17.

```
C:\Users\Mrunal>node
Welcome to Node.js v16.17.0.
Type ".help" for more information.
> .editor
// Entering editor mode (Ctrl+D to finish, Ctrl+C to cancel)
let n = 5
let string = ""
for(i = 1; i <=n ; i++){
 for(j=0 ; j < i ; j++){
  string += "*"
 }
 string += "\n"
}
console.log(string)
*
**
***
****
*****
```

11.

```
> .editor
// Entering editor mode (Ctrl+D to finish, Ctrl+C to cancel)
var t1 = 0
var t2 = 1
console.log(t1)
console.log(t2)
for(i = 2; i < 10 ; i ++){
 var t3 = t1+t2
 console.log(t3)
 t1 = t2
 t2 = t3
}
0
1
1
2
3
5
8
13
21
34
34
```

12.

Opening REPL Terminal:

```
C:\Users\Mrunal>node
Welcome to Node.js v16.17.0.
Type ".help" for more information.
```

Arithmetic Operators:

```
> 10 + 15
25
> var a = 24
undefined
> var b = 17
undefined
> a - b
7
> a * b
408
> a / b
1.411764705882353
> a % b
7
```

Logical Operators:

```
> var x = true
undefined
> var y = false
undefined
> x && y
false
> x || y
true
> !x
false
```

Concatenation of two strings:

```
> "hello " + "Experiment " + 12
'hello Experiment 12'
```

Using the Variable to obtain the last value:

```
> a + b
41
> var sum = _
undefined
> console.log(sum)
41
```

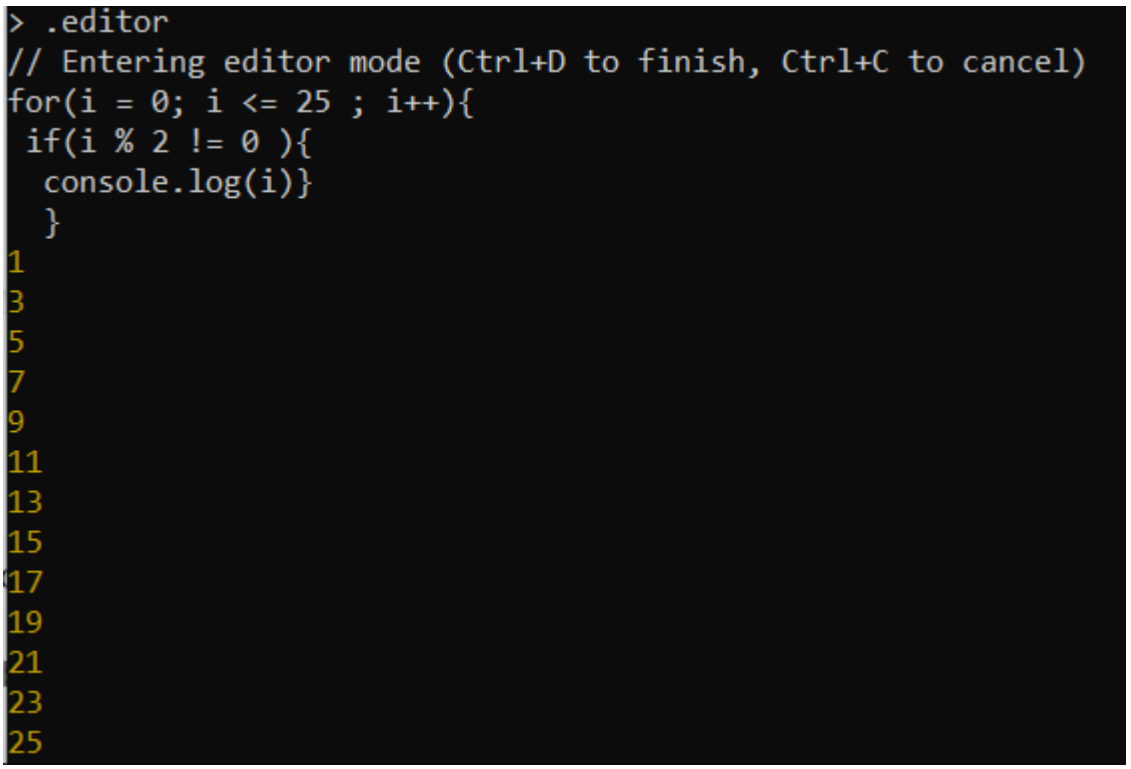
To print the table of a given number:

```
> .editor
// Entering editor mode (Ctrl+D to finish, Ctrl+C to cancel)
var a = 7
for(i = 1; i <= 10; i++){
 console.log(a," x ",i," = ", a*i)
 }
7  x  1  =  7
7  x  2  =  14
7  x  3  =  21
7  x  4  =  28
7  x  5  =  35
7  x  6  =  42
7  x  7  =  49
7  x  8  =  56
7  x  9  =  63
7  x  10  =  70
```

Even Series:

```
> .editor
// Entering editor mode (Ctrl+D to finish, Ctrl+C to cancel)
for(i = 0; i <=25; i++){
 if(i%2 == 0){
  console.log(i)}
  }
0
2
4
6
8
10
12
14
16
18
20
22
24
```

Odd Series:



13.

Reading From Stream.

Program:

```
var fs = require("fs");
var data = “”;
var readerStream = fs.createReadStream('input.txt');
readerStream.setEncoding('UTF8');
readerStream.on('data', function(chunk) {
   data += chunk;
});
readerStream.on('end',function() {
   console.log(data);
});
readerStream.on('error', function(err) {
   console.log(err.stack);
});
console.log("Program Ended");
```

Writing to a Stream:

Program:

```
var fs = require("fs");
var data = 'Experiment 13: Writing to a Stream';
```

```
var writerStream = fs.createWriteStream('writeStreamOutput.txt');

writerStream.write(data,'UTF8');

writerStream.end();

writerStream.on('finish', function() {
  console.log("Write completed.");
});

writerStream.on('error', function(err) {
  console.log(err.stack);
});

console.log("Program Ended");
```

14.

1.Creating a Web Server using Node
Server.js
Program:

```
var http = require('http');
var fs = require('fs');
var url = require('url');

http.createServer( function (request, response) {

  var pathname = url.parse(request.url).pathname;

  console.log("Request for " + pathname + " received.");

  fs.readFile(pathname.substr(1), function (err, data) {
    if (err) {
      console.log(err);

      response.writeHead(404, {'Content-Type': 'text/html'});
    } else {

      response.writeHead(200, {'Content-Type': 'text/html'});

      response.write(data.toString());
    }

    response.end();
  });
}).listen(8081);
```

```
console.log('Server running at http://127.0.0.1:8081/');
```

Index.html
Program:

```
<html>
  <head>
    <title>HTML Page</title>
  </head>

  <body>
    Hello World!
  </body>
</html>
```

3.Creating Web client using Node:

Client.js

Program:

```
var http = require('http');

var options = {
  host: 'localhost',
  port: '8081',
  path: '/index.html'
};

var callback = function(response) {

  var body = '';
  response.on('data', function(data) {
    body += data;
  });

  response.on('end', function() {
    console.log(body);
  });
}

var req = http.request(options, callback);
req.end();
```

15.

```
const express = require('express');
const app = express();
app.get('/',(req, res) =>{
    res.send("Landing Page");
})
//Simple request time logger for a specific route
app.use('/home', (req, res, next) => {
  console.log('A new request received at ' + Date.now() + ' at
HomePage');
  next();
});
app.use('/about', (req, res, next) => {
  console.log('A new request received at ' + Date.now() + ' at
AboutPage');
  next();
});

app.get('/home', (req, res) => {
  res.send('Home Page');
});

app.get('/about', (req, res) => {
  res.send('About Page');
});

app.listen(3001, () => console.log('Example app listening on port
3001!'));
```

```
const express = require('express')
const router = express.Router()

// middleware that is specific to this router
router.use((req, res, next) => {
  console.log('Time: ', Date.now())
  next()
})
// define the home page route
router.get('/', (req, res) => {
  res.send('Birds home page')
})
// define the about route
router.get('/about', (req, res) => {
  res.send('About birds')
})

module.exports = router
```

hen, load the router module in the app:

```
const birds = require('./birds')

// ...

app.use('/birds', birds)
```