**1. concat() Exercises  
Q1: Given the following arrays:**

**let arr1 = [1, 2, 3];  
let arr2 = [4, 5, 6];  
Use the concat() method to create a new array that combines both arr1 and arr2.**

let arr1 = [1, 2, 3];

let arr2 = [4, 5, 6];

let combinedArr = arr1.concat(arr2);

console.log(combinedArr); // Output: [1, 2, 3, 4, 5, 6]

**Q2:  
Given:**

**let names1 = ["Alice", "Bob"];  
let names2 = ["Charlie", "David"];  
let names3 = ["Eve", "Frank"];  
Write a TypeScript statement that merges all three arrays into a single array.**

let names1 = ["Alice", "Bob"];

let names2 = ["Charlie", "David"];

let names3 = ["Eve", "Frank"];

let mergedNames = names1.concat(names2, names3);

console.log(mergedNames); // Output: ["Alice", "Bob", "Charlie", "David", "Eve", "Frank"]

**Q3:  
What will be the output of the following code?**

**let arr1 = [true, false];  
let arr2 = ["Yes", "No"];  
let arr3 = arr1.concat(arr2, [1, 2, 3]);  
console.log(arr3);**

let arr1 = [true, false];

let arr2 = ["Yes", "No"];

let arr3 = arr1.concat(arr2, [1, 2, 3]);

console.log(arr3); // Output: [true, false, "Yes", "No", 1, 2, 3]

**2. copyWithin() Exercises  
Q4: Given the following array:**

**let numbers = [10, 20, 30, 40, 50];  
Use the copyWithin() method to replace the first two elements with the last two elements of the array.**

let numbers = [10, 20, 30, 40, 50];

numbers.copyWithin(0, 3, 5);

console.log(numbers); // Output: [40, 50, 30, 40, 50]

**Q5: What will be the output of the following code?**

**let fruits = ["apple", "banana", "cherry", "date", "elderberry"];  
fruits.copyWithin(1, 3, 5);  
console.log(fruits);**

let fruits = ["apple", "banana", "cherry", "date", "elderberry"];

fruits.copyWithin(1, 3, 5);

console.log(fruits);

// Output: ["apple", "date", "elderberry", "date", "elderberry"]

**Q6:  
Given:**

**let arr = [1, 2, 3, 4, 5, 6];  
Use copyWithin() to copy elements starting from index 2 and place them at index 0.**

 let arr = [1, 2, 3, 4, 5, 6];

arr.copyWithin(0, 2);

console.log(arr); // Output: [3, 4, 5, 6, 5, 6]

**3. every() Exercises  
Q7:  
Given the following array:**

**let numbers = [2, 4, 6, 8, 10];  
Use the every() method to check if all elements in the array are even numbers.**

let numbers = [2, 4, 6, 8, 10];

let allEven = numbers.every(num => num % 2 === 0);

console.log(allEven); // Output: true

**Q8:  
What will be the output of the following code?**

**let words = ["apple", "banana", "avocado"];  
let result = words.every(word => word.startsWith("a"));  
console.log(result);**

let words = ["apple", "banana", "avocado"];

let result = words.every(word => word.startsWith("a"));

console.log(result); // Output: False

**Q9:  
Write a TypeScript function that takes an array of numbers as input and returns true if all numbers in the array are positive. Use the every() method inside the function.**

function areAllPositive(numbers: number[]): boolean {

return numbers.every(num => num > 0);

}

let numbers = [1, 2, 3, 4];

console.log(areAllPositive(numbers)); // Output: true

let numbers2 = [1, -2, 3, 4];

console.log(areAllPositive(numbers2)); // Output: false

**4. fill() Exercises**

**Q10: Given the array:  
let arr = [1, 2, 3, 4, 5];  
Use the fill() method to replace all elements with 0.**

let arr = [1, 2, 3, 4, 5];

arr.fill(0);

console.log(arr); // Output: [0, 0, 0, 0, 0]

**Q11:  
What will be the output of the following code?**

**let arr = ["A", "B", "C", "D", "E"];  
arr.fill("X", 1, 4);  
console.log(arr);**

let arr = ["A", "B", "C", "D", "E"];

arr.fill("X", 1, 4);

console.log(arr); // Output: ["A", "X", "X", "X", "E"]

**Q12:  
Use the fill() method to update only the last three elements of the following array with "done".**

**let status = ["pending", "pending", "pending", "pending", "pending"];**

let status = ["pending", "pending", "pending", "pending", "pending"];

status.fill("done", 2);

console.log(status); // Output: ["pending", "pending", "done", "done", "done"]