IN LAB

LAB TASK 1:

```
phonebook ={}

phonebook["Ali"] = {"Phone" : "03056663224",
            "Email" : "ali@gmail.com" }
phonebook["bilal"] = {"Phone" : "0307666555",
            "Email" : "bilal@gmail.com" }
phonebook["ahmad"] = {"Phone" : "03036333333",
            "Email" : "ahmad@gmail.com" }

print(phonebook)
```

```
MINUSX/.spyder-py3')

In [2]: runfile('C:/Users/MINUSX/.spyder-py3/guessgame.py', wdir='C:/Users/
MINUSX/.spyder-py3')
{'Ali': {'Phone': '03056663224', 'Email': 'ali@gmail.com'}, 'Zain': {'Phone':
'0307666555', 'Email': 'bilal@gmail.com'}, 'Shoaib': {'Phone': '03036333333',
'Email': 'ahmad@gmail.com'}}

In [3]:
```

LAB TASK 2:

```
phonebook ={}

phonebook["Ali"] = {"Phone" : "03053533224",
            "Email" : "ali@gmail.com" }
phonebook["bilal"] = {"Phone" : "03021466555",
            "Email" : "bilal@gmail.com" }
phonebook["ahmad"] = {"Phone" : "0303634533",
            "Email" : "ahmad@gmail.com" }

print(phonebook)
for name, record in phonebook.items():

    print("{}'s phone number is {}, and email is {}".format(name, record ["Phone"],
record["Email"]))
```

```
bilal's phone number is 0307666555, and email is bilal@gmail.com
ahmad's phone number is 03036333333, and email is ahmad@gmail.com

In [4]: runfile('C:/Users/MINUSX/.spyder-py3/guessgame.py', wdir='C:/Users/
MINUSX/.spyder-py3')
{'Ali': {'Phone': '03053533224', 'Email': 'ali@gmail.com'}, 'bilal': {'Phone':
'03021466555', 'Email': 'bilal@gmail.com'}, 'ahmad': {'Phone': '0303634533',
'Email': 'ahmad@gmail.com'}}
Ali's phone number is 03053533224, and email is ali@gmail.com
bilal's phone number is 03021466555, and email is bilal@gmail.com
ahmad's phone number is 0303634533, and email is ahmad@gmail.com

In [5]:
```

LAB TASK 3:

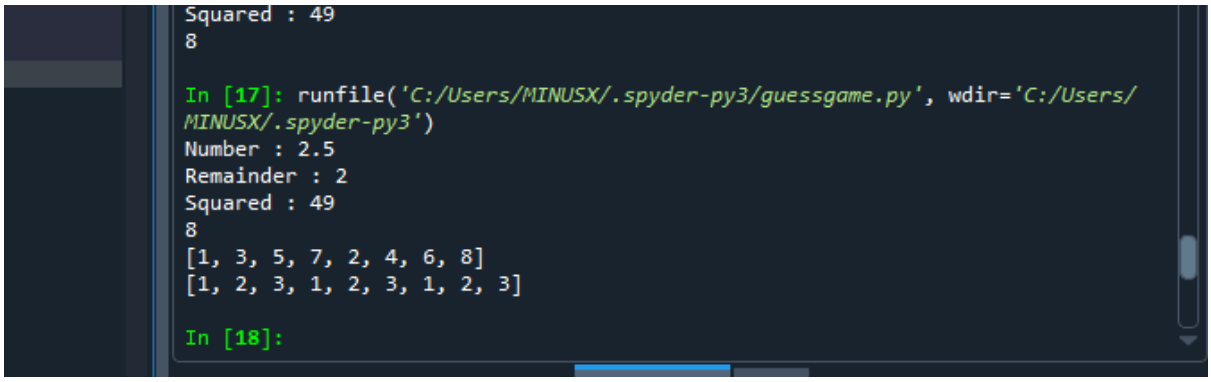
```
phonebook = {}

phonebook [ "John"] = {"Phone": "012 794 794",

"Email":"john@email.com"}

phonebook [ "Jill"] = {"Phone": "012 345 345",

"Email": "jill@email.com"}

phonebook [ "Joss"] = {"Phone": "012 321 321", "Email": "joss@email.com"}
print (phonebook)

del phonebook [ "John"]
for name, record in phonebook.items():

    print("{}'s phone number is {}, and their email is {}".format(name, record[ "Phone"], record[
"Email"]))
# Pop returns the record, and deletes it

jill_record = phonebook.pop("Jill")

print(jill_record)

for name, record in phonebook.items():

    print("{}'s phone number is {}, and their email is {}".format(name, record[ "Phone"], record[
"Email"]))

del phonebook [ "John"]
```

```
In [9]: runfile('C:/Users/MINUSX/.spyder-py3/guessgame.py', wdir='C:/Users/
MINUSX/.spyder-py3')
{'John': {'Phone': '012 794 794', 'Email': 'john@email.com'}, 'Jill':
{'Phone': '012 345 345', 'Email': 'jill@email.com'}, 'Joss': {'Phone': '012
321 321', 'Email': 'joss@email.com'}}
Jill's phone number is 012 345 345, and their email is jill@email.com
Joss's phone number is 012 321 321, and their email is joss@email.com
{'Phone': '012 345 345', 'Email': 'jill@email.com'}
Joss's phone number is 012 321 321, and their email is joss@email.com
Traceback (most recent call last):
```

LAB TASK 4:

```
#Arithmatic Operators
```

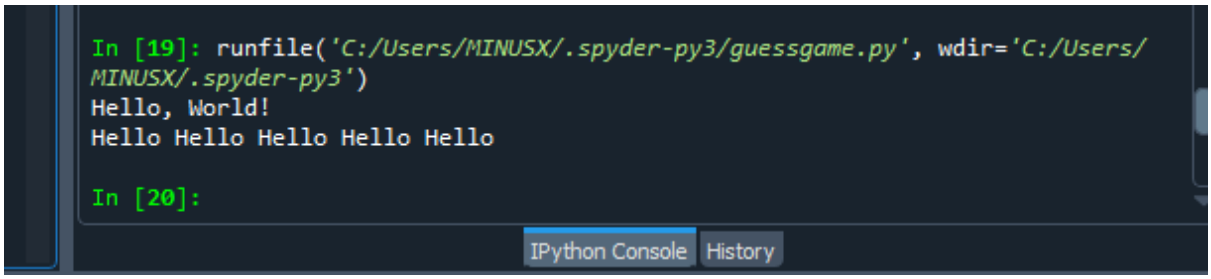
```
number = 1 +2 * 3 / 4.0


print('Number :', number)

remainder = 11 % 3

print('Remainder :',  remainder)

# power
squared = 7 ** 2

print('Squared :',squared)

cubed = 2 ** 3
print(cubed)


#%%
# List Operators


even_numbers = [2, 4, 6, 8]

uneven_numbers = [1, 3, 5, 7]

all_numbers = uneven_numbers + even_numbers


print(all_numbers)


print([1, 2, 3] * 3)
```



LAB TASK 5

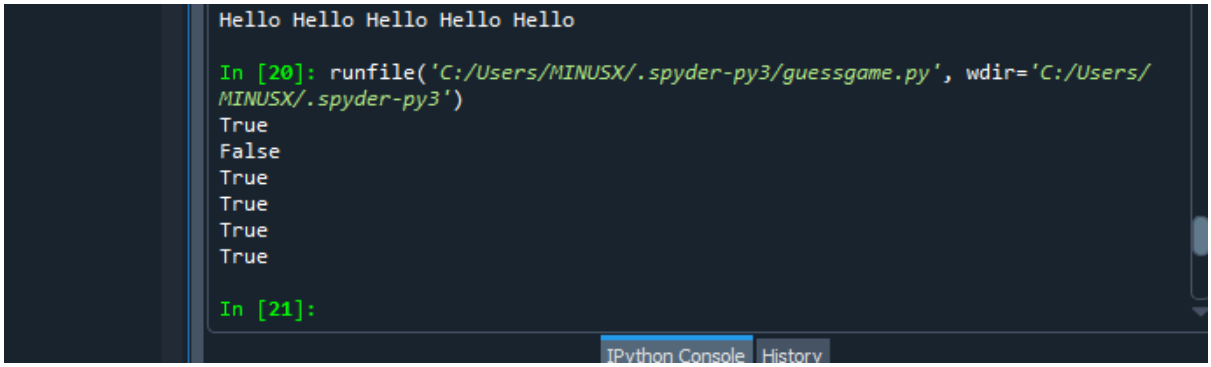
```
# Define the two strings
```

```
greeting = "Hello, World!"

repeated_hello = "Hello " * 5

# Print the strings
print(greeting)

print(repeated_hello)
```



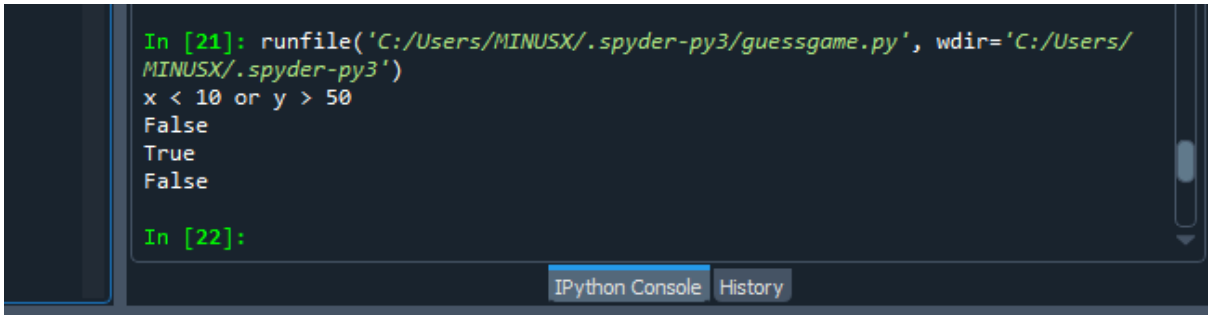
LAB TASK 6

```
x = 2
print(x == 2)
print(x == 3)
print(x < 3)
name = "John"
4

print(name == "John" and x == 2)

# Using `or`

print(name == "John" or name == "Jill")

# Using in on lists

print(name in ["John", "Jill", "Jess"])
```



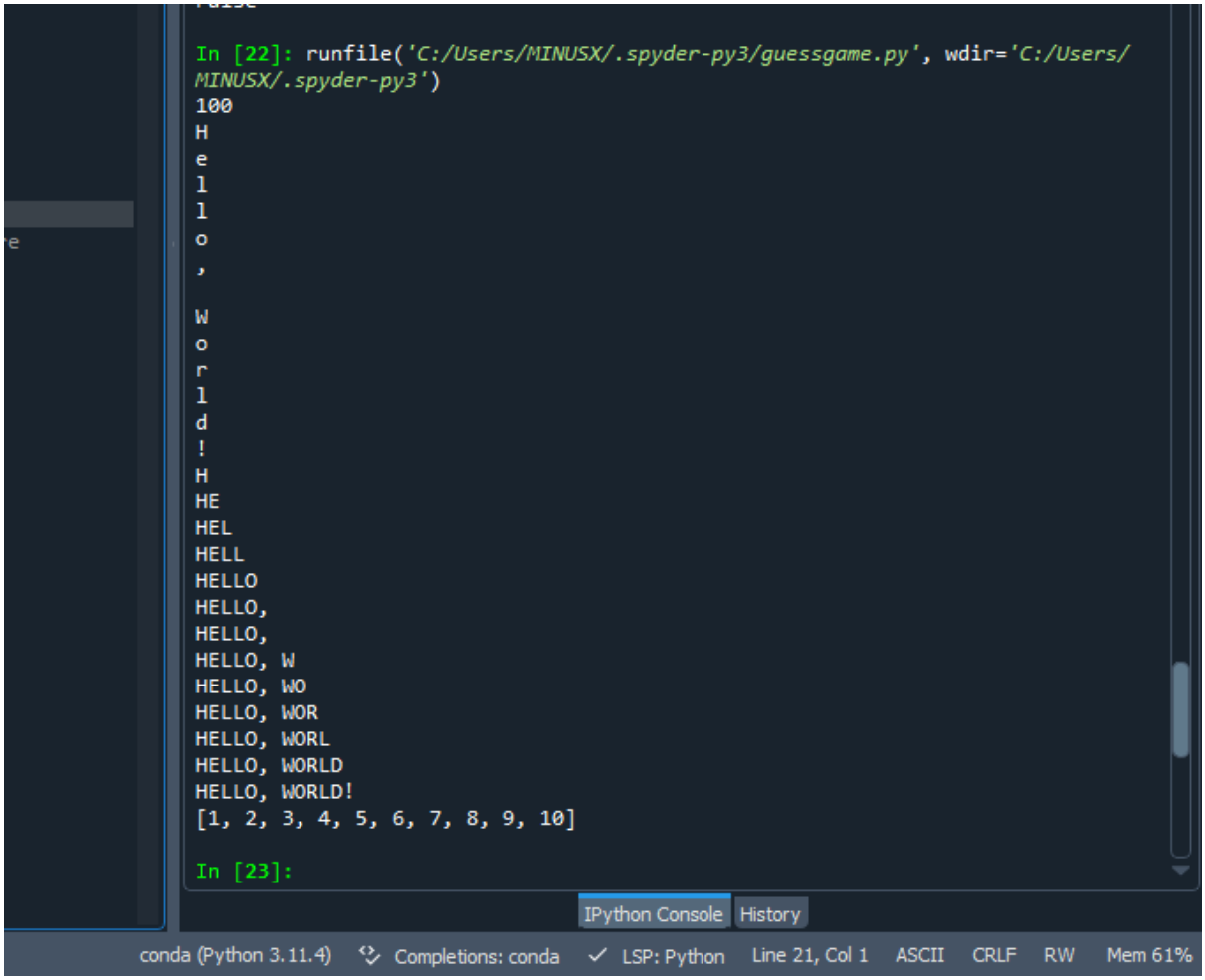
LAB TASK 7

```
x = 2

y = 10

if x > 2:
    print("x > 2")

elif x == 2 and y > 50:

    print("x == 2 and y > 50")

elif x < 10 or y > 50:
    print("x < 10 or y > 50")
else:
    print("Nothing worked.")


name_list1 = ["John", "Jill"]

name_list2 = ["John", "Jill"]

print (not (name_list1 == name_list2))

# Using `is`

name2 = "John"

print(name_list1 == name_list2)

print(name_list1 is name_list2)
```



LAB TASK 8

```
numeric_data  = [10, 20, 30, 40, 50]

for number in numeric_data:

    result = number*2 # Perform some operation (e.g., multiplication) print (result) Print the result
print(result)
#Sample string

text = "Hello, World!"

#Using a for loop to read and print each character in the string

for char in text:

    print(char)

new_text = ""

for char in text:

    new_text += char.upper() # Convert letters to uppercase else: new_text += char #Keep non-letter
characters as they are
    print(new_text)

#Writing Numeric Data

numeric_data = []

for i in range(1,11):
    numeric_data.append(i)


print(numeric_data)
```



LAB TASK 9

```
count = 1

while count <= 5:

    print(count)

count += 1

# 2. For Strings

# Using a while loop to print each character of a string text= "Hello"

text = "Hello"
index = 0

while index < len(text):

    print(text[index])
    index += 1
```

```
student_grades = {"Alice": 92, "Bob": 85, "Charlie": 78}

keys = list(student_grades.keys()) # Get the keys as a list

index = 0

while index < len (keys):

    key = keys[index]

value = student_grades[key]
print (f" (key): {value}")

index += 1
```

Post LAB

TASK

```
def calculate_grade(grade):
    if grade >= 90:
        return "Excellent"
    elif grade >= 80:
        return "Very Good"
    elif grade >= 70:
        return "Good"
    else:
        return "Needs Improvement"

def calculate_average_grade(grades):
    total_grade = sum(grades)
    average_grade = total_grade / len(grades)
    return average_grade

def main():
    students = ["Ali", "Bilal", "Abdullah", "ahmad", "sana", "uzair", "usman"]
    grades = [85, 78, 92, 68, 70, 88, 79]

    while True:
        print("1. Calculate and display average grade for all students")
        print("2. Categorize each student's grade")
        print("3. Search for a specific student's grade")
        print("4. Exit")
        choice = int(input("Enter your choice: "))

        if choice == 1:
            print("Students and their grades:")
```

```
        for i in range(len(students)):
            print(f"{students[i]} - {grades[i]}")
        average_grade = calculate_average_grade(grades)
        print(f"Average grade: {average_grade:.2f}")

    elif choice == 2:
        print("Categorized grades:")
        for i in range(len(students)):
            print(f"{students[i]} - {calculate_grade(grades[i])}")

    elif choice == 3:
        student_name = input("Enter student name to search for grade: ")
        if student_name in students:
            index = students.index(student_name)
            print(f"Grade for {student_name}: {grades[index]}")
        else:
            print("Student not found.")

    elif choice == 4:
        break

    else:
        print("Invalid choice. Please try again.")

if __name__ == "__main__":
    main()
```