{CODE} THREADS

#5
Interesting Python Tips

#1

" You can use the **enumerate** function to iterate over a list and also get the index of each element "

Example:

```python
my_top_ten_movies = ["The Shawshank Redemption", "The Godfather",
                     "The Dark Knight", "12 Angry Men",
                     "Schindlers List", "Forrest Gump", "Inception",
                     "Saving Private Ryan", "Joker",
                     "Parasite", "WALL·E"]

for rank, movie in enumerate(my_top_ten_movies):
    print(f"{rank} : {movie}")
```

Output:

```
#mycomputer@terminal:~$python code_thread_tip1.py
0 : The Shawshank Redemption
1 : The Godfather
2 : The Dark Knight
3 : 12 Angry Men
4 : Schindlers List
5 : Forrest Gump
6 : Inception
7 : Saving Private Ryan
8 : Joker
9 : Parasite
10 : WALL·E
```

#2

" You can use the <u>zip</u> function to iterate over multiple lists at the same time "

Example:

```python
american_english_words = ['color', 'labor', 'defense',
                          'theater', 'fiber']
british_english_words = ['colour', 'labour', 'defence',
                         'theatre', 'fibre']

for american_word, british_word in \
        zip(american_english_words, british_english_words):
    print(f"American: {american_word}, British: {british_word}")
```

Output:

```
#mycomputer@terminal:~$python code_thread_tip2.py
American: color, British: colour
American: labor, British: labour
American: defense, British: defence
American: theater, British: theatre
American: fiber, British: fibre
```

#3

" You can use the <u>any</u> and <u>all</u> functions to check if any or all elements of an iterable are true, respectively "

Example:

```python
numbers = [12, 22, 9, 18, -1, 19, 20, 25]

if any(x < 0 for x in numbers):
    print("There is at least one negative number in the list")

if all(x > 0 for x in numbers):
    print("All the numbers in the list are positive")
```

Output:

```
#mycomputer@terminal:~$python code_thread_tip3.py
There is at least one negative number in the list
```

#4

" You can use the <u>defaultdict</u> class from the collections module to create a dictionary with a default value for non-existent keys. This can be useful if you want to avoid having to check whether a key exists before accessing it "

Example:

```python
from collections import defaultdict

# Create a defaultdict with a default value of 0
my_dict = defaultdict(int)

# You can now access keys without checking whether they exist
print('output is', my_dict[5])  # prints 0
```

Output:

```
#mycomputer@terminal:~$python code_thread_tip4.py
output is 0
```

#5

" In Python 3, you can use the print function to write to a file by redirecting the output to a file using the file keyword argument "

Example:

```python
# Open a file for writing
with open("output.txt", "w") as f:
    # Write to the file using the print function
    print("Hello, world!", file=f)

# "output.txt" will now contain the string "Hello, world!"
```

For more Interesting tips please follow CodeThreads Dev

For more insightful articles, visit codethreads.dev

Happy Coding!
with
codethreads.dev

https://codethreads.dev