PYTHON : DAY-6

Escape sequences in strings:

Escape sequences allow special characters in strings.

BASIC ESCAPES :

Sequence	Meaning
`\n`	newline
`\t`	tab spacing
`\\`	print a backslash

EXAMPLE :

```
print('Days\tTopics\tExercises')
print('This is a backslash symbol (\\)')
```

String formatting

Python provides multiple ways to insert values into strings.

A. Old Style `%` Formatting

strings:

```
name = "dermin has two nicknames , which are %s and %s" %
(name_1, name_2)
```

Strings + Numbers:

```
formated_string = "the area of the circle with radius %d is %.2f." % (radius, area)
```

Formats used :

Format	Use
`%s`	string
`%d`	integer
`%.2f`	float up to 2 decimals

B. `str.format()` — New Style Formatting

Introduced in Python 3:

```
full_name = "the one learning python is {}{}".format(first_name, last_name)
```

with expressions :

```
print("{} + {} = {}".format(a, b, a+b))
```

C. f-Strings (Python 3.6+):

Most recommended modern way:

```
print(f"{a} + {b} = {a + b}")
```

Fast , clean , readable

String Slicing:

slicing format :

```
string[start:end:step]
```

EXAMPLE :

```
s = "hello python"
print(s[0:5])  # hello
print(s[6:])   # python
print(s[::-1]) # reverse
```

Other slicing :

```
language = "python"
print(language[0::2])  # pto
print(language[2:6:2]) # to
```

Common String Methods :

Method	Meaning
`capitalize()`	First letter uppercase
`count()`	Count occurrences
`endswith()`	Check suffix
`expandtabs()`	Convert `\t` to spaces
`find()`	Returns index of first occurrence
`rfind()`	Returns index of last occurrence
`replace()`	Replace part of string
`strip()`	Removes characters from edges
`title()`	Capitalizes each word
`swapcase()`	Swap UPPER↔lower
`startswith()`	Check prefix
`join()`	Join list into string

Example :

```
s = "dermin is not so cool"
print(s.capitalize()) # Dermin is not so cool
```

Example of **join():**

```
web_tech = ["HTML", "CSS" , "JAVA"]
result = "#".join(web_tech)
print(result)  # HTML#CSS#JAVA
```

Character Classification Methods:

These return `True` or `False`

Method	Checks if...
`isalnum()`	alphanumeric (A-Z, 0-9)
`isalpha()`	alphabet only (A-Z)
`isdecimal()`	decimal digits only
`isdigit()`	digits (incl. unicode numbers)
`isnumeric()`	numeric values (fractions, superscripts etc.)
`isidentifier()`	valid Python identifier
`islower()`	all lowercase
`isupper()`	all uppercase

Examples:

```
"3monkeys".isalnum()  # True
"three monkeys".isalnum() # False (space not allowed)
```

Identifier example :

```
"three_lil_monkeys".isidentifier() # True
"3lilmonkeys".isidentifier()       # False
```

RADIUS CALCULATOR :

```
radius = float(input("enter the radius :"))
pi = 3.14
area = pi * radius**2
```

Formatted result :

```
result = "the area of the circle with radius {} is {}".format(radius, area)
```

Summary :

✔️ Escape characters

✔️ Old style `%` formatting

✔️ `.format()` method

✔️ f-strings

✔️ String slicing

✔️ Built-in string methods

✔️ Character test methods

✔️ Practical input/output example