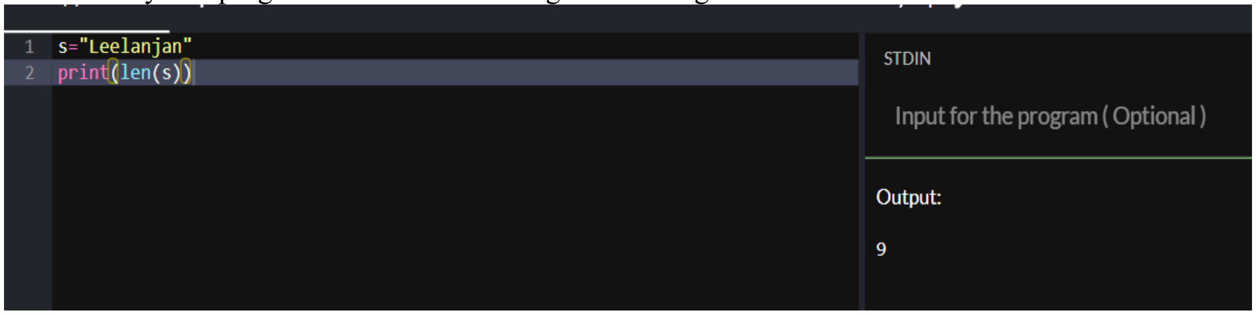
1. Write a Python program to calculate the length of a string.



```
s="Leelanjan"
print(len(s))
```

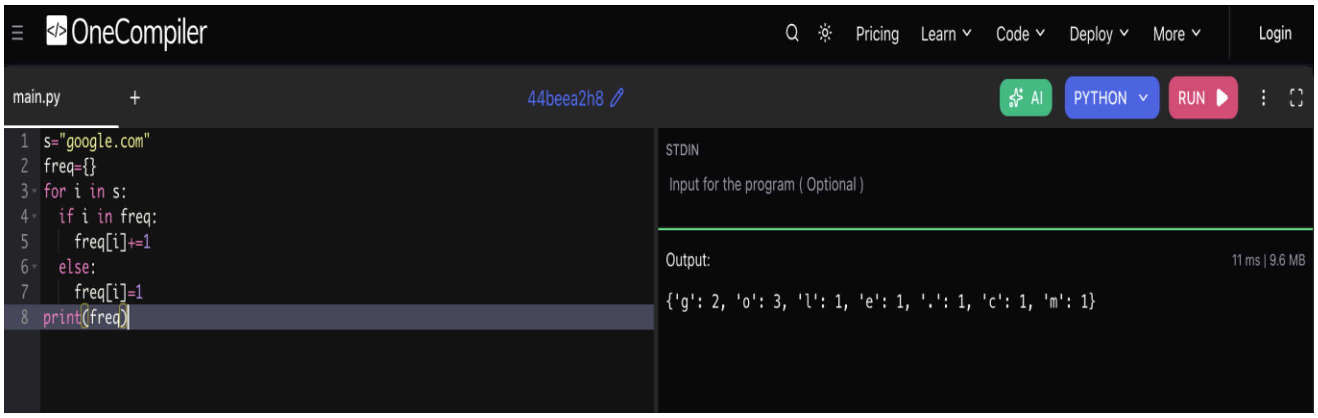
STDIN

Input for the program (Optional)

Output:

```
9
```

2. Write a Python program to count the number of characters (character frequency) in a string.
Sample String : google.com'
Expected Result : {'o': 3, 'g': 2, '.': 1, 'e': 1, 'l': 1, 'm': 1, 'c': 1}



```
s="google.com"
freq={}
for i in s:
  if i in freq:
    freq[i]+=1
  else:
    freq[i]=1
print(freq)
```

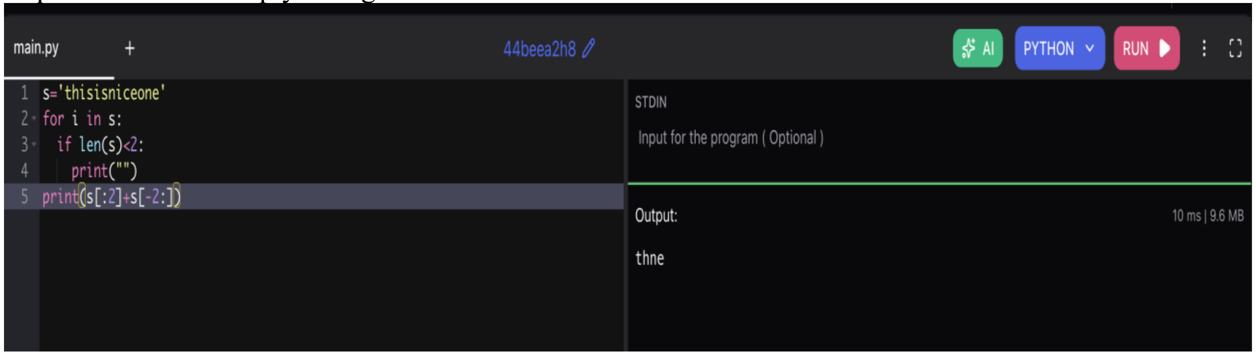
STDIN

Input for the program (Optional)

Output: 11 ms | 9.6 MB

```
{'g': 2, 'o': 3, 'l': 1, 'e': 1, '.': 1, 'c': 1, 'm': 1}
```

3. Write a Python program to get a string made of the first 2 and the last 2 chars from a given a string. If the string length is less than 2, return instead of the empty string.
Sample String : 'thisisniceone'
Expected Result : 'thne'"
Sample String : 'ab'
Expected Result : 'abab'
Sample String : 'f'
Expected Result : Empty String



```
s='thisisniceone'
for i in s:
  if len(s)<2:
    print("")
print(s[:2]+s[-2:])
```

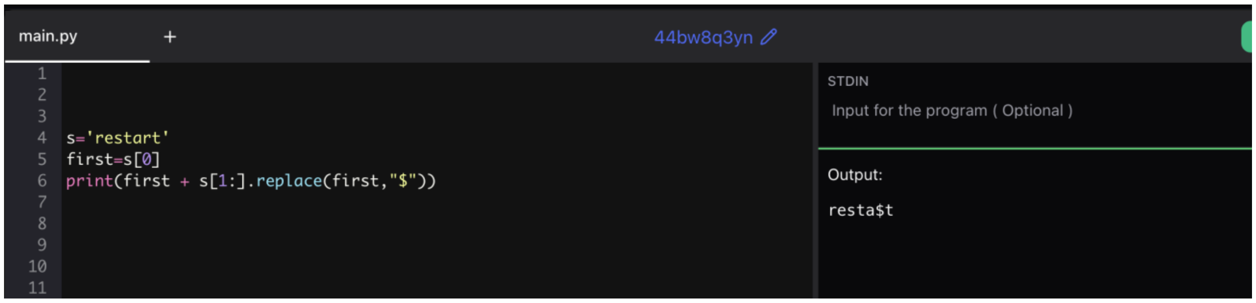
STDIN

Input for the program (Optional)

Output: 10 ms | 9.6 MB

```
thne
```

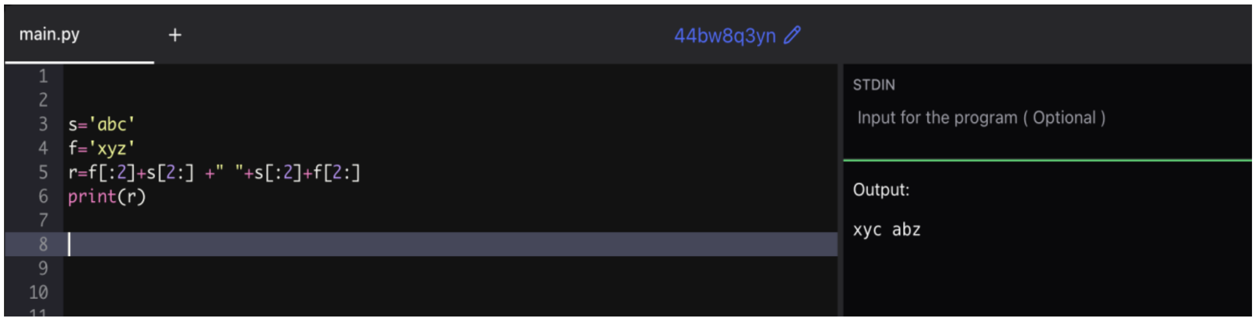
4. Write a Python program to get a string from a given string where all occurrences of its first char have been changed to '$', except the first char itself.
Sample String : 'restart'
Expected Result : 'resta$t'



```python
s='restart'
first=s[0]
print(first + s[1:].replace(first,"$"))
```

```
Output:
resta$t
```

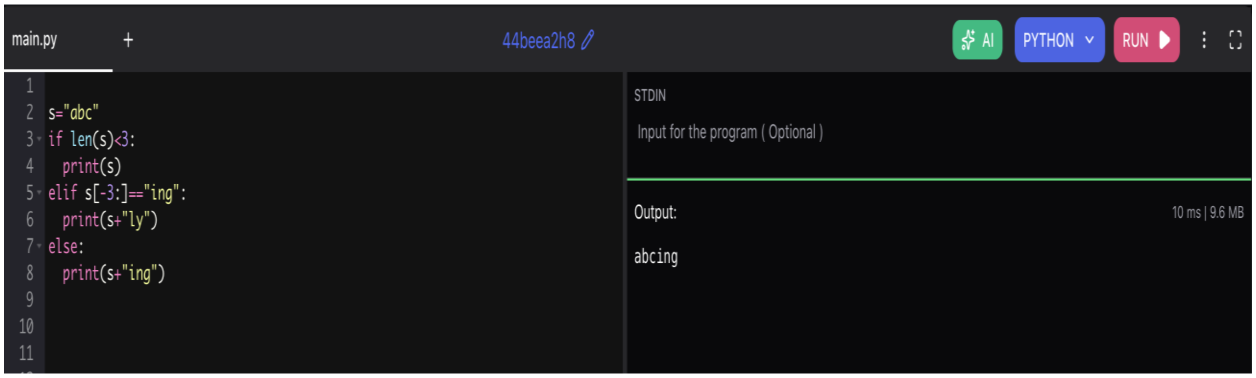
5. Write a Python program to get a single string from two given strings, separated by a space and swap the first two characters of each string.
Sample String : 'abc', 'xyz'
Expected Result : 'xyc abz'



```python
s='abc'
f='xyz'
r=f[:2]+s[2:] +" "+s[:2]+f[2:]
print(r)
```

```
Output:
xyc abz
```

6. Write a Python program to add 'ing' at the end of a given string (length should be at least 3). If the given string already ends with 'ing' then add 'ly' instead. If the string length of the given string is less than 3, leave it unchanged.
Sample String : 'abc'
Expected Result : 'abcing'
Sample String : 'string'
Expected Result : 'stringly'

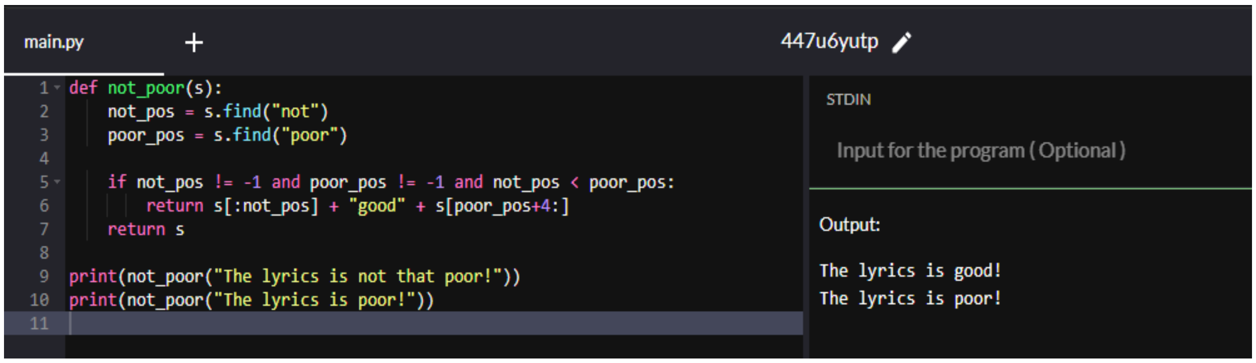


```python
s="abc"
if len(s)<3:
  print(s)
elif s[-3:]=="ing":
  print(s+"ly")
else:
  print(s+"ing")
```

```
Output:
abcing
```

7. Write a Python program to find the first appearance of the substring 'not' and 'poor' from a given string, if 'not' follows the 'poor', replace the whole 'not'...'poor' substring with 'good'. Return the resulting string.

Sample String : 'The lyrics is not that poor!'
'The lyrics is poor!'
Expected Result : 'The lyrics is good!'
'The lyrics is poor!'



main.py + 447u6yutp

```python
def not_poor(s):
    not_pos = s.find("not")
    poor_pos = s.find("poor")

    if not_pos != -1 and poor_pos != -1 and not_pos < poor_pos:
        return s[:not_pos] + "good" + s[poor_pos+4:]
    return s

print(not_poor("The lyrics is not that poor!"))
print(not_poor("The lyrics is poor!"))
```

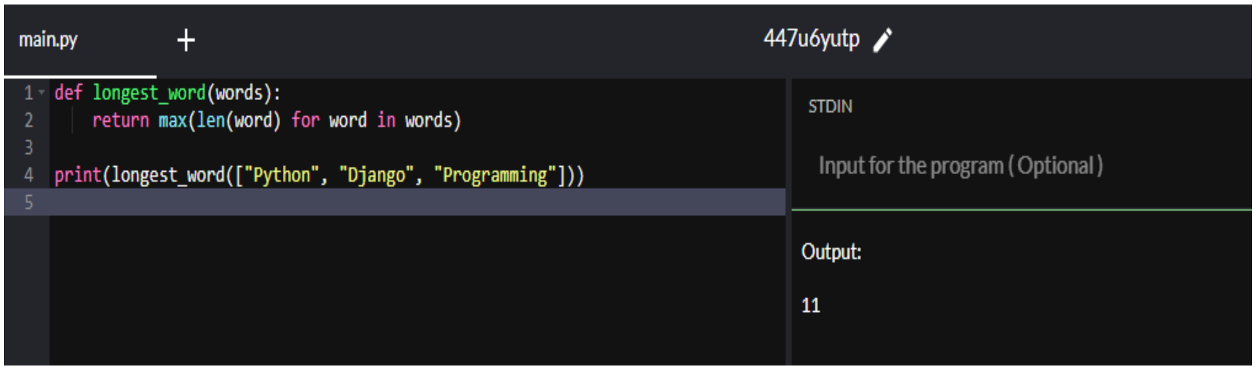
STDIN

Input for the program (Optional)

Output:

```
The lyrics is good!
The lyrics is poor!
```

8. Write a Python function that takes a list of words and returns the length of the longest one.



main.py + 447u6yutp

```python
def longest_word(words):
    return max(len(word) for word in words)

print(longest_word(["Python", "Django", "Programming"]))
```

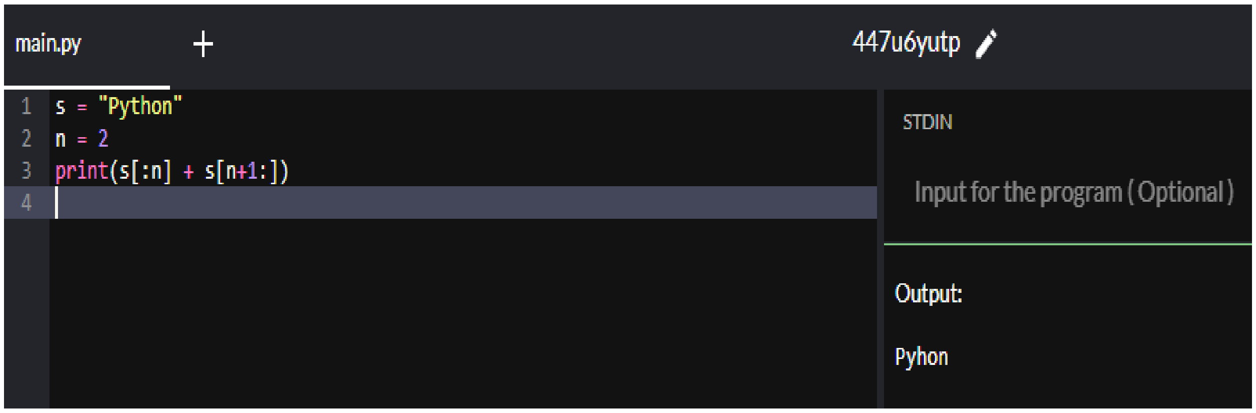
STDIN

Input for the program (Optional)

Output:

```
11
```

9. Write a Python program to remove the nth index character from a nonempty string.



main.py + 447u6yutp

```python
s = "Python"
n = 2
print(s[:n] + s[n+1:])
```

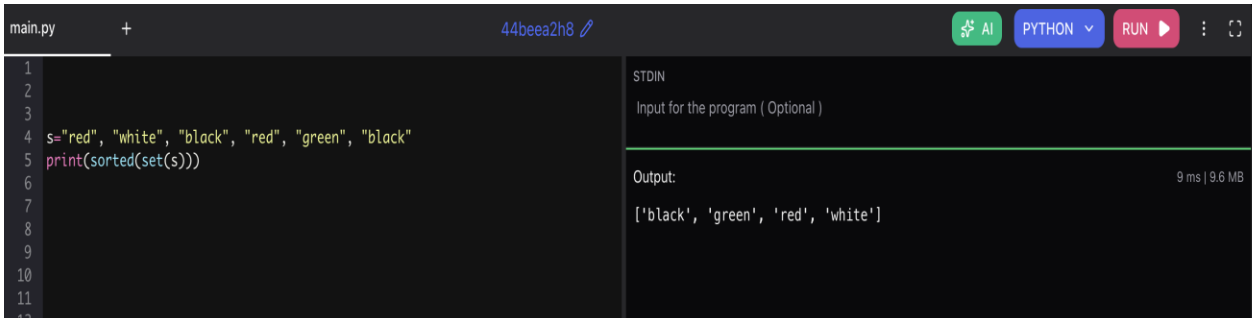
STDIN

Input for the program (Optional)

Output:

```
Pyhon
```

10. Write a Python program that accepts a comma separated sequence of words as input and prints the unique words in sorted form (alphanumerically).
Sample Words : red, white, black, red, green, black
Expected Result : black, green, red, white

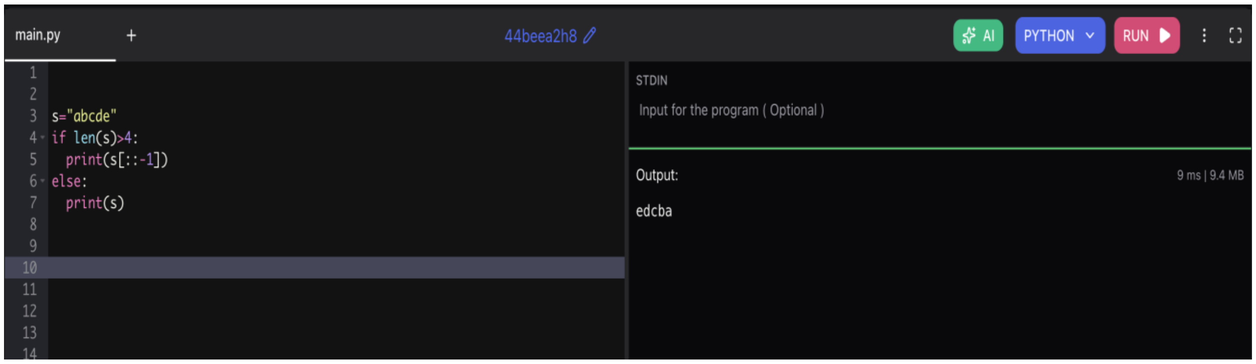


```python
s="red", "white", "black", "red", "green", "black"
print(sorted(set(s)))
```

Output:

```
['black', 'green', 'red', 'white']
```

11. Write a Python function to reverses a string if it's length is a multiple of 4.

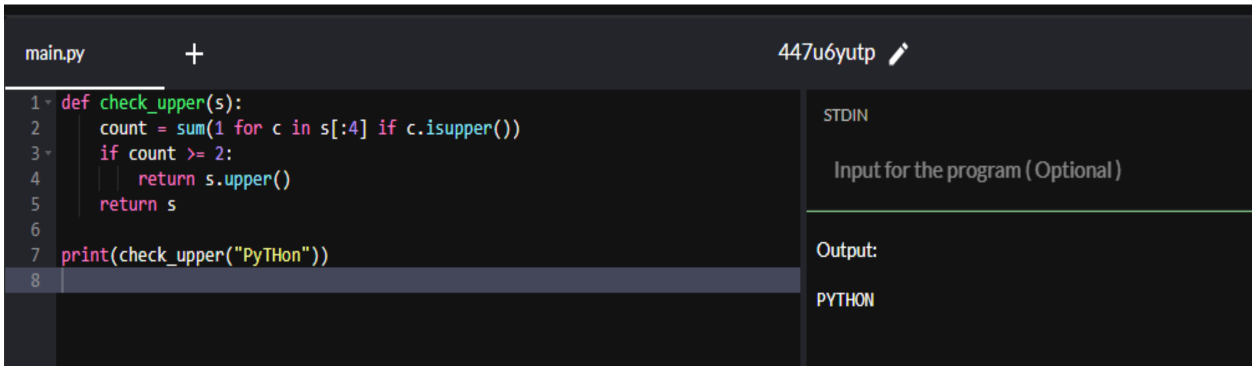


```python
s="abcde"
if len(s)>4:
  print(s[::-1])
else:
  print(s)
```

Output:

```
edcba
```

12. Write a Python function to convert a given string to all uppercase if it contains at least 2 uppercase characters in the first 4 characters.



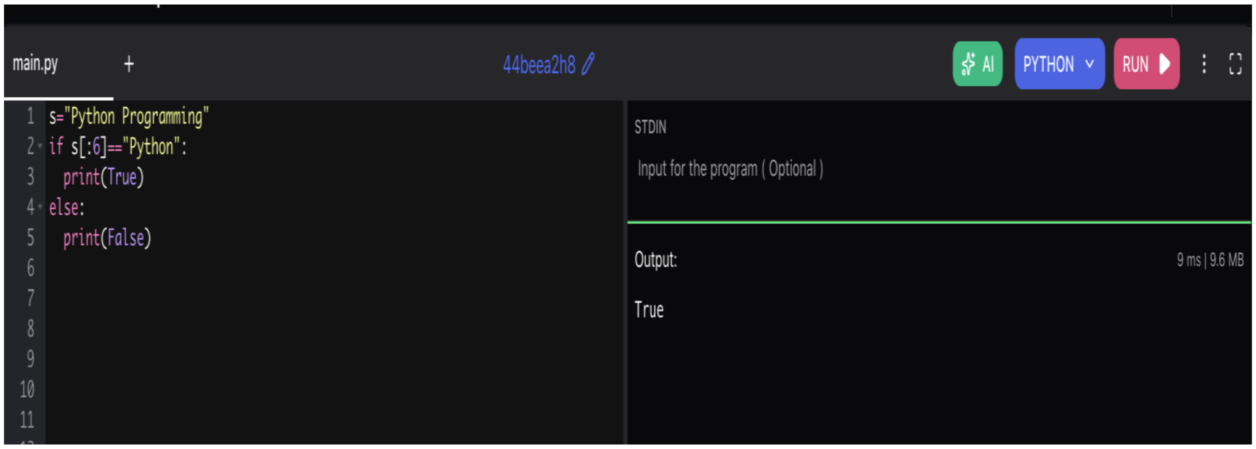
```python
def check_upper(s):
    count = sum(1 for c in s[:4] if c.isupper())
    if count >= 2:
        return s.upper()
    return s

print(check_upper("PyTHon"))
```

Output:

```
PYTHON
```

13. Write a Python program to check whether a string starts with specified characters.


main.py + 44beea2h8 AI PYTHON RUN

```python
s="Python Programming"
if s[:6]=="Python":
  print(True)
else:
  print(False)
```

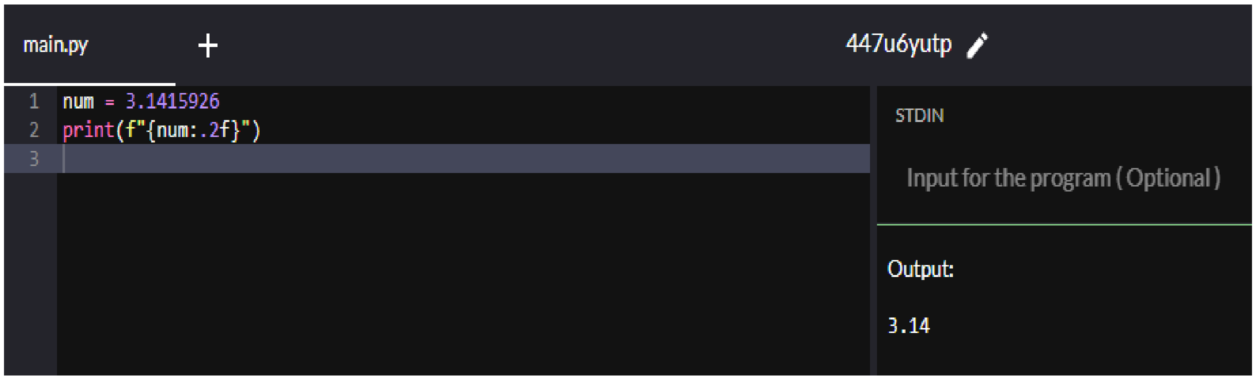
STDIN

Input for the program (Optional)

Output: 9 ms | 9.6 MB

```
True
```

14. Write a Python program to print the following floating numbers upto 2 decimal places.
3.1415926


main.py + 447u6yutp

```python
num = 3.1415926
print(f"{num:.2f}")
```

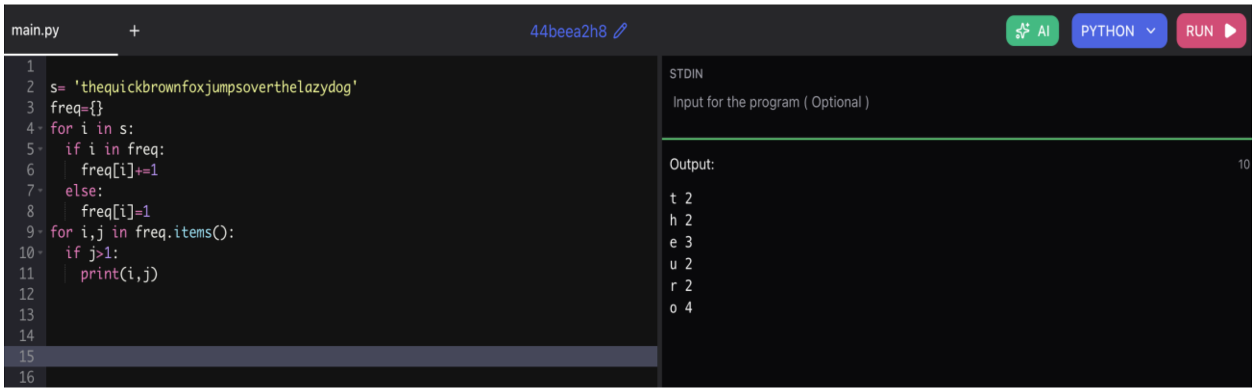
STDIN

Input for the program (Optional)

Output:

```
3.14
```

15. Write a Python program to count repeated characters in a string.
Sample string: 'thequickbrownfoxjumpsoverthelazydog'
Expected output :
o 4
e 3
u 2
h 2
r 2
t 2


main.py + 44beea2h8 AI PYTHON RUN

```python

s= 'thequickbrownfoxjumpsoverthelazydog'
freq={}
for i in s:
  if i in freq:
    freq[i]+=1
  else:
    freq[i]=1
for i,j in freq.items():
  if j>1:
    print(i,j)
```

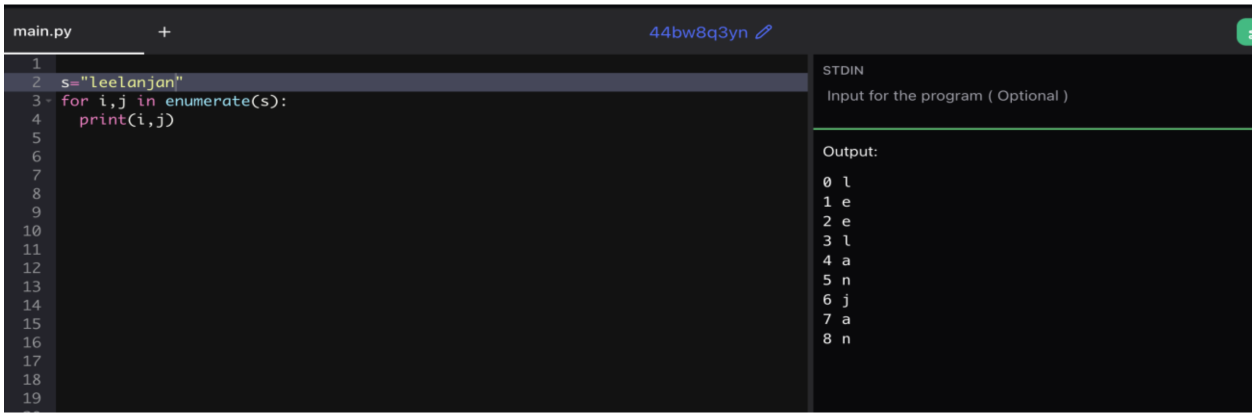
STDIN

Input for the program (Optional)

Output: 10

```
t 2
h 2
e 3
u 2
r 2
o 4
```

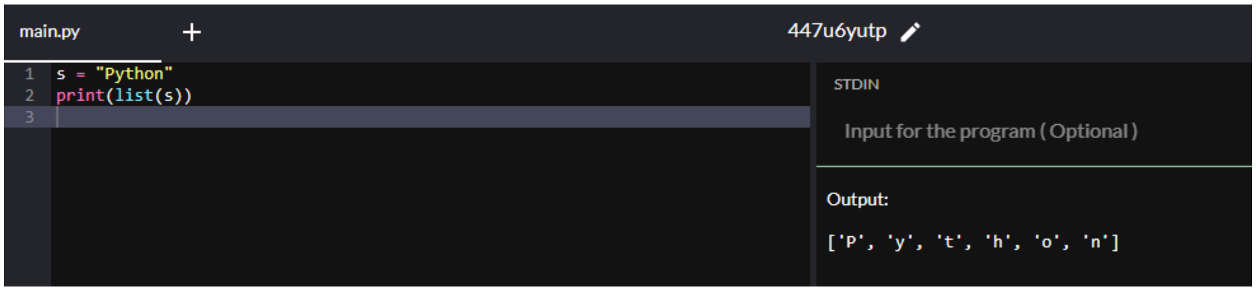
16. Write a Python program to print the index of the character in a string.



```python
s="leelanjan"
for i,j in enumerate(s):
  print(i,j)
```

```
0 l
1 e
2 e
3 l
4 a
5 n
6 j
7 a
8 n
```

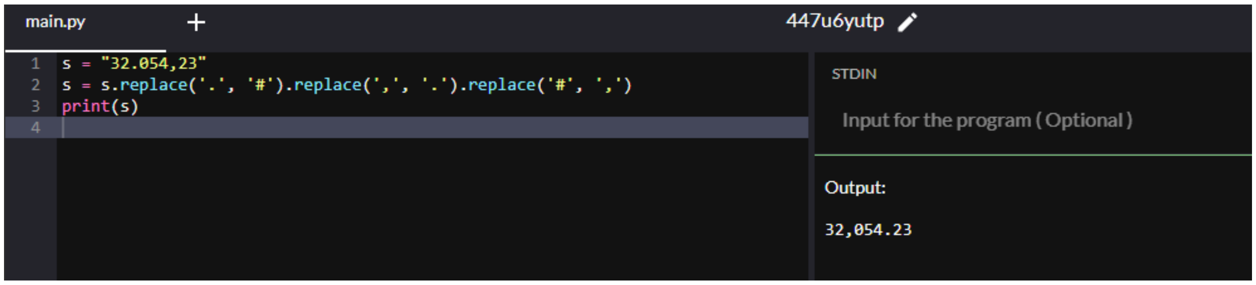
17. Write a Python program to convert a string in a list.



```python
s = "Python"
print(list(s))
```

```
['P', 'y', 't', 'h', 'o', 'n']
```

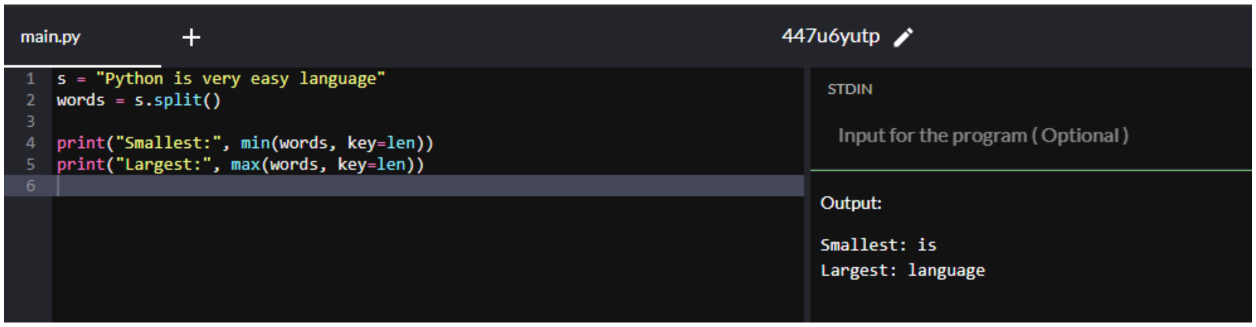
18. Write a Python program to swap comma and dot in a string.
Sample string: "32.054,23"
Expected Output: "32,054.23"



```python
s = "32.054,23"
s = s.replace('.', '#').replace(',', '.').replace('#', ',')
print(s)
```

```
32,054.23
```

19. Write a Python program to find smallest and largest word in a given string.

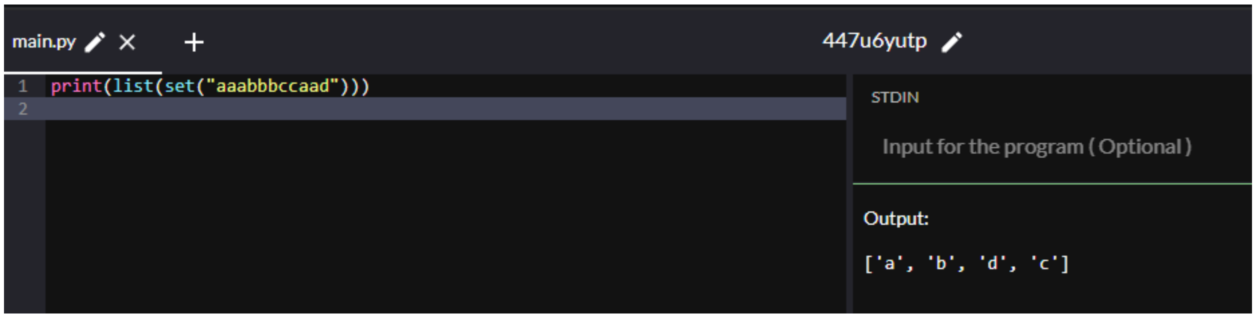


```python
s = "Python is very easy language"
words = s.split()

print("Smallest:", min(words, key=len))
print("Largest:", max(words, key=len))
```

```
Smallest: is
Largest: language
```

20. Write a Python program to remove all consecutive duplicates of a given string.



```python
print(list(set("aaabbbccaad")))
```

```
['a', 'b', 'd', 'c']
```