

# String ??

??

## Concatenate

```
secret_password = 'jhk7GSH8ds'  
print('Password hint: the third letter of your password is ' + secret_password[2])
```

```
# Escaping characters  
introduction = 'Hello, I\'m John!'  
print(introduction)  
  
# Joining strings  
user_age = 28  
user_name = 'John'  
greeting = user_name + ', you are ' + str(user_age) + '!'  
print(greeting)
```

```
s = 'String'  
s += ' Concatenation'  
print(s)
```

```
# Using % NOTE: %%  
s1, s2, s3 = 'Python', 'String', 'Concatenation'  
s = '%s %s %s' % (s1, s2, s3)  
print(s)  
  
# Using format()  
s1, s2, s3 = 'Python', 'String', 'Concatenation'  
s = '{} {} {}'.format(s1, s2, s3)  
print(s)  
  
# Using f-string  
s1, s2, s3 = 'Python', 'String', 'Concatenation'  
s = f'{s1} {s2} {s3}'
```

```
print(s)
```

# Parsing

## split()

- `.split()` : convert a string into a list or multiple variables
- `.split(delimiter)` : convert a string into a list by specified delimiter, default is space.

```
"This is another example".split()
# Return ['This', 'is', 'another', 'example']
```

```
test = "How-much-wood-would-a-woodchuck-chuck"
print(test.split("-")) # prints ['How', 'much', 'wood', 'would', 'a', 'woodchuck', 'chuck']
```

```
removed_users = "wjaffrey jsoto abernard jhill awilliam"
print("before .split():", removed_users)
removed_users = removed_users.split()
print("after .split():", removed_users)
```

```
with open("update_log.txt", "r") as file:
    updates = file.read()
updates = updates.split()
```

```
msg = "2024/12/11|Hello World|aaa@bb.com"
date, title, emails = msg.split("|")
print(date)
```

## join()

`.join()` : convert a list into a string

```
approved_users = ["elarson", "bmoreno", "tshah", "sgilmore", "eraab"]
print("before .join():", approved_users)
approved_users = ",".join(approved_users)
print("after .join():", approved_users)
```

```
with open("update_log.txt", "r") as file:
    updates = file.read()
updates = updates.split()
```

```
updates = " ".join(updates)
with open("update_log.txt", "w") as file:
    file.write(updates)
```

```
# List List
strings = ' '.join(my_list)

# List List
strings = '\n\n'.join(my_list)
```

```
def list_elements(list_name, elements):
    return "The " + list_name + " list includes: " + ", ".join(elements)

print(list_elements("Printers", ["Color Printer", "Black and White Printer", "3-D Printer"]))
# Should print "The Printers list includes: Color Printer, Black and White Printer, 3-D Printer"
```

## index()

`.index()` : get the index of specified character

```
string = "Hello, World"
print(string.index('w'))
```

```
def replace_domain(email, old_domain, new_domain):
    if "@" + old_domain in email:
        index = email.index("@" + old_domain)
        new_email = email[:index] + "@" + new_domain
        return new_email
    return email
```

## replace()

`.replace(old,new)` : Returns a new string where all occurrences of old have been replaced by new

```
test = "How much wood would a woodchuck chuck"
print(test.replace("wood", "plastic")) # prints "How much plastic would a plasticchuck chuck"
```

## Slicing

- Format: string [includ-index : exclude-index]
- Character Index: beginning with zero
- string[-2]: the last two characters

```
string1 = "Greetings, Earthlings"
print(string1[0])  # Prints "G"
print(string1[4:8]) # Prints "ting"
print(string1[11:]) # Prints "Earthlings"
print(string1[:5]) # Prints "Greet"

print(string1[-10:]) # Prints "Earthlings" again
```

```
phonenumber = "2025551212"

# The first 3 digits are the area code:
area_code = "(" + phonenumber[:3] + ")"
# area_code is (202)

# the numbers 4-6 from the list:
exchange = phonenumber[3:6]
# exchange is 555

# the last four numbers:
line = phonenumber[-4:]
# line is 1212
```

## Formating

```
name = "Manny"
number = len(name) * 3
print("Hello {}, your lucky number is {}".format(name, number))
```

```
name = "Manny"
print("Your lucky number is {number}, {name}".format(name=name, number=len(name)*3))
```

```
price = 7.5
with_tax = price * 1.09
print(price, with_tax)
print("Base price: ${:.2f}. With Tax: ${:.2f}".format(price, with_tax))
```

- `{:>3}` ?????3 ???
- `{:>6.2f}` ?????6 ??????? 2 ?
- `{:10,.2f}` 10 ??????????? 2 ?
- `{:.2s}` 2 ?????

```
def to_celsius(x):
    return (x-32)*5/9

for x in range(0,101,10):
    print("{:>3} F | {:>6.2f} C".format(x, to_celsius(x)))
```

```
0 F | -17.78 C
10 F | -12.22 C
20 F | -6.67 C
30 F | -1.11 C
40 F | 4.44 C
50 F | 10.00 C
60 F | 15.56 C
70 F | 21.11 C
80 F | 26.67 C
90 F | 32.22 C
100 F | 37.78 C
```

## f-strings

```
name = "Micah"
print(f'Hello {name}')
```

```
item = "Purple Cup"
amount = 5
price = amount * 3.25
print(f'Item: {item} - Amount: {amount} - Price: {price:.2f}')
```

## More methods

- ???????????? methods
- `.capitalize()` : ????

## strip()

`.strip()` , `.lstrip()` , `.rstrip()`

```
" yes ".strip()    # Return 'yes'
" yes ".lstrip()   # Return 'yes '
" yes ".rstrip()   # Return ' yes'

# Multiple methods
' yes '.upper().strip() # Return 'YES'
```

## count()

`.count()`

```
"The number of times e occurs in this string is 4".count("e")
# Return 4
```

## endswith()

`.endswith()`

```
"Forest".endswith("rest")
# Return True
```

## isnumeric(), isalpha()

`.isnumeric()` , `.isalpha()`

```
"Forest".isnumeric()    # Return False
"12345".isnumeric()     # Return True
"xyzy".isalpha()        # Return True
```

---

Revision #49

Created 15 November 2023 14:52:05 by Admin

Updated 11 December 2024 11:36:11 by Admin