callback

Tuesday, March 15, 2022 2:48 PM

- When a function takes a callback there are few different way to pass the call back

```
const  values= [ 2, 3, 9, 8, 11, 22, 49, 70 ];
```

1. Write a function separately and pass it as parameter

```
function  printResult( value ) {

    console.log(value);

}

values.forEach( printResult);
```

2. If you don't have a function you can create anonymous function

```
values.filter (  function(value) {   return  value>20; } );
```

3. Or you can use a lambda function

```
values.filter (  value =>  value>20 );
```

4. Suppose you already function that can do your work

- Find all prime number using isPrime function

```
values. filter  (   n =>   isPrime( n ) );
```

- If your lambda is taking a parameter and simply passing to another function you don't need the lambda, you can simply pass the function

```
values. filter  (  isPrime );
```

Remember

- A lambda function is still a function
- You are passing a function and not calling it in
- The function is called by filter function when i it.