Question1
Create a function that takes a string and returns a string in which each character is repeated once.
Examples
double_char("String") ➞ "SSttrriinngg"
double_char("Hello World!") ➞ "HHeellllloo WWoorrlldd!!"
double_char("1234!_ ") ➞ "11223344!!__ "

Question2
Create a function that reverses a boolean value and returns the string "boolean expected"
if another variable type is given.
Examples
reverse(True) ➞ False
reverse(False) ➞ True
reverse(0) ➞ "boolean expected"
reverse(None) ➞ "boolean expected"

Question3
Create a function that returns the thickness (in meters) of a piece of paper after folding it n number of times. The paper starts off with a thickness of 0.5mm.
Examples
num_layers(1) ➞ "0.001m"
Paper folded once is 1mm (equal to 0.001m)
num_layers(4) ➞ "0.008m"
Paper folded 4 times is 8mm (equal to 0.008m)
num_layers(21) ➞ "1048.576m"
Paper folded 21 times is 1048576mm (equal to 1048.576m)

Question4

Create a function that takes a single string as argument and returns an ordered list containing
the indices of all capital letters in the string.
Examples
index_of_caps("eDaBiT") ➞ [1, 3, 5]
index_of_caps("eQuINoX") ➞ [1, 3, 4, 6]
index_of_caps("determine") ➞ []
index_of_caps("STRIKE") ➞ [0, 1, 2, 3, 4, 5]
index_of_caps("sUn") ➞ [1]

Question5
Using list comprehensions, create a function that finds all even numbers from 1 to the given number.
Examples
find_even_nums(8) ➞ [2, 4, 6, 8]
find_even_nums(4) ➞ [2, 4]
find_even_nums(2) ➞ [2]

Answer 1:

Here is the Python function to repeat each character in a string once:

python
```
def double_char(string):
    return ''.join([char * 2 for char in string])
```

The `join()` function concatenates all the characters returned by the list comprehension, which multiplies each character in the input string by 2.

Answer 2:

Here is the Python function to reverse a boolean value:

python
```
def reverse(arg):
    if type(arg) == bool:
        return not arg
    else:
        return "boolean expected"
```

The function first checks if the input argument is a boolean type. If it is, the function returns the opposite value (i.e., `True` becomes `False`, and vice versa). If the input is not a boolean type, the function returns the string "boolean expected".

Answer 3:

Here is the Python function to compute the thickness of a folded paper:

python
```
def num_layers(n):
    thickness = 0.5 / 1000  # convert mm to m
    folded_thickness = thickness * (2 ** n)
    return '{:.3f}m'.format(folded_thickness)
```

The function computes the thickness of a single layer of paper in meters, which is 0.5 mm (as given in the problem). Then, it uses the formula `folded_thickness = thickness * (2 ** n)` to compute the thickness of the folded paper after `n` folds. Finally, it formats the result to have 3 decimal places and appends the unit "m" to indicate meters.

Answer 4:

Here is the Python function to find the indices of capital letters in a string:

python

```
def index_of_caps(string):
    return [index for index, char in enumerate(string) if
char.isupper()]
```

The list comprehension iterates over each character in the input string and uses the `isupper()` method to check if the character is uppercase. If it is, the index of that character is added to the output list using the `enumerate()` function.

Answer 5:

Here is the Python function to find all even numbers from 1 to the given number:

python

```
def find_even_nums(n):
    return [num for num in range(1, n+1) if num % 2 == 0]
```

The list comprehension generates all numbers from 1 to `n` using the `range()` function, and filters out the odd numbers using the modulo operator `%`. The resulting list contains only the even numbers.