed, 1 insertion(+)

15841099NGYOV3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (test)
$ git push dhrushi test
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 277 bytes | 277.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:Dhrushi-EI/GitAssignment.git
 835f000..8738cb test -> test

15841099NGYOV3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (test)
$ git branch
  feature
* main
  test

15841099NGYOV3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (test)
$ git branch --delete test
error: Cannot delete branch 'test' checked out at 'C:/Users/158410/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment'

15841099NGYOV3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (test)
$ git checkout main
Switched to branch 'main'

15841099NGYOV3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (main)
$ git branch --delete test
error: The branch 'test' is not fully merged.
If you are sure you want to delete it, run 'git branch -D test'.

15841099NGYOV3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (main)
$ git branch -D test
Deleted branch test (was 8738cb).

```

The screenshot shows a GitHub repository page for 'Dhrushi-EI / GitAssignment'. The repository has 1 branch and 0 tags. The 'Code' tab is selected, showing a commit history. The first commit is 'Dhrushi-EI commit5' (first commit) made 2 hours ago. The second commit is 'commit5' made 3 minutes ago. There is a note to 'Add a README'.

Assignment3: Pulling out changes that you are not supposed to push in git repository.

1. Create a new branch in your repository
2. Add some files <File1.txt> & <File2.txt> in it and commit and push your changes
3. Edit file <File1.txt> and add this file again in GIT as you have made some changes in it
4. Now as you don't want to make this changes so backing out all the changes in your working directory
5. Now your changes are unstaged, so checkout the file to discard any changes made in <File1.txt>
6. Edit files <File1.txt> & <File2.txt> again and commit in same branch and don't push your changes
7. Reset all the changes that you have not pushed yet and reset your branch with latest master branch

```
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (main)
$ git branch develop
Switched to branch 'develop'
```

```
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (develop)
$ touch File{1,2}.txt
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (develop)
$ git add File{1,2}.txt
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (develop)
$ git commit -m "commit6"
[develop 7a9e49c] commit6
 2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 File1.txt
 create mode 100644 File2.txt
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (develop)
$ git push dhrushi develop
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 242 bytes | 242.00 KiB/s, done.
Total 2 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'develop' on GitHub by visiting:
remote:   https://github.com/Dhrushi-EI/GitAssignment/pull/new/develop
remote:
To github.com:Dhrushi-EI/GitAssignment.git
 * [new branch]      develop -> develop
```

```
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (develop)
$ git add File1.txt
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (develop)
$ git status
On branch develop
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   File1.txt

158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (develop)
$ git commit -m "commit8"
[develop 8e5410d] commit8
 1 file changed, 1 insertion(+), 1 deletion(-)

158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (develop)
$ git push dhrushi develop
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 278 bytes | 278.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:Dhrushi-EI/GitAssignment.git
  9e3c8bc..8e5410d  develop -> develop
```

```
158410@9NGYOV3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (develop)
$ git reset --hard
HEAD is now at 8e5410d commit8
```

```
158410@9NGYOV3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (develop)
$ git reset --hard 8e541
HEAD is now at 8e5410d commit8
```

```
158410@9NGYOV3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (develop)
$ git status
On branch develop
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
    (use "git restore <file>..." to discard changes in working directory)
      modified:   File1.txt

no changes added to commit (use "git add" and/or "git commit -a")

158410@9NGYOV3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (develop)
$ git checkout develop -- File1.txt
```

```
158410@9NGYOV3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (develop)
$ git add File{1,2}.txt

158410@9NGYOV3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (develop)
$ git commit -m "commit9"
[develop 8197e05] commit9
 2 files changed, 2 insertions(+), 1 deletion(-)
```

```
$ git log
commit 8197e05e497216ee2c34ccb7a6084bdf81c2ad76 (HEAD -> develop)
Author: dhruvi <dhruvi.virani@einfochips.com>
Date:   Wed Mar 29 15:13:18 2023 +0530

  commit9

commit 8e5410d676e68d29aaaf9d9722ecb9d0d7d0d0543 (dhruvi/develop)
Author: dhruvi <dhruvi.virani@einfochips.com>
Date:   Wed Mar 29 14:57:41 2023 +0530

  commit8

commit 9e3c8bcd86dc485c637b1d049266344340a7428a
Author: dhruvi <dhruvi.virani@einfochips.com>
Date:   Wed Mar 29 14:55:44 2023 +0530

  commit7

commit 7a9e49c6a0e102dde41a1c39bee0bd042951b674
Author: dhruvi <dhruvi.virani@einfochips.com>
Date:   Wed Mar 29 14:47:52 2023 +0530

  commit6

commit 361d0185fedbae6ea63483ac0a500c93d42e0676 (main)
Author: dhruvi <dhruvi.virani@einfochips.com>
Date:   Wed Mar 29 13:15:13 2023 +0530

  commit3

commit 99dd0f40935005b9d3fb31c16d73d271a16b0411
Author: dhruvi <dhruvi.virani@einfochips.com>
Date:   Wed Mar 29 12:32:08 2023 +0530

  first commit

commit 2ad9d716566d739b8ac3c401bb5cd07f44054295
Author: dhruvi <dhruvi.virani@einfochips.com>
Date:   Wed Mar 29 12:12:07 2023 +0530

  my new commit

commit 6777e215dc6ffd985f0227471eef87fa3da8af13
Author: dhruvi <dhruvi.virani@einfochips.com>
Date:   Wed Mar 29 12:04:32 2023 +0530
```

```
158410@9NGY0V3 MINGW64 ~\OneDrive - Arrow Electronics, Inc\Desktop\practice_assignment (develop)
$ git reset --hard 361d0
HEAD is now at 361d018 commit3
```

Assignment4: Pulling out changes that you are not supposed to push in git repository.

1. Add new remote in GIT. You can use below repository as new remote
 - a. <https://github.com/ishabhatt/training.git>
2. Create new branch (Test_Branch_1) in your repository from ishabhatt's master branch
3. Add new files Test_GIT_Assi1.txt & Test_GIT_Assi2.txt in newly created branch
4. Commit your changes
5. Push your branch in your repository <YourRepo>
6. Create pull request to merge your changes in ishabhatt's repository

```
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/practice_assignment (develop)
$ git clone git@github.com:Dhruvi-EI/training.git
Cloning into 'training'...
remote: Enumerating objects: 12, done.
remote: Counting objects: 100% (12/12), done.
remote: Compressing objects: 100% (8/8), done.
receiving objects: 100% (12/12), done.
remote: Total 12 (delta 0), reused 7 (delta 0), pack-reused 0
```

```
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/training (main)
$ git branch Test_Branch_1
```

```
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/training (main)
$ touch Test_GIT_Assi{1,2}.txt
```

```
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/training (main)
$ git add Test_GIT_Assi{1,2}.txt
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/training (main)
$ git commit -m "new files added"
[main bf3df6b] new files added
 2 files changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 Test_GIT_Assi1.txt
 create mode 100644 Test_GIT_Assi2.txt
```

```
158410@9NGY0V3 MINGW64 ~/OneDrive - Arrow Electronics, Inc/Desktop/training (main)
$ git push origin Test_Branch_1
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 284 bytes | 284.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:Dhruvi-EI/training.git
 1391e4b..bf3bc48  Test_Branch_1 -> Test_Branch_1
```

Chapter 9: Full Automation of Websites using Robot Framework

9.1 Automation on PhpTravels Website:

→ Manual Test Cases Of Phptravels website

A	B	C	D	E	F
Testcase Number	Summary	Pre-condition	Description	Steps	Expected Result
1	Logo	Open Website: https://phptravels.net	Validate logo of the phptravels website	1. Open the https://phptravels.net website 2. Click On logo	The website should be opened It should redirect to the home page
2	Header	Open Website: https://phptravels.net	Validate Header Section and it's functionality of the phptravels website	1. Open the https://phptravels.net website 2. Click on hotels href in header section 3. Click on flights href in header section 4. Click on tours href in header section 5. Click on visa href in header section 6. Click on transfers href in header section 7. Click on Blogs href in header section 8. Click on offers href in header section	The website should be opened It should navigate to the page where heading is 'SEARCH FOR BEST HOTELS' page of h2 tag. It should navigate to the page where heading is 'SEARCH FOR BEST FLIGHTS' page of h2 tag. It should navigate to the page where heading is 'FIND THE BEST TOURS PACKAGES' page It should navigate to the page where heading is 'SUBMIT YOUR VISA TODAY!' page of h2 It should navigate to the page where heading is 'BOOK YOUR TRANSFER TODAY' page of h2 It should navigate to the page where heading is 'PHPTRAVELS BLOG' page of h2 tag. It should navigate to the page where heading is 'PHPTRAVELS OFFERS' page of h2 tag.
3	Ghandi/Company	Open Website: https://phptravels.net	Validate Ghandi Drop-down options in the header	1. Open the https://phptravels.net website 2. Mouse Hover on Ghandi drop-down in header section 3. Click on 'Ghandi' in the header section.	The website should be opened Drop-down options should be visible and the options should be About Us, Terms of Use, FAQ, How to Book and it should navigate to page where heading should be 'About Us' link. It should navigate to its respective page.
4					
5					
6					
7					
8					
9					
10					
11					
12					
13					
14					

				4. Click on 'About-us' drop-down option of Ghandi drop-down button	It should navigate to page where heading is 'About-Us' which is a h3 tag.	
15				5. Click on 'Terms of Use' drop-down option of Ghandi drop-down button	It should navigate to page where heading is 'Terms Of Use' which is a h3 tag.	
16				6. Click on 'FAQ' drop-down option of Ghandi drop-down button	It should navigate to page where heading is 'FAQ' which is a h3 tag.	
17				7. Click on 'How to Book' drop-down option of Ghandi drop-down button	It should navigate to page where heading is 'How To Book' which is a h3 tag.	
18	4	English Drop-down	Open Website: https://phptravels.net	Validate English drop-down button in the header section	1. Open the https://phptravels.net website	
19					2. Click on English Drop-down button in header section	
20					3. Verify drop-down options of languages	
21					By clicking on any of the language, it should navigate to respective language page.	
22		5		Validate USD drop-down button in the header section	1. Open the https://phptravels.net website	
23					2. Click on USD drop-down button in header section and verify drop-down options	
24	6	Account Drop-down	Open Website: https://phptravels.net	Validate Account drop-down button in the header section	1. Open the https://phptravels.net website	
25					2. Click on Account drop-down button	
26					3. Click on Customer login option and Agents login from the drop-down list of account button	
27					4. Click on Customer signup option and Agents signup from the drop-down list of account button	
28					5. Click on Suppliers login option from the drop-down list of account button	
29					6. Click on Suppliers login option from the drop-down list of account button	
30					The website should be opened	
31	7	Heading 'Let's book your next trip!'	Open Website: https://phptravels.net	Validate heading 'Let's book your next trip!'	1. Open the https://phptravels.net website	
					2. Validate heading	
					The heading should be h2 tag and it should be Let's book your next trip!'	

	8	Hotels in Tablist	Open Website: https://phptravels.net/	Validate Hotels section in Tablist and it's functionality of phptravels website	1. Open the https://phptravels.net website	The website should be opened
32					2. Click on 'Search by City' button.	'Please Enter 3 or more characters' text should be visible
33					3. Enter input	Drop-down options for entered input should be visible
34					4. Click on Placeholder where today's date is selected by default	It should open table of calender and by-default today's date is selected.
35					5. Select any Checkin date	It should open the table of calender for checkout date
36					6. Click on travellers box	It should open drop-down having options for rooms,
37					7. Select the nationality drop-down option where India is selected by default	It should open drop-down list of countries
38					8. Click on Search button having search image	It should navigate to page where options for particular chosen destination and dates
39						
	9	Flights in Tablist	Open Website: https://phptravels.net/	Validate Hotels section in Tablist and it's functionality of phptravels website	1. Open the https://phptravels.net website	The website should be opened
40					2. Click on 'Flying From' and enter the input	It should open the drop-down options of the entered input
41					3. Click on 'Destination' and enter the input	It should open the drop-down options of the entered input
42					4. Click on departure date box	It should open calender
43				Validate For One Way radio Button	5. Click on travellers box under passengers section	It should open the drop-down options that includes adults,
44					6. Click on Search button having search image	It should navigate to page where options for particular chosen destination and dates
45					7. Click on Swap arrow Button	It should swap Flying From and To Destination
46				Validate Round Trip radio button	1. Click on Round Way radio button	It will open Return way date box that opens calender while
47						
	10	Tours in Tablist	Open Website: https://phptravels.net/	Validate Tours section in Tablist and it's functionality of phptravels website	1. Open the https://phptravels.net website	The website should be opened
48					2. Click on 'Search by City' button under origin label	'Please Enter 3 or more characters' text should be visible
49					3. Enter input	Drop-down options for entered input should be visible
50					4. Click on date box under date label	It should open table of calender for the particular month
51					5. Click on travellers box	options that includes adults and child.
52					6. Click on Search button having search image	where options for particular chosen destination and dates hotels are visible.
53						
	11	Transfers in Tablist	Open Website: https://phptravels.net/	Validate Transfer section in Tablist and it's functionality of phptravels website	1. Open the https://phptravels.net website	The website should be opened
54					2. Click on 'Search by City' button under origin label	'Please Enter 3 or more characters' text should be visible
55					3. Click on 'Search by City' button under destination label	'Please Enter 3 or more characters' text should be visible
56						

				4. Click on box under 'From Date' 5. Click on box under 'To Date' 6. Click on travellers box 7. Click on Search button having search image	It should open table of calender and by-default today's date is selected. It should open table of calender and by-default It should open drop-down options that includes adults It should navigate to page where options for particular chosen destination and dates
57					
58					
59					
60					
61	12	Visa in Tablist	Open Website: https://phptravels.net/	Validate Visa section in Tablist and it's functionality of phptravels website	1. Open the https://phptravels.net website 2. Click on Select country drop-down box under 'From Country' label 3. Click on Select country drop-down box under 'From Country' label 4. Click on date box under 'date' label
62					The website should be opened It should open drop-down list of countries It should open drop-down list of countries calender and by-default today's date is selected.
63					
64	13	Heading 'Top Flight Destinations'	Open Website: https://phptravels.net/	Validate heading 'Top Flight Destinations'	1. Open the https://phptravels.net website 2. Validate heading
65					The website should be opened The heading should be h2 tag and it should be Top Flight Destinations
66	14	Heading 'Featured Hotels'	Open Website: https://phptravels.net/	Validate heading 'Featured Hotels'	1. Open the https://phptravels.net website 2. Validate heading
67					The website should be opened The heading should be h2 tag and it should be Featured Hotels
68	15	Heading 'Featured Tours'	Open Website: https://phptravels.net/	Validate heading 'Featured Tours'	1. Open the https://phptravels.net website 2. Validate heading 3. Click on View More tours Button
69					The website should be opened and it should be Featured Tours It should redirect to a page where heading is 'FIND THE BEST TOURS PACKAGES' which
70					
71	16	Heading 'Book Your Transfer Today'	Open Website: https://phptravels.net/	Validate heading 'Book Your Transfer Today'	1. Open the https://phptravels.net website 2. Validate heading
72					The website should be opened The heading should be h2 tag and it should be Book Your Transfer Today
73	17	Heading 'New Offers'	Open Website: https://phptravels.net/	Validate heading 'New Offers'	1. Open the https://phptravels.net website 2. Validate heading
74					The website should be opened The heading should be h2 tag and it should be New Offers
75	18	View Offers Button	Open Website: https://phptravels.net/	Validate Button 'View Offers'	1. Open the https://phptravels.net website 2. Click on the 'View More'
76					The website should be opened It should redirect to where the heading 'PHPTRAVELS Offers' is of h2 tag.
77	19	Heading 'Latest on blogs'	Open Website: https://phptravels.net/	Validate heading 'Latest on blogs'	1. Open the https://phptravels.net website 2. Validate heading
78					The website should be opened and it should be Latest on blogs
79	20	View More Button	Open Website: https://phptravels.net/	Validate 'View More' Button	1. Open the https://phptravels.net website 2. Click on 'View More' Button
80					The website should be opened It should navigate to 'PHPTRAVELS BLOG' page
81	21	'Need Help? Contact us' container	Open Website: https://phptravels.net/	Validate 'Need Help? Contact us' box	1. Open the https://phptravels.net website
82					The website should be opened

				2. Click on 'Need Help? Contact us' box in the container	It should redirect to 'Contact Us' page which is of h2 tag.
83	22	Subscribe to see secret Deals	Open Website: https://phptravels.net/	Validate 'Subscribe to see secret Deals'	1.Open the https://phptravels.net website The website should be opened
84					2. Validate heading
85	23	Enter email address	Open Website: https://phptravels.net/	Validate 'Enter email address' placeholder	1.Open the https://phptravels.net website The website should be opened 2.Validate 'Enter email address' box. Placeholder should be 'Enter your email'
86					
87	24	Footer logo	Open Website: https://phptravels.net/	Validate logo in the footer section	1.Open the https://phptravels.net website The website should be opened 2. Click On logo It should redirect to the home page from the top
88					
89					
90	25	Validate mail	Open Website: https://phptravels.net/	Validate mail in the footer section	1.Open the https://phptravels.net website The website should be opened 2. Validate the mail given under the footer logo The mail should be 'info@travelagency.com'
91	26	Contact Us	Open Website: https://phptravels.net/	Validate 'Contact Us' href under the footer logo	1.Open the https://phptravels.net website The website should be opened 2. Validate the 'Contact Us' given under the footer logo It should redirect to 'Contact Us' page which is of h2 tag.
92					
93	27	Company section	Open Website: https://phptravels.net/	Validate 'Company' section in the footer	1.Open the https://phptravels.net website The website should be opened 2. Click on the 'Company' heading which is a href It should redirect to 'Company' page which is of h3 3. Click on 'About Us' It should redirect to 'About Us' page which is of h3 tag. 4. Click on 'Terms Of Use' It should navigate to page where heading is 'Terms Of Use' which is a h3 tag. 5. Click on 'Cookies policy' It should navigate to page where heading is 'Cookies policy' which is a h3 tag. 6. Click on 'Privacy Policy' It should navigate to page where heading is 'Privacy Policy' which is a h3 tag.
94					
95					
96					
97					
98					
99	28	Support section	Open Website: https://phptravels.net/	Validate 'Support' section in the footer	1.Open the https://phptravels.net website The website should be opened 2. Click on the 'Support' heading which is a href It should redirect to 'Supprt' page which is of h3 tag. 3. Click on 'Become Supplier' It should navigate to page where heading is 'Privacy Policy' which is a h3 tag. 4. Click on 'FAQ' It should navigate to page where heading is 'FAQ' which 5. Click on 'Booking Tips' It should navigate to page where heading is 'Booking Tips' which is a h3 tag. 6. Click on 'How to Book' It should navigate to page where heading is 'How to
100					
101					
102					
103					
104					
105					
106	29	Services section	Open Website: https://phptravels.net/	Validate 'Services' section in the footer	1.Open the https://phptravels.net website The website should be opened 2. Click on the 'Services' heading which is a href It should redirect to 'Services' page which is of h3 tag. 3. Click on 'File A Claim' It should navigate to page where heading is 'File A Claim' which is a h3 tag. 4. Click on 'PHPTRAVELS OFFERS' It should navigate to page where heading is 'PHPTRAVELS OFFERS' which is a h3 tag.
107					
108					
109					

			Sign up needed			
110				5. Click on 'Add Your Business'	It should navigate to page where heading is 'Signup'	
111				3. Click on 'Careers And Jobs'	It should navigate to page where heading is 'Careers And Jobs' which is a h3 tag.	
112	30	Facebook icon	Open Website: https://phptravels.net/	Validate Facebook icon in the footer section	1. Open the https://phptravels.net website	The website should be opened
113					2. Click on the icon	phptravels account in the facebook
114	31	Twitter icon	Open Website: https://phptravels.net/	Validate Twitter icon in the footer section	1. Open the https://phptravels.net website	The website should be opened
115					2. Click on the icon	It should redirect to the phptravels account in the
116	32	Youtube icon	Open Website: https://phptravels.net/	Validate Youtube icon in the footer section	1. Open the https://phptravels.net website	The website should be opened
117					2. Click on the icon	It should redirect to the phptravels account in the
118	33	Whatsapp icon	Open Website: https://phptravels.net/	Validate Whatsapp icon in the footer section	1. Open the https://phptravels.net website	The website should be opened
119					2. Click on the icon	It should redirect to the phptravels account in the
120	34	Instagram logo	Open Website: https://phptravels.net/	Validate Instagram icon in the footer section	1. Open the https://phptravels.net website	The website should be opened
121					2. Click on the icon	It should redirect to the phptravels account in the
122	35	PHPTRAVELS href	Open Website: https://phptravels.net/	Validate PHPTRAVELS href in the footer section	1. Open the https://phptravels.net website	The website should be opened
123					2. Click on the href	phptravels 'Startup Your Online Travel Agency Today!' page where heading is h1 tag
124	36	Top-up Button	Open Website: https://phptravels.net/	Validate Top-up button	1. Open the https://phptravels.net website	The website should be opened
125					2. Click on the top-up button	It should navigate to top of the home page

Code :

TestCase File:

*** Settings ***

Resource/Common/testCase_resources.robot

#Suite Setup Open The browser

#Suite TearDown Close Browser Session

#Author: Dhruvi.Virani

*** Test Cases ***

Validate Phptravels Website

Open The Browser

Validate Logo Of The Phptravels Website

Validate Header Section
Validate Ghandi Drop-down Option In The Header section
Validate English dropdown In The Header Section
Validate USD Drop-down In The Header Section
Validate Account Drop-down In The Header section
Validate Heading
Validate Hotels In Tablist
Validate Flights In Tablist
Validate Tours In Tablist
Validate Transfers In Tablist
Validate Visa In Tablist
Validate Heading 'Top Flight Destinations'
Validate Heading 'Featured Hotels'
Validate Heading 'Featured Tours'
Validate Heading 'Book Your Transfer Today'
Validate Heading 'Latest on blogs'
Validate View More Button
Validate Need Help Box
Validate Subscribe To See Secret Deals Heading
Validate Enter Email Address Box
Validate Footer Logo
#Validate Footer Mail
Validate Footer Contact Us
Validate Footer Company Section
Validate Footer Support Section
Validate Footer Services Section
Validate Footer Facebook Icon
Validate Footer Twitter Icon
Validate Footer Youtube Icon
Validate Footer Whatsapp Icon
Validate Footer Instagram Icon
Validate Footer PHPTRAVELS href
Validate Top-Up Button
Close Browser Section

*** Keywords ***

Open The Browser
 Open Browser \${PhpUrl} Chrome
 Maximize Browser Window
 Wait Until Element Is Visible \${PhpMainHeading}

Validate Logo Of The Phptravels Website
 Wait Until Element Is Visible \${PhpLogo}

Get Title

Validate Header Section

```
Wait Until Element Is Visible ${Header}
Get Text ${Header}
Click Element ${HeaderHotels}
Wait Until Element Is Visible ${WaitHotelsHeading}
Go Back
Wait Until Element Is Visible ${PhpMainHeading}
Click Element ${HeaderFlights}
Wait Until Element Is Visible ${WaitFlightsHeading}
Go Back
Wait Until Element Is Visible ${PhpMainHeading}
Click Element ${HeaderTours}
Wait Until Element Is Visible ${WaitToursHeading}
Go Back
Wait Until Element Is Visible ${PhpMainHeading}
Click Element ${HeaderTransfer}
Wait Until Element Is Visible ${WaitTransferHeading}
Go Back
Wait Until Element Is Visible ${PhpMainHeading}
Click Element ${HeaderVisa}
Wait Until Element Is Visible ${WaitVisaHeading}
Go Back
Wait Until Element Is Visible ${PhpMainHeading}
Click Element ${HeaderBlogs}
Wait Until Element Is Visible ${WaitBlogHeading}
Go Back
Wait Until Element Is Visible ${PhpMainHeading}
```

Validate Ghandi Drop-down Option In The Header section

```
Mouse Over ${HeaderGhandi}
Wait Until Element Is Visible ${GhandiDrop-down}
Click Element ${HeaderGhandi}
Wait Until Element Is Visible ${GhandiHeading}
Go Back
Wait Until Element Is Visible ${PhpMainHeading}
Mouse Over ${HeaderGhandi}
Wait Until Element Is Visible ${GhandiDrop-down}
Click Element ${GhandiFAQ}
Go Back
Wait Until Element Is Visible ${PhpMainHeading}
Mouse Over ${HeaderGhandi}
```

Wait Until Element Is Visible \${GhandiDrop-down}
Click Element \${GhandiTermsOfUse}
Go Back
Wait Until Element Is Visible \${PhpMainHeading}
Mouse Over \${HeaderGhandi}
Wait Until Element Is Visible \${GhandiDrop-down}
Click Element \${GhandiHowToBook}
Go Back
Wait Until Element Is Visible \${PhpMainHeading}
Mouse Over \${HeaderGhandi}
Wait Until Element Is Visible \${GhandiDrop-down}
Click Element \${GhandiAboutUs}
Go Back
Wait Until Element Is Visible \${PhpMainHeading}

Validate English dropdown In The Header Section
Click Element \${HeaderEnglishSection}
Wait Until Element Is Visible \${EnglishLanguages}
Click Element \${EnglishTurkish}
Wait Until Element Is Visible \${PhpMainHeading}
Click Element \${HeaderEnglishSection}
Wait Until Element Is Visible \${EnglishLanguages}
Click Element \${EnglishLanguage}
Wait Until Element Is Visible \${PhpMainHeading}

Validate USD Drop-down In The Header Section
Click Element \${USDButton}
Wait Until Element Is Visible \${EnglishLanguages}
Click Element \${USDEuro}
Wait Until Element Is Visible \${PhpMainHeading}
Click Element \${USDButton}
Wait Until Element Is Visible \${EnglishLanguages}
Click Element \${USD}
Wait Until Element Is Visible \${PhpMainHeading}

Validate Account Drop-down In The Header section
Set Selenium Timeout 10 seconds
Click Element \${AccountButton}
Wait Until Element Is Visible \${EnglishLanguages}
Click Element \${CustomerLogin}
Wait Until Element Is Visible \${CustomerLoginHeading}
Go Back
Wait Until Element Is Visible \${PhpMainHeading}

```
Click Element ${AccountButton}
Wait Until Element Is Visible ${EnglishLanguages}
Click Element ${CustomerSignIn}
Wait Until Element Is Visible ${CustomerLoginHeading}
Go Back
Wait Until Element Is Visible ${PhpMainHeading}
Click Element ${AccountButton}
Wait Until Element Is Visible ${EnglishLanguages}
Click Element ${SupplierLogin}
Switch Window NEW
Wait Until Element Is Visible ${SupplierLoginHeading}
${GetTitle}= Get Text ${SupplierLoginHeading}
Log ${GetTitle}
Switch Window MAIN
```

Validate Heading

```
Wait Until Element Is Visible ${PhpMainHeading}
Element Text Should Be ${PhpMainHeading} Let's book your next trip!
${title}= Get Text ${PhpMainHeading}
Log ${title}
```

Validate Hotels In Tablist

```
Sleep 1
Element Text Should Be ${BodyHotelsText} Hotels
Click Element ${ClickHotels}
Input Text ${InputHotels} Ahmedabad
Wait Until Element Is Visible ${HotelsFlagIn}
Click Element ${HotelsFlagIn}
Click Element ${HotelsTravellers}
Click Element ${HotelsSearch}
Wait Until Element Is Visible ${HotelsAfterSearchPage}
Go Back
Wait Until Element Is Visible ${PhpMainHeading}
```

Validate Flights In Tablist

```
Element Text Should Be ${BodyFlightsText} Flights
Click Element ${ClickFlights}
Click Element ${FlightsRoundTrip}
Click Element ${FlightsInputFlyingFrom}
Input Text ${FlightsInputFlyingFrom} Ahmedabad
Wait Until Element Is Visible ${FlightsSourceWaitElementVisible}
Click Element ${FlightsSourceWaitElementVisible}
Click Element ${FlightsInputDestination}
```

```
Input Text ${FlightsInputDestination} Dubai
Wait Until Element Is Visible ${FlightsDestinationWaitElementVisible}
Click Element ${FlightsDestinationWaitElementVisible}
Click Element ${FlightsSwapButton}
Click Element ${FlightsTravellers}
Click Element ${FlightsSubmit}
Wait Until Element Is Visible ${FlightsAfterSearch}
Go Back
Wait Until Element Is Visible ${PhpMainHeading}
```

Validate Tours In Tablist

```
Set Selenium Timeout 10 seconds
Element Text Should Be ${BodyToursText} Tours
Click Element ${ClickTours}
Click Element ${ToursSelectCountry}
Input Text ${ToursInput} Ahmedabad
Wait Until Element Is Visible ${HotelsFlagIn}
Click Element ${HotelsFlagIn}
Click Element ${ToursSubmitButton}
Wait Until Element Is Visible ${ToursWaitAfterSubmit}
Go Back
Wait Until Element Is Visible ${PhpMainHeading}
```

Validate Transfers In Tablist

```
Element Text Should Be ${BodyTransferText} Transfers
Click Element ${ClickTransfer}
Click Element ${TransferSelectCountry}
Input Text ${TransferInput} Ahmedabad
Wait Until Element Is Visible ${HotelsFlagIn}
Click Element ${HotelsFlagIn}
Click Element ${TransferClickAfterInput}
Input Text ${TransferInputTextDestination} Dubai
Wait Until Element Is Visible ${TransferWaitUntilDestinationIsVisible}
Click Element ${TransferWaitUntilDestinationIsVisible}
Click Element ${TransferSubmit}
Wait Until Element Is Visible ${TransferWaitAfterSubmit}
Go Back
Wait Until Element Is Visible ${PhpMainHeading}
```

Validate Visa In Tablist

```
Element Text Should Be ${BodyVisaText} Visa
Click Element ${ClickVisa}
execute javascript window.scrollTo(0,300)
```

```
Wait Until Element Is Visible ${VisaScroll}
Click Element ${VisaSelectCountrySource}
Input Text ${VisaInputSource} India
Wait Until Element Is Visible //li[text()='India']
Click Element ${VisaInputText}
Click Element ${VisaSelectCountryDestination}
Input Text ${VisaInputDestination} American Samoa
Wait Until Element Is Visible ${VisaInputTextDest}
Click Element ${VisaInputTextDest}
Click Element ${VisaSubmitButton}
Wait Until Element Is Visible ${VisaWaitAfterSubmit}
Go Back
Wait Until Element Is Visible ${PhpMainHeading}
```

Validate Heading 'Top Flight Destinations'

```
Wait Until Element Is Visible ${ValidateHeading1}
Element Text Should Be ${TopFlightDestinations} Top Flight Destinations
${Heading1}= Get Text ${ValidateHeading1}
Log ${Heading1}
```

Validate Heading 'Featured Hotels'

```
Wait Until Element Is Visible ${ValidateHeading2}
Element Text Should Be ${FeaturedHotels} Featured Hotels
${Heading2}= Get Text ${ValidateHeading2}
Log ${Heading2}
```

Validate Heading 'Featured Tours'

```
Wait Until Element Is Visible //h2[@class='sec__title_left']
Element Text Should Be css=h2[class='sec__title_left'] Featured Tours
${Heading3}= Get Text //h2[@class='sec__title_left']
Log ${Heading3}
```

Validate Heading 'Book Your Transfer Today'

```
Wait Until Element Is Visible ${ValidateHeading3}
Element Text Should Be ${BookYourTransferToday} Book Your Transfer Today
${Heading4}= Get Text ${ValidateHeading3}
Log ${Heading4}
```

Validate Heading 'Latest on blogs'

```
Wait Until Element Is Visible ${ValidateHeading4}
Element Text Should Be ${LatestOnBlogs} Latest on blogs
${Heading5}= Get Text ${ValidateHeading4}
Log ${Heading5}
```

Validate View More Button
execute javascript window.scrollTo(0,4000)
Sleep 3

Click Element \${ViewButton}
Wait Until Element Is Visible \${AfterViewButton}
Go Back
Wait Until Element Is Visible \${ViewButton}

Validate Need Help Box
execute javascript window.scrollTo(0,4250)

Sleep 3
Click Element \${NeedHelpBox}
Wait Until Element Is Visible \${AfterViewButton}
Go Back
Wait Until Element Is Visible \${PhpMainHeading}

Validate Subscribe To See Secret Deals Heading

execute javascript window.scrollTo(0,4500)
Sleep 1
Element Text Should Be \${SubscribeHeading} Subscribe to see Secret Deals
Wait Until Element Is Visible \${SubscribeHeading}
\${SectionHeading}= Get Text \${SubscribeHeading}
Log \${SectionHeading}

Validate Enter Email Address Box

execute javascript window.scrollTo(0,4500)
Sleep 1
Element Text Should Be \${FooterEmailText} Enter email addresss
\${email}= Get Text \${FooterEmail}
Log \${email}
Element Text Should Be \${SubscribeInput} Subscribe to see Secret Deals
Input Text \${FooterEmail} abc@gmail.com
Click Element \${SubscribeButton}

Validate Footer Logo

execute javascript window.scrollTo(0,4600)
Sleep 1
Click Element \${FooterLogo}
Sleep 2

Validate Footer Mail

Switch Window MAIN

```
execute javascript window.scrollTo(0,4700)
Sleep 1
#Wait Until Element Is Visible css=a[href='mailto:info@travelagency.com']
Click Element //a[@href='mailto:info@travelagency.com']
Sleep 8
Close Window
```

Validate Footer Contact Us

```
Switch Window MAIN
execute javascript window.scrollTo(0,4800)
Sleep 1
Click Element ${FooterContactUs}
Wait Until Element Is Visible ${WaitAfterContactUs}
Go Back
Sleep 2
```

Validate Footer Company Section

```
execute javascript window.scrollTo(0,4800)
Sleep 1
Click Element ${FooterCompany}
Wait Until Element Is Visible ${CompanyHeadings}
Go Back
Sleep 1
Click Element ${FooterAboutUs}
Wait Until Element Is Visible ${CompanyHeadings}
Go Back
Sleep 1
Click Element ${FooterTermsOfUse}
Wait Until Element Is Visible ${CompanyHeadings}
Go Back
Sleep 1
Click Element ${FooterCookiesPolicy}
Wait Until Element Is Visible ${CompanyHeadings}
Go Back
Sleep 1
Click Element ${FooterPrivacyPolicy}
Wait Until Element Is Visible ${CompanyHeadings}
Go Back
Sleep 1
```

Validate Footer Support Section

```
execute javascript window.scrollTo(0,4800)
Sleep 1
```

```
Click Element ${FooterSupport}
Wait Until Element Is Visible ${CompanyHeadings}
Go Back
Sleep 1
Click Element ${Footersupplier}
Wait Until Element Is Visible ${CompanyHeadings}
Go Back
Sleep 1
Click Element ${FooterFAQ}
Wait Until Element Is Visible ${CompanyHeadings}
Go Back
Sleep 1
Click Element ${FooterBookingTips}
Wait Until Element Is Visible ${CompanyHeadings}
Go Back
Sleep 1
Click Element ${FooterHowToBook}
Wait Until Element Is Visible ${CompanyHeadings}
Go Back
Sleep 1
```

```
Validate Footer Services Section
execute javascript window.scrollTo(0,4800)
Sleep 1
Click Element css=a[href='services']
Wait Until Element Is Visible ${CompanyHeadings}
Go Back
Switch Window Main
execute javascript window.scrollTo(0,4850)
Sleep 1
Wait Until Element Is Visible ${FooterFileAClaim}
Click Element ${FooterFileAClaim}
Wait Until Element Is Visible ${CompanyHeadings}
Go Back
Switch Window MAIN
execute javascript window.scrollTo(0,4850)
Sleep 1
Wait Until Element Is Visible ${FooterOffers}
Click Element ${FooterOffers}
Wait Until Element Is Visible ${OfferHeading}
Go Back
Switch Window MAIN
execute javascript window.scrollTo(0,4850)
```

```
Sleep 1
Wait Until Element Is Visible ${SignupSupplier}
Click Element ${SignupSupplier}
Wait Until Element Is Visible ${SupplierSignupHeading}
Go Back
Switch Window MAIN
execute javascript window.scrollTo(0,4850)
Sleep 1
Wait Until Element Is Visible ${FooterCareers}
Click Element ${FooterCareers}
Wait Until Element Is Visible ${CompanyHeadings}
Go Back
Sleep 1
```

```
Validate Footer Facebook Icon
execute javascript window.scrollTo(0,5000)
Sleep 1
Click Element ${FooterFacebook}
Switch Window NEW
Wait Until Element Is Visible ${FacebookWait}
Click Element ${FacebookWait}
Wait Until Element Is Visible ${BackFromFacebook}
Close Window
```

```
Validate Footer Twitter Icon
Switch Window MAIN
execute javascript window.scrollTo(0,5000)
Sleep 1
Click Element ${FooterTwitter}
Switch Window NEW
Wait Until Element Is Visible ${TwitterWait}
Close Window
```

```
Validate Footer Youtube Icon
Switch Window MAIN
execute javascript window.scrollTo(0,5000)
Sleep 1
Click Element ${FooterYoutbe}
Switch Window NEW
Wait Until Element Is Visible ${YoutubeWait}
Close Window
```

```
Validate Footer Whatsapp Icon
```

```
Switch Window MAIN
execute javascript window.scrollTo(0,5000)
Sleep 1
Click Element ${FooterWhatsapp}
Switch Window NEW
Wait Until Element Is Visible ${WhatsappWait}
Close Window
```

```
Validate Footer Instagram Icon
Switch Window MAIN
execute javascript window.scrollTo(0,5000)
Sleep 1
Click Element ${FooterInstagram}
Switch Window NEW
Wait Until Element Is Visible ${InstagramWait}
Close Window
```

```
Validate Footer PHPTRAVELS href
Switch Window MAIN
Click Element ${FooterPHPtravels}
execute javascript window.scrollTo(0,5000)
Sleep 1
Click Element ${PhptravelsPage}
Switch Window NEW
Wait Until Element Is Visible ${PhpWait}
Close Window
```

```
Validate Top-Up Button
Switch Window MAIN
execute javascript window.scrollTo(0,5000)
Sleep 1
Click Element ${Top-Up}
Wait Until Element Is Visible ${PhpMainHeading}
```

```
Close Browser Section
Close Browser
```

Variables locators file:

*** Variables ***

```
 ${PhpUrl} https://phptravels.net/
 ${PhpMainHeading} css=h2[class='pb-1']
 ${PhpLogo} css=div[class='logo']
 ${Header} css=div[class='w-100']
 ${HeaderHotels} css=a[class='active_hotels waves-effect']
 ${WaitHotelsHeading} //h2[text()='SEARCH FOR BEST HOTELS']
 ${HeaderFlights} css=a[class='active_flights waves-effect']
 ${WaitFlightsHeading} //h2[text()='SEARCH FOR BEST FLIGHTS']
 ${HeaderTours} css=a[class='active_tours waves-effect']
 ${WaitToursHeading} //h2[text()='Find the best tours packages']
 ${HeaderTransfer} css=a[class='active_cars waves-effect']
 ${WaitTransferHeading} (//h2[text()='Book Your Transfer Today'])[1]
 ${HeaderVisa} css=a[class='active_visa waves-effect']
 ${WaitVisaHeading} //h2[text()='Submit Your Visa Today!']
 ${HeaderBlogs} css=a[class='active_blog waves-effect']
 ${WaitBlogHeading} //h2[text()='PHPTRAVELS Blog']
 ${HeaderGhandi} (//li[@class='footm'])[1]
 ${GhandiDrop-down} (//ul[@class='dropdown-menu-item'])[1]
 ${GhandiHeading} css=h3[class='title stroke-shape']
 ${GhandiFAQ} (//a[text()='FAQ'])[1]
 ${GhandiTermsOfUse} (//a[text()='Terms of Use'])[1]
 ${GhandiHowToBook} (//a[text()='How to Book'])[1]
 ${GhandiAboutUs} (//a[text()='About Us'])[1]
 ${HeaderEnglishSection} (//button[@class='btn btn-default dropdown-toggle waves-effect'])[1]
 ${EnglishLanguages} css=ul[class='dropdown-menu show']
 ${EnglishTurkish} //a[text()=' Turkish']
 ${EnglishLanguage} //a[text()=' English']
 ${USDButton} //button[@id='currency']
 ${USDEuro} //a[text()=' EUR']
 ${USD} //a[text()=' USD']
 ${AccountButton} //button[@id='ACCOUNT']
 ${CustomerLogin} //a[text()='Customer Login']
 ${CustomerLoginHeading} css=h5[class='modal-title title']
 ${CustomerSignIn} //a[text()='Customer Signup']
 ${SupplierLogin} css=a[href='https://phptravels.net/api/supplier']
 ${SupplierLoginHeading} css=h1[class='display-5 mb-0']
 ${BodyHotelsText} //span[@class='text']
 ${ClickHotels} //span[@id='select2-hotels_city-container']
 ${InputHotels} css=input[class='select2-search__field']
 ${HotelsFlagIn} css=i[class='flag in']
```

```
 ${HotelsTravellers}  (/a[@class='dropdown-toggle dropdown-btn travellers waves-effect'])[1]
 ${HotelsSearch}   (/div[@class='btn-search text-center'])[1]
 ${HotelsAfterSearchPage}  css=a[class='mt-5 mb-2 btn btn-primary text-center loading_effect waves-effect']
 ${BodyFlightsText}  (/span[@class='text'])[2]
 ${ClickFlights}   (/span[@class='d-none d-lg-block d-xl-block'])[2]
 ${FlightsRoundTrip} //input[@id='round-trip']
 ${FlightsInputFlyingFrom}  (/input[@placeholder='Flying From'])[1]
 ${FlightsSourceWaitElementVisible} //div[text()='Ahmedabad, India']
 ${FlightsInputDestination}  (/input[@placeholder='To Destination'])[1]
 ${FlightsDestinationWaitElementVisible} //strong[text()=' Dubai Intl']
 ${FlightsSwapButton}  css=div[class='swap-places waves-effect shadow']
 ${FlightsTravellers}  (/p[text()='Travellers '])[2]
 ${FlightsSubmit}  css=button[id='flights-search']
 ${FlightsAfterSearch}  css=div[class='card-header']
 ${BodyToursText}  (/span[@class='text'])[3]
 ${ClickTours}   (/span[@class='d-none d-lg-block d-xl-block'])[3]
 ${ToursSelectCountry}  //span[@id='select2-tours_city-container']
 ${ToursInput}  css=input[class='select2-search__field']
 ${ToursSubmitButton}  (/button[@id='submit'])[2]
 ${ToursWaitAfterSubmit}  css=a[class='mt-5 mb-2 btn btn-primary text-center loading_effect waves-effect']
 ${BodyTransferText}  (/span[@class='text'])[4]
 ${ClickTransfer}   (/span[@class='d-none d-lg-block d-xl-block'])[4]
 ${TransferSelectCountry}  css=span[id='select2-carfrom-container']
 ${TransferInput}  css=input[class='select2-search__field']
 ${TransferClickAfterInput}  css=span[id='select2-carto-container']
 ${TransferInputTextDestination}  css=input[aria-controls='select2-carto-results']
 ${TransferWaitUntilDestinationIsVisible}  css=li[class='select2-results__option select2-results__option--highlighted']
 ${TransferSubmit}  (/button[@id='submit'])[3]
 ${TransferWaitAfterSubmit} //strong[text()='Back To Search']
 ${BodyVisaText}  (/span[@class='text'])[5]
 ${ClickVisa}  //span[text()='visa']
 ${VisaScroll}  css=i[class='la la-bullhorn']
 ${VisaSelectCountrySource}  (/span[text()='Select Country'])[1]
 ${VisaInputSource}  css=input[class='select2-search__field']
 ${VisaInputText}  //li[text()='India']
 ${VisaSelectCountryDestination}  (/span[text()='Select Country'])
 ${VisaInputDestination}  css=input[class='select2-search__field']
 ${VisaInputTextDest}  //li[text()='American Samoa']
 ${VisaSubmitButton}  (/button[@id='submit'])[4]
 ${VisaWaitAfterSubmit}  css=h2[class='sec__title_list text-center my-5']
```

`\${ValidateHeading1} //h2[text()='Top Flight Destinations']
`\${TopFlightDestinations} (//h2[@class='sec__title line-height-55'])[1]
`\${ValidateHeading2} //h2[text()='Featured Hotels']
`\${FeaturedHotels} css=h2[class='sec__title_left']
`\${ValidateHeading3} //h2[text()='Book Your Transfer Today']
`\${BookYourTransferToday} (//h2[@class='sec__title line-height-55'])[3]
`\${ValidateHeading4} //h2[text()='Latest on blogs']
`\${LatestOnBlogs} (//h2[@class='sec__title line-height-55'])[4]
`\${ViewButton} css=a[class='theme-btn waves-effect']
`\${AfterViewButton} css=h2[class='sec__title text-white']
`\${NeedHelpBox} css=i[class='la la-phone']
`\${SubscribeHeading} css=h2[class='sec__title font-size-30 text-white']
`\${FooterEmailText} css=label[class='label-text text-white']
`\${FooterEmail} css=input[placeholder='Enter your email']
`\${SubscribeInput} css=h2[class='sec__title font-size-30 text-white']
`\${SubscribeButton} //button[text()='Subscribe']
`\${FooterLogo} (//a[@href='https://phptravels.net/'])[2]
`\${FooterContactUs} //strong[text()='Contact Us']
`\${WaitAfterContactUs} css=h2[class='sec__title text-white']
`\${FooterCompany} css=a[href='company']
`\${CompanyHeadings} css=h3[class='title stroke-shape']
`\${FooterAboutUs} (//a[text()='About Us'])[2]
`\${FooterTermsOfUse} (//a[text()='Terms of Use'])[2]
`\${FooterCookiesPolicy} //a[text()='Cookies policy']
`\${FooterPrivacyPolicy} //a[text()='Privacy Policy']
`\${FooterSupport} css=a[href='supprt']
`\${Footersupplier} (//a[text()='Become Supplier'])[1]
`\${FooterFAQ} (//a[text()='FAQ'])[2]
`\${FooterBookingTips} (//a[text()='Booking Tips'])[1]
`\${FooterHowToBook} (//a[text()='How to Book'])[2]
`\${FooterService} css=a[href='services']
`\${FooterFileAClaim} //a[text()='File a claim']
`\${FooterOffers} (//a[@href='https://phptravels.net/offers'])[3]
`\${OfferHeading} css=h2[class='sec__title text-white']
`\${SignupSupplier} (//a[@href='https://phptravels.net/signup-supplier'])[2]
`\${SupplierSignupHeading} css=h5[class='modal-title title']
`\${FooterCareers} (//a[@href='https://phptravels.net/careers'])
`\${FooterFacebook} css=i[class='lab la-facebook']
`\${FacebookWait} css=div[class='x92rtbv x10l6tqk x1tk7jg1 x1vjfegm']
`\${BackFromFacebook} css=h1[class='x1heor9g x1qlqyl8 x1pd3egz x1a2a7pz']
`\${FooterTwitter} css=i[class='lab la-twitter']
`\${TwitterWait} (//span[@class='css-901oao css-16my406 r-poiln3 r-bcqeeo r-qvutc0'])[5]
`\${FooterYoutbe} css=i[class='lab la-youtube']

```

${YoutubeWait}  (/img[@class='style-scope yt-img-shadow'])[2]
${FooterWhatsapp}  css=i[class='lab la-whatsapp']
${WhatsappWait}  css=a[id='action-button']
${FooterInstagram}  css=i[class='lab la-instagram']
${InstagramWait}  css=div[class='_aagx']
${FooterPHPtravels}  css=button[id='cookie_stop']
${PhptravelsPage}  //a[@href='http://www.phptravels.com']
${PhpWait}  css=h1[class='hero-text-1']
${Top-Up}  css=i[class='la la-angle-up']

```

9.2 Automation on Make My Trip Website

→ Manual Testcases Of Make My Trip Website

	A	B	C	D	E	F
Testcase Number	Summary	Pre-condition	Description	Steps	Expected Result	
1	Logo	https://www.makemytrip.com/	Validate logo of the makemytrip.com website	1. Open the website https://www.makemytrip.com 2. Click On logo	The website should be opened It should redirect to the home page.	
2	Flights tab	https://www.makemytrip.com/	Validate Flights in Header Section and it's functionality of the makemytrip.com website	1. Open the website https://www.makemytrip.com 2. Click on round trip radio button in Flights section of home page. 3. Click on 'From' box. 4. Click on 'To' box. 5. Click on 'Departure' box. 6. Click on 'Travellers' box. 7. Click on 'Search' box. 8. Click on 'Multi City' radio button in Flights section of home page.. 9. Click on 'Add Another City' button. 10. Mouse hover on 'Armed Forces Fares' Button. 11. Mouse hover on 'Student Fares' Button.	The website should be opened It should display return date section. It should open a drop-down of popular cities. It should open a drop-down of popular cities. It will open a calendar and selecting departure date will automatic select return date. It will open a drop-down with options adults, children, infants and choose travellers class. It will redirect to available flights page. It will display options to select another city/cities. It will ask to select another city. It should display information about 'Armed Forces Fares' in the form of paragraph. It should display information about 'Student Fares' in the form of paragraph.	
3						
4						
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						

15				12. Mouse hover on 'Senior citizen Fares' Button.	It should display information about 'Senior citizen Fares' in the form of paragraph.
16				13. Mouse hover on 'Doctors & Nurses' Button.	It should display information about 'Doctors & Nurses' in the form of paragraph.
17				14. Click on options given in trending searches.	It will redirect to the page of the available flights for particular search.
3	Hotels tab	https://www.makemytrip.com/	Validate in Hotels Header Section and it's functionality of the makemytrip.com website	1. Open the website https://www.makemytrip.com	The website should be opened
18				2. Click on hotels tab.	It will open a container having 'Book Domestic and International Property Online. To list your property click here' heading.
19				2. Click on 'City, Property name or Location'	It will drop-down of suggestions.
20				3. Click on 'Check-in' box.	It will open a calender and by selecting check-in date will automatic open the option to select check-out date.
21				4. Click on 'Rooms and Guests' box.	It will open drop-down with options rooms, adults and children.
22				5. Click on 'Price per night' box.	It will open a drop-down with options adults, children, infants and choose travellers class.
23				6. Click on 'Search' button.	It will redirect to available hotels page.
24				7. Click on 'Group Deals' radio button.	It will remove the 'prices per night' box option.
25				8. Click on 'Click Here'	It will redirect to page where 'LIST YOUR HOTEL' button is present.
26				9. "Offers" section"	h2 tag having text "Offers
27				10. "Download" Section	h3 tag should be "Download the app now!"
28					
4	Homestays tab	https://www.makemytrip.com/	Validate in Home Stays Header Section and it's functionality of the makemytrip.com website	1. Open the website https://www.makemytrip.com	The website should be opened
29				2. Click on Homestays tab.	It will open a container having 'Book your ideal Homestay - Villas, Apartments & more.'
30				2. Click on 'City, Property name or Location'	It will drop-down of suggestions.
31				3. Click on 'Add Check-in date' box.	It will open a calender and by selecting check-in date will automatic open the option to select check-out date.
32					

				4. Click on 'Guests'	It will open a drop-down with options of children and adults.
				5. Click on 'Travelling For'	It will open a drop-down having options work and leisure.
				6. Click on 'Search' button.	It will redirect to available Home Stays page.
				7. 'Find Your Wonderland' section	It will display some homestays options available.
				8. "Offers" section"	h2 tag having text "Offers
33					
34					
35					
36					
37					
38					
39					
40					
41					
42					
43					
44					
45					
46				1. Open the website https://www.makemytrip.com	The website should be opened
47				2. Click on Holiday Packages tab.	It will open a container to search for the destinations.
48				3. Click on 'From City' box.	It will open a drop-down of top searches .
49				4. Click on 'To City' box.	It will open a drop-down of top searches .
50				5. Click on 'Departure date'.	It will open a calender and by selecting check-in date will automatic open the option to select check-out date.
				6. Click on 'Guests'.	It will open a drop-down having options of adults and children.
				7. Click on 'Filters'.	It will open a drop-down to choose duration of the trip.
				8. Click on 'Search' button.	It will redirect to available Holiday packages page.
				7. Click on 'Destinations'.	It will open the container showing different destinations.
				8. Click on any destination.	It will navigate to a page of that particular destination.
				9. Click on 'Emerging'.	It will shows emerging destinations.
				10. Click on 'Honeymoon' tab.	It will show destinations and by clicking on any of them will navigate to particular destination page.
				11. Click on 'Featured' tab.	It will open the container showing 'Explore By Theme' and 'Best Selling Holidays' options.

				12. Click on 'Explore By theme'.	It will show some options to select such as luxury, unique stays, adventure, etc.
51				13. Click on 'Best-Selling Holidays'.	It will show destinations and by clicking on any of them will navigate to particular destination page.
52				14. Click on 'Travel Guidelines'.	It will show domestic travel guidelines and international travel guidelines.
53				15. Click on 'Domestic'.	It will open 'India Travel Guidelines' page.
54				16. Click on 'International'.	It will open 'COVID-19 certification' page.
55				17.Price Drop section	h2 tag contains text "Biggest Price Drop!"
56				18.Jackpot deals section	h2 tag text "Jackpot Deals on Top Selling Packages"
57				19.International destination section	h2 tag will be "International Destinations!"
58				20.Explore the unexplore section	h2 tag text will be "International Destinations!"
59				21."Offers" section	h2 tag having text "Offers"
60				22."Cities" section	It redirects to the packages of the respective cities
61	6 Trains tab	https://www.makemytrip.com/	Validate in Trains Header Section and it's functionality of the makemytrip.com website	1. Open the website https://www.makemytrip.com	The website should be opened
62				2. Click on 'Trains'.	It will open a container to select from and destination cities.
63				3. Click on 'From' box.	It will open a drop-down showing popular cities.
64				4. Click on 'To' box.	It will open a drop-down showing popular cities.
65				5. Click on 'Travel Date'.	It will open a calendar and by selecting check-in date will automatically open the option to select check-out date.
66					

					6. Click on 'Class'. 7. Click on 'Search' button.	It will display drop-down of Class such as sleeper class, Third AC, Second AC, etc. It will redirect to available Trains page.
67					8. "Offers" section"	h2 tag having text "Offers
68					9. Click on 'Check PNR Status'	It will display a box where 10 digit PNR number has to be entered.
69					10. Click on 'Live Train Status'.	It will ask to enter train number, your stop and train start date.
70						
71						
72	7	Buses tab	https://www.makemytrip.com/	Validate in Buses Header Section and it's functionality of the makemytrip.com website	1. Open the website https://www.makemytrip.com	The website should be opened
73					2. Click on 'Buses'.	It will open a container to select from and To cities.
74					3. Click on 'From' box.	It will open a drop-down showing popular cities.
75					4. Click on 'To' box.	It will open a drop-down showing popular cities.
76					5. Click on 'Travel Date'.	It will open a calendar and by selecting check-in date will automatically open the option to select check-out date.
77					6. Click on 'Search' button.	It will redirect to available Buses page.
78					7. Click on 'Buses from destination To'.	It will redirect to particular available buses.
79	8	Validate Cabs tab	Cabs tab should be open	Verify Cabs tab And its functionalities	1.Validate Cabs tab	Text should be "Cabs"
80					2.Click on Cabs tab	a tag having text as "Cab booking"
81					3.Click on "Outstation Round Trip"	"Drop time" option should be available
82					4.Click on "From" city	Able to take a input for city
83					5.Click on "To" city	Able to take a input for city
84					6.Click on "Departure date"	Calendar should be open to select the date
85					7.Click on "Return date"	Calendar should be open to select the date
86					8.Click on "pickup time"	Dropdown will be open and able to change the pick up time
87	9	Validate Forex tab	Forex tab should be open	Verify Forex tab And its functionalities	1.Validate Forex tab	Text should be "Forex"
88					2.Click on Forex tab	Forex currency page will be open

				3. Validate the page	p tag text should be "Global INR Card"
89				4.Click "Get Started"	Page will be open in same tab for Forex buying
90				5."Offers" section	h2 tag having text "Offers"
91				6."Download" Section	h3 tag should be "Download the app now!"
92					
93	10	Validate Charter Flights tab	Charter Flights tab should be open	Verify Charter Flights and its functionalities	1.Validate charter Flights tab Text should be "Charter Flights" 2.Click on charter Flights tab h1 tag text should be "Book Charter Planes in India" 3.Check the radio button for "Round Trip" return date option should be available 4.Click on "From" city Able to take a input for city 5.Click on "To" city Able to take a input for city 6.Click on "Departure date" Calendar should be open to select the date 7.Click on "Return date" Calendar should be open to select the date 8.Click on "Passengers" Able to change the passengers count 9.Click on "Search" button It should redirects to the page for the related search information
94					
95	11	Validate Activities tab	Activities tab should be open	Verify Activities tab and its functionalities	1.Validate tab ActivitiesText should be "Activities" 2.Click on Activities tab Page should be open having Activities search box 3.Click on search box Able to take input for the city or the activities 4.Click on "Search" button It should redirects to the page for the related search information 5."Offers" section h2 tag having text "Offers"
96					
97	12	Validate Footer section	https://www.makemytrip.com/	Verify all links present in the footer	1.Click on twitter icon It will redirect to twitter page of makemytrip 2.Click on Facebook icon It will redirect to facebook page of makemytrip 3.Click on country name available in footer It will redirect to the home page
98					
99	13	Validate Header bar	Website Should be opened https://www.makemytrip.com/	Validate all links present in the navigation bar	1.Validate click on the Super Offers and verify the link It should redirects to the new tab 2.Hover Over myBiz Pop Up should be open where text "Planning a Business Trip ?" 3.Click On myBiz Pop Up should be open for login or sign up 4.Close the pop up opened in myBiz Pop up should be closed 5.Click on MyTrip In Navigation Pop up should be open for login personal account or with MyBiz account 6.Click on Login or Create account Pop up should be open for login personal account or with MyBiz account 7.Click on the country/language/currency tab Dropdown should be open in which we can all this things
100					
101					
102					
103					
104					
105					
106					
107					
108					
109					
110					
111					
112					
113					
114					
115					
116					

Code:

Testcase file:

*** Settings ***

Resource/Common/testCase_resources.robot

#Suite Setup Open The browser

#Suite TearDown Close Browser Session

#Author: Dhruvi.Virani

*** Test Cases ***

Validate MakeMyTrip Website

- Open The Browser
- Validate Logo
- Validate Header Bar
- Validate Flights Tab
- Validate Hotels Tab
- Validate Home Stays Tab
- Validate Holiday Packages Tab
- Validate Trains Tab
- Validate Buses Tab
- Validate Cabs Tab
- Validate Forex Tab
- Validate Charter Flights Tab
- Validate Activities Tab
- Validate Footer Section

Close The Browser

*** Keywords ***

Open The Browser

- Set Selenium Timeout 10 seconds

- Open Browser https://www.makemytrip.com/ }
- Chrome

- Maximize Browser Window

- Sleep 3

- Wait Until Element Is Visible

css=img[src='//imgak.mmtcdn.com/pwa_v3/pwa_hotel_assets/header/mmtLogoWhite.png']

Validate Logo

- Click Element

css=img[src='//imgak.mmtcdn.com/pwa_v3/pwa_hotel_assets/header/mmtLogoWhite.png']

- Wait Until Element Is Visible

css=img[src='//imgak.mmtcdn.com/pwa_v3/pwa_hotel_assets/header/mmtLogoWhite.png']

Validate Header Bar

- Click Element css=p[class='font10 userPerText appendTop3']

- Switch Window New

- Wait Until Element is Visible //span[text()='FILTERS']

- Switch Window Main

- Wait Until Element Is Visible

css=img[src='//imgak.mmtcdn.com/pwa_v3/pwa_hotel_assets/header/mmtLogoWhite.png']

- Click Element //p[text()='Business Travel Solution']

- Sleep 3

- Wait Until Element Is Visible css=span[data-cy='myBizLoginClose']

```
Click Element css=span[data-cy='myBizLoginClose']
Click Element css=span[class='mbizUse__close']
Wait Until Element Is Visible ${logo}
Click Element css=span[class='latoBold capText font11']
Click Element //button[text()='Apply']
```

Validate Flights Tab

```
Click Element //li[text()='Round Trip']
Click Element //p[@class='code makeRelative'][1]
Wait Until Element Is Visible //p[text()='Mumbai, India']
Click Element //p[text()='Mumbai, India']
Click Element //p[@class='code makeRelative'][1]
Wait Until Element Is Visible //p[text()='Bangkok, Thailand']
Click Element //p[text()='Bangkok, Thailand']
Click Element //p[@class='blackText font20 code lineHeight36'][2]
Click Element //div[@class='dateInnerCell']/p[text()='1'][1]
Click Element //div[@class='dateInnerCell']/p[text()='15'][1]
#Click Element //li[text()='First Class']
#Click Element css=button[class='primaryBtn btnApply pushRight']
Click Element css=a[class='primaryBtn font24 latoBold widgetSearchBtn ']
Wait Until Element Is Visible css=p[class='font24 blackFont whiteText appendBottom20
journey-title']
Go Back
Wait Until Element Is Visible ${logo}
Mouse Over //p[text()='Armed Forces ']
Mouse Over //p[text()='Student ']
Mouse Over //p[text()='Senior Citizen ']
Mouse Over //p[text()='Doctors & Nurses ']
execute javascript window.scrollTo(0,500)
Wait Until Element Is Visible //font[text()='Offers']
execute javascript window.scrollTo(0,1200)
Wait Until Element Is Visible css=h3[class='appDnldCnt__heading']
Element Text Should Be css=h3[class='appDnldCnt__heading'] Download App Now !
execute javascript window.scrollTo(1200,0)
```

Validate Hotels Tab

```
Click Element css=span[class='chNavIcon appendBottom2 chSprite chHotels']
Wait Until Element Is Visible //span[text()='City, Property name or Location']
Click Element //span[text()='City, Property name or Location']
Scroll Element Into View //span[text()='City, Property name or Location']
Wait Until Element Is Visible //p[text()='Mumbai'][1]
Click Element //p[text()='Mumbai'][1]
Click Element //span[@class='lbl_input latoBold appendBottom5'][4]
```

```
Wait Until Element Is Visible //li[text()='₹2500-₹5000']
Click Element //li[text()='₹2500-₹5000']
Click Element css=button[id='hsw_search_button']
Wait Until Element Is Visible //p[text()='Showing Properties in Mumbai']
Go Back
Wait Until Element Is Visible ${logo}
execute javascript window.scrollTo(0,500)
Wait Until Element Is Visible //font[text()='Offers']
execute javascript window.scrollTo(0,1200)
Wait Until Element Is Visible css=h3[class='appDnldCnt__heading']
Element Text Should Be css=h3[class='appDnldCnt__heading'] Download App Now !
execute javascript window.scrollTo(1200,0)
```

Validate Home Stays Tab

```
Wait Until Element Is Visible ${logo}
Click Element //span[text()='Homestays']
Sleep 1
Click Element //span[text()='City, Property name or Location']
#Wait Until Element Is Visible css=div[class='hsw_inputBox selectHtlCity ']
Click Element css=input[class='hsw_inputField font30 lineHeight36 latoBlack']
Click Element css=input[id='checkin']
Click Element //div[text()='15'][1]
Click Element //div[text()='25'][1]
Click Element css=button[class='primaryBtn btnApply']
Wait Until Element Is Visible //p[text()='Select a Reason (optional)']
Click Element //p[text()='Select a Reason (optional)']
Wait Until Element Is Visible //li[text()='Leisure']
Click Element //li[text()='Leisure']
Click Element css=button[id='hsw_search_button']
Wait Until Element Is Visible //p[text()='Showing Properties in Goa']
Go Back
execute javascript window.scrollTo(0,500)
Sleep 1
Wait Until Element Is Visible //p[text()='Find Your Wonderland']
execute javascript window.scrollTo(0,2000)
Wait Until Element Is Visible //font[text()='Offers']
execute javascript window.scrollTo(2000,0)
```

Validate Holiday Packages Tab

```
Wait Until Element Is Visible ${logo}
Click Element //span[text()='Holiday Packages']
Sleep 1
Click Element css=input[id='fromCity']
```

```
Wait Until Element Is Visible //li[text()='Chennai']
Click Element //li[text()='Chennai']
Click Element //span[text()='To City/Country/Category']
Wait Until Element Is Visible //div[text()='Kashmir']
Click Element //div[text()='Kashmir']
#Click Element css=span[class='appendBottom5 field-label down-arrow']
Click Element (//p[text()='30'])[1]
Sleep 1
Click Element //button[text()='APPLY']
execute javascript window.scrollTo(0,100)
Sleep 1
Click Element //div[@class='action-container']/button
Click Element css=button[class='primaryBtn font24 latoBlack widgetSearchBtn lineHeight29']
Wait Until Element is Visible //h1[text()='Kashmir']
Go Back
Wait Until Element Is Visible ${logo}
execute javascript window.scrollTo(100,0)
Sleep 1
Click Element //span[text()='Destinations']
Click Element //li[text()='Emerging']
Click Element //a[text()='Gujarat']
Switch Window New
Wait Until Element Is Visible css=h1[class='topHeading']
Switch window Main
Wait Until Element Is Visible ${logo}
Click Element //span[text()='Honeymoon']
Click Element //a[text()='Maldives']
Switch Window New
Wait Until Element Is Visible //h1[text()='Maldives']
Switch window Main
Wait Until Element Is Visible ${logo}
Click Element //span[text()='Featured']
Wait Until Element Is Visible //a[text()='Adventure']
Click Element //a[text()='Adventure']
Switch Window New
Wait Until Element Is Visible //h1[text()='Adventure']
Switch window Main
Wait Until Element Is Visible ${logo}
Click Element //span[text()='Travel Guidelines']
Click Element //li[text()='International']
Click Element //a[text()='International Travel Guidelines']
Wait Until Element Is Visible ${logo}
execute javascript window.scrollTo(0,2000)
```

```
Wait Until Element Is Visible //h2[text()='Jackpot Deals on Top Selling Packages']  
execute javascript window.scrollTo(0,2500)  
Wait Until Element Is Visible //h2[text()='International Destinations!']  
execute javascript window.scrollTo(0,2850)  
Wait Until Element Is Visible //h2[text()='Explore the Unexplored']  
execute javascript window.scrollTo(0,3100)  
Wait Until Element Is Visible //font[text()='Offers']  
execute javascript window.scrollTo(3100,0)
```

Validate Trains Tab

```
Click Element //span[text()='Trains']  
Sleep 1  
Click Element //span[text()='From']  
Wait Until Element Is Visible //span[text()='Kolkata Howrah Junction']  
Click Element //span[text()='Kolkata Howrah Junction']  
Wait Until Element Is Visible //span[text()='Delhi']  
Click Element //span[text()='Delhi']  
Click Element //(div[text()='12'])[1]  
Click Element //li[text()='Third AC']  
Click Element css=a[class='primaryBtn font24 latoBold widgetSearchBtn']  
Wait Until Element Is Visible css=button[class=' primaryBtn font18 latoBlack  
widgetSearchBtn']  
Go To https://www.makemytrip.com/  
Wait Until Element Is Visible ${logo}  
Click Element //span[text()='Trains']  
Sleep 1  
Click Element //span[text()='CHECK PNR STATUS']  
Element Should Be Visible css=input[placeholder='Enter 10 digit PNR']  
Click Element //span[text()='LIVE TRAIN STATUS']  
Element Should Be Visible css=label[class='lbl_input makeFlex column latoBold']  
execute javascript window.scrollTo(0,250)  
Wait Until Element Is Visible //font[text()='Offers']
```

Validate Buses Tab

```
Click Element css=span[class='chNavIcon appendBottom2 chSprite chBuses']  
Sleep 1  
Click Element //span[text()='From']  
Wait Until Element Is Visible //span[text()='Pune, Maharashtra']  
Click Element //span[text()='Pune, Maharashtra']  
Wait Until Element Is Visible //span[text()='Delhi, Delhi']  
Click Element //span[text()='Delhi, Delhi']  
Click Element //(div[text()='15'])[1]  
Click Element css=button[id='search_button']
```

Wait Until Element Is Visible css=button[class='primaryBtn font18 latoBlack widgetSearchBtn']
Go Back
Wait Until Element Is Visible \${logo}

Validate Cabs Tab

Click Element css=span[class='chNavIcon appendBottom2 chSprite chCabs']
Sleep 1
Click Element //li[text()='OUTSTATION ROUND TRIP']
Click Element //span[text()='From']
Wait Until Element Is Visible //span[text()='Bangalore']
Click Element //span[text()='Bangalore']
Click Element //span[text()='To']
Wait Until Element Is Visible //span[text()='Mumbai']
Click Element //span[text()='Mumbai']
Click Element //span[text()='DEPARTURE']
Click Element //(div[text()='11'])[1]
Click Element //(div[text()='26'])[1]
Click Element //a[text()='Search']
Wait Until Element Is Visible css=button[class='bluePrimarybtn widgetSearchBtn']
Go Back
Wait Until Element Is Visible \${logo}

Validate Forex Tab

Click Element //span[text()='Forex']
Wait Until Element Is Visible //p[text()='Global INR Card']
Click Element css=a[class='primaryBtn font24 latoBlack widgetSearchBtn']
Wait Until Element Is Visible css=a[class='navbar-brand navbar-logo']
Go Back
Wait Until Element Is Visible \${logo}

Validate Charter Flights Tab

Click Element css=span[class='chNavIcon appendBottom2 chSprite chCharterFlights']
Sleep 1
Click Element //li[text()='Round Trip']
Click Element css=a[class='primaryBtn font24 latoBold widgetSearchBtn']
Wait Until Element Is Visible //p[text()='Sorry, No Charters found for this search']
Go Back
Wait Until Element Is Visible \${logo}

Validate Activities Tab

Click Element //span[text()='Activities']
Wait Until Element Is Visible //input[@placeholder='Activities, Tours or Destination'][1]

Validate Footer Section

```
execute javascript window.scrollTo(0,3000)
Sleep 1
execute javascript window.scrollTo(0,5000)
Sleep 1
#Wait Until Element Is Visible css=span[class='twiiterIcon landingSprite']
Click Element css=span[class='twiiterIcon landingSprite']
Wait Until Element Is Visible css=svg[class='r-13v1u17 r-4qtqp9 r-yyyyoo r-16y2uox r-8kz0gk
r-dnmrzs r-bnwqim r-1plcrui r-lrvibr r-lrsllp']
Sleep 1
Go To https://www.makemytrip.com/
Wait Until Element Is Visible ${logo}
execute javascript window.scrollTo(0,3000)
Sleep 1
execute javascript window.scrollTo(0,5000)
Sleep 1
Click Element css=span[class='facebookIcon landingSprite']
Wait Until Element Is Visible css=div[class='x92rtbv x10l6tqk x1tk7jg1 x1vjfegm']
Click Element css=div[class='x92rtbv x10l6tqk x1tk7jg1 x1vjfegm']
Wait Until Element Is Visible //h1[@class='x1heor9g x1qlqyl8 x1pd3egz x1a2a7pz']
Go Back
Wait Until Element Is Visible ${logo}
```

Close The Browser

```
Close Browser
```