

**Foundation Programme in Higher Education**

**Module** : DOC334 Introduction to Programming in

Python-P2

**Module Leader** : Dr. Damitha Karunaratne

**Type of assignment:** Individual

**Submission Date** : 23rd April, 2021 before 5.00pm

**Student Name** : I. N. Perera

**Student ID** : 20200216

# Acknowledgement

The final output of this assignment required a lot of guidance and assistance from my lecturers Dr. Damitha Karunaratne and Mr Nishan Saliya who conducted the tutorial sessions and the self-study sessions. I thank them for giving me this opportunity to do this assignment on time. I’m incredibly grateful to them for providing me useful tips that I would never consider.

# Introduction

Python is a broadly used general-purpose high level programming language, invented by Guido Van Rossum in 1991 and auxiliary 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. Modular programming refers to a procedure of contravening a large unwieldy programming task into separate, smaller, more manageable subtasks or modules. Individual modules can then be mended together similarly to building blocks to generate a superior application. This assignment contains the python programmes in text form and screenshots.

Table of Content

[Acknowledgement 2](#_Toc69846351)

[Introduction 3](#_Toc69846352)

[Tables 5](#_Toc69846353)

[Your Task 6](#_Toc69846354)

[Tasks to Complete 7](#_Toc69846355)

[Description of the Problem 9](#_Toc69846356)

[The Process of the Game 9](#_Toc69846357)

[Solution 10](#_Toc69846358)

[Algorithm of the Game. 10](#_Toc69846359)

[Developing the Programme as modules and packages. 11](#_Toc69846360)

[Developing the programme which checks the possibilities, calculates the marks, and calculates the total game plays. 12](#_Toc69846361)

[Developing the programme which appends the game play history to the database. 17](#_Toc69846362)

[Complete python programme of ProcessingTheGame.py including all the steps and the function which appends the Game play history to the database (InsertionOfTheHistory.py) 21](#_Toc69846363)

[Developing the program which shows the rules to the human Player 29](#_Toc69846364)

[The Python Programme 29](#_Toc69846365)

[Developing the Main Python Programme which runs the entire programme 31](#_Toc69846366)

[Python Programme 32](#_Toc69846367)

[Folder structure of the Entire Programme 33](#_Toc69846368)

[33](#_Toc69846369)

# Tables

# Your Task

You are to create a Python 3.x program which mimics this game. A single player must be able to play this game with a computer. You can create a Python program which runs in the console. You can decide how the menus and the console interface will look like. You must clearly state your assumptions.

# Tasks to Complete

1. You must use proper Python 3.x program constructs such as packages, modules, functions, variables, data structures, etc. to develop this program.

2. The player must be able to do below tasks

• View the past game play history

1. This is stored by using a either a file base or a database (a DB will carry more marks)

2. If a DB is used, username must be “root” with no passwords and the server must be “localhost” 3. You should also provide means to import/restore your current database for marking purposes. 4. If a text file is used, the extension must be of .TXT

• Play the game with the PC as many times as he/she wants

1. Players must also have the ability to exit the game as they desire at any time

• Your program should keep track of wins, losses and draws

• Your program must be able to display stats like below

1. Total game plays

2. Total wins by human

3. Total wins by PC

4. Total draws

3. You can use any external package which will help you to develop the game

• Such package uses must be explained in the report

• Proper instructions must be given to do the installation

• Links where you install/download the packages must also be provided in the report

4. A challenge activity will be to display the result in a HTML file so you can see the past game plays in a web browser.

• Hint: pretty table can be used for this

• You’ll get extra 10 marks for completing this task

5. You are to develop a console-based application with a suitable menu system to complete this task

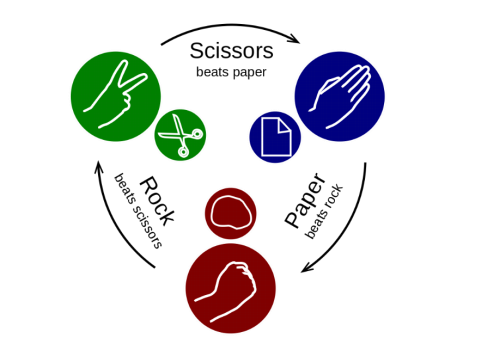
# Description of the Problem

The task requires to generate a python 3x program which helps to run the game “Rock Paper Scissors”. The game should be created using functions, modules packages variables and Data structures. The question also requires to display statistics like

1. Total game plays
2. Total wins by the Human player
3. Total wins by the PC
4. Total number of draws.

Finally the question requires to develop a console based application with a suitable menu system to complete this task.

## The Process of the Game



**Figure 1| Process of the game**

**Figure SEQ Figure \\* ARABIC 1| Process of the Game**

* Rock beats Scissors
* Paper Beats Rock
* Scissors beat Paper

# Solution

## Algorithm of the Game.

1) START

2) Getting the Human Player’s name

3) Getting the choice by the Human Player from the given three choices (Rock, Paper or scissors)

4) Getting the choice by the computer from the given three choices randomly. (Rock, Paper or scissors)

5) Displaying the result after each game

6) Getting the choice on playing again from the human player, if the choice on playing again is “yes” repeat steps number 2,3,4 Else display the final result and insert data into the database or to the text file.

7) STOP

## Developing the Programme as modules and packages.

2) Getting the Human Player’s name –Main Programme

3) Getting the choice by the Human Player from the given three choices (Rock, Paper or scissors) – In another python program which checks the possibilities (Could be done in a package)

4) Getting the choice by the computer from the given three choices randomly. (Rock, Paper or scissors) – In another python program which checks the possibilities (Could be done in a package)

5) Displaying the result after each game – In another python program which checks the possibilities (Could be done in a package)

6) Getting the choice on playing again from the human player, if the choice on playing again is yes repeat steps number 2,3,4 Else display the final result and insert data into the database or to the text file. – In another python Programme which inserts the data to a database. (In a module)

**Developing the programme which checks the possibilities, calculates the marks, and calculates the total game plays**.

This python programme is inside a folder named “Package” which will be imported to the main programme.

**Step one**

Defining the function and giving it a suitable name

def ProcessingTheGame(choiceList,PlayerName):

**Step two**

Declaring the variables

#creating and initializing the variables

Score\_Of\_the\_Computer=0

Score\_Of\_the\_HumanPlayer=0

computer\_wins=0

HumanPlayer\_wins=0

Total\_game\_plays=0

Total\_draws = 0

**Step three**

Getting the user choice on playing as:

userChoice = input("Do you want to play ? (Yes/No): ")

**Step Four**

1. Checking the possibilities
2. Getting the Computer’s random choice from the choice list
3. Calculating total number of draw games
4. Calculating total number of games played
5. Wins by the computer
6. Wins by the human player
7. Score of the human player
8. Score of the Computer
9. Displaying the results after each turn
10. Determining the winner
11. Inserting the Game play history to the database – Included in another module and imported to this programme

**If using a file base system the code should be:**

file\_object = open('GameHistory.txt','a+')

file\_object.write("============================================ " + "\n")

file\_object.write(" "+ "\n")

file\_object.write("Name of the human player:"+str (PlayerName) + "\n")

file\_object.write("The Game TIES"+ "\n")

file\_object.write("Wins by the human player(Score of the Human Player):" +str (HumanPlayer\_wins)+ "\n")

file\_object.write("Wins by the computer(Score of the Computer): " +str (computer\_wins)+ "\n")

file\_object.write("Total number of draws: " +str( Total\_draws)+"\n")

file\_object.write("Total Number of Games played:" +str(Total\_game\_plays)+"\n")

file\_object.write(" "+ "\n")

file\_object.write("===================================" + "\n")

#Close file

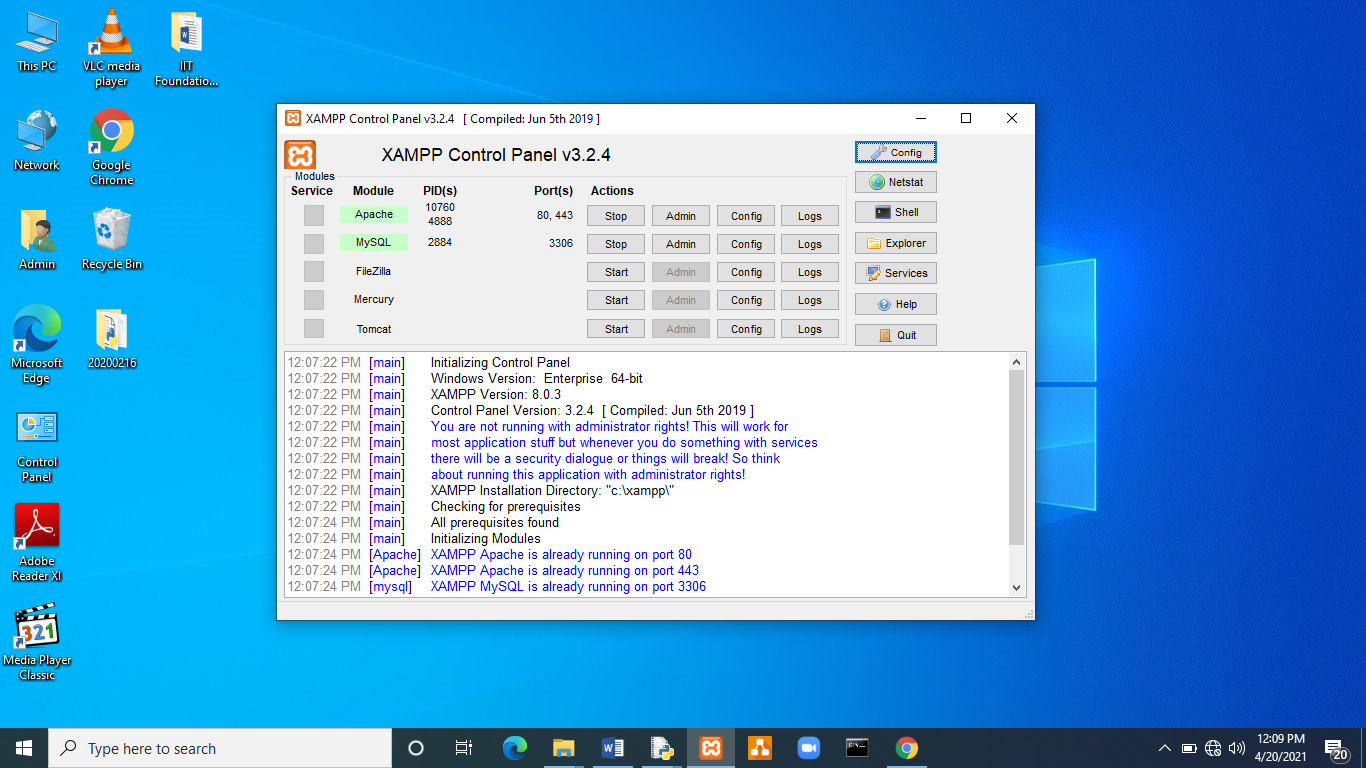
file\_object.close()

Once the user enters “No” as the choice on playing the above code should be repeated in a while loop, along with if and elifs. The data about the game would be added to a text file called “GameHistory” . The text file opens in a+ mode. Which allows to read and append data to the text file. If the file is not existing, it creates a file named “GameHistory”

**If using a database system:**

**Step1**

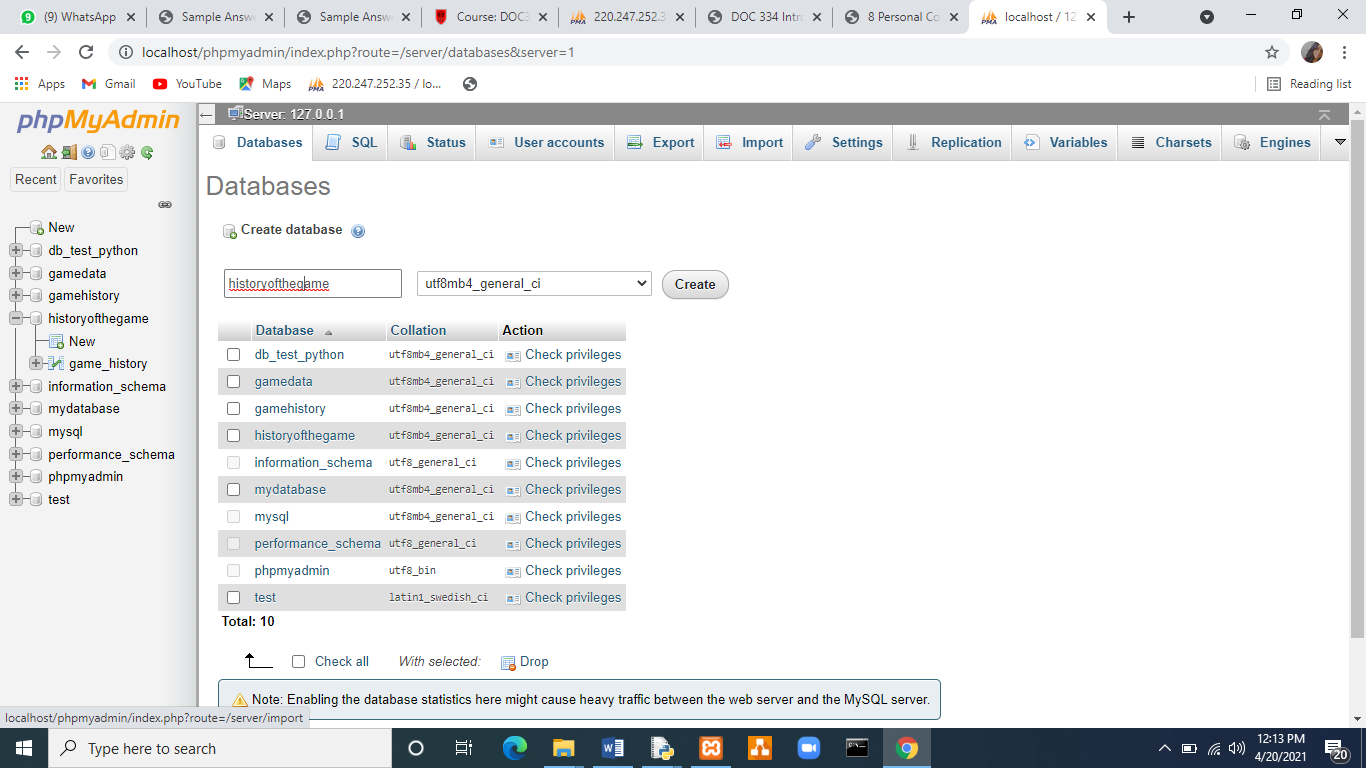
The XAMPP server should run with MySQL and Apache services.



**Figure 2| XAMPP Server runs**

**Step2**

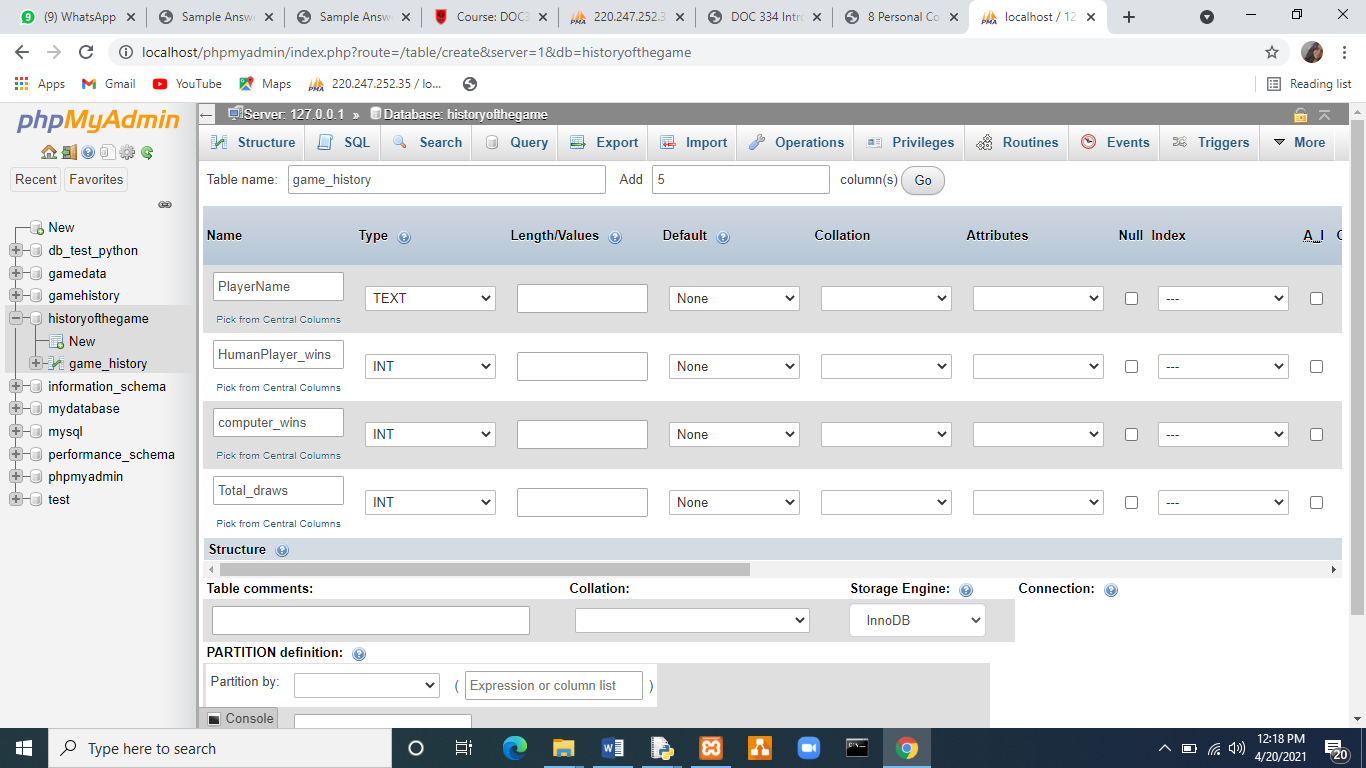
Creating a new database



**Figure 3| Creating a new database called “historyofthegame”**

**Step3**

Creating table in the database (**historyofthegame”)**



**Figure 4| Creating the table called “game\_hstory” with column names and types**

**Assumptions:** When considering the file based system and the data base system I assumed that the file base system would make the python program complex and the database system would make the database system convenient and simple.

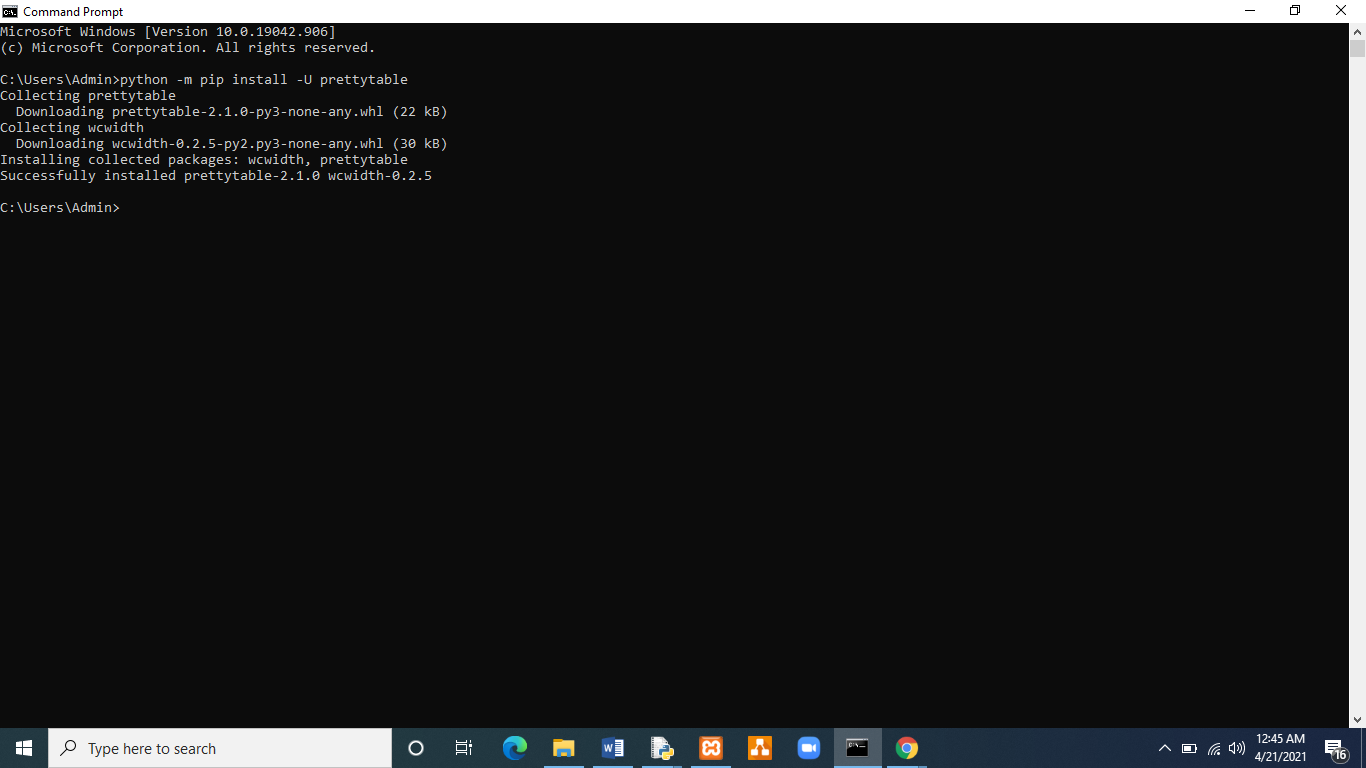
Therefore the programme which appends data to the database could be done as a function and could be imported to the main programme which runs the entire game by importing modules and packages.

Creating a HTML file so that the past game play history details could be visible in it

**Step1**

Installing prettytable using command prompt

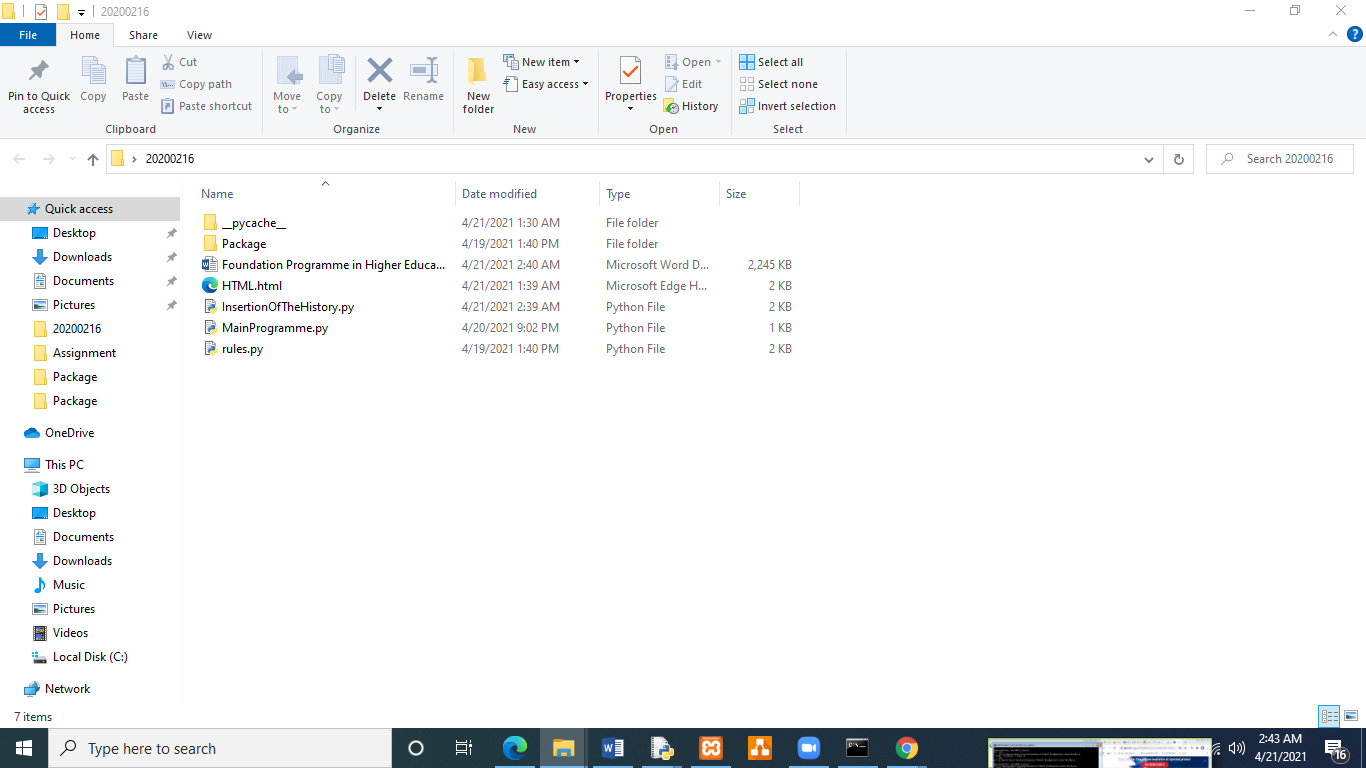
python -m pip install -U prettytable



**Figure 5| Download successful**

**Creating an html file named HTML to satisfy the last task of the assignment**

The file extension should be .html



**Figure 6| Created a html file named HTML**

**Developing the programme which appends the game play history to the database**.

Name of the python programme (Saved as): InsertionOfTheHistory.py

def InsertionOfTheGameDetails(PlayerName,HumanPlayer\_wins,computer\_wins,Total\_draws,Total\_game\_plays):

"Insertion of the Game details"

import mysql.connector

from prettytable import from\_db\_cursor

import mysql.connector

#Open a database connection with a dictionary

conDict={"host":"localhost",

"user":"root",

"database":"historyofthegame",

"password":""

}

db=mysql.connector.connect(\*\*conDict)

#Prepare a cursor object using cursor () method

cursor=db.cursor()

cursor.execute ("SELECT \* FROM game\_history")

GamePlays = from\_db\_cursor(cursor)

print(GamePlays)

data=GamePlays.get\_html\_string(attributes={"name":"GameHistory","class":"table"},format=True)

File\_object=open(“HTML.html”, “w”)

File\_object.write(data)

#committing the change

#executing sql query using execute () method

updateText="INSERT INTO game\_history VALUES ('"+PlayerName+"','"+HumanPlayer\_wins+"','"+computer\_wins+"','"+Total\_draws+"','"+Total\_game\_plays+"')"

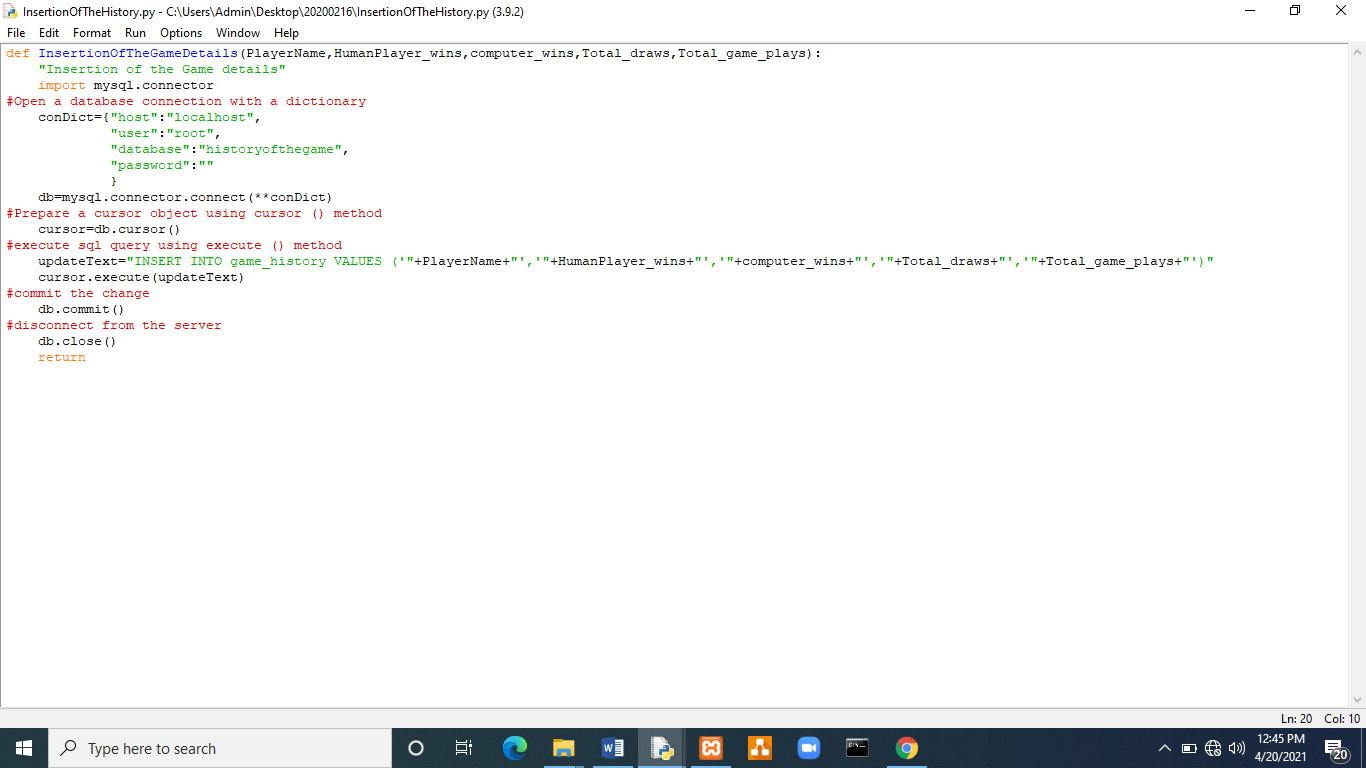
cursor.execute(updateText)

db.commit()

#disconnecting from the server

db.close()

return



The above program which appends the data to the database (InsertionOfTheHistory module which contains “InsertionOfTheGameDetails” ) should be imported once the Human Player inputs “No” as the choice on playing again.

**Figure 7| Screen shot of the “InsertionOfTheHistory.py**”

Complete python programme of ProcessingTheGame.py including all the steps and the function which appends the Game play history to the database (InsertionOfTheHistory.py)

def ProcessingTheGame(choiceList,PlayerName):

"Processing the game"

#Creating and initializing the variables

Score\_Of\_the\_Computer=0

Score\_Of\_the\_HumanPlayer=0

computer\_wins=0

HumanPlayer\_wins=0

Total\_game\_plays=0

Total\_draws = 0

#importing random

import random

print("") #To get a clean output

userChoice = input("Do you want to play ? (Yes/No): ")

print("") #To get a clean output

#Processing the game

while (userChoice == "Yes"):

#Getting the choice from the Computer

Computer = random.choice(choiceList)

Player2= input("Enter your choice here(Rock,Paper,Scissors): ")

if (Computer==Player2):

print("YOU and the Computer chose the same,Game TIED")

Total\_draws = Total\_draws + 1 #Calculating the draw games

Total\_game\_plays=Total\_game\_plays+1 #Calculating the total no. of games

continue

while (Player2 == "Scissors"):

if (Computer == "Paper"):

print("You chose Scissors and the Computer chose Paper, You WON!")

Score\_Of\_the\_HumanPlayer=Score\_Of\_the\_HumanPlayer+1 #Score of the Human player

HumanPlayer\_wins=HumanPlayer\_wins+1 #Wins by the Human player

Total\_game\_plays=Total\_game\_plays+1 #Calculating the total no. of games

elif (Computer == "Rock"):

print("YOU chose Scissors and the Computer chose Rock, You LOSE!")

Score\_Of\_the\_Computer=Score\_Of\_the\_Computer+1 #Score of the Computer

computer\_wins=computer\_wins+1 #Wins by the Computer

Total\_game\_plays=Total\_game\_plays+1 #Calculating the total no. of games

break

while (Player2 == "Paper"):

if(Computer == "Scissors"):

print("You chose Paper and the Computer chose Scissors, you Lose!")

Score\_Of\_the\_Computer=Score\_Of\_the\_Computer+1 #Score of the Computer

computer\_wins=computer\_wins+1 #Wins by the Computer

Total\_game\_plays=Total\_game\_plays+1 #Calculating the total no. of games

elif (Computer=="Rock"):

print("You chose Paper and the Computer chose Rock You WON!")

Score\_Of\_the\_HumanPlayer=Score\_Of\_the\_HumanPlayer+1 #Score of the Human player

HumanPlayer\_wins=HumanPlayer\_wins+1 #Wins by the Human player

Total\_game\_plays=Total\_game\_plays+1 #Calculating the total no. of games

break

while (Player2=="Rock"):

if (Computer=="Paper"):

print("You chose Rock and the Computer chose Paper,YOU LOSE!")

Score\_Of\_the\_Computer=Score\_Of\_the\_Computer+1 #Score of the Computer

computer\_wins=computer\_wins+1 #Wins by the Computer

Total\_game\_plays=Total\_game\_plays+1 #Calculating the total no. of games

elif (Computer =="Scissors"):

print( "You chose Rock and the Computer chose Scissors! You WON!!”)

Score\_Of\_the\_HumanPlayer=Score\_Of\_the\_HumanPlayer+1 #Score of the Human player

HumanPlayer\_wins=HumanPlayer\_wins+1 #Wins by the Human player

Total\_game\_plays=Total\_game\_plays+1 #Calculating the total no. of games

break

print("")#A space is given to get a clear output

print("Wins by the human: " +str (HumanPlayer\_wins))#Converting to a string

print("Wins by the computer: " +str (computer\_wins))#Converting to a string

print("Total number of draws: " +str( Total\_draws)) # converting to a string

print("Total Number of Games played:" +str(Total\_game\_plays))#Converting to a string

userChoice=input("Do you want to play again? (Yes/No): ")#Asking the human player's choice on playing again

print("")#A space is given to get a clear output

while userChoice == "No":

if Score\_Of\_the\_Computer==Score\_Of\_the\_HumanPlayer:

print("")#A space is given to get a clear output

print("Marks are Tied")

print("Score of the computer: ",Score\_Of\_the\_Computer)

print("Your Score: ",Score\_Of\_the\_HumanPlayer)

print("No. of Wins by You: " +str (HumanPlayer\_wins))#Converting to a string

print("No. of Wins by the computer: " +str (computer\_wins))#Converting to a string

print("Total number of draws: " +str( Total\_draws)) # converting to a string

print("Total number of games played:" +str(Total\_game\_plays))#Converting to a string

print("")#A space is given to get a clear output

import InsertionOfTheHistory

InsertionOfTheHistory.InsertionOfTheGameDetails (str(PlayerName),str(HumanPlayer\_wins),str(computer\_wins),str(Total\_draws),str(Total\_game\_plays))

elif Score\_Of\_the\_Computer>Score\_Of\_the\_HumanPlayer:#Determining the winner

print("")#A space is given to get a clear output

print("Bad luck! Computer won!")

print("Score of the computer: ",Score\_Of\_the\_Computer)

print("Your Score: ",Score\_Of\_the\_HumanPlayer)

print("No. of wins by you: " +str (HumanPlayer\_wins))#Converting to a string

print("No. of Wins by the computers: " +str (computer\_wins))#Converting to a string

print("Total number of draws: " +str( Total\_draws)) # converting to a string

print("Total number of games played:" +str(Total\_game\_plays))#Converting to a string

print("")#A space is given to get a clear output

import InsertionOfTheHistory

InsertionOfTheHistory.InsertionOfTheGameDetails (str(PlayerName),str(HumanPlayer\_wins),str(computer\_wins),str(Total\_draws),str(Total\_game\_plays))

else:

print("")#A space is given to get a clear output

print("YOU WON!! ")

print("Score of the computer: ",Score\_Of\_the\_Computer)

print("Your Score: " ,Score\_Of\_the\_HumanPlayer)

print("Wins by You: " +str (HumanPlayer\_wins))#Converting to a string

print("Wins by the computer: " +str (computer\_wins))#Converting to a string

print("Total number of draws: " +str( Total\_draws)) # converting to a string

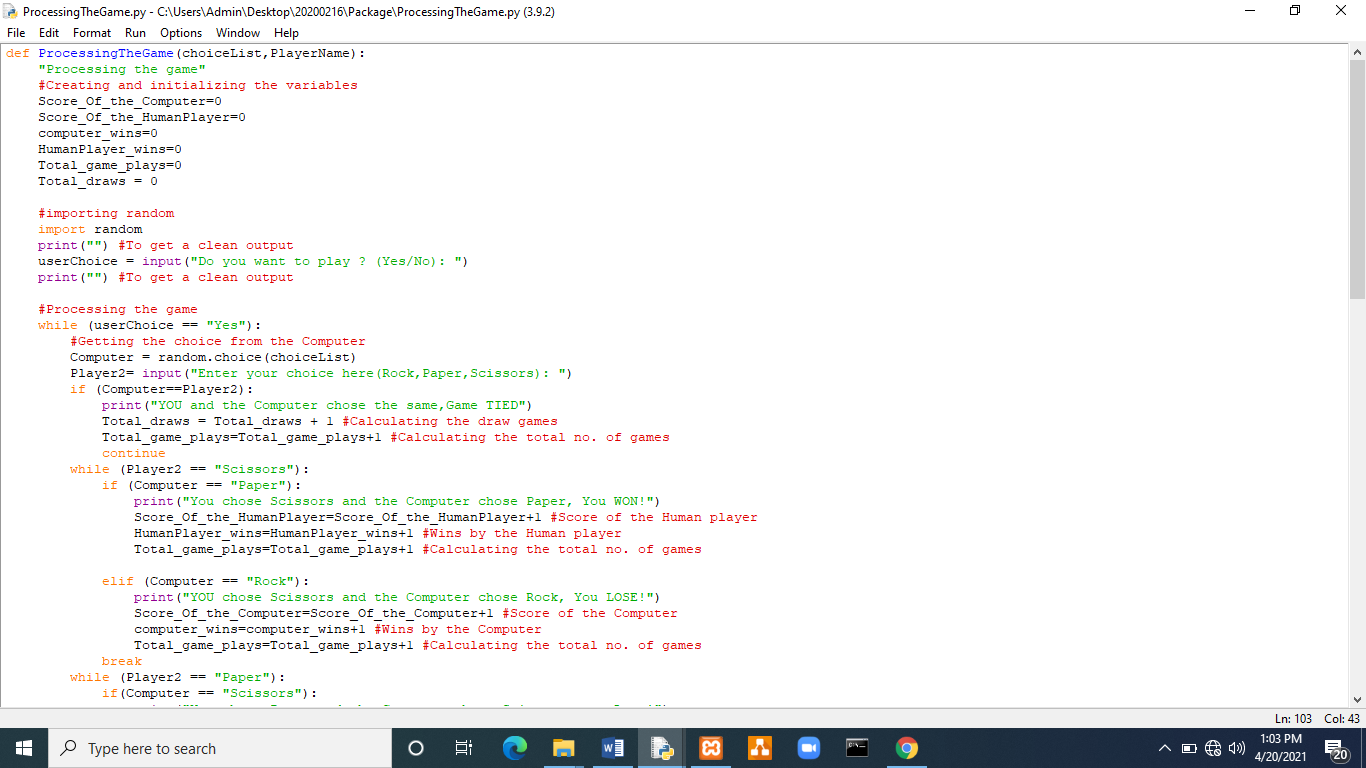
print("Total number of games played:" +str(Total\_game\_plays))#Converting to a string

print("")#A space is given to get a clear output

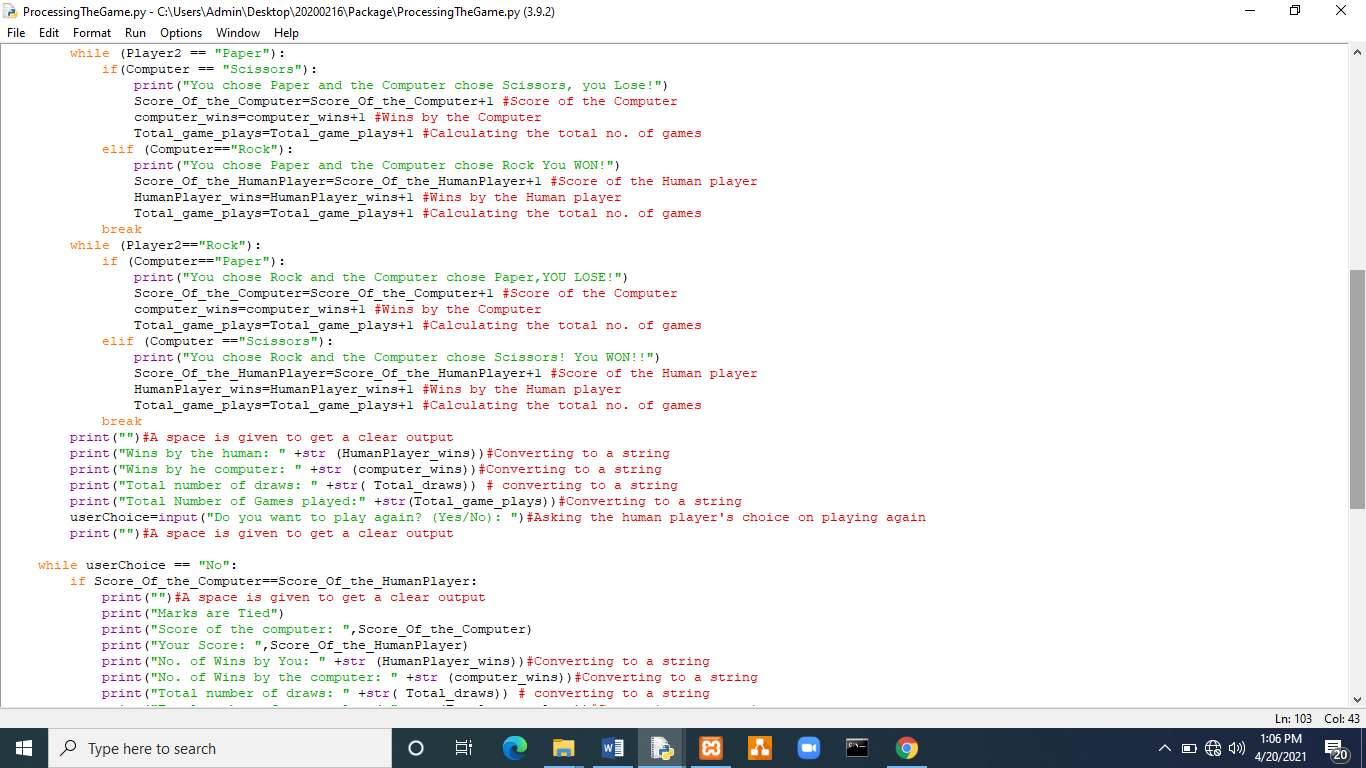
import InsertionOfTheHistory

InsertionOfTheHistory.InsertionOfTheGameDetails (str(PlayerName),str(HumanPlayer\_wins),str(computer\_wins),str(Total\_draws),str(Total\_game\_plays))

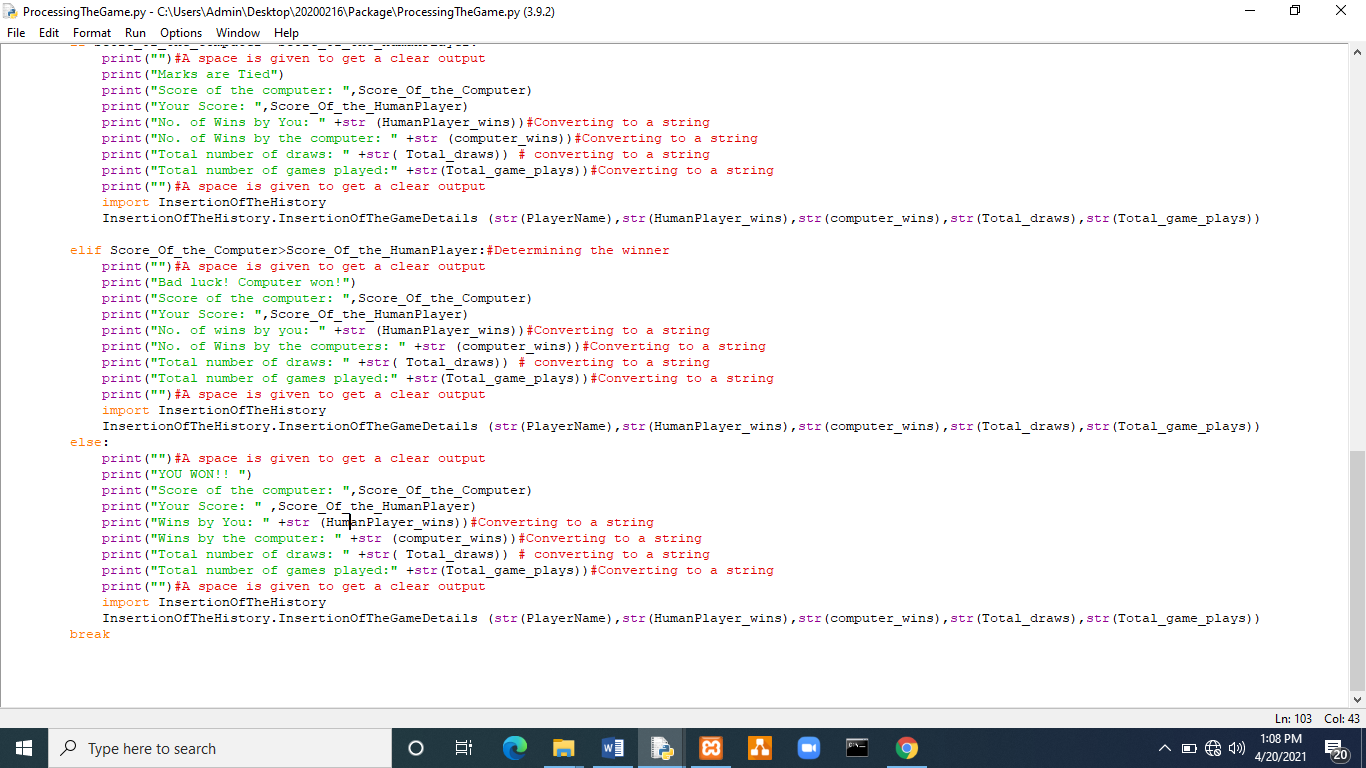
break



**Figure 8**| **ProcessingTheGame.py**



**Figure 9| ProcesssingTheGame.py Contd.**



**Figure 10| ProcesssingTheGame.py Contd.**

## Developing the program which shows the rules to the human Player

This python program is done as a module which will be called by the Main Programme which runs the complete game. This programme would display an unpretentious welcome message, the rules and regulations and a “Good Luck” message which gives the human player an impression, of a computer game. This is the program which the **MainProgramme.py** imports firstly. The **MainProgramme.py** calls the entire **rules module** and then calls the **TheRulesOfTheGame()** which is inside **rules** (module)

The Python Programme

def RulesOfTheGame():

print(" Hello! ")

print(" These are the rules and the regulations of the game! MAKE SURE YOU FOLLOW THEM!! ")

print(“ You should enter your choice as 'Rock','Paper','Scissors' ")

print(" Please type the first letter of your choice in capital ")

print(" The wins by the computer, the total wins by you, total number of draws and the total number of games played will be displayed at the end ")

print(" Always remember, That Scissors wins Over Paper ")

print(" Paper wins over Rock ")

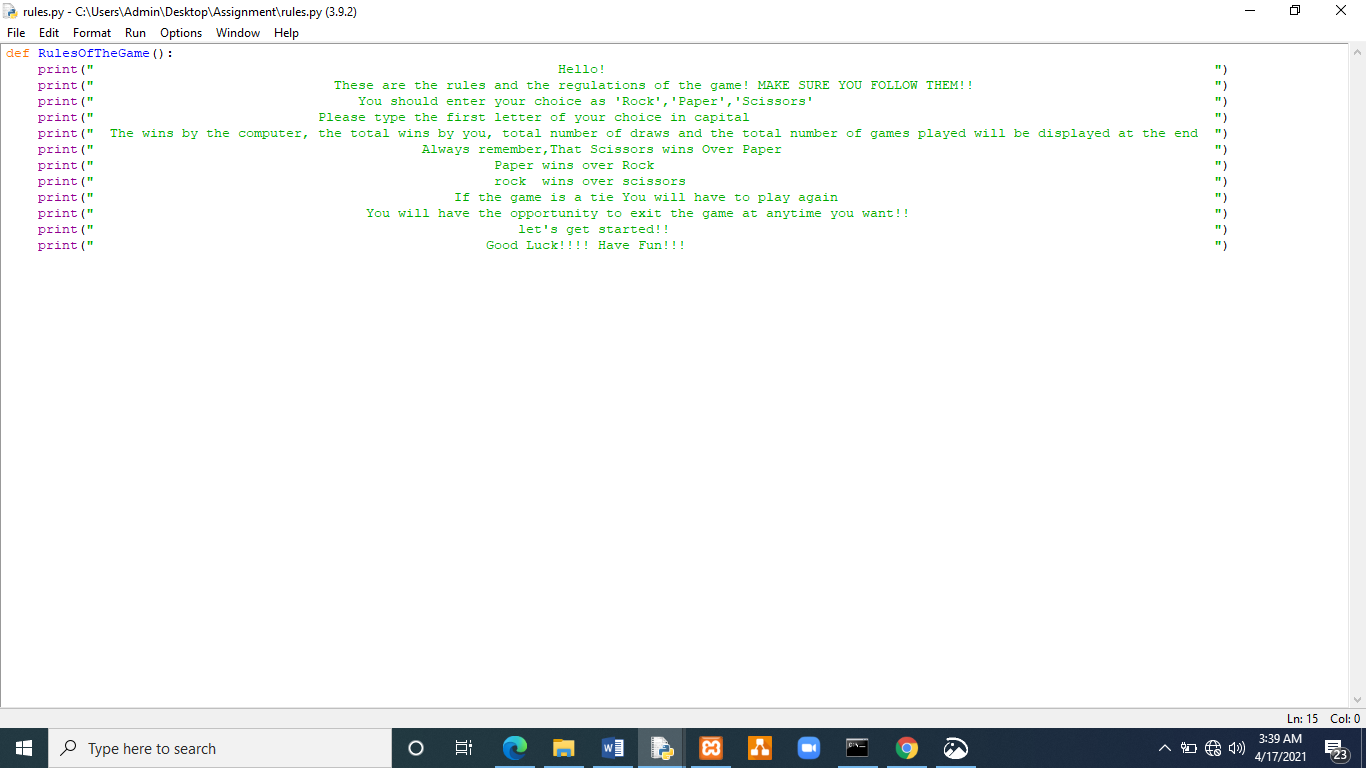
print(" rock wins over scissors ")

print(" If the game is a tie You will have to play again ")

print(“ You will have the opportunity to exit the game at any time you want!! ")

print(" let's get started!! ")

print(" Good Luck!!!! Have Fun!!! ")



**Figure 11| Screen Shot of the rules.py**

## Developing the Main Python Programme which runs the entire programme

This is the final programme and the main programme which imports all the modules and packages created behalf of the game. The programme inaugurates by importing the program which was created as a package in a separate folder named “Package”. It imports the package named “package” and the module named “ProcessingTheGame” which contains inside it.

#Calling from a package

import Package.ProcessingTheGame

Secondarily the variables would be initialized. The list of choices ("Rock","Paper","Scissors") as choiceList= ("Rock","Paper","Scissors") and the player’s name as PlayerName = "" and getting the name of the player.

#Creating and initializing the variables

choiceList =("Rock","Paper","Scissors")

PlayerName = ""

#Getting the player's name (input)

PlayerName=input("Hello! Enter your Name: ")

Next the entire “rules” module is imported, and the “RulesOfTheGame” which is inside the rules module would be called

#Calling the entire "rules" module

import rules

#Calling "RulesOfTheGame" inside "rules"

rules.RulesOfTheGame()

#Calling "ProcessingTheGame" inside ProcessingTheGame

Package.ProcessingTheGame.ProcessingTheGame(choiceList,PlayerName)

Python Programme

The MainProgramme.py inside 20200216

This is the main program, which imports modules, packages, functions and run the program.

#Calling from a package

import Package.ProcessingTheGame

#Creating and initializing the variables

choiceLis t= ("Rock","Paper","Scissors")

PlayerName = ""

#Getting the player's name (input)

PlayerName=input("Hello! Enter your Name :")

#Calling the entire "rules" module

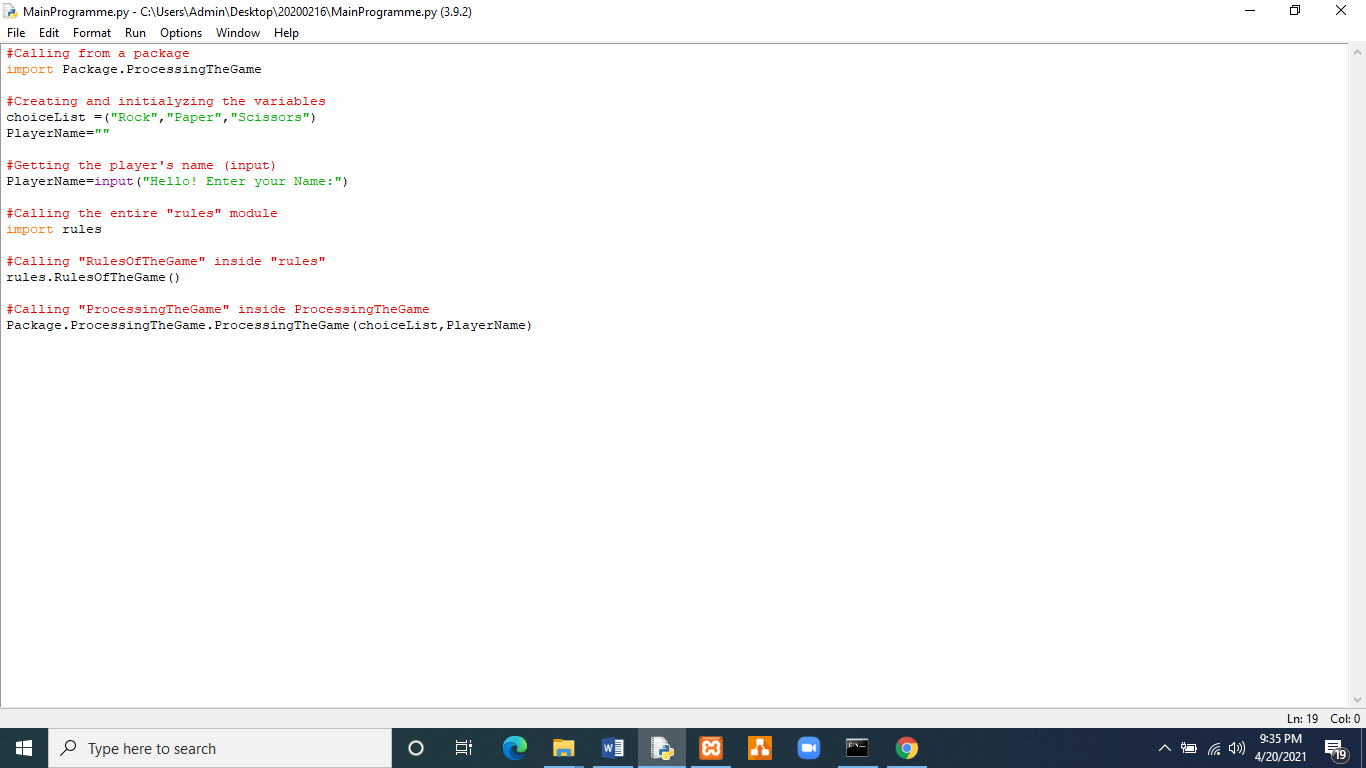
import rules

#Calling "RulesOfTheGame" inside "rules"

rules.RulesOfTheGame()

#Calling "ProcessingTheGame" inside ProcessingTheGame

Package.ProcessingTheGame.ProcessingTheGame(choiceList,PlayerName)



**Figure 12| Screen shot of MainProgramme.py**

## Folder structure of the Entire Programme

Folder saved in the Desktop

**“20200216”**

## 

The main python program saved in the **“20200216”** folder

“**MainProgramme.py”**

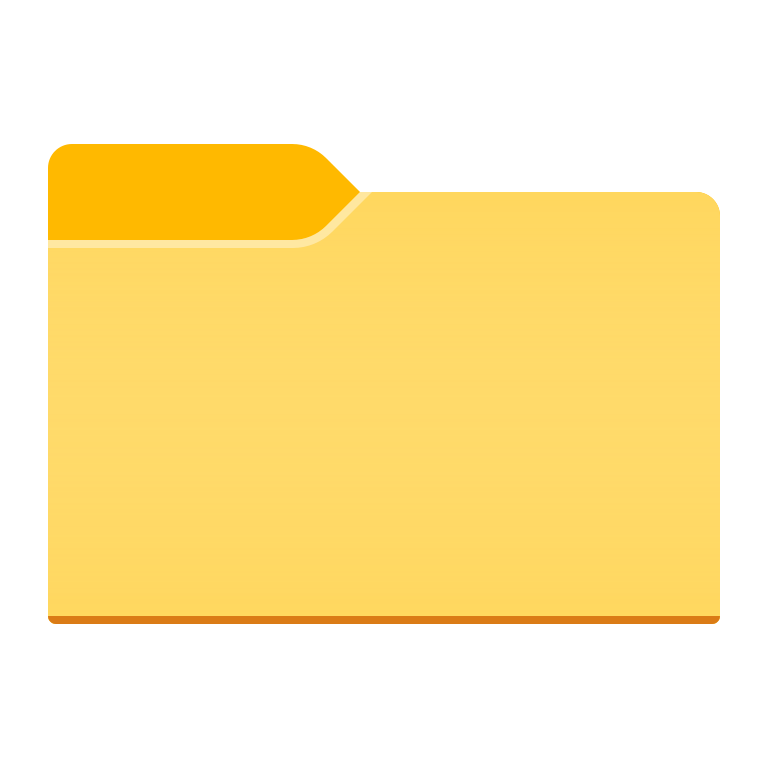
The module saved in the “**20200216**” folder

“**rules.py**”

**Figure SEQ Figure \\* ARABIC 2| View of the “Assignment” folder**

The sub folder saved in the “**20200216**” folder called,

“**Package**”





**Figure SEQ Figure \\* ARABIC 3| View of the “Package” Package**

The html file saved in the “**20200216**” folder

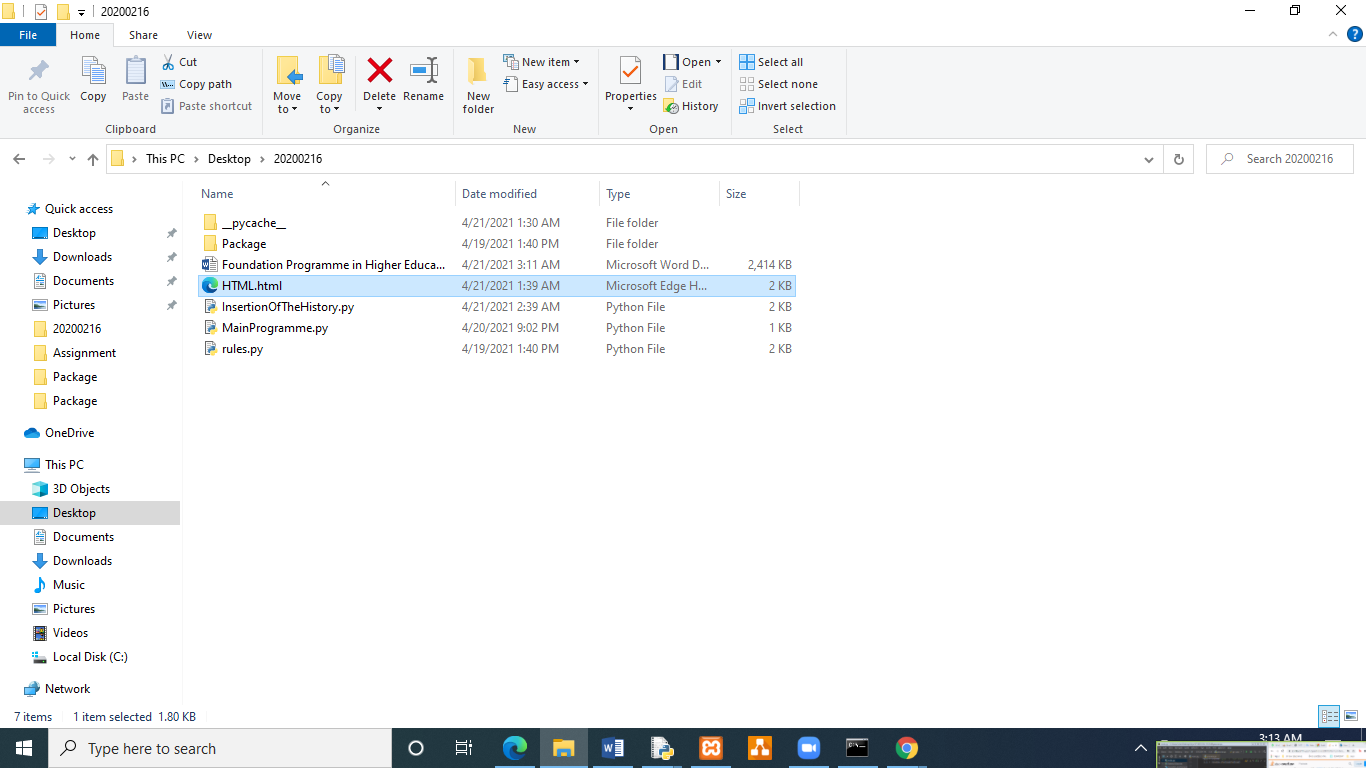
“HTML.html”

The module saved in the “**20200216**” folder

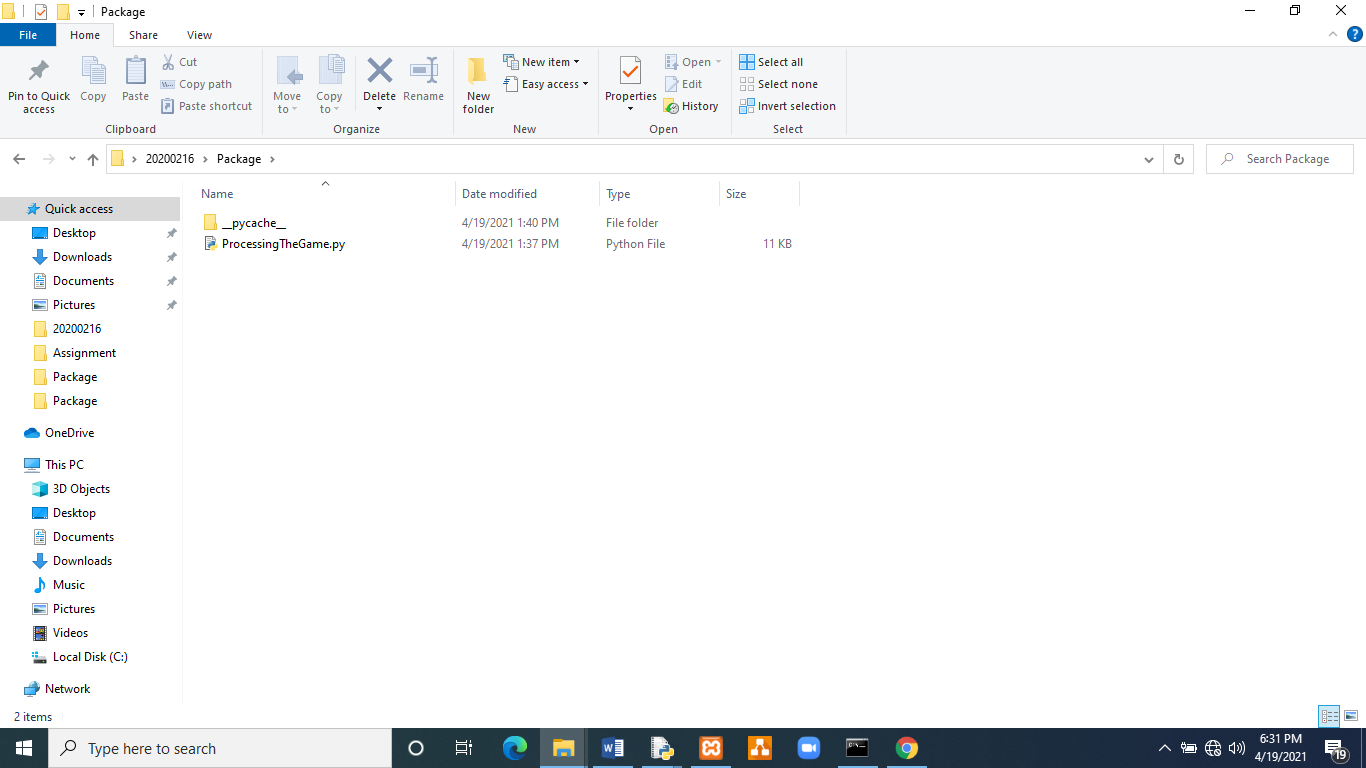
“InsertionOfTheHistory.py”

The module which is inside the Package called, “**Package**”

“**ProcessingTheGame.py**”



**Figure 13| Folder Structure of the Entire Programme**





**Figure 14| Folder Structure of the Entire Programme Contd.**

# Test Cases

**Test Case 1 (the inputs)**

Name of the player: Kumara

Choice on playing: Yes

Kumara’s 1st choice: Scissors

Choice on playing: Yes

Kumara’s 2nd choice: Gun 🡪 A choice which is not available in the choice list

Choice on playing: Yes

Kumara’s 3rd choice: Rock

Choice on playing: Yes

Kumara’s 4thchoice: Paper

Choice on playing: Yes

Kumara’s 5st choice: Scissors

Choice on playing: Yes

Kumara’s 6th choice: Rock

Choice on playing: Yes

Kumara’s 8th choice: Paper

Choice on playing: Yes

Kumara’s 9th choice: Scissors

Choice on playing: Yes

Kumara’s 10th choice: Paper

Choice on playing: No

**Expected output**

Hello!

These are the rules and the regulations of the game! MAKE SURE YOU FOLLOW THEM!!

You should enter your choice as 'Rock','Paper','Scissors'

Please type the first letter of your choice in capital

The wins by the computer, the total wins by you, total number of draws and the total number of games played will be displayed at the end

Always remember, That Scissors wins Over Paper

Paper wins over Rock

rock wins over scissors

If the game is a tie You will have to play again

You will have the opportunity to exit the game at any time you want!!

let's get started!!

Good Luck!!!! Have Fun!!!

Prompting the choice on playing

Prompting the choice from the choice list

Displaying the result as a message and

Wins by the human:

Wins by the computer:

Total number of draws:

Total Number of Games played:

Do you want to play again? (Yes/No): **after each game**

And an update in the database after refreshing it as below

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PlayerName | HumanPlayer\_wins | computer\_wins | Total\_draws | Total\_game\_plays |
| Kumara | 3 | 5 | 1 | 9 |

**Actual output**

Hello! Enter your Name:Kumara

Hello!

These are the rules and the regulations of the game! MAKE SURE YOU FOLLOW THEM!!

You should enter your choice as 'Rock','Paper','Scissors'

Please type the first letter of your choice in capital

The wins by the computer, the total wins by you, total number of draws and the total number of games played will be displayed at the end

Always remember,That Scissors wins Over Paper

Paper wins over Rock

rock wins over scissors

If the game is a tie You will have to play again

You will have the opportunity to exit the game at anytime you want!!

let's get started!!

Good Luck!!!! Have Fun!!!

Do you want to play ? (Yes/No): Yes

Enter your choice here(Rock,Paper,Scissors): Scissors

YOU and the Computer chose the same,Game TIED

Enter your choice here(Rock,Paper,Scissors): Gun

Wins by the human: 0

Wins by the computer: 0

Total number of draws: 1

Total Number of Games played:1

Do you want to play again? (Yes/No): Yes

Enter your choice here(Rock,Paper,Scissors): Rock

YOU and the Computer chose the same,Game TIED

Enter your choice here(Rock,Paper,Scissors): Paper

You chose Paper and the Computer chose Scissors, you Lose!

Wins by the human: 0

Wins by the computer: 1

Total number of draws: 2

Total Number of Games played:3

Do you want to play again? (Yes/No): Yes

Enter your choice here(Rock,Paper,Scissors): Scissors

You chose Scissors and the Computer chose Paper, You WON!

Wins by the human: 1

Wins by the computer: 1

Total number of draws: 2

Total Number of Games played:4

Do you want to play again? (Yes/No): Yes

Enter your choice here(Rock,Paper,Scissors): Rock

You chose Rock and the Computer chose Scissors! You WON!!

Wins by the human: 2

Wins by the computer: 1

Total number of draws: 2

Total Number of Games played:5

Do you want to play again? (Yes/No): Yes

Enter your choice here(Rock,Paper,Scissors): Paper

You chose Paper and the Computer chose Rock You WON!

Wins by the human: 3

Wins by the computer: 1

Total number of draws: 2

Total Number of Games played:6

Do you want to play again? (Yes/No): Yes

Enter your choice here(Rock,Paper,Scissors): Scissors

YOU chose Scissors and the Computer chose Rock, You LOSE!

Wins by the human: 3

Wins by the computer: 2

Total number of draws: 2

Total Number of Games played:7

Do you want to play again? (Yes/No): Yes

Enter your choice here(Rock,Paper,Scissors): Paper

YOU and the Computer chose the same,Game TIED

Enter your choice here(Rock,Paper,Scissors): Paper

YOU and the Computer chose the same,Game TIED

Enter your choice here(Rock,Paper,Scissors): Paper

YOU and the Computer chose the same,Game TIED

Enter your choice here(Rock,Paper,Scissors): Paper

You chose Paper and the Computer chose Scissors, you Lose!

Wins by the human: 3

Wins by the computer: 3

Total number of draws: 5

Total Number of Games played:11

Do you want to play again? (Yes/No): No

Marks are Tied

Score of the computer: 3

Your Score: 3

No. of Wins by You: 3

No. of Wins by the computer: 3

Total number of draws: 5

Total number of games played:11

+------------+------------------+---------------+-------------+------------------+

| PlayerName | HumanPlayer\_wins | computer\_wins | Total\_draws | Total\_game\_plays |

+------------+------------------+---------------+-------------+------------------+

+------------+------------------+---------------+-------------+------------------+

>>>

**Description**

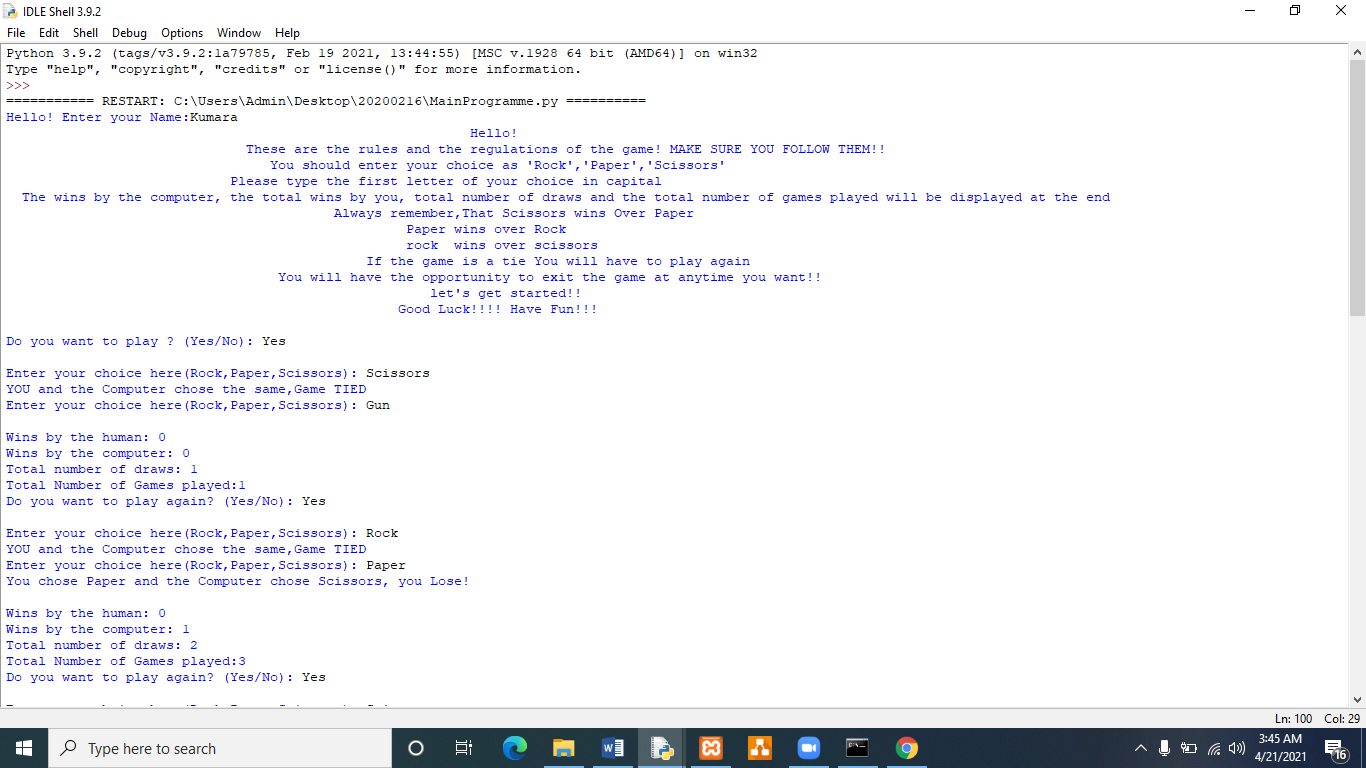
The result of the game is 100% similar to the expected output. The resulting message is highlighted. Finally the game lay history is visible to the human player. Kumara the human player has Gun as the 2nd choice, the programme smoothly allows the user to enter the next choice without any error

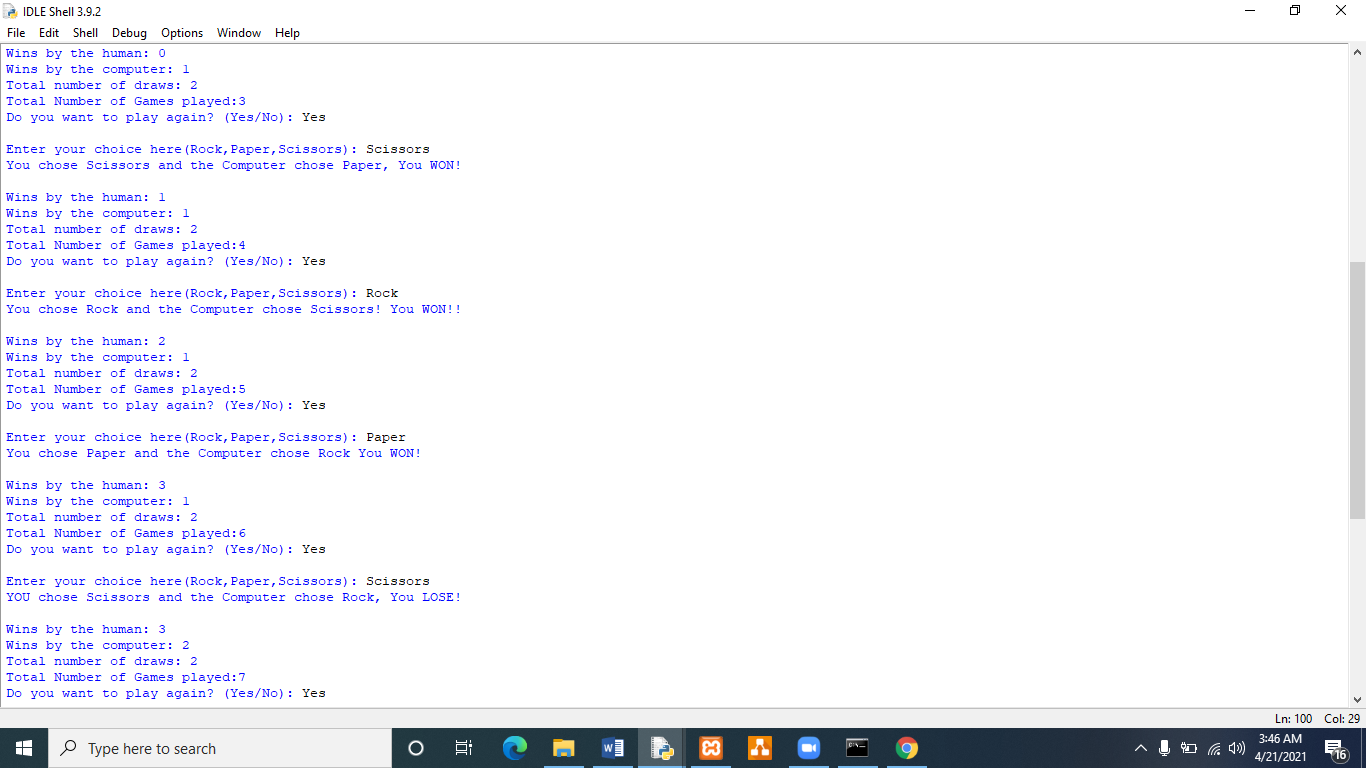
**Remarks**

Test case pass

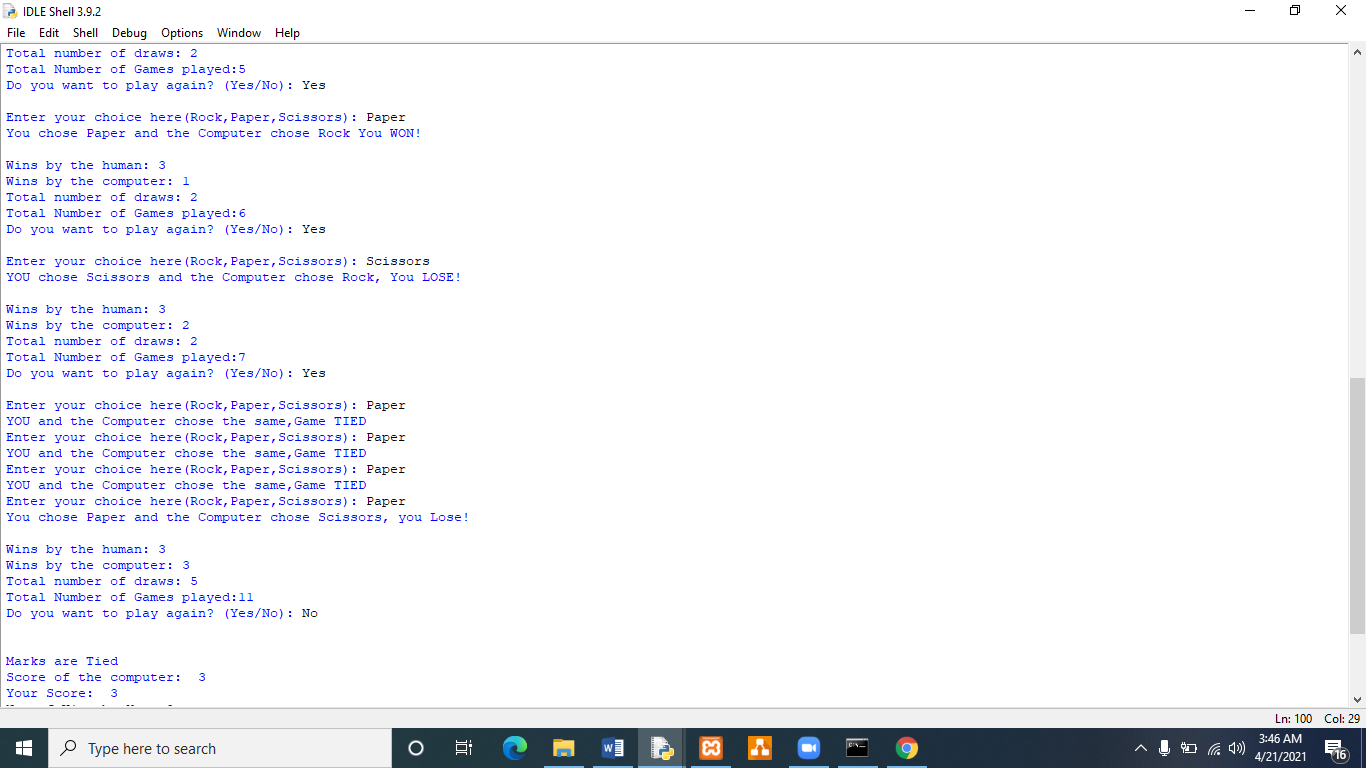
**Screen shot of Test Case 01**

**Figure 15| Screen shot of test case one**

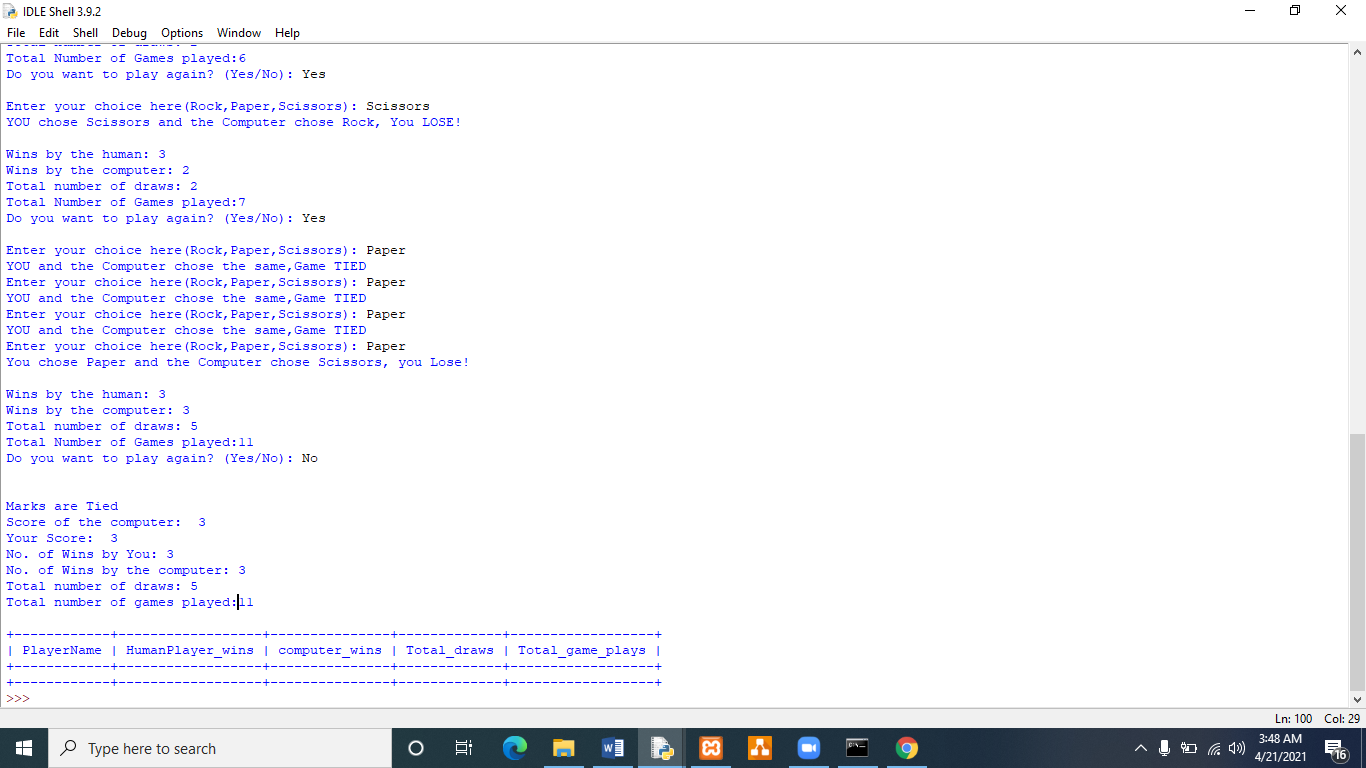




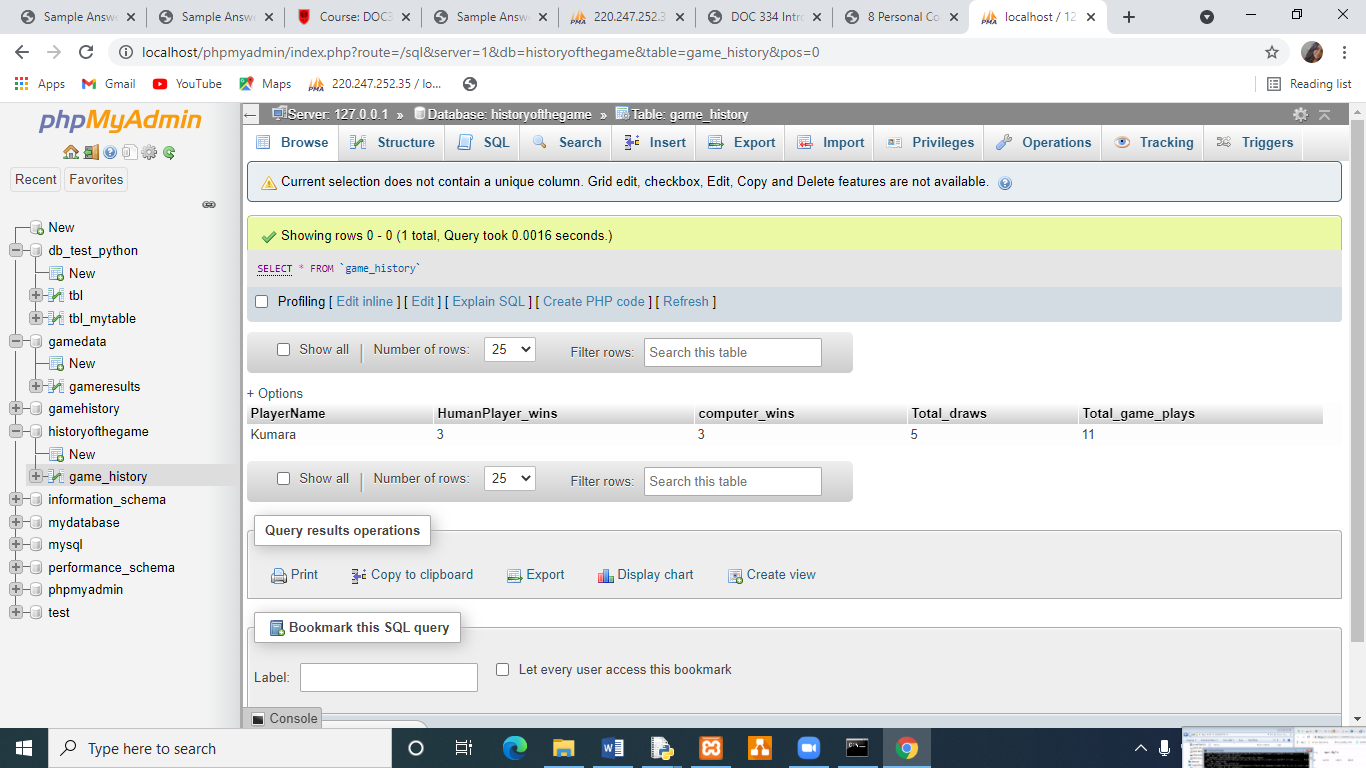
**Figure 16| Contd.**



**Figure 17| Contd.**



**Figure 18| Test case 01 Contd.**



**Figure 19| New record of data is added to the database**

**Test Case 02(Inputs)**

Human Player’s name: Sandamali

1st Choice: Rock

2nd Choice:Paper

3rd Choice: Scissors

**Expected output**

Hello!

These are the rules and the regulations of the game! MAKE SURE YOU FOLLOW THEM!!

You should enter your choice as 'Rock','Paper','Scissors'

Please type the first letter of your choice in capital

The wins by the computer, the total wins by you, total number of draws and the total number of games played will be displayed at the end

Always remember, That Scissors wins Over Paper

Paper wins over Rock

rock wins over scissors

If the game is a tie You will have to play again

You will have the opportunity to exit the game at any time you want!!

let's get started!!

Good Luck!!!! Have Fun!!!

Prompting the choice on playing

Prompting the choice from the choice list

Displaying the result as a message and

Wins by the human:

Wins by the computer:

Total number of draws:

Total Number of Games played:

Do you want to play again? (Yes/No): **after each game**

And an update in the database after refreshing it as below

**Actual Output**

Hello! Enter your Name:Sandamali

Hello!

These are the rules and the regulations of the game! MAKE SURE YOU FOLLOW THEM!!

You should enter your choice as 'Rock','Paper','Scissors'

Please type the first letter of your choice in capital

The wins by the computer, the total wins by you, total number of draws and the total number of games played will be displayed at the end

Always remember,That Scissors wins Over Paper

Paper wins over Rock

rock wins over scissors

If the game is a tie You will have to play again

You will have the opportunity to exit the game at anytime you want!!

let's get started!!

Good Luck!!!! Have Fun!!!

Do you want to play ? (Yes/No): Yes

Enter your choice here(Rock,Paper,Scissors): Rock

You chose Rock and the Computer chose Paper,YOU LOSE!

Wins by the human: 0

Wins by the computer: 1

Total number of draws: 0

Total Number of Games played:1

Do you want to play again? (Yes/No): Yes

Enter your choice here(Rock,Paper,Scissors): Paper

YOU and the Computer chose the same,Game TIED

Enter your choice here(Rock,Paper,Scissors): Paper

You chose Paper and the Computer chose Rock You WON!

Wins by the human: 1

Wins by the computer: 1

Total number of draws: 1

Total Number of Games played:3

Do you want to play again? (Yes/No): Yes

Enter your choice here(Rock,Paper,Scissors): Scissors

YOU and the Computer chose the same,Game TIED

Enter your choice here(Rock,Paper,Scissors): Scissors

YOU chose Scissors and the Computer chose Rock, You LOSE!

Wins by the human: 1

Wins by the computer: 2

Total number of draws: 2

Total Number of Games played:5

Do you want to play again? (Yes/No): No

Bad luck! Computer won!

Score of the computer: 2

Your Score: 1

No. of wins by you: 1

No. of Wins by the computers: 2

Total number of draws: 2

Total number of games played:5

+------------+------------------+---------------+-------------+------------------+

| PlayerName | HumanPlayer\_wins | computer\_wins | Total\_draws | Total\_game\_plays |

+------------+------------------+---------------+-------------+------------------+

| Kumara | 3 | 3 | 5 | 11 |

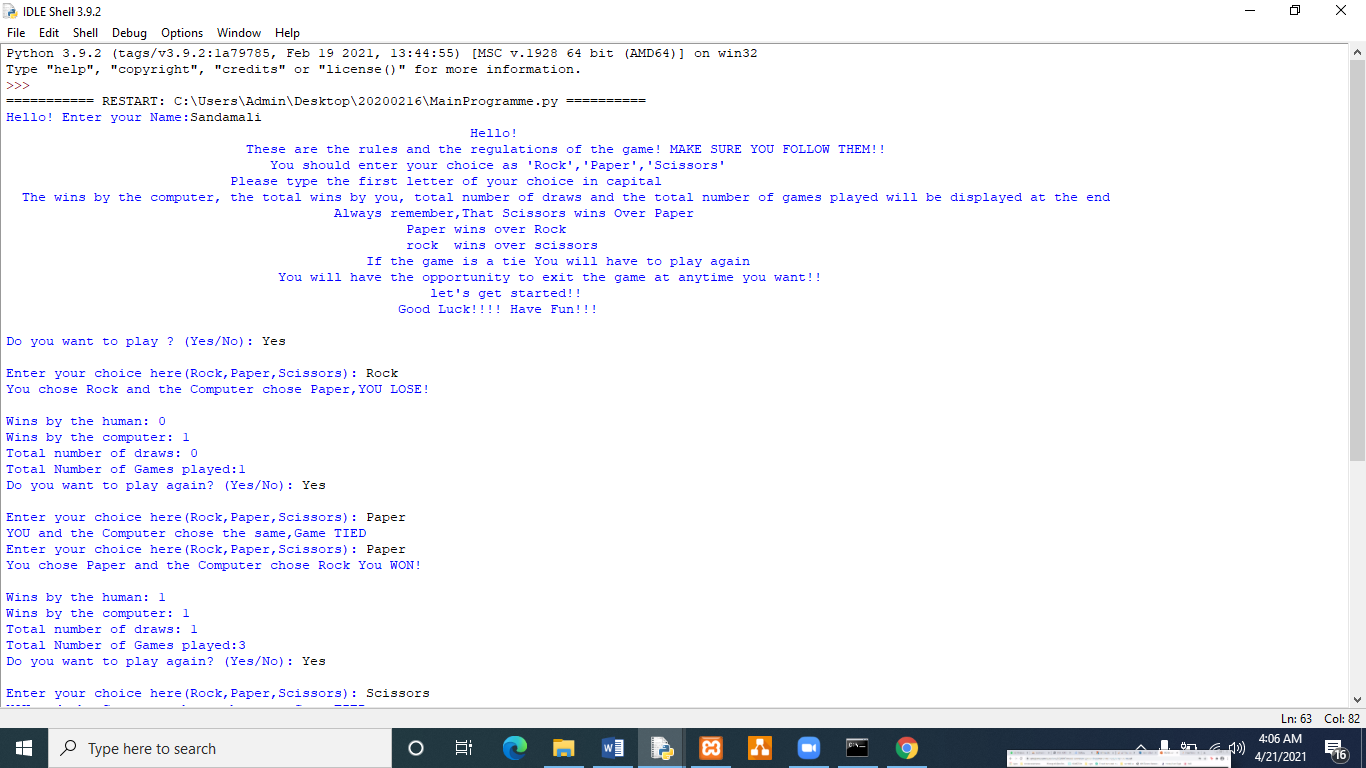
+------------+------------------+---------------+-------------+------------------+

>>>

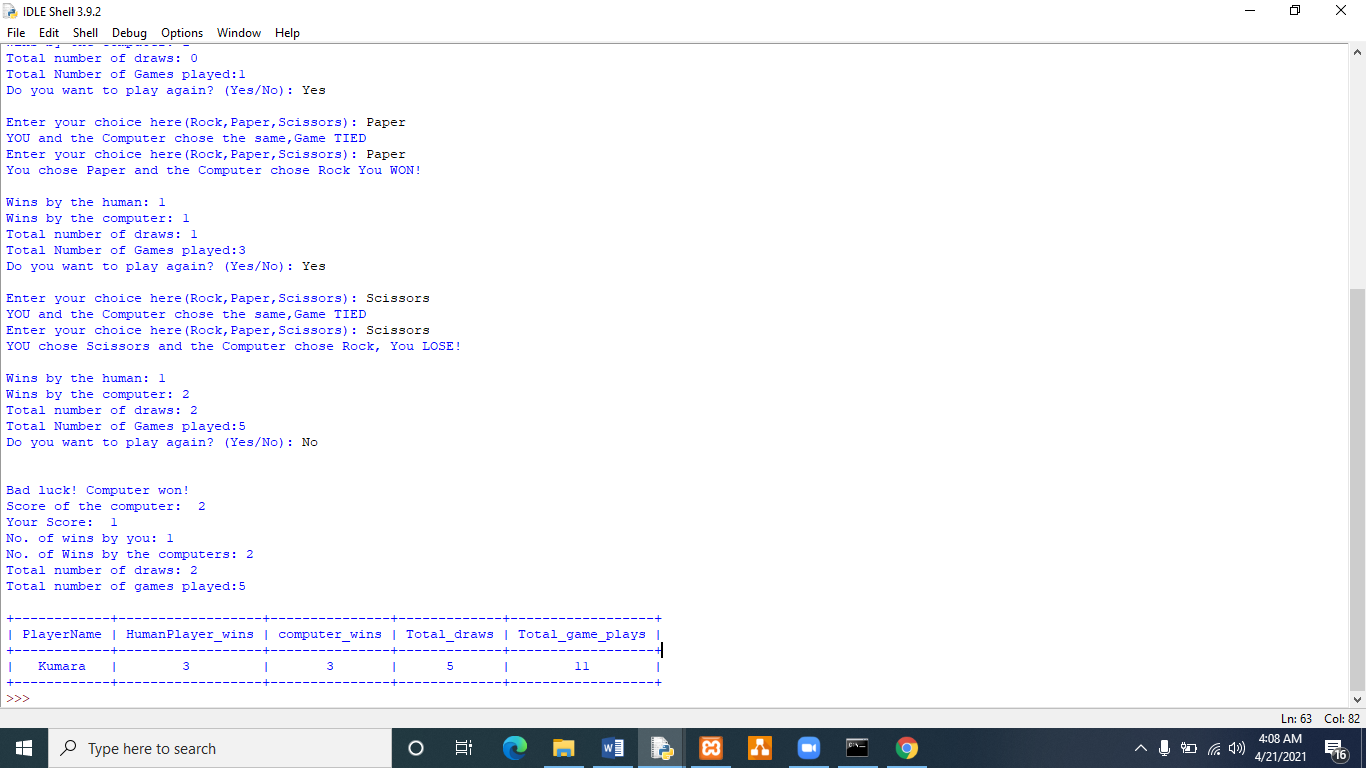
**Description**

Paper wins over Rock, Paper wins over Rock and Rock wins over Scissors. The result of the game is 100% similar to the expected output. The resulting message is highlighted. Finally the game lay history is visible to the human player.

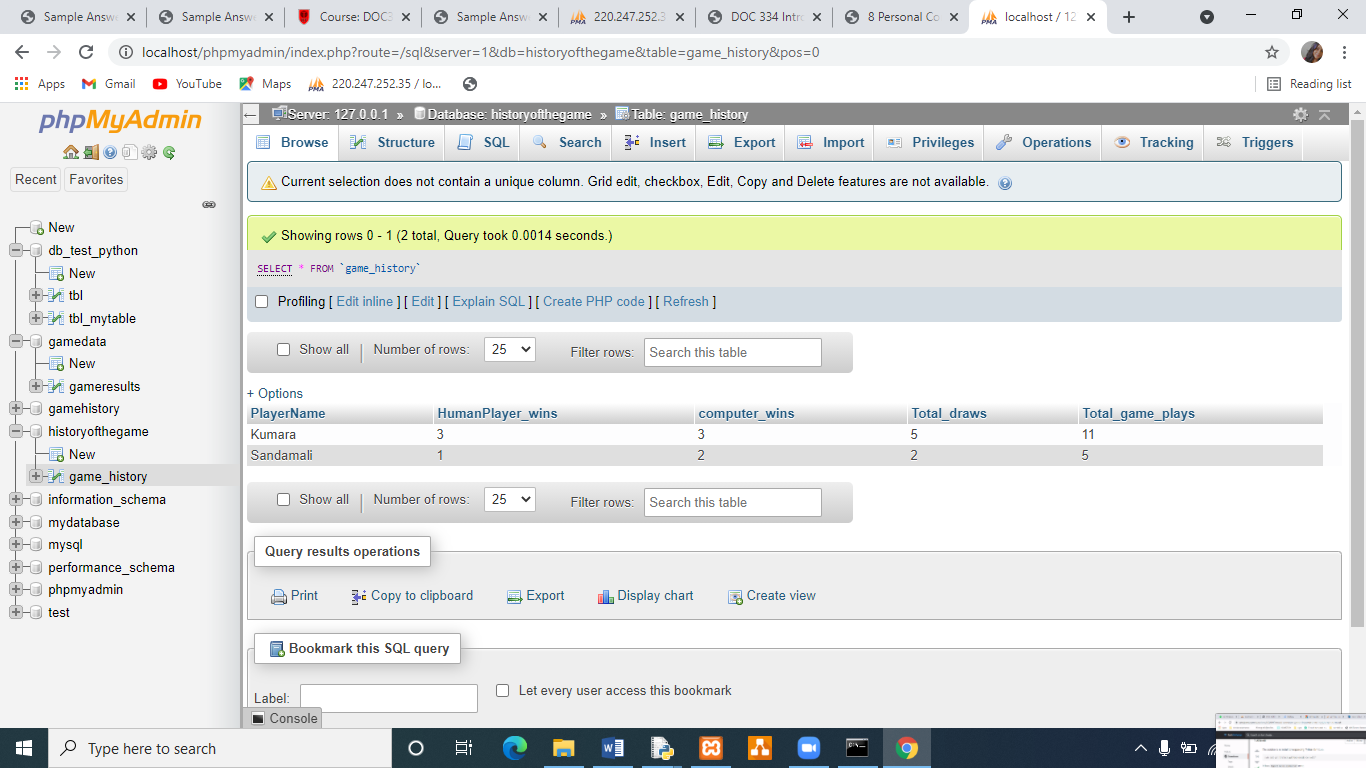
**Remarks**

Test case pass

**Figure 20| Test case 2**



**Figure 21| Test case 2 Contd.**



**Figure 22| Test Case 2 Data base update with new record.**

**Test Case 03**

(inputs)

Human Player Name: Hiranya Fernando

1st choice =Paper

2nd choice = Scissors

3rd choice =Paper

**Expected output**

Hello!

These are the rules and the regulations of the game! MAKE SURE YOU FOLLOW THEM!!

You should enter your choice as 'Rock','Paper','Scissors'

Please type the first letter of your choice in capital

The wins by the computer, the total wins by you, total number of draws and the total number of games played will be displayed at the end

Always remember, That Scissors wins Over Paper

Paper wins over Rock

rock wins over scissors

If the game is a tie You will have to play again

You will have the opportunity to exit the game at any time you want!!

let's get started!!

Good Luck!!!! Have Fun!!!

Prompting the choice on playing

Prompting the choice from the choice list

Displaying the result as a message and

Wins by the human:

Wins by the computer:

Total number of draws:

Total Number of Games played:

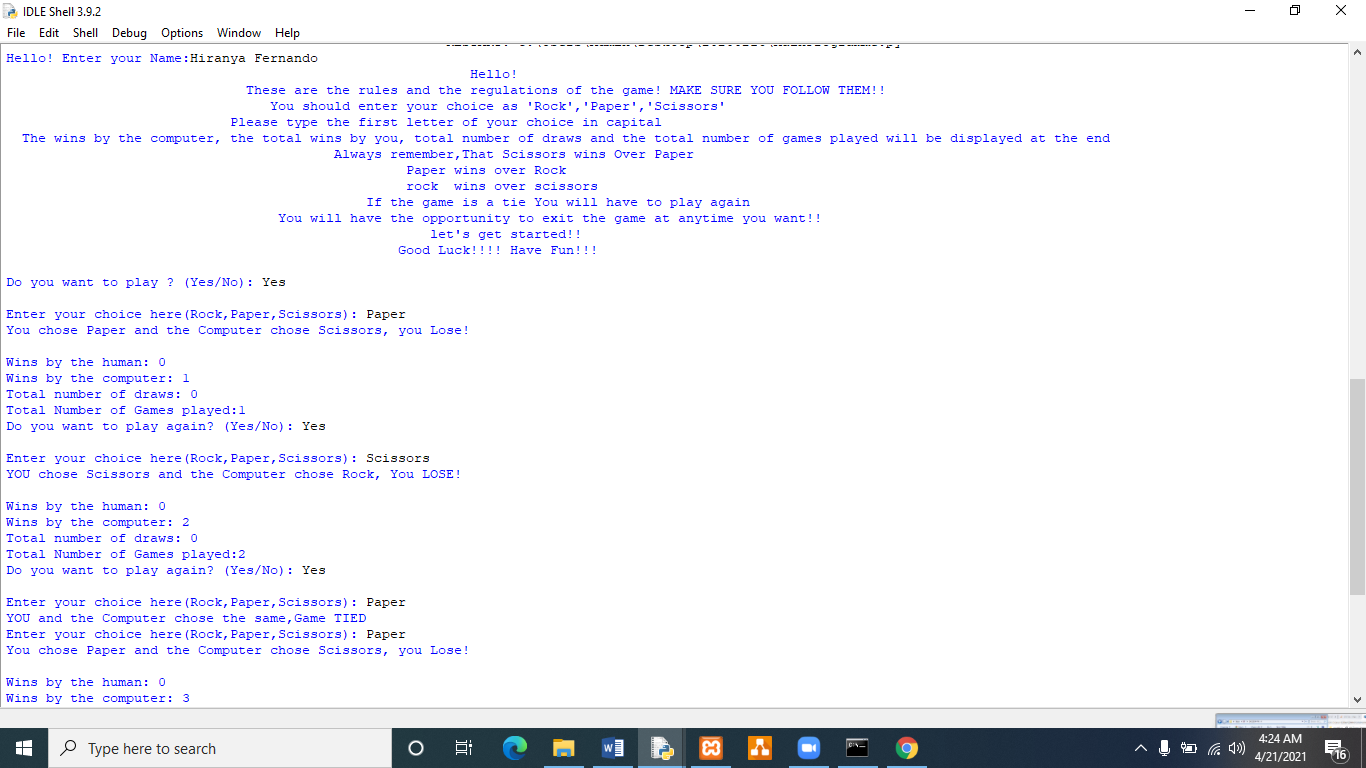
Do you want to play again? (Yes/No): **after each game**

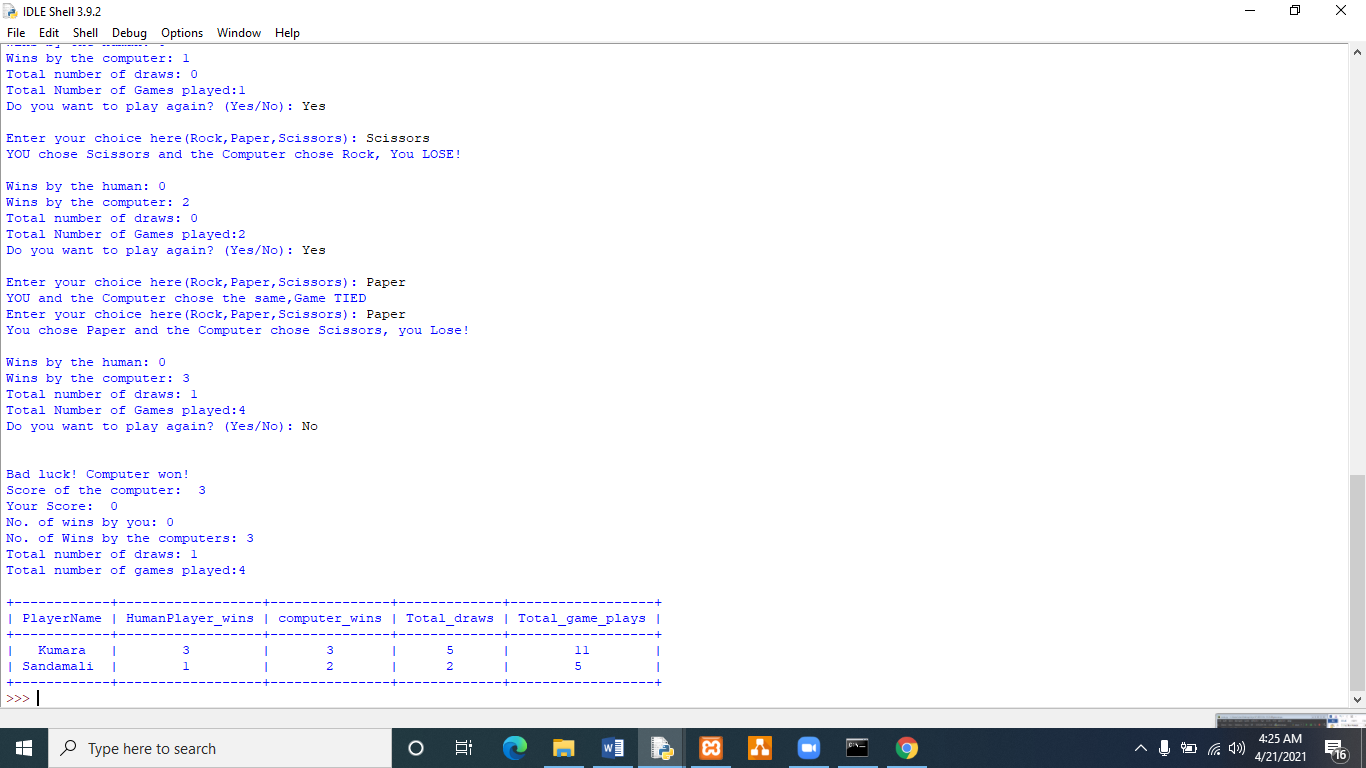
Updating the database and the HTML.html file

Displaying the past game play history to the player

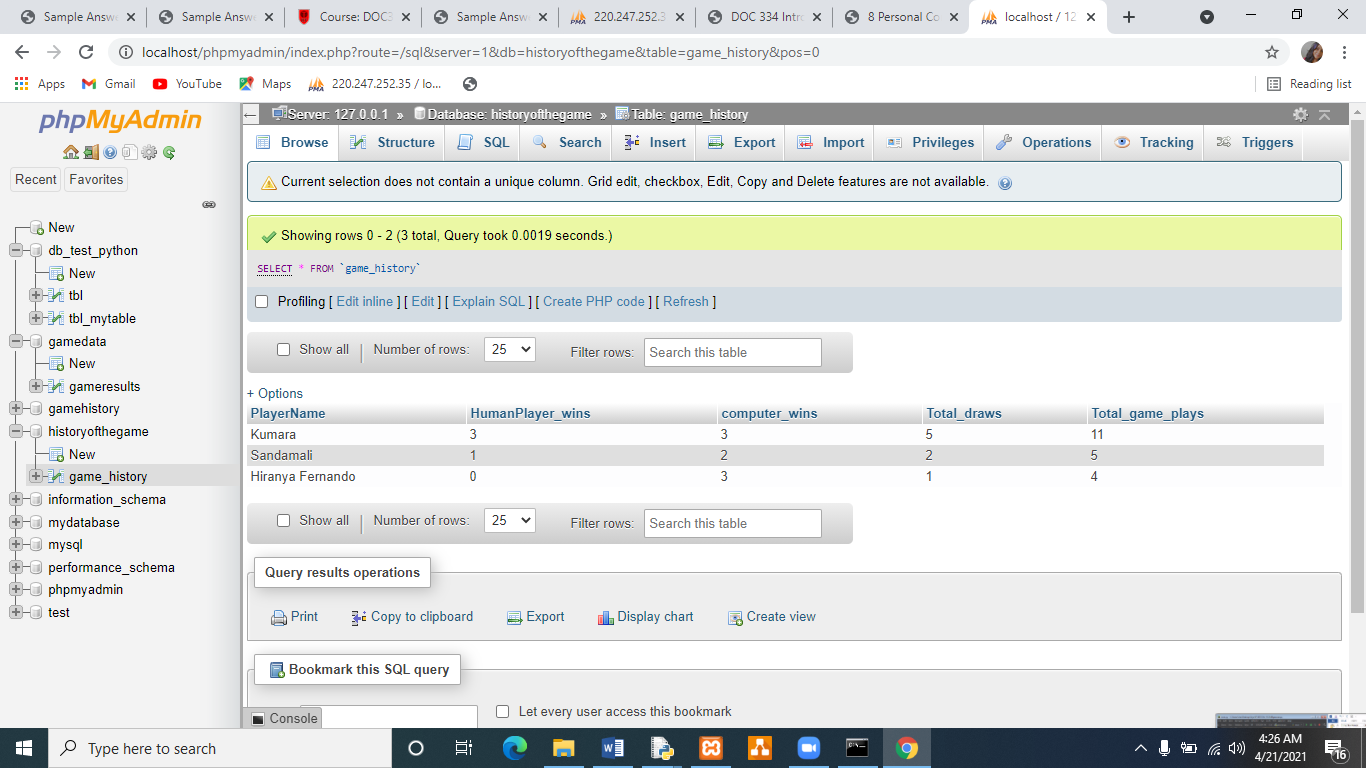
**Actual Output**

**Figure 23| Screen shot of Test Case 03**

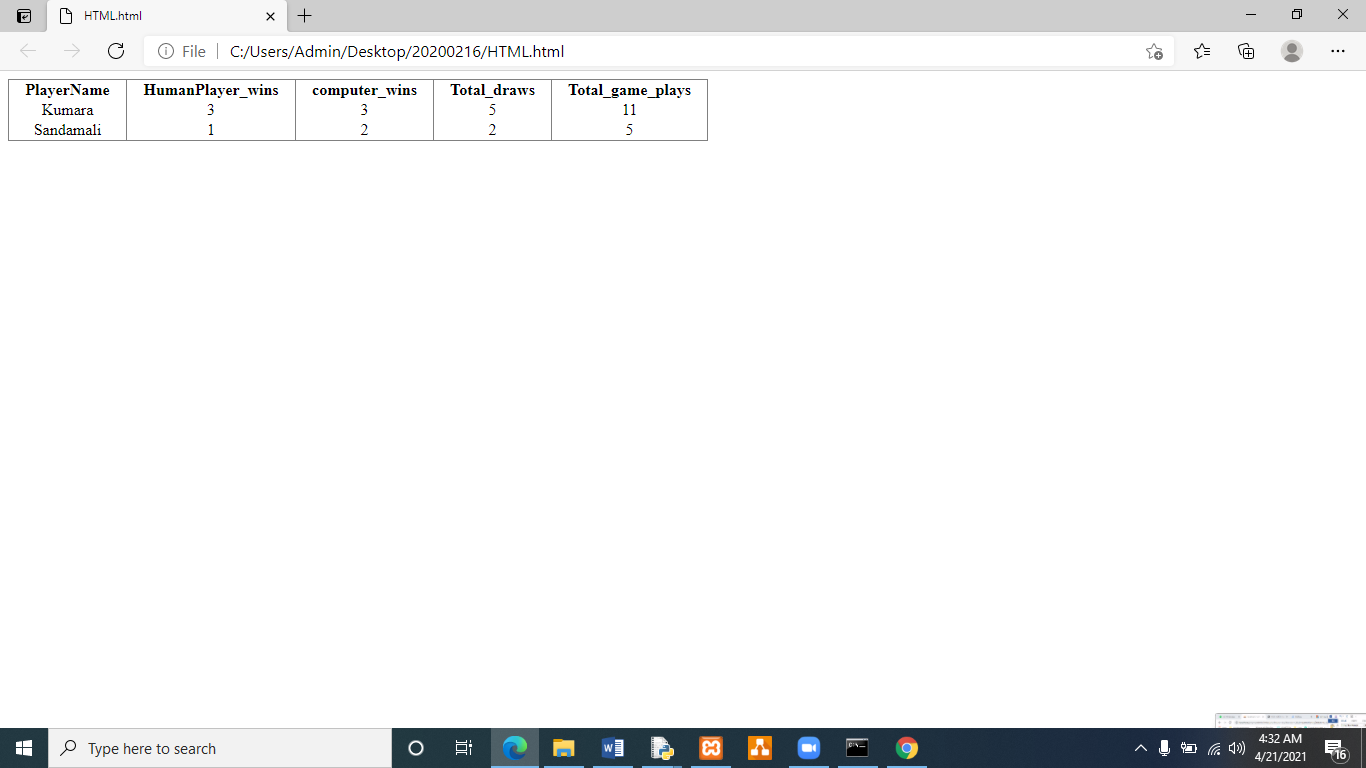




**Figure 24| Screen shots of test case 3 Contd.**



**Figure 25| Screen shot of the database (Updated with a new record)**



**Figure 26|Updates the HTML.html file**

**Description**

Scissors wins over paper, Rock wins over Scissors and Scissors wins over paper. Determines the winner and updates the database. Allows the user to check the past game play history and also updates the HTML.html file

**Remarks**

Test Case Pass

**Test Case 04**

Human Player’s Name: Peter Gunaratne

1st choice: Rock

2nd Choice: Rock

3rd Choice: Rock

**Expected output**

Hello!

These are the rules and the regulations of the game! MAKE SURE YOU FOLLOW THEM!!

You should enter your choice as 'Rock','Paper','Scissors'

Please type the first letter of your choice in capital

The wins by the computer, the total wins by you, total number of draws and the total number of games played will be displayed at the end

Always remember, That Scissors wins Over Paper

Paper wins over Rock

rock wins over scissors

If the game is a tie You will have to play again

You will have the opportunity to exit the game at any time you want!!

let's get started!!

Good Luck!!!! Have Fun!!!

Prompting the choice on playing

Prompting the choice from the choice list

Displaying the result as a message and

Wins by the human:

Wins by the computer:

Total number of draws:

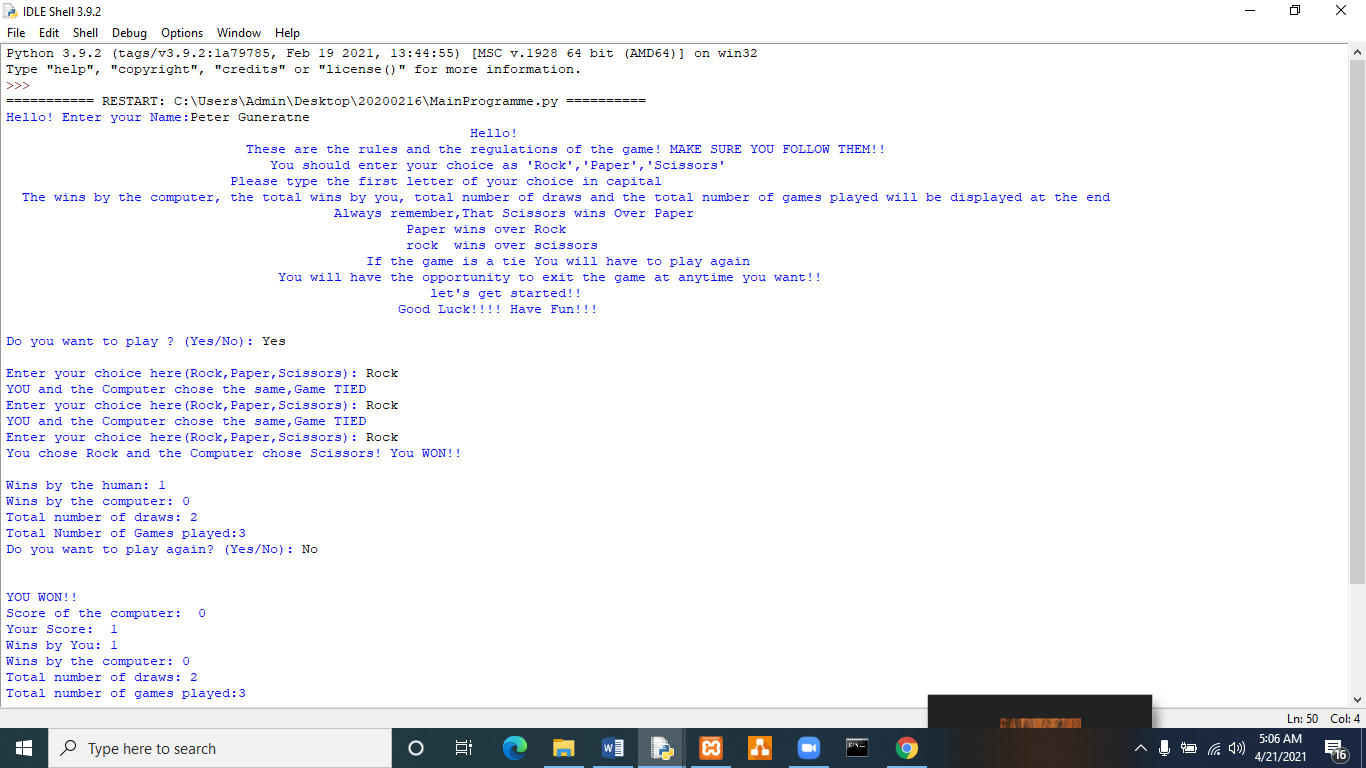
Total Number of Games played:

Do you want to play again? (Yes/No): **after each game**

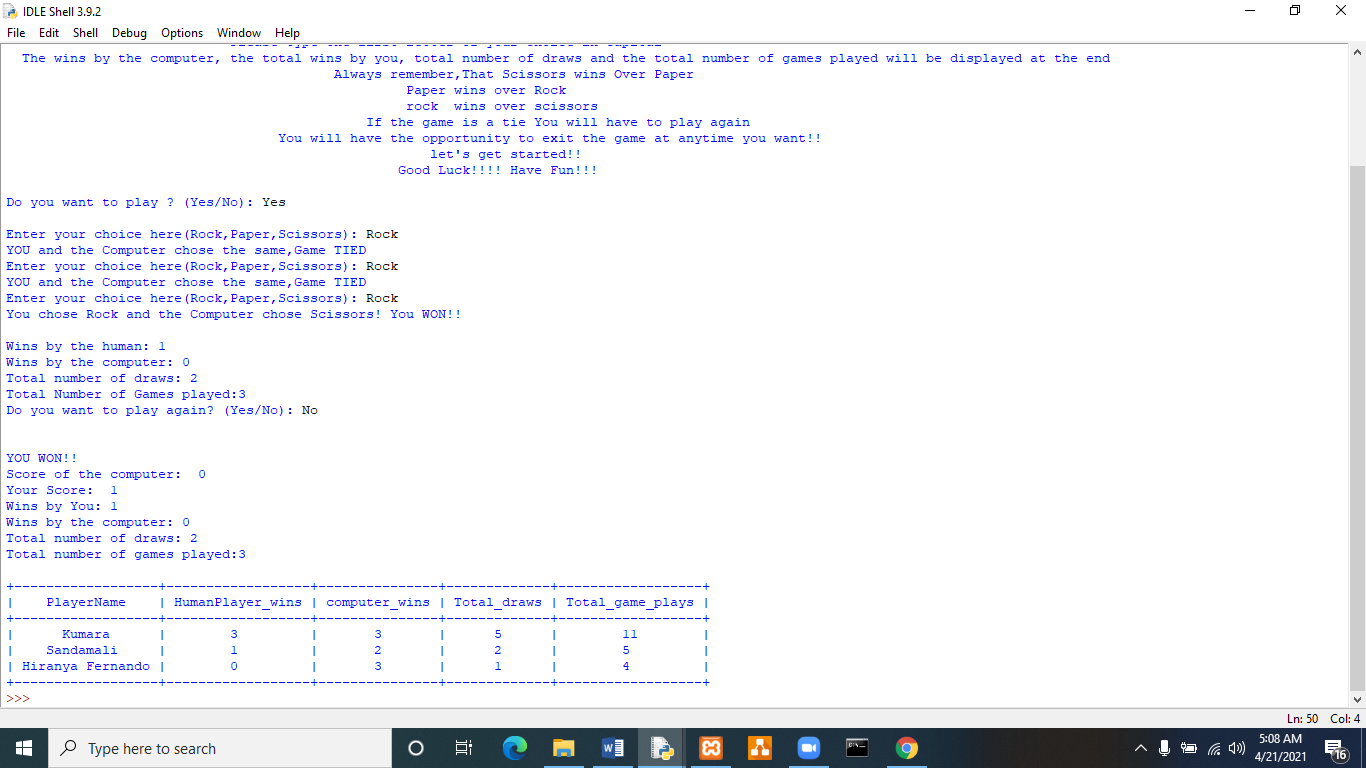
Updating the database and the HTML.html file

Displaying the past game play history to the player

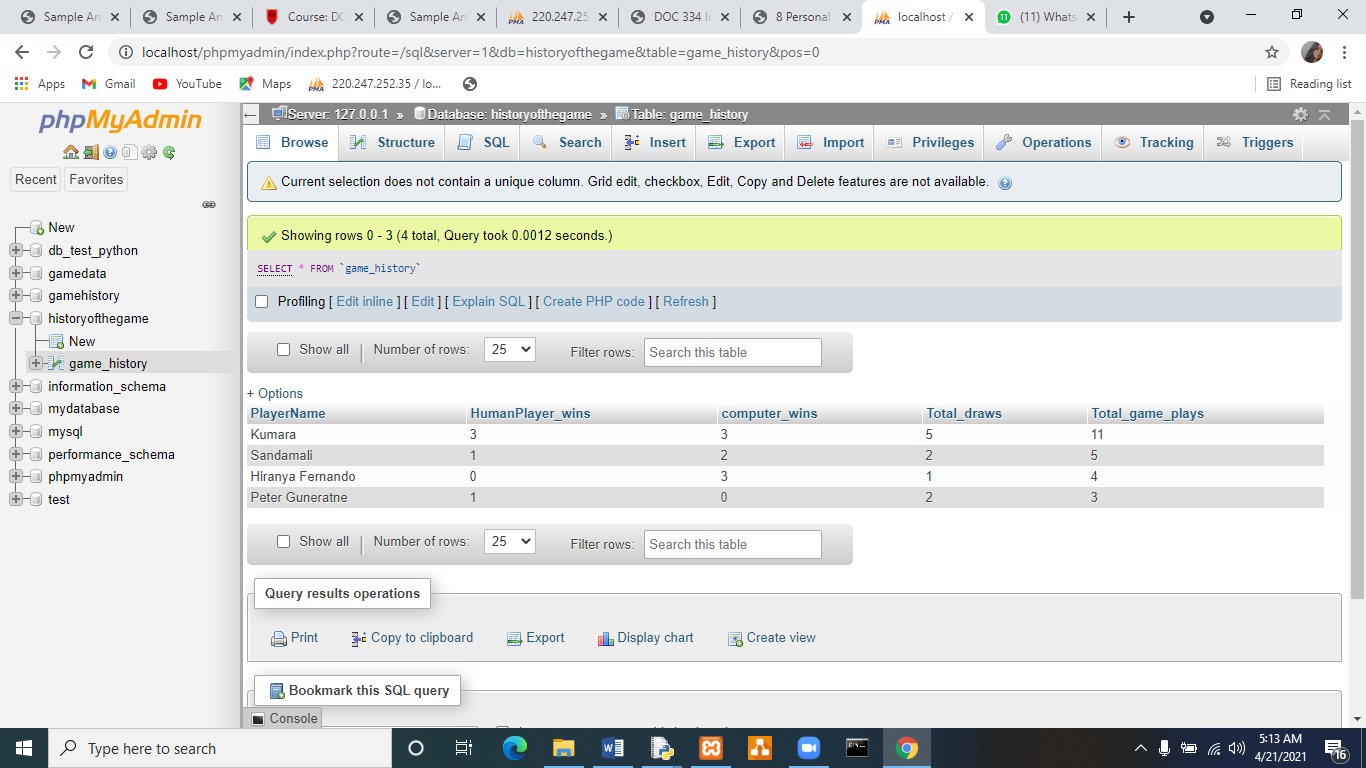
**Actual output**



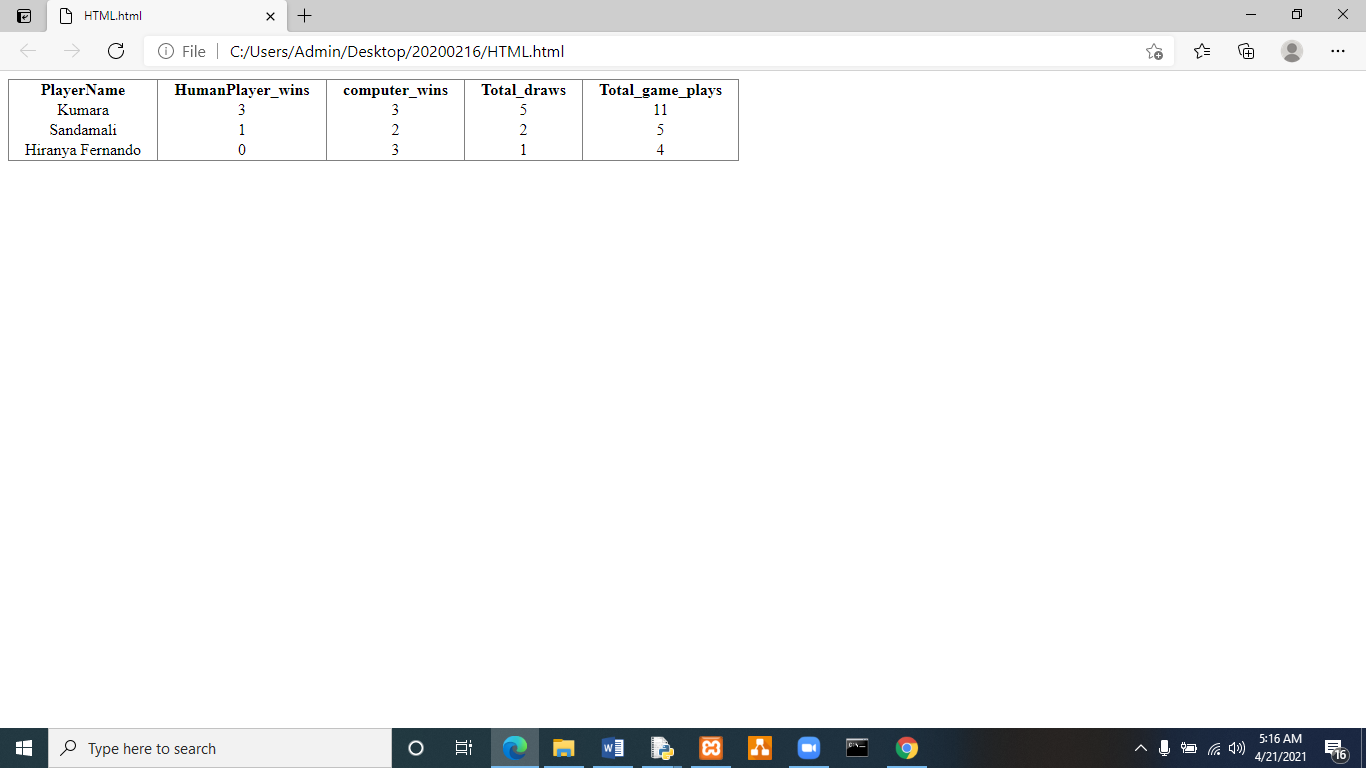
**Figure 27| Test Case 04 Actual output**



**Figure 28| Test Case 04 Actual output**



**Figure 29| Data base is updated with the new record**



**Figure 30| HTML.html file -updated**

**Description**

Rock wins over scissors. The programme determines the winner and updates the database. Allows the user to check the past game play history and also updates the HTML.html file.

**Remarks**

Test case pass

**Test Case 05**

Human Player’s Name: Devdun Tissera

1st choice: Rock

2nd choice: Paper

3rd choice: Scissors

4th choice; Scissors

5th choice: Paper

6th choice: Rock

**Expected output**

Hello!

These are the rules and the regulations of the game! MAKE SURE YOU FOLLOW THEM!!

You should enter your choice as 'Rock','Paper','Scissors'

Please type the first letter of your choice in capital

The wins by the computer, the total wins by you, total number of draws and the total number of games played will be displayed at the end

Always remember, That Scissors wins Over Paper

Paper wins over Rock

rock wins over scissors

If the game is a tie You will have to play again

You will have the opportunity to exit the game at any time you want!!

let's get started!!

Good Luck!!!! Have Fun!!!

Prompting the choice on playing

Prompting the choice from the choice list

Displaying the result as a message and

Wins by the human:

Wins by the computer:

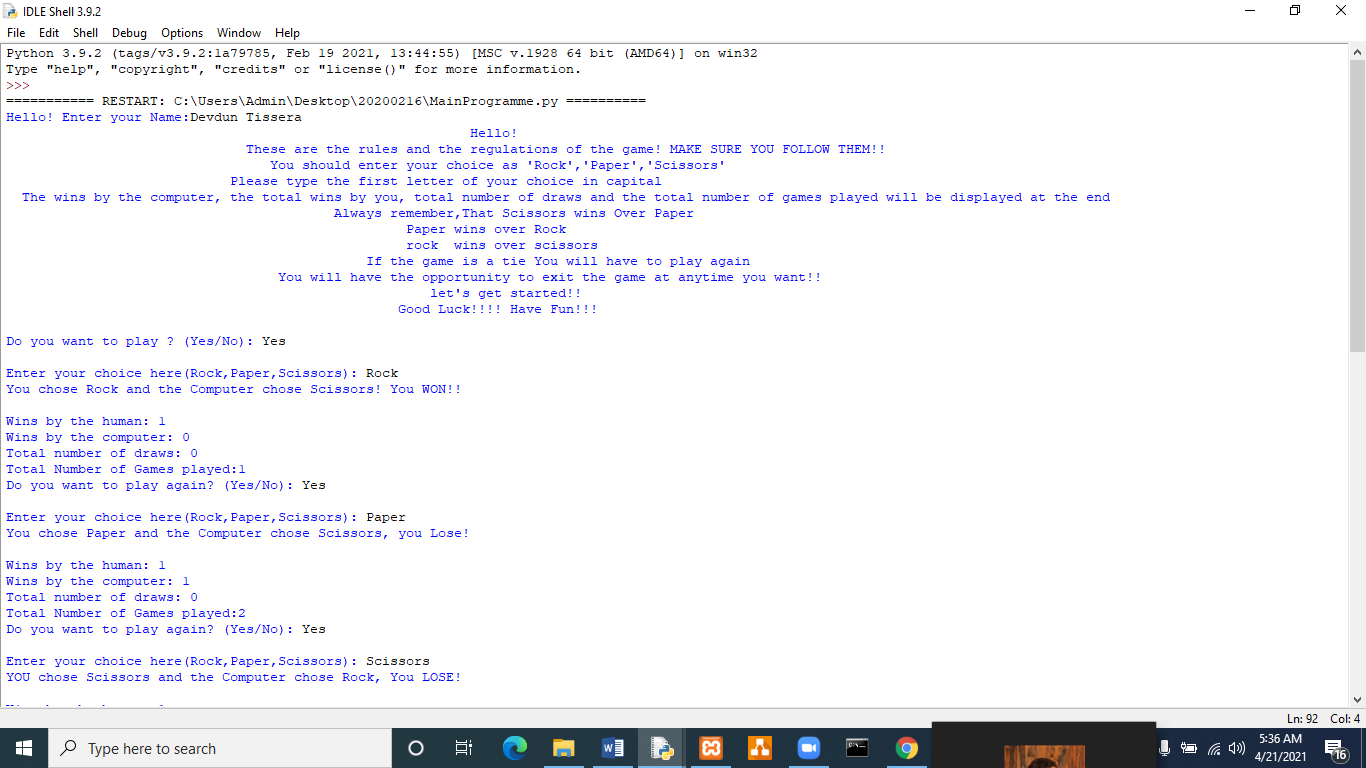
Total number of draws:

Total Number of Games played:

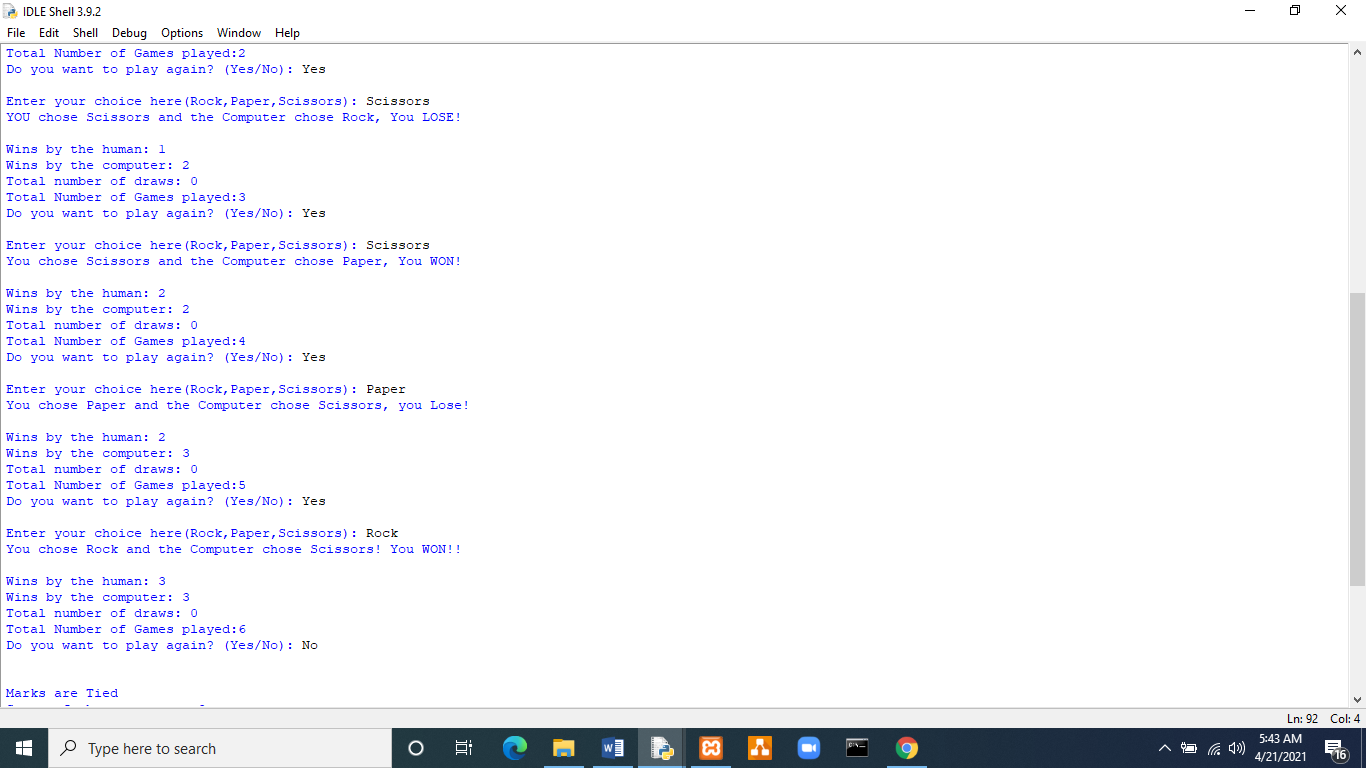
Do you want to play again? (Yes/No): **after each game**

Updating the database and the HTML.html file

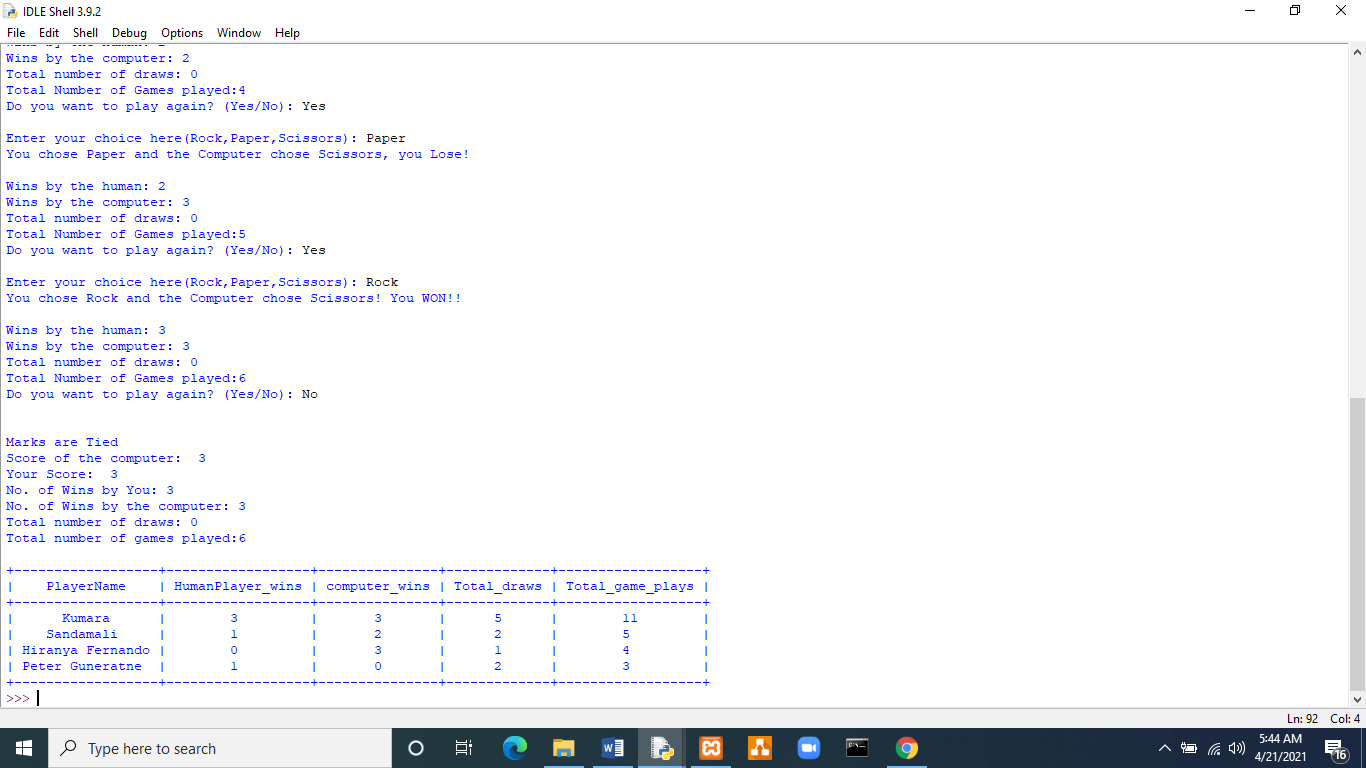
Displaying the past game play history to the player



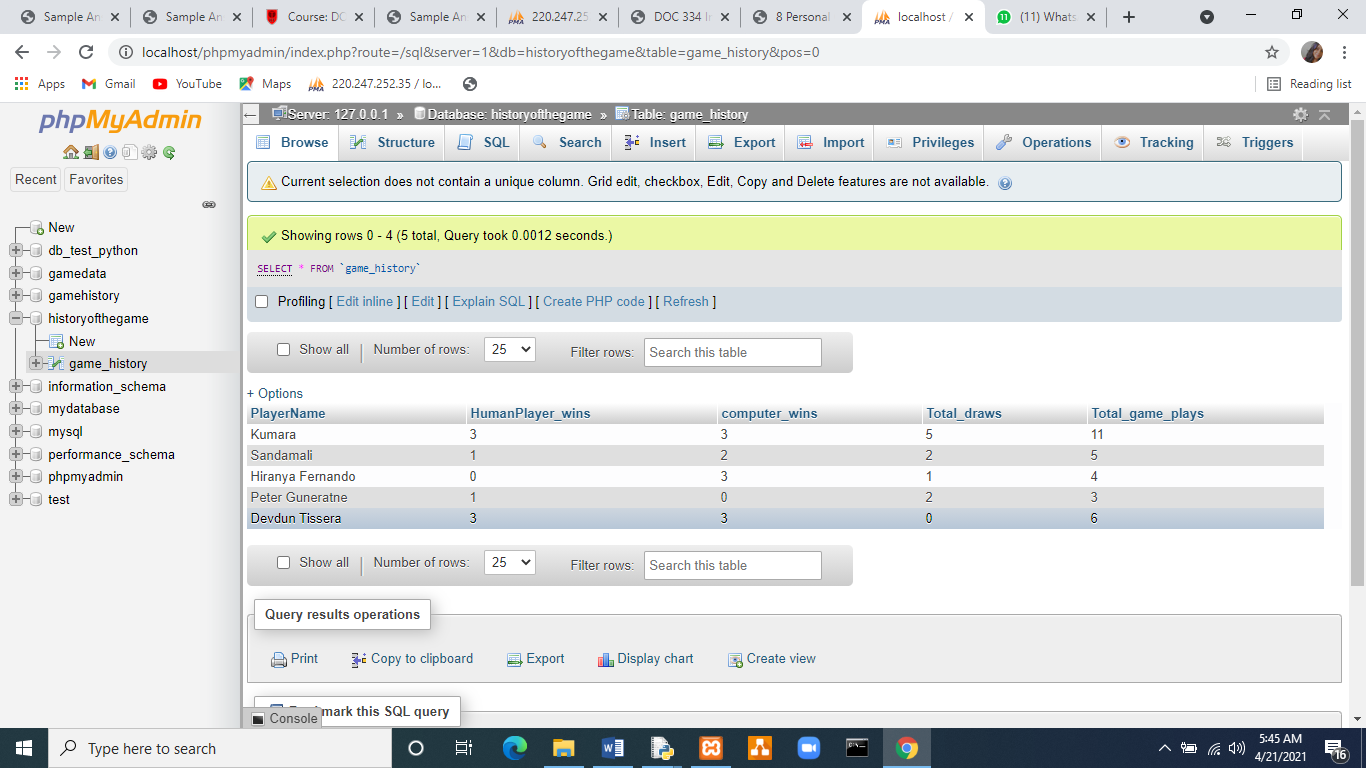
**Figure 31| Test Case 5**



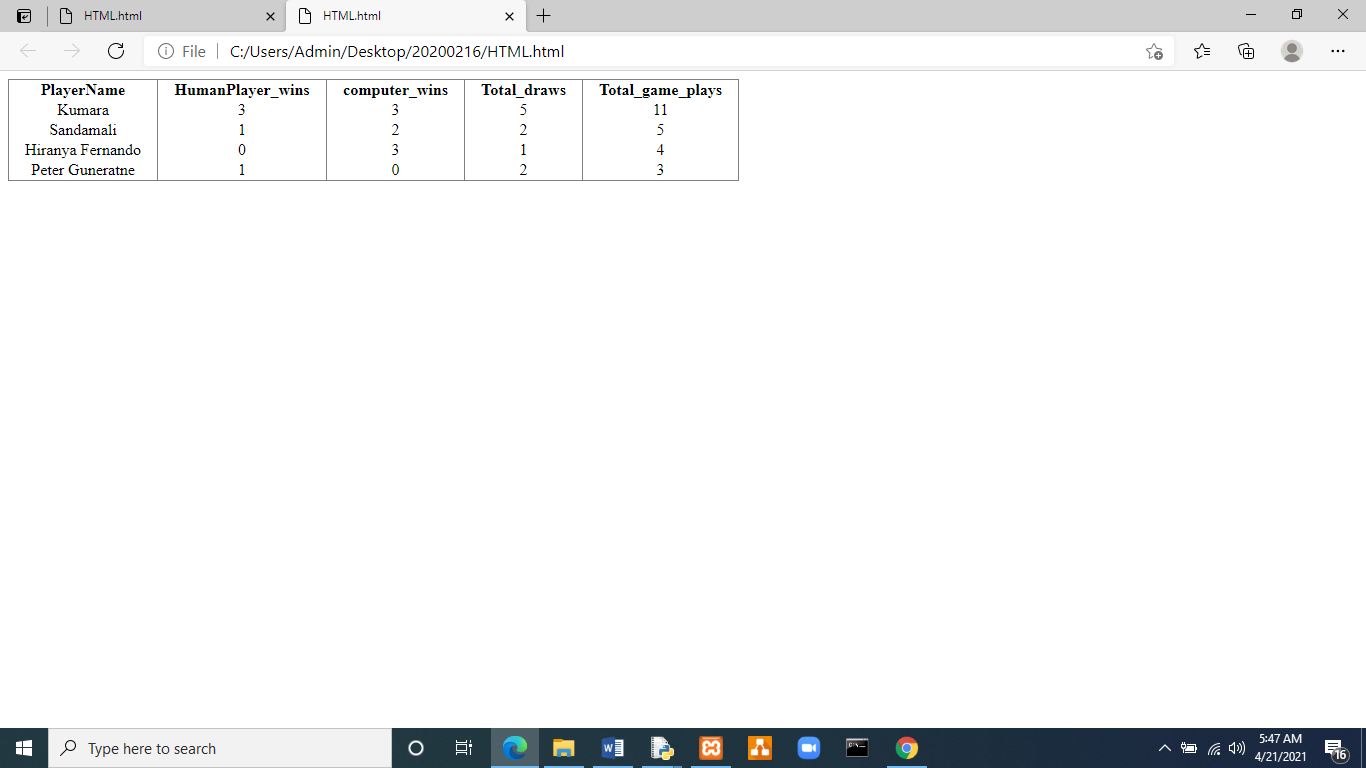
**Figure 32| Test Case 5 Contd.**



**Figure 33| Test Case 05 Contd. (Actual output)**



**Figure 34| The updated database**



**Figure 35| The updated HTML.html file**

**Description**

Rock wins over scissors. Scissors win over paper, Rock wins over Scissors and Scissors win over Paper. The programme determines the winner and updates the database. Allows the user to view the past game play history and also updates the HTML.html file. This test case is a Game play without draw games.

**Remarks**

Test case pass

# The Programme in the Console...