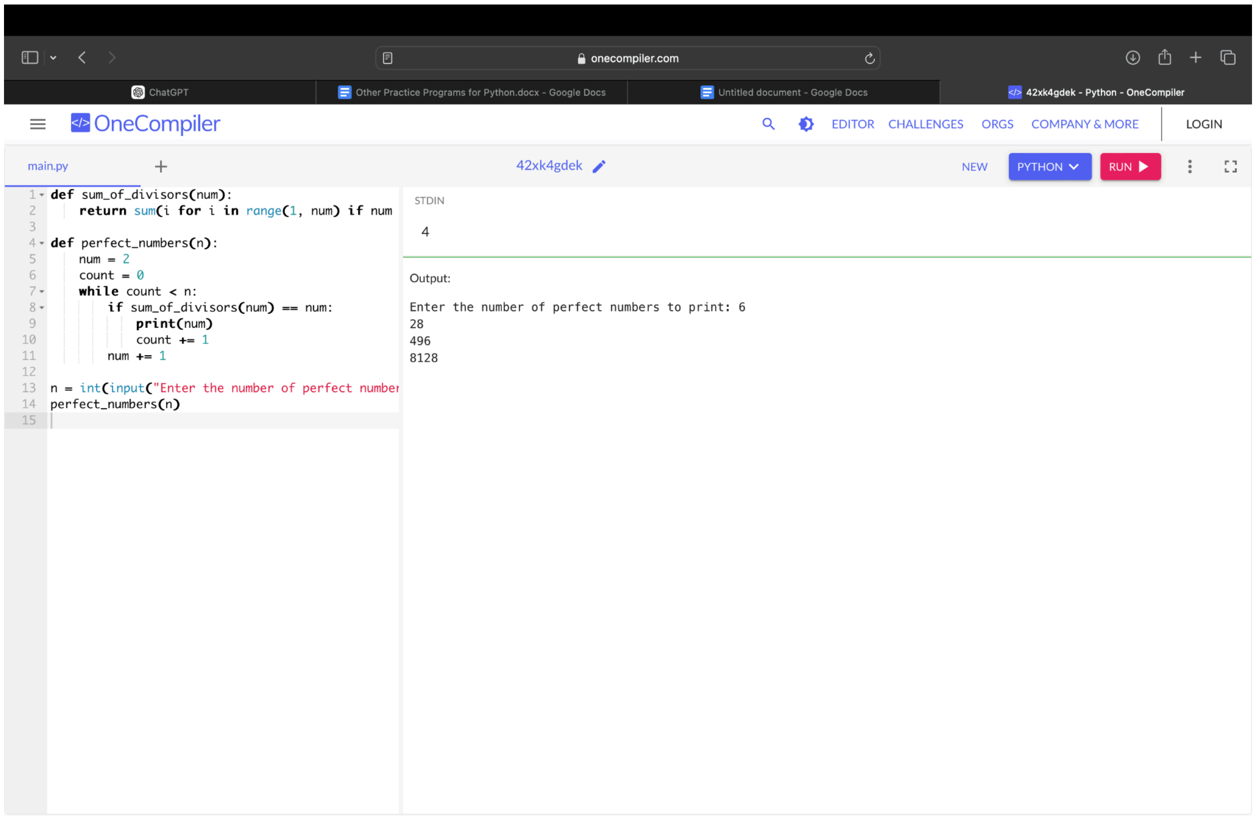
21. Write a program to print the first n perfect numbers. (Hint Perfect number means a positive integer that is equal to the sum of its proper divisors)

PROGRAM:

```
def sum_of_divisors(num):
    return sum(i for i in range(1, num) if num % i == 0)

def perfect_numbers(n):
    num = 2
    count = 0
    while count < n:
        if sum_of_divisors(num) == num:
            print(num)
            count += 1
        num += 1

n = int(input("Enter the number of perfect numbers to print: "))
perfect_numbers(n)
```

OUTPUT:


onecompiler.com
ChatGPT
Other Practice Programs for Python.docx - Google Docs
Untitled document - Google Docs
42xk4gdek - Python - OneCompiler
OneCompiler
EDITOR
CHALLENGES
ORGS
COMPANY & MORE
LOGIN
main.py
42xk4gdek
NEW
PYTHON
RUN

```
def sum_of_divisors(num):
    return sum(i for i in range(1, num) if num

def perfect_numbers(n):
    num = 2
    count = 0
    while count < n:
        if sum_of_divisors(num) == num:
            print(num)
            count += 1
        num += 1

n = int(input("Enter the number of perfect number
perfect_numbers(n)
```

STDIN

```
4
```

Output:

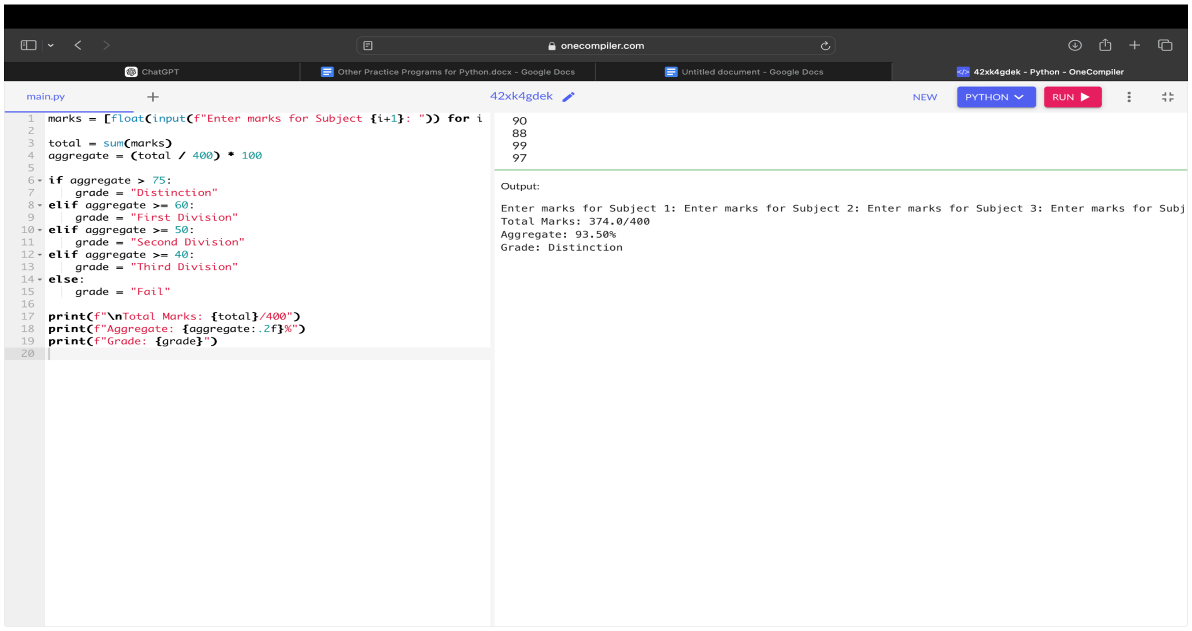
```
Enter the number of perfect numbers to print: 6
28
496
8128
```

22. Write a program to enter the marks of a student in four subjects. Then calculate the total and aggregate, display the grade obtained by the student. If the student scores an aggregate greater than 75%, then the grade is Distinction. If aggregate is 60>= and <75, then the grade is First Division. If aggregate is 50 >= and <60, then the grade is Second Division. If aggregate is 40>= and <50, then the grade is Third Division. Else the grade is Fail.

PROGRAM:

```
marks = [float(input(f"Enter marks for Subject {i+1}: ")) for i in range(4)]
total = sum(marks)
aggregate = (total / 400) * 100
if aggregate > 75:
    grade = "Distinction"
elif aggregate >= 60:
    grade = "First Division"
elif aggregate >= 50:
    grade = "Second Division"
elif aggregate >= 40:
    grade = "Third Division"
else:
    grade = "Fail"
print(f"\nTotal Marks: {total}/400")
print(f"Aggregate: {aggregate:.2f}%")
print(f"Grade: {grade}")
```

OUTPUT:


onecompiler.com
ChatGPT
Other Practice Programs for Python.docx - Google Docs
Untitled document - Google Docs
42xk4gdek - Python - OneCompiler
main.py
42xk4gdek
NEW
PYTHON
RUN

```
marks = [float(input(f"Enter marks for Subject {i+1}: ")) for i

total = sum(marks)
aggregate = (total / 400) * 100

if aggregate > 75:
    grade = "Distinction"
elif aggregate >= 60:
    grade = "First Division"
elif aggregate >= 50:
    grade = "Second Division"
elif aggregate >= 40:
    grade = "Third Division"
else:
    grade = "Fail"

print(f"\nTotal Marks: {total}/400")
print(f"Aggregate: {aggregate:.2f}%")
print(f"Grade: {grade}")
```

```
90
88
99
97
```

Output:

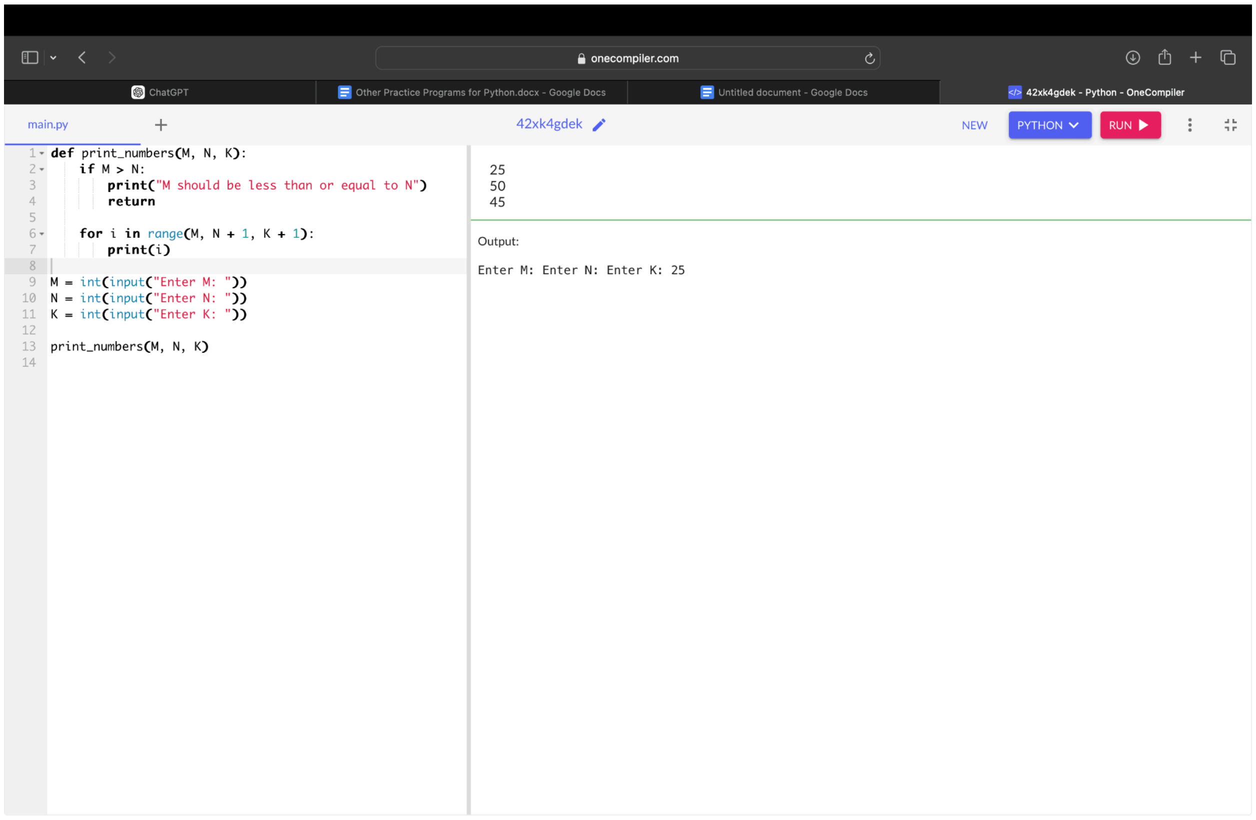
```
Enter marks for Subject 1: Enter marks for Subject 2: Enter marks for Subject 3: Enter marks for Subj
Total Marks: 374.0/400
Aggregate: 93.50%
Grade: Distinction
```

23. Write a program to print the numbers from M to N by skipping K numbers in between?

PROGRAM:

```
def print_numbers(M, N, K):
    if M > N:
        print("M should be less than or equal to N")
        return

    for i in range(M, N + 1, K + 1):
        print(i)

M = int(input("Enter M: "))
N = int(input("Enter N: "))
K = int(input("Enter K: "))

print_numbers(M, N, K)
```

OUTPUT:


onecompiler.com
ChatGPT
Other Practice Programs for Python.docx - Google Docs
Untitled document - Google Docs
42xk4gdek - Python - OneCompiler
main.py
42xk4gdek
NEW
PYTHON
RUN

```
def print_numbers(M, N, K):
    if M > N:
        print("M should be less than or equal to N")
        return

    for i in range(M, N + 1, K + 1):
        print(i)

M = int(input("Enter M: "))
N = int(input("Enter N: "))
K = int(input("Enter K: "))

print_numbers(M, N, K)
```

```
25
50
45
```

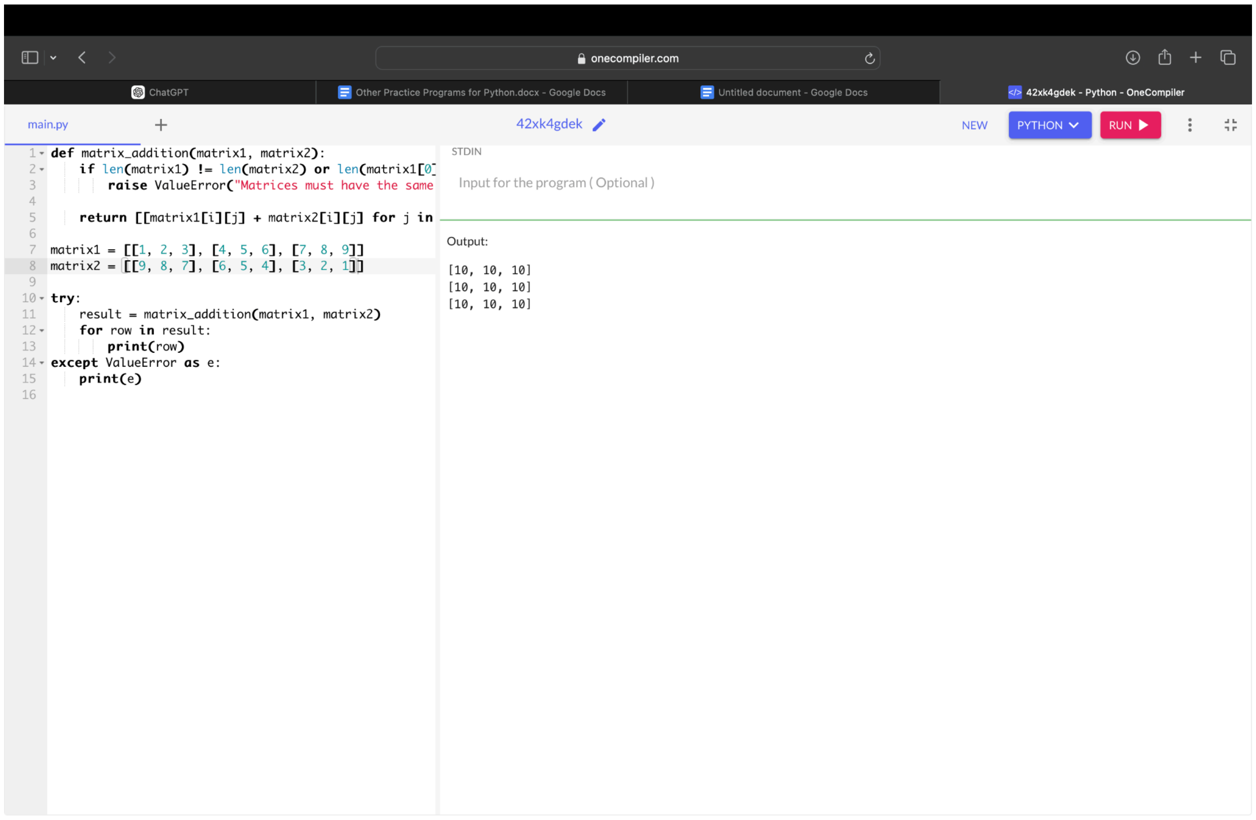
```
Output:

Enter M: Enter N: Enter K: 25
```

24. Write a program for matrix addition?

PROGRAM:

```
def matrix_addition(matrix1, matrix2):
    if len(matrix1) != len(matrix2) or len(matrix1[0]) != len(matrix2[0]):
        raise ValueError("Matrices must have the same dimensions.")

    return [[matrix1[i][j] + matrix2[i][j] for j in range(len(matrix1[0]))] for i in
range(len(matrix1))]

matrix1 = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
matrix2 = [[9, 8, 7], [6, 5, 4], [3, 2, 1]]

try:
    result = matrix_addition(matrix1, matrix2)
    for row in result:
        print(row)
except ValueError as e:
    print(e)
```

OUTPUT:


onecompiler.com
ChatGPT
Other Practice Programs for Python.docx - Google Docs
Untitled document - Google Docs
42xk4gdek - Python - OneCompiler
main.py
42xk4gdek
NEW
PYTHON
RUN
STDIN
Input for the program (Optional)
Output:

```
def matrix_addition(matrix1, matrix2):
    if len(matrix1) != len(matrix2) or len(matrix1[0
        raise ValueError("Matrices must have the same

    return [[matrix1[i][j] + matrix2[i][j] for j in

matrix1 = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
matrix2 = [[9, 8, 7], [6, 5, 4], [3, 2, 1]]

try:
    result = matrix_addition(matrix1, matrix2)
    for row in result:
        print(row)
except ValueError as e:
    print(e)
```

```
[10, 10, 10]
[10, 10, 10]
[10, 10, 10]
```

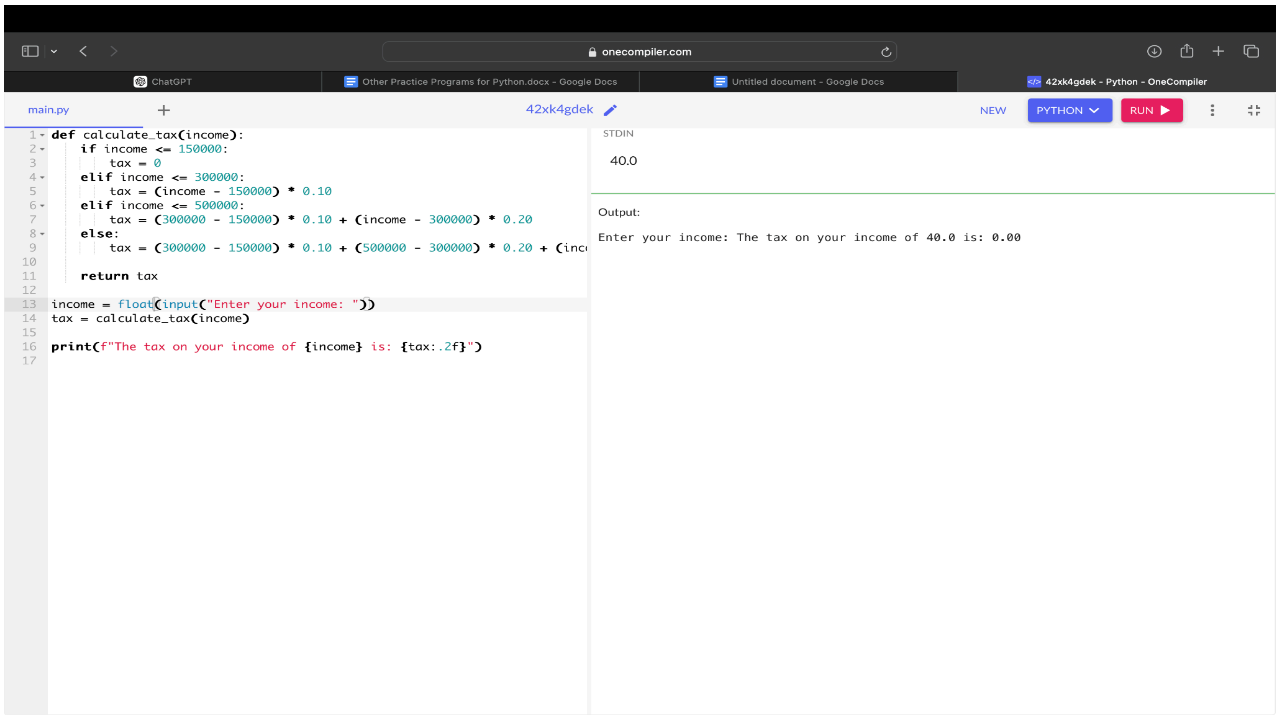
25. Write a program to calculate tax given the following conditions:

a. If income is less than or equal to 1,50,000 then no tax
b. If taxable income is 1,50,001 – 3,00,000 the charge 10% tax
c. If taxable income is 3,00,001 – 5,00,000 the charge 20% tax
d. If taxable income is above 5,00,001 then charge 30% tax

PROGRAM:

```
def calculate_tax(income):
    if income <= 150000:
        tax = 0
    elif income <= 300000:
        tax = (income - 150000) * 0.10
    elif income <= 500000:
        tax = (300000 - 150000) * 0.10 + (income - 300000) * 0.20
    else:
        tax = (300000 - 150000) * 0.10 + (500000 - 300000) * 0.20 + (income - 500000) *
0.30

    return tax
income = float(input("Enter your income: "))
tax = calculate_tax(income)

print(f"The tax on your income of {income} is: {tax:.2f}")
```

OUTPUT:



```
def calculate_tax(income):
    if income <= 150000:
        tax = 0
    elif income <= 300000:
        tax = (income - 150000) * 0.10
    elif income <= 500000:
        tax = (300000 - 150000) * 0.10 + (income - 300000) * 0.20
    else:
        tax = (300000 - 150000) * 0.10 + (500000 - 300000) * 0.20 + (inc

    return tax

income = float(input("Enter your income: "))
tax = calculate_tax(income)

print(f"The tax on your income of {income} is: {tax:.2f}")
```

STDIN

```
40.0
```

Output:

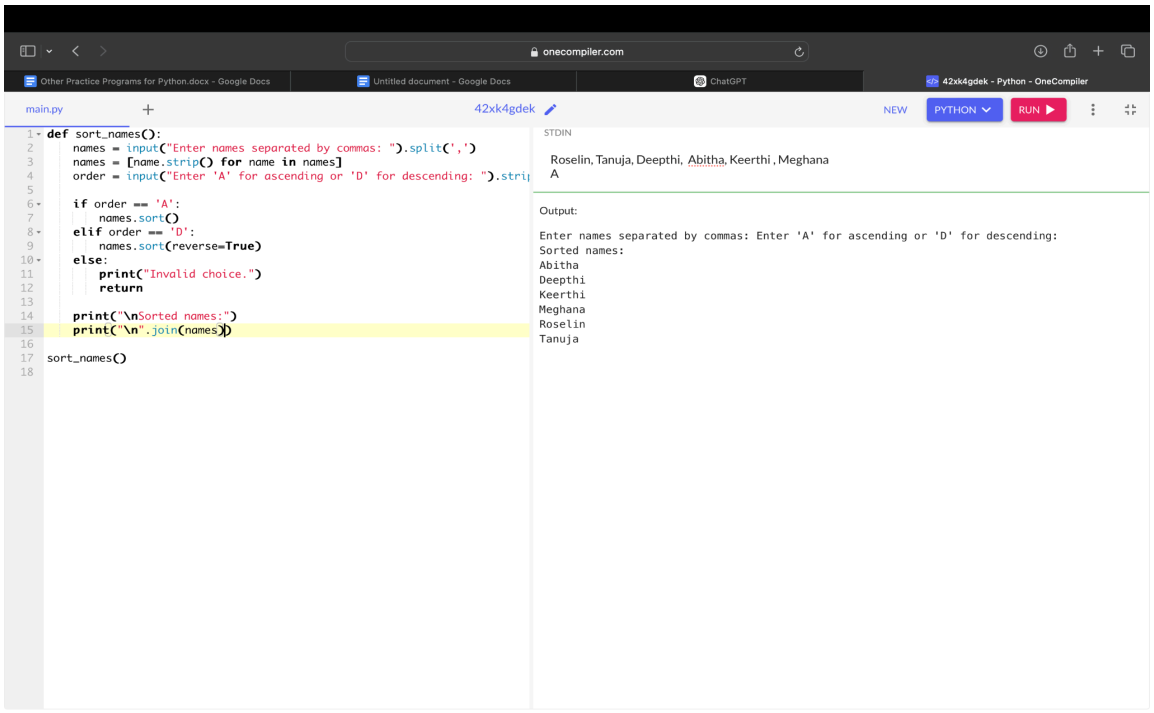
```
Enter your income: The tax on your income of 40.0 is: 0.00
```

26. *Write a program that would sort a list of names in alphabetical order Ascending or Descending, choice from the user?*

PROGRAM:

```
def sort_names():
    names = input("Enter names separated by commas: ").split(',')
    names = [name.strip() for name in names]
    order = input("Enter 'A' for ascending or 'D' for descending: ").strip().upper()

    if order == 'A':
        names.sort()
    elif order == 'D':
        names.sort(reverse=True)
    else:
        print("Invalid choice.")
        return

    print("\nSorted names:")
    print("\n".join(names))

sort_names()
```

OUTPUT:


onecompiler.com
Other Practice Programs for Python.docx - Google Docs
Untitled document - Google Docs
ChatGPT
42xk4gdek - Python - OneCompiler
main.py
42xk4gdek
NEW
PYTHON
RUN

```
def sort_names():
    names = input("Enter names separated by commas: ").split(',')
    names = [name.strip() for name in names]
    order = input("Enter 'A' for ascending or 'D' for descending: ").stri

    if order == 'A':
        names.sort()
    elif order == 'D':
        names.sort(reverse=True)
    else:
        print("Invalid choice.")
        return

    print("\nSorted names:")
    print("\n".join(names))

sort_names()
```

STDIN

```
Roselin, Tanuja, Deepthi,  Abitha, Keerthi , Meghana
A
```

Output:

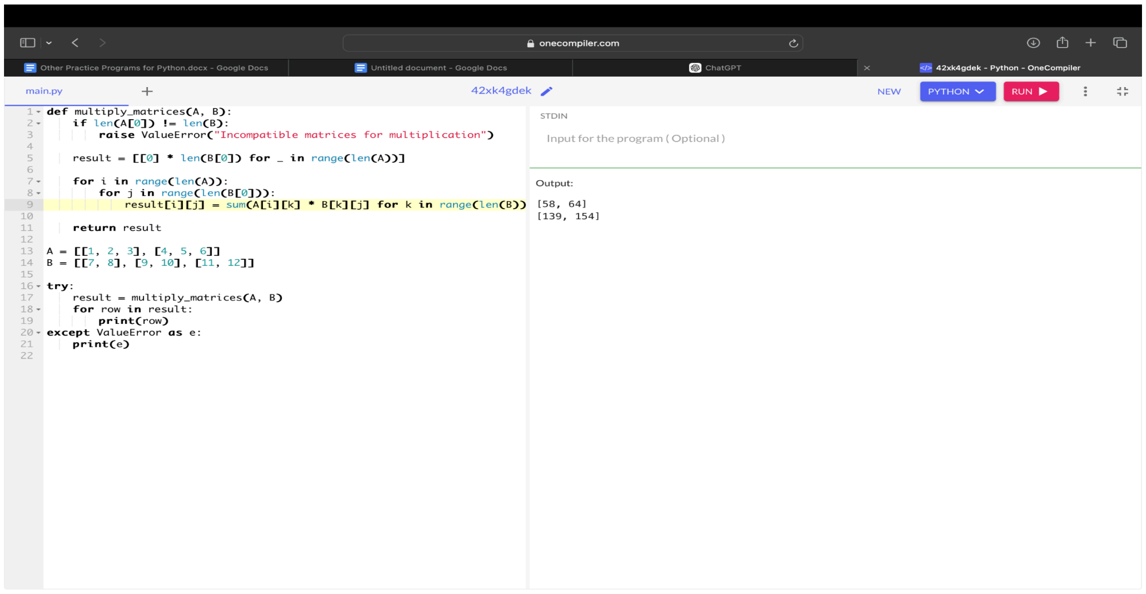
```
Enter names separated by commas: Enter 'A' for ascending or 'D' for descending:
Sorted names:
Abitha
Deepthi
Keerthi
Meghana
Roselin
Tanuja
```

27. Write a program for matrix multiplication?

PROGRAM:

```
def multiply_matrices(A, B):
    if len(A[0]) != len(B):
        raise ValueError("Incompatible matrices for multiplication")

    result = [[0] * len(B[0]) for _ in range(len(A))]

    for i in range(len(A)):
        for j in range(len(B[0])):
            result[i][j] = sum(A[i][k] * B[k][j] for k in range(len(B)))

    return result

A = [[1, 2, 3], [4, 5, 6]]
B = [[7, 8], [9, 10], [11, 12]]

try:
    result = multiply_matrices(A, B)
    for row in result:
        print(row)
except ValueError as e:
    print(e)
```

OUTPUT:


onecompiler.com
Other Practice Programs for Python.docx - Google Docs
Untitled document - Google Docs
ChatGPT
42xk4gdek - Python - OneCompiler
main.py
42xk4gdek
NEW
PYTHON
RUN
STDIN
Input for the program (Optional)

```
def multiply_matrices(A, B):
    if len(A[0]) != len(B):
        raise ValueError("Incompatible matrices for multiplication")

    result = [[0] * len(B[0]) for _ in range(len(A))]

    for i in range(len(A)):
        for j in range(len(B[0])):
            result[i][j] = sum(A[i][k] * B[k][j] for k in range(len(B)))

    return result

A = [[1, 2, 3], [4, 5, 6]]
B = [[7, 8], [9, 10], [11, 12]]

try:
    result = multiply_matrices(A, B)
    for row in result:
        print(row)
except ValueError as e:
    print(e)
```

Output:

```
[58, 64]
[139, 154]
```

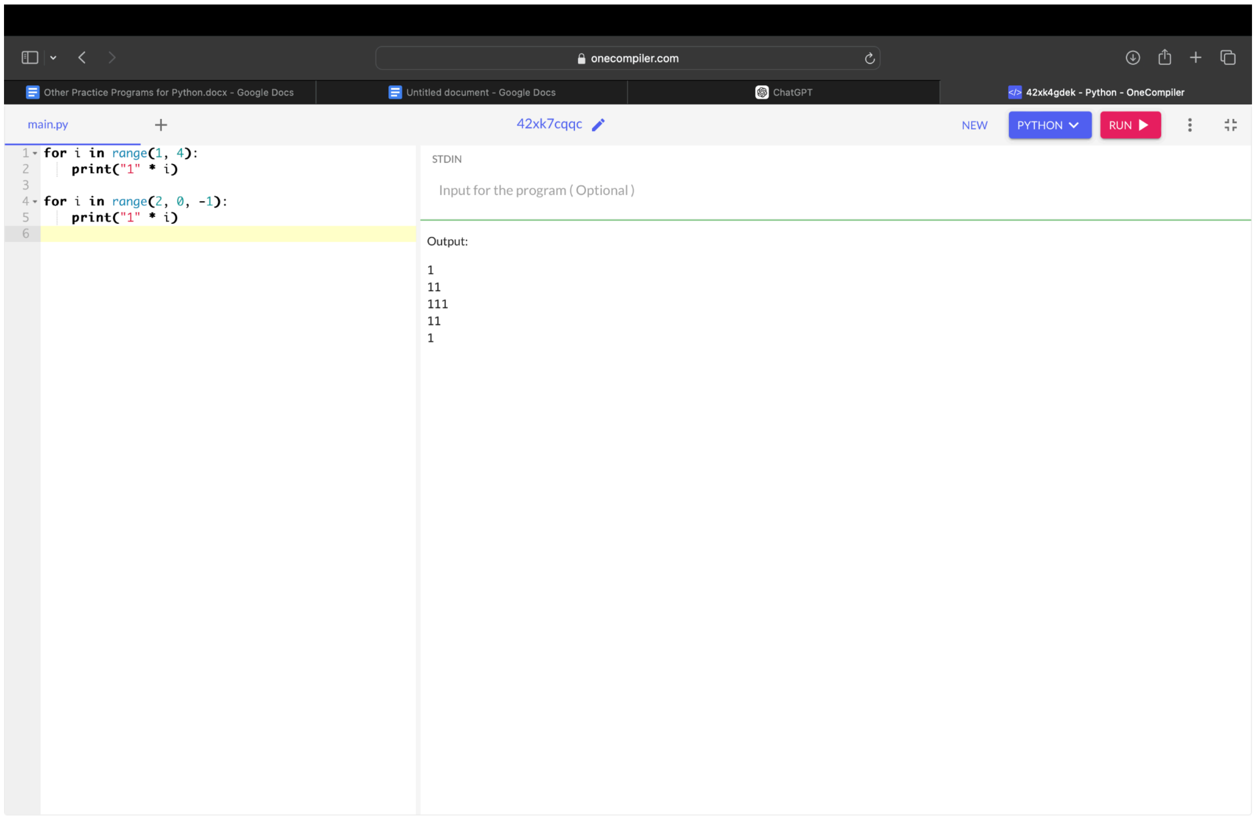
29. Write a program to print the following pattern

```
1

11

111

11

1
```

PROGRAM:

```
for i in range(1, 4):
    print("1" * i)

for i in range(2, 0, -1):
    print("1" * i)
```

OUTPUT:



onecompiler.com

Other Practice Programs for Python.docx - Google Docs | Untitled document - Google Docs | ChatGPT | 42xk4gdek - Python - OneCompiler

main.py | 42xk7cqqc | NEW | PYTHON | RUN

```
for i in range(1, 4):
    print("1" * i)

for i in range(2, 0, -1):
    print("1" * i)
```

STDIN

Input for the program (Optional)

Output:

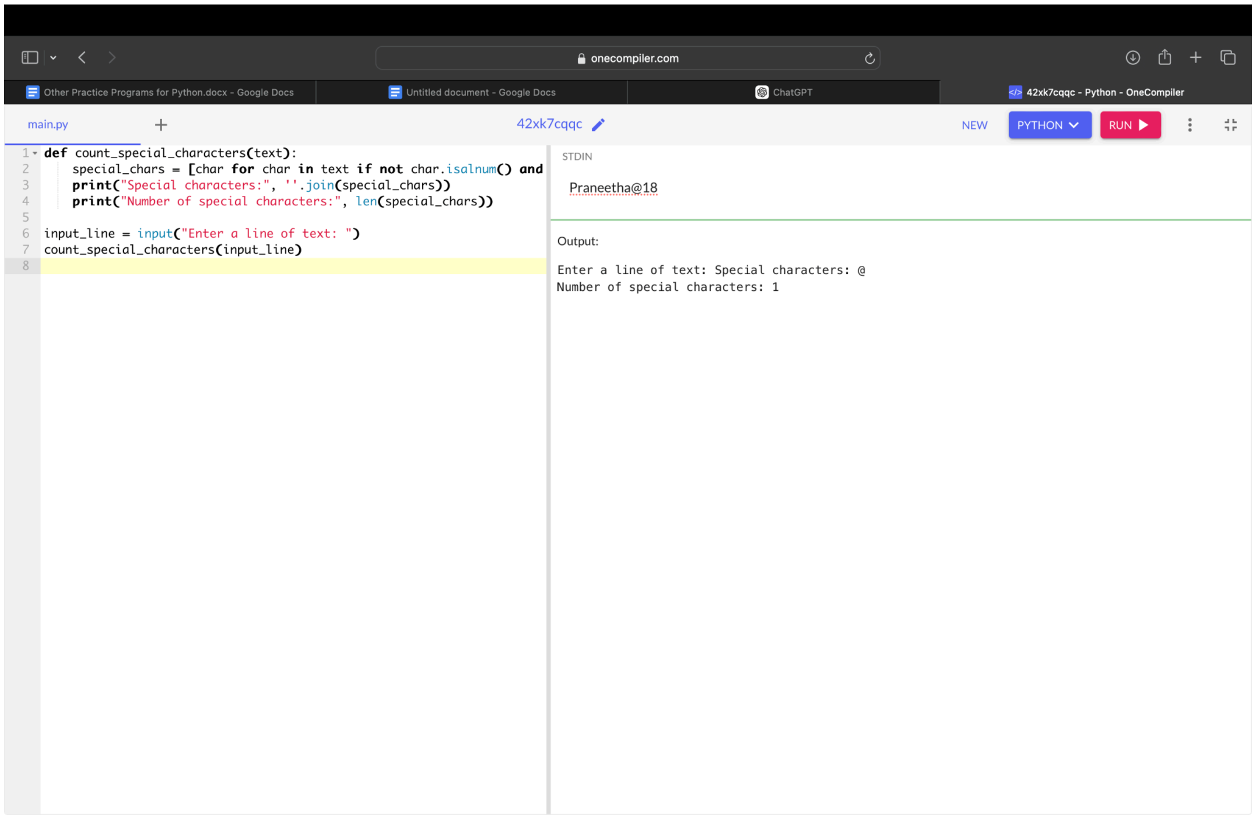
```
1
11
111
11
1
```

30. Write a program to print the special characters separately and print number of Special characters in the line?

PROGRAM:

```
def count_special_characters(text):
    special_chars = [char for char in text if not char.isalnum() and not char.isspace()]
    print("Special characters:", ''.join(special_chars))
    print("Number of special characters:", len(special_chars))

input_line = input("Enter a line of text: ")
count_special_characters(input_line)
```

OUTPUT:


onecompiler.com
Other Practice Programs for Python.docx - Google Docs
Untitled document - Google Docs
ChatGPT
42xk7cqqc - Python - OneCompiler
main.py
42xk7cqqc
NEW
PYTHON
RUN
STDIN
Output:

```
def count_special_characters(text):
    special_chars = [char for char in text if not char.isalnum() and
    print("Special characters:", ''.join(special_chars))
    print("Number of special characters:", len(special_chars))

input_line = input("Enter a line of text: ")
count_special_characters(input_line)
```

STDIN

Praneetha@18

Output:

```
Enter a line of text: Special characters: @
Number of special characters: 1
```