

## hello1.js (Page 1 of 1)

```
1: //-----
2: // hello1.js
3: // Author: Bob Dondero
4: //-----
5:
6: 'use strict';
7:
8: process.stdout.write('hello, world\n');
9:
10: // Alternative:
11: // console.log('hello, world'); // Always appends a newline.
12:
```

## hello2.js (Page 1 of 1)

```
1: //-----
2: // hello2.js
3: // Author: Bob Dondero
4: //-----
5:
6: 'use strict';
7:
8: function main() {
9:   process.stdout.write('hello, world\n');
10: }
11:
12: if (require.main === module)
13:   main();
```

## square.js (Page 1 of 1)

```
1: //-----  
2: // square.js  
3: // Author: Bob Dondero  
4: //-----  
5:  
6: 'use strict';  
7:  
8: // Function definition statement  
9: function square1(i) {  
10:   return i * i;  
11: }  
12:  
13: // Function definition expression  
14: let square2 = function(i) {  
15:   return i * i;  
16: };  
17:  
18: // Arrow function definition expression  
19: let square3 = (i) => {return i * i;};  
20: let square4 = (i) => i * i;  
21: let square5 = i => i * i;  
22:  
23: function main() {  
24:   let sqr1 = square1(5);  
25:   let sqr2 = square2(5);  
26:   let sqr3 = square3(5);  
27:   let sqr4 = square4(5);  
28:   let sqr5 = square5(5);  
29:  
30:   process.stdout.write(String(sqr1) + '\n');  
31:   process.stdout.write(String(sqr2) + '\n');  
32:   process.stdout.write(String(sqr3) + '\n');  
33:   process.stdout.write(String(sqr4) + '\n');  
34:   process.stdout.write(String(sqr5) + '\n');  
35: }  
36:  
37: if (require.main === module)  
38:   main();
```

## squareroot.js (Page 1 of 1)

```
1: //-----  
2: // squareroot.js  
3: // Author: Bob Dondero  
4: //-----  
5:  
6: 'use strict';  
7:  
8: function main() {  
9:   let sqrtOfTwo = Math.sqrt(2.0);  
10:   process.stdout.write(String(sqrtOfTwo) + '\n');  
11: }  
12:  
13: if (require.main === module)  
14:   main();
```

## circle1.js (Page 1 of 1)

```

1: //-----
2: // circle1.js
3: // Author: Bob Dondero
4: //-----
5:
6: // Before running this program you must install the readline-sync
7: // module. You can do that by issuing this command:
8: // npm install readline-sync
9:
10: 'use strict';
11:
12: const readlineSync = require('readline-sync');
13:
14: function main() {
15:   let line = readlineSync.question("Enter the circle's radius:\n");
16:   let radius = Number(line);
17:
18:   let diam = 2 * radius;
19:   let circum = Math.PI * diam;
20:
21:   process.stdout.write('A circle with radius ' + String(radius) +
22:     ' has diameter ' + String(diam) + '\n');
23:   process.stdout.write('and circumference ' + String(circum) + '\n');
24: }
25:
26: if (require.main === module)
27:   main();

```

## circle2.js (Page 1 of 1)

```

1: //-----
2: // circle2.js
3: // Author: Bob Dondero
4: //-----
5:
6: // Before running this program you must install the readline-sync
7: // module. You can do that by issuing this command:
8: // npm install readline-sync
9:
10: 'use strict';
11:
12: const readlineSync = require('readline-sync');
13:
14: function main() {
15:   try {
16:     let line = readlineSync.question(
17:       "Enter the circle's radius:\n");
18:     if (line === '')
19:       throw new Error('Missing number');
20:
21:     if (isNaN(line))
22:       throw new Error('Not a number');
23:     let radius = Number(line);
24:
25:     let diam = 2 * radius;
26:     let circum = Math.PI * diam;
27:
28:     process.stdout.write('A circle with radius ' + String(radius) +
29:       ' has diameter ' + String(diam) + '\n');
30:     process.stdout.write('and circumference ' + String(circum) +
31:       '\n');
32:   }
33:   catch (e) {
34:     process.stderr.write(String(e) + '\n');
35:   }
36: }
37:
38: if (require.main === module)
39:   main();

```

## euclidclient1.js (Page 1 of 2)

```

1: //-----
2: // euclidclient1.js
3: // Author: Bob Dondero
4: //-----
5:
6: 'use strict';
7:
8: const readlineSync = require('readline-sync');
9:
10: //-----
11:
12: function gcd(i, j) {
13:   if ((i === 0) && (j === 0))
14:     throw new Error('Computation is undefined');
15:
16:   i = Math.abs(i);
17:   j = Math.abs(j);
18:   while (j !== 0) {
19:     let temp = i % j;
20:     i = j;
21:     j = temp;
22:   }
23:   return i;
24: }
25:
26: //-----
27:
28: function lcm(i, j) {
29:   if ((i === 0) || (j === 0))
30:     throw new Error('Computation is undefined');
31:
32:   i = Math.abs(i);
33:   j = Math.abs(j);
34:   return (i / gcd(i, j)) * j;
35: }
36:
37: //-----
38:
39: function readInt(prompt) {
40:   let line = readlineSync.question(prompt);
41:   if (line === '')
42:     throw new Error('Missing integer');
43:   if (isNaN(line))
44:     throw new Error('Not a number');
45:   let n = Number(line);
46:   if (!Number.isInteger(n))
47:     throw new Error('Not an integer');
48:   return n;
49: }
50:
51: //-----
52:
53: function main() {
54:   try {
55:     let i = readInt('Enter the first integer:\n');
56:     let j = readInt('Enter the second integer:\n');
57:
58:     let myGcd = gcd(i, j);
59:     process.stdout.write('gcd: ' + String(myGcd) + '\n');
60:
61:     let myLcm = lcm(i, j);
62:     process.stdout.write('lcm: ' + String(myLcm) + '\n');
63:   }
64:   catch (e) {
65:     process.stderr.write(e + '\n');

```

## euclidclient1.js (Page 2 of 2)

```

66:   }
67: }
68:
69: if (require.main === module)
70:   main();
71:

```