JS-A2#3

Task 1	Watch the lesson video carefully twice (at least). Then answer the following a. Find, findIndex and filter questions solved in the video i. Element greater than x ii. Names of people and ch iii. Convert the code into fns iv. Employees with dept and city b. Sort questions solved in the video i. Sort array in ascending order of length of strings ii. Sort array in descending order of sales value iii. Filters array on mincut and sorts in descending order c. Explain find, findIndex and filter. Explain their syntax and return values also. d. What does find and findIndex return if no element in the array matches. e. What is the spread operator. Why do we need it. Explain with example. f. Explain sort function in arrays. Explain it's syntax. g. Does sort create a new array or does it sort the array in place. What should be done if the original array is not to be changed.

Task 2.1	Given an array of numbers, find the number and its index for a) equal to 10 b) greater than 30 c) multiple of 3 Create suitable test cases and test properly
Task 2.2	Given an array of numbers, filter the array a) equal to 10 b) greater than 30 c) multiple of 3 Create suitable test cases and test properly

Task 3.1	Given an array of names, find the name and its index for a) length of name is greater than 5 b) starts with P c) starts with the character provided as a parameter Create suitable test cases and test properly
Task 3.2	Given an array of names, filter the array for a) length of name is greater than 5 b) starts with P c) starts with the character provided as a parameter Create suitable test cases and test properly

Task 4.1	Given an array of employees where each employee is a JSON with name and age, find the employee and its index for a) age less than 35 b) name of the employee starts with S c) name of the employee starts with the character provided as parameter Create suitable test cases and test properly
Task 4.2	Filter the above array of employees for a) age less than 35 b) name of the employee starts with S

JS-A2#3

	c) name of the employee starts with the character provided as parameter

Task 5.1	Given an array of string, find the first string which ends in t. Also find its index. Also filter the array. For e.g. if the array was ['Hello', 'React', 'Java', 'Python', 'Hibernate', 'JavaScript'] the output should be React and index should be 1 and the output of filter should be ['React', 'JavaScript']
Task 5.2	Given a character ch, find the first string which ends in ch. Also finds its index. Also filterthe array. So, for the above array if ch = 'a', the output should be Java and 2 and filter should be ['Java'] If ch="b" the output should be undefined, -1 and []
Task 5.3	Given a character ch, find the first string has the ch in it. Also finds its index and filter the array. So, for the above array if ch = 'y', the output should be Python, 3 and ['Python'] If ch='a' the output should be React, 1 and ['React', 'Java', 'Hibernate', 'JavaScript'].

	Given an array of JSON representing students, [{"name":"Jack","maths":55,"english":60,"science":62} {"name":"Anita","maths":62,"english":65,"science":56} {"name":"Thomas","maths":68,"english":72,"science":75} {"name":"Steve","maths":51,"english":56,"science":68} {"name":"Julia","maths":47,"english":77,"science":72} {"name":"Mary","maths":72,"english":55,"science":81}]
Task 6.1	Find the first student who has scored less than 50 in Maths. Also find its index.
Task 6.2	Find the first student whose total marks in more than 200. Also find its index.
Task 6.3	Find the first student who scored more in maths than english. Also find its index.
Task 6.4	Given a number n, find the first student who scored more than n in all subjects. Also find its index.
Task 6.5	Filter the array for students who scored less than 50 in Maths.
Task 6.6	Filter the array for students with total marks more than 200. Also find its index.
Task 6.7	Filter the array for students who scored more in maths than english.
Task 6.8	Given a number n, filter the array for students who scored more than n in all subjects.
Task 6.9	Given two numbers n1 and n2 as input, filter the array for those whose marks in maths is more than n1 and marks in science is more than n2.

	Create an array of the JSONs {"name":"Jack","country":"USA","age":35} {"name":"Amit","country":"India","age":38} {"name":"Edward","country":"USA","age":41} {"name":"Vishal","country":"India","age":30} {"name":"Annie","country":"USA","age":27} {"name":"Nick","country":"France","age":32} {"name":"Francis","country":"France","age":44} {"name":"Preeti","country":"India","age":25}

JS-A2#3

	{"name":"Sophie","country":"France","age":29} {"name":"Harpreet","country":"India","age":48} {"name":"Bob","country":"USA","age":21}
Task 7.1	Filter the array for those whose country is India. Also find and findIndex the first JSON whose country is India.
Task 7.2	Filter the array for those whose country is India and age is less than 35. Also find and findIndex the first JSON whose country is India and age is less than 35.
Task 7.3	Filter the array for those whose names has less than 6 characters. Also find and findIndex the first JSON whose names has less than 6 characters.
Task 7.4	Filter the array for those whose country is either India or USA. Also find and findIndex the first JSON whose country is either India or USA.
Task 7.5	Given a number minAge, filter the array for those whose age is greater than minAge. Also find and findIndex the first JSON whose age is greater than minAge.
Task 7.6	Given a string str as input, filter the array for those whose country is one specified in str.

	Create an array of the JSONs. {"name":"Mary","marks":[72,65,55,71]} {"name":"Anita","marks":[66,70,75,53]} {"name":"Edward","marks":[44,54,64,58]} {"name":"Thomas","marks":[62,55,65,81]} {"name":"Robin","marks":[41,44,47,49]} {"name":"Sophia","marks":[71,73,67,77]} {"name":"Bruce","marks":[52,57,61,64]}
Task 8.1	Filter the array for those whose total marks is less than 250.
Task 8.2	Filter the array for those who have atleast 1 mark more than 70
Task 8.3	Filter the array for those who have scored 2 or more marks greater than 70.
Task 8.4	Given a number avg, filter the array for those whose average marks is greater than avg.

Task 9	Define a function that takes an array of JSON, a string and a boolean as parameters. If the the boolean is true, filter and return the array whose name match the given string. If the boolean is false, filter and return the array who tech match the given string. [{"name":"Jack","tech":"Android"},{"name":"Mary","tech":"React"}, {"name":"Bob","tech":"Angular"},{"name":"Steve","tech":"Spring"}, {"name":"James","tech":"Android"},{"name":"Julia","tech":"Android"}, {"name":"Michel","tech":"React"},{"name":"Bill","tech":"Angular"}, {"name":"Sonny","tech":"Spring"},{"name":"Martins","tech":"React"}, {"name":"Bruce","tech":"Angular"},{"name":"Sam","tech":"Spring"}]

Task 10	**Given an array of string, create a string as shown. Use join.**	
Test Cases	**Input**	**Expected Output**
	["Hi","Hello","Bye"]	Hi#Hello#Bye
	["A","B","C","X","Y"]	A#B#C#X#Y
	["AB","!2","EF"]	AB#12#EF

JS-A2#3

Task 11	Define a function takes an array and separator as parameters and returns a string with the array elements in a string separated by the separator. Use join.	
Test Cases	**Input**	**Expected Output**
	["Hi","Hello","Bye"], '**'	Hi**Hello**Bye
	["A","B","C"], 'Level:'	ALevel:BLevel:CLevel:
	["AB","!2","EF"], '#'	AB#12#EF

Task 12	Given an array of numbers like [12,5,-6,10,-9,4,0,-23,34] a) Sort the numbers in ascending order b) sort the numbers in descending order c) sort the numbers based on their absolute value in ascending order Create suitable test cases and test properly

Task 13	Given an array of strings like ["JavaScript","Hello","React","Java","Python","C","Node"] a) Sort the array in ascending order as in dictionary b) Sort the array in descending order as in dictionary c) Sort the array in ascending order of their length d) Sort the array in ascending order of the number of 'a' in them Create suitable test cases and test properly

	Given an array of JSON like {"name":"Jack","maths":55,"english":60,"science":62} {"name":"Anita","maths":62,"english":65,"science":56} {"name":"Thomas","maths":68,"english":58,"science":54} {"name":"Steve","maths":51,"english":56,"science":68} {"name":"Julia","maths":47,"english":77,"science":72} {"name":"Mary","maths":72,"english":55,"science":60}
Task 14.1	Sort the array in ascending order of name.
Task 14.2	Sort the array in descending order of name.
Task 14.3	Sort the array in ascending order of total marks in 3 subjects.
Task 14.4	Sort the array in descending order marks in maths

	Given an array of JSON like {"name":"Jack","country":"USA","age":35} {"name":"Amit","country":"India","age":38} {"name":"Edward","country":"USA","age":41} {"name":"Vishal","country":"India","age":30} {"name":"Annie","country":"USA","age":27} {"name":"Preeti","country":"India","age":25} {"name":"Sophie","country":"France","age":29} {"name":"Harpreet","country":"India","age":48} {"name":"Bob","country":"USA","age":21}
Task 15.1	Sort the array in ascending order of name
Task 15.2	Sort the array in descending order of age.

JS-A2#3

Task 15.3	Sort the array in ascending order of country and name, i.e. it should be sorted by country and within country sorted by name.
Task 15.4	Sort the array in ascending order of country and age, i.e. it should be sorted by country and within country sorted by age.
Task 15.5	Define a function that takes a country as parameter and returns an array of people in that country sorted by age in descending order.

	Given an array of JSON like {"name":"Mary","marks":[72,65,55,71]} {"name":"Anita","marks":[66,70,75,53]} {"name":"Edward","marks":[44,54,64,58]} {"name":"Thomas","marks":[62,55,65,81]} {"name":"Robin","marks":[41,44,47,49]} {"name":"Sophia","marks":[71,73,67,77]} {"name":"Bruce","marks":[52,57,61,64]}
Task 16.1	Sort the array in descending order of sum of marks array.
Task 16.2	Sort the array in ascending order of maximum value in marks array.
Task 16.3	Given a number cutoff, count the number of elements in the marks array greater than cutoff. Sort the JSON array in descending order of this count.
Task 16.4	Define a function that takes a minTotal as parameter and returns an array of JSON whose total marks is greater than minTotal sorted by total score in descending order.
Task 16.5	Define a function that takes a minScore as parameter and returns an array of JSON in which all the entries in marks array is greater than minScore sorted by name in ascending order.

Task 17	Given an array of string as input. Count the number of a,b or c in the string. Sort the array in the ascending order of this count. If the count is the same, the string with smaller length appears earlier.	
Test Cases	**Input**	**Expected Output**
	[“aa”,”car”,”is”,”going”]	["is","going","car","aa"]
	[“q”,”w1”,”c2”,”cat”,"abcd","e34","a",]	[“q”,”w1”,”e34”,”a”,”c2”,”cat”,"abcd"]
	[“bear”,”bell”,”cab”,”hello”]	["hello","bell","bear","cab"]

Task 18	Given an array of string as input. Each string represents a number around which other characters have been put. So the string "ab4n#56b2Y" actually represents the number 4562. Sort the array in the ascending order of the number each string in it represents.	
Test Cases	**Input**	**Expected Output**
	[“c101”,”mB24x”,”a4b6n7”,"zzz2z"]	["zzz2z","mB24x","c101","a4b6n7"]
	[“8”,”9”,”10”,”aX2”]	["aX2","8","9","10"]
	[“b1b2b3a”,”a125","a80zz"]	["a80zz","b1b2b3a","a125"]

Task 19	Given an array of string. Fllter the array for strings that have at least 1 a, b or c.	
Test Cases	**Input**	**Expected Output**
	[“aa”,”car”,”is”,”going”]	["car","aa"]
	[“q”,”w1”,”c2”,”cat”,"abcd","e34","a",]	[”a”,”c2”,”cat”,"abcd"]

JS-A2#3

	[“bear”,”bell”,”cab”,”hello”]	"bell","bear","cab"]

Task 20	Given an array of string as input. Filter the array for strings that have at least 1 uppercase alphabet.	
Test Cases	**Input**	**Expected Output**
	["name","Java","Master","tech"]	["Java","Master"]
	[“Hello”,”how”,”are”,”you”]	["Hello"]
	[“JS”,”ANGULAR","REACT","HTML"]	["HTML","ANGULAR","JS","REACT"]
	["12","abc","##"]	[]

	Given an array of JSONs of persons with their name and city [{"name":"Mary","city":"London"}, {"name":"Anita","city":"Paris" }, {"name":"Edward","city":"New York"}, {"name":"Thomas","city":"Rome"}, {"name":"Robin","city":"Seattle"}, {"name":"Sophia","city":"Los Angeles"}, {"name":"Bruce","city":"Delhi"}] You have also been given an array of JSON which tells the continent of each city. [{"city":"London","continent":"Europe"}, {"city":"Delhi","continent":"Asia"}, {"city":"Seattle","continent":"North America"}, {"city":"Paris","continent":"Europe"}, {"city":"New York","continent":" North America"}, {"city":"Rome","continent":"Europe"}, {"city":"Bengaluru","continent":"Asia"}, {"city":"Los Angeles","continent":"North America"}]
Task 21.1	Define a function that takes personArray, continentArray and continent as parameters and returns a array of persons in that continent. Hint: Filter the first array based on the continent provided as parameter. Use find in the filter function to determine the continent of the person.
Task 21.2	Sort the personArray in ascending order of their continent. Hint: In the sorting function, use find to determine the continent of the person.