

Exercises

1.

Write a function
printItems(list)
that:

- Prints every item in a list

2.

Write a function
sumItems(list)
that:

- Sums every item in a list

Example: `sumItems[1, 2, 3] => 6`

3.

Write a function
generateList(value, len)
that:

- Returns an array containing the same **value** repeated **len** times

Example: `generateList('x', 3) => ['x', 'x', 'x']`

4.

Write a function

generateRange(n, m)

that:

- Returns an array containing the specified range of numbers in increasing order

Example: `generateRange(2, 5)` => `[2, 3, 4, 5]`

Example: `generateRange(5, 2)` => `[2, 3, 4, 5]`

5.

Write a function
repeatList(list, n)
that:

- Returns you list repeated **n** times

Example: `repeatList([1, 2, 3], 2)`
=> `[1, 2, 3, 1, 2, 3]`

6.

Write a function
repeatItems(list, n)
that:

- Returns a list with every item repeated **n** times

Example: `repeatItems([1, 2, 3], 2)`
=> `[1, 1, 2, 2, 3, 3]`

7.

Write a function

mergeLists(list1, list2)

that:

- Merges **list1** and **list2** and returns it as an array
- Only adds the items from **list2** that does not already exist in **list1**

Example: `mergeLists([1, 1, 2, 3], [3, 4, 5]) =>`
`[1, 1, 2, 3, 4, 5]`

8.

Write a function
isEqual(list1, list2)
that:

- Checks if **list1** is equal to **list2** and returns either **true** or **false**

Example: `isEqual([1, 2, 3], [1, 2, 3]) => true`

Example: `isEqual([1, 2, 3], [2, 3, 4]) => false`

9.

Write a function

isSublistOf(list1, list2)

that:

- Checks if **list2** is a sublist of **list1** and returns either **true** or **false**

Example: `isSublistOf([1, 2, 3, 4], [2, 3]) => true`

Example: `isSublistOf([1, 2, 3, 4], [1, 4]) => false`

Example: `isSublistOf([1, 2, 3, 4], [5, 6]) => false`

10.

Write a function

flipCase(str)

that:

- Flips the case of each character in a string

Examples:

```
flipCase("Hello World") => "hELLO wORLD"
```

```
flipCase("HaHaHa") => "hAhAhA"
```

11.

Write a function
removeDuplicates(list)
that:

- Removes any duplicate items from a list

Examples:

`removeDuplicates([1, 1, 2, 2, 3]) => [1, 2, 3]`

`removeDuplicates([1, 2, 2, 2, 3]) => [1, 2, 3]`

12.

Write a function
getDuplicates(list)
that:

- Returns any duplicate items in a list

Examples:

`getDuplicates([1, 1, 2, 2, 3]) => [1, 2]`

`getDuplicates([1, 2, 2, 3, 3]) => [2, 3]`

`getDuplicates([2, 2, 2, 3, 3]) => [2, 3]`

13.

Write a function

sortList(list)

that:

- Returns a sorted list (without using [].sort!)

Examples:

```
sortList([5, 4, 3, 2, 1]) => [1, 2, 3, 4, 5]
```

```
sortList([1, 1, 9, 4, 3]) => [1, 1, 3, 4, 9]
```