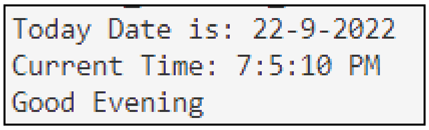
Practical: 5

Que:1 Create an event emitter instance and register a listener which calculates current date and time and display greeting based on time.

```
const event = require('events');
const e = new event.EventEmitter();
e.on('clicked',function(){
    const date = new Date()
    var hour = date.getHours();
    var min = date.getMinutes();
    console.log("Today Date is: "+date.getDate()+
    "-"+(parseInt(date.getMonth())+1)+"-"+date.getFullYear());
    if(hour >= 12)
    {
        console.log("Current Time: "
        +(hour%12)+":"+min+":"+date.getSeconds()+" PM");
    }
    else{
        console.log("Current Time: "
        +(hour%12)+":"+min+":"+date.getSeconds()+" AM");
    }
    if(hour>=6 && hour<=11 && min>=0 && min<=59)
    {
        console.log("Good Morning");
    }
    else if(hour>=12 && hour<=15 && min>=0 && min<=59)
    {
        console.log("Good Afternoon");
    }
    else if(hour>=16 && hour<=19 && min>=0 && min<=59)
    {
        console.log("Good Evening");
    }
    else
    {
        console.log("Good Night");
    }
});
e.emit('clicked');
```

Output:

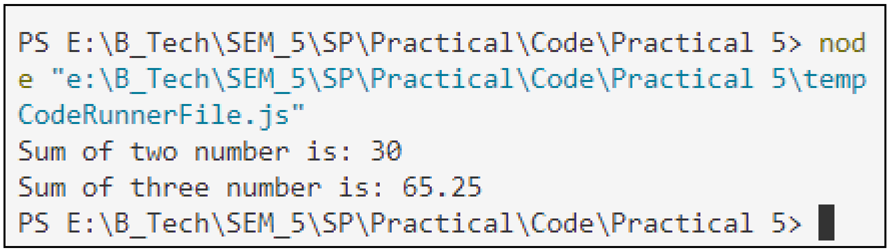


```
Today Date is: 22-9-2022
Current Time: 7:5:10 PM
Good Evening
```

Que:2 Create an event and register two listeners to perform sum of two values with different arguments (use different data types).

```
const event = require('events');
const e = new event.EventEmitter();
e.once('event-1',(num1,num2)=>{
    console.log("Sum of two number is: "+(num1+num2));
});
e.once('event-2',(num1, num2,num3)=>{
    console.log("Sum of three number is: "+(num1+num2+num3));
});
e.emit('event-1',10,20);
e.emit('event-2',10,20.25,35);
```

Output:



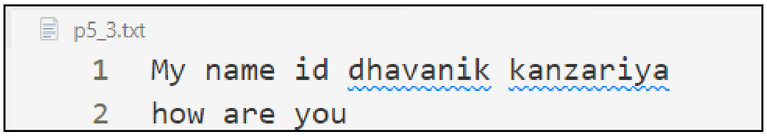
```
PS E:\B_Tech\SEM_5\SP\Practical\Code\Practical 5> nod
e "e:\B_Tech\SEM_5\SP\Practical\Code\Practical 5\temp
CodeRunnerFile.js"
Sum of two number is: 30
Sum of three number is: 65.25
PS E:\B_Tech\SEM_5\SP\Practical\Code\Practical 5> █
```

Que:3 Create event which performs following tasks synchronously:

a. Read file content
b. Display content of file
c. Find out number of words
d. Display numbers of words and finishing message `const event = require('events');`

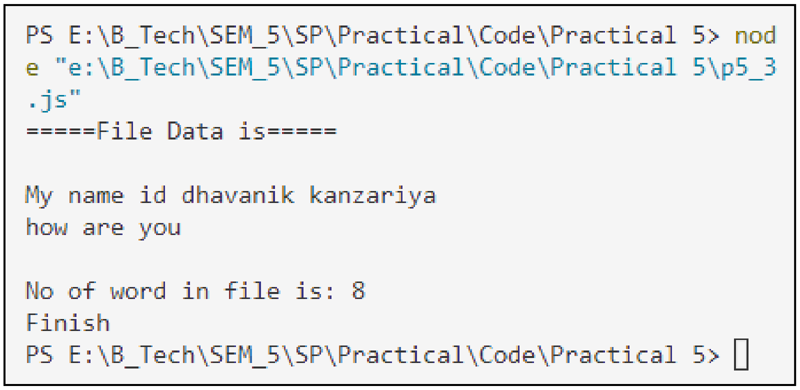
```
const fs = require('fs');
const e = new event.EventEmitter();

e.once('event-read',(fname)=>{
    fs.readFile(fname,function(err,data){
        if(err)
        {
            console.log(err.message);
        }
        else{
            console.log("=====File Data is=====\n")
            console.log(data.toString()+"\n");
            e.emit('event-word-cal',data.toString());
        }
    });
});
e.once('event-word-cal',(data)=>{
    var str = data.replace('\r\n',' ').split(' ');
    console.log("No of word in file is: "+(str.length));
    console.log("Finish");
});
e.emit('event-read','./p5_3.txt');
```

Output:



p5_3.txt

```
My name id dhavanik kanzariya
how are you
```



```
PS E:\B_Tech\SEM_5\SP\Practical\Code\Practical 5> nod
e "e:\B_Tech\SEM_5\SP\Practical\Code\Practical 5\p5_3
.js"
=====File Data is=====

My name id dhavanik kanzariya
how are you

No of word in file is: 8
Finish
PS E:\B_Tech\SEM_5\SP\Practical\Code\Practical 5> 
```

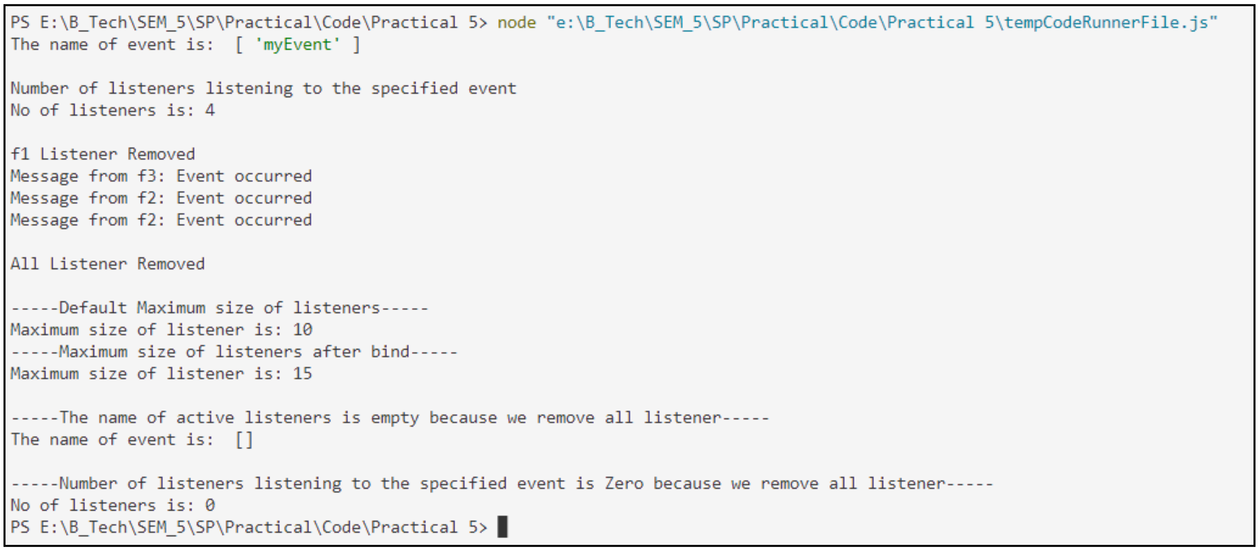
Que:4 Do as directed:

- **a.** Remove a specific listener.
- **b.** Remove all listeners.
- **c.** How to get the maximum size of listener that will bind with a specific event?
- **d.** How to change the maximum size of listeners to bind with specific events?
- **e.** Find out the name of active listeners for the specified event.
- **f.** Find out the number of listeners listening to the specified event.
- **g.** How to change the execution position of listeners?

```
const EventEmitter = require('events');
var eventEmitter = new EventEmitter();

var f1= (msg) => {
    console.log("Message from f1: " + msg);
};

var f2 = (msg) => {
    console.log("Message from f2: " + msg);
};

var f3 = (msg) => {
    console.log("Message from f3: " + msg);
};

eventEmitter.on('myEvent', f1);
eventEmitter.on('myEvent', f2);
eventEmitter.on('myEvent', f2);
//Change the execution position of listener by using prependListener
eventEmitter.prependListener('myEvent', f3);

//get name of all active listener
console.log("The name of event is: ",eventEmitter.eventNames());

//get number of listeners listening to the specified
console.log("\nNumber of listeners listening to the specified event")
console.log("No of listeners is: "+eventEmitter.listenerCount('myEvent'))

//Remove a specific listener.
eventEmitter.removeListener('myEvent', f1);
console.log("\nf1 Listener Removed");
eventEmitter.emit('myEvent', "Event occurred");

//Remove all listeners.
eventEmitter.removeAllListeners('myEvent');
console.log("\nAll Listener Removed\n");
eventEmitter.emit('myEvent', "Event occurred");

//get maximum size of listener
console.log("-----Default Maximum size of listeners-----")
console.log("Maximum size of listener is: "+eventEmitter.getMaxListeners())
```

```
//set maximum size of listener
eventEmitter.setMaxListeners(15);
console.log("-----Maximum size of listeners after bind-----")
console.log("Maximum size of listener is: "+eventEmitter.getMaxListeners())

//get name of all active listener
console.log("\n-----The name of active listeners is empty because we remove
all listener-----");
console.log("The name of event is: ",eventEmitter.eventNames());

//get number of listeners listening to the specified event after remove it
console.log("\n-----Number of listeners listening to the specified event is
Zero because we remove all listener-----")
console.log("No of listeners is: "+eventEmitter.listenerCount('myEvent'))
```

Output:



```
PS E:\B_Tech\SEM_5\SP\Practical\Code\Practical 5> node "e:\B_Tech\SEM_5\SP\Practical\Code\Practical 5\tempCodeRunnerFile.js"
The name of event is:  [ 'myEvent' ]

Number of listeners listening to the specified event
No of listeners is: 4

f1 Listener Removed
Message from f3: Event occurred
Message from f2: Event occurred
Message from f2: Event occurred

All Listener Removed

-----Default Maximum size of listeners-----
Maximum size of listener is: 10
-----Maximum size of listeners after bind-----
Maximum size of listener is: 15

-----The name of active listeners is empty because we remove all listener-----
The name of event is:  []

-----Number of listeners listening to the specified event is Zero because we remove all listener-----
No of listeners is: 0
PS E:\B_Tech\SEM_5\SP\Practical\Code\Practical 5> ▌
```