

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:

```

euclid.js (Page 1 of 1)

```

1: //-----
2: // euclid.js
3: // Author: Bob Dondero
4: //-----
5:
6: 'use strict';
7:
8: //-----
9:
10: function gcd(i, j) {
11:
12:   if ((i === 0) && (j === 0))
13:     throw new Error('Computation is undefined');
14:
15:   i = Math.abs(i);
16:   j = Math.abs(j);
17:   while (j !== 0) {
18:     let temp = i % j;
19:     i = j;
20:     j = temp;
21:   }
22:   return i;
23: }
24:
25: //-----
26:
27: function lcm(i, j) {
28:
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: module.exports = { gcd, lcm };
40:

```

euclidclient2.js (Page 1 of 1)

```

1: //-----
2: // euclidclient2.js
3: // Author: Bob Dondero
4: //-----
5:
6: 'use strict';
7:
8: const readlineSync = require('readline-sync');
9: const euclid = require('./euclid.js');
10:
11: //-----
12:
13: function readInt(prompt) {
14:   let line = readlineSync.question(prompt);
15:   if (line === '')
16:     throw new Error('Missing integer');
17:   if (isNaN(line))
18:     throw new Error('Not a number');
19:   let n = Number(line);
20:   if (!Number.isInteger(n))
21:     throw new Error('Not an integer');
22:   return n;
23: }
24:
25: //-----
26:
27: function main() {
28:   try {
29:     let i = readInt('Enter the first integer:\n');
30:     let j = readInt('Enter the second integer:\n');
31:
32:     let myGcd = euclid.gcd(i, j);
33:     process.stdout.write('gcd: ' + String(myGcd) + '\n');
34:
35:     let myLcm = euclid.lcm(i, j);
36:     process.stdout.write('lcm: ' + String(myLcm) + '\n');
37:   }
38:   catch (e) {
39:     process.stderr.write(String(e) + '\n');
40:   }
41: }
42:
43: if (require.main === module)
44:   main();

```

fraction1.js (Page 1 of 2)

```

1: //-----
2: // fraction1.js
3: // Author: Bob Dondero
4: //-----
5:
6: 'use strict';
7:
8: const euclid = require('./euclid.js');
9:
10: function create(num=0, den=1) {
11:
12:   if (arguments.length > 2)
13:     throw new Error('Too many arguments');
14:
15:   if (den === 0)
16:     throw new Error('Denominator cannot be zero');
17:
18:   let f = {};
19:
20:   f._num = num;
21:   f._den = den;
22:
23:   if (f._den < 0) {
24:     f._num *= -1;
25:     f._den *= -1;
26:   }
27:   if (f._num === 0)
28:     f._den = 1;
29:   else {
30:     let gcden = euclid.gcd(f._num, f._den);
31:     f._num /= gcden;
32:     f._den /= gcden;
33:   }
34:
35:   return f;
36: }
37:
38: function toString(f1) {
39:   if (f1._den === 1)
40:     return String(f1._num);
41:   return String(f1._num) + '/' + String(f1._den);
42: }
43:
44: function compareTo(f1, f2) {
45:   if ((f1._num * f2._den) < (f2._num * f1._den))
46:     return -1;
47:   if ((f1._num * f2._den) > (f2._num * f1._den))
48:     return 1;
49:   return 0;
50: }
51:
52: function negate(f1) {
53:   return create(-f1._num, f1._den);
54: }
55:
56: function add(f1, f2) {
57:   let newNum = (f1._num * f2._den) + (f2._num * f1._den);
58:   let newDen = f1._den * f2._den;
59:   return create(newNum, newDen);
60: }
61:
62: function subtract(f1, f2) {
63:   let newNum = (f1._num * f2._den) - (f2._num * f1._den);
64:   let newDen = f1._den * f2._den;
65:   return create(newNum, newDen);

```

fraction1.js (Page 2 of 2)

```

66: }
67:
68: function multiply(f1, f2) {
69:   let newNum = f1._num * f2._num;
70:   let newDen = f1._den * f2._den;
71:   return create(newNum, newDen);
72: }
73:
74: function divide(f1, f2) {
75:   let newNum = f1._num * f2._den;
76:   let newDen = f1._den * f2._num;
77:   return create(newNum, newDen);
78: }
79:
80: module.exports = { create, toString, compareTo, negate, add,
81:   subtract, multiply, divide };

```

fractionlclient.js (Page 1 of 2)

```

1: //-----
2: // fractionlclient.js
3: // Author: Bob Dondero
4: //-----
5:
6: 'use strict';
7:
8: const readlineSync = require('readline-sync');
9: const fraction = require('./fraction1.js');
10:
11: //-----
12:
13: function readInt(prompt) {
14:   let line = readlineSync.question(prompt);
15:   if (line === '')
16:     throw new Error('Missing integer');
17:   if (isNaN(line))
18:     throw new Error('Not a number');
19:   let n = Number(line);
20:   if (!Number.isInteger(n))
21:     throw new Error('Not an integer');
22:   return n;
23: }
24:
25: //-----
26:
27: function main() {
28:   try {
29:     let n1 = readInt('Numerator 1: ');
30:     let d1 = readInt('Denominator 1: ');
31:     let n2 = readInt('Numerator 2: ');
32:     let d2 = readInt('Denominator 2: ');
33:
34:     let f1 = fraction.create(n1, d1);
35:     let f2 = fraction.create(n2, d2);
36:
37:     process.stdout.write('f1: ' + fraction.toString(f1) + '\n');
38:     process.stdout.write('f2: ' + fraction.toString(f2) + '\n');
39:
40:     if (f1 === f2)
41:       process.stdout.write('f1 is identical to f2\n');
42:     else
43:       process.stdout.write('f1 is not identical to f2\n');
44:
45:     let compare = fraction.compareTo(f1, f2);
46:     if (compare < 0)
47:       process.stdout.write('f1 is less than f2\n');
48:     if (compare > 0)
49:       process.stdout.write('f1 is greater than f2\n');
50:     if (compare === 0)
51:       process.stdout.write('f1 is equal to f2\n');
52:
53:     let f3;
54:
55:     f3 = fraction.negate(f1);
56:     process.stdout.write('-f1: ' + fraction.toString(f3) + '\n');
57:
58:     f3 = fraction.add(f1, f2);
59:     process.stdout.write('f1 + f2: ' + fraction.toString(f3) + '\n');
60:
61:     f3 = fraction.subtract(f1, f2);
62:     process.stdout.write('f1 - f2: ' + fraction.toString(f3) + '\n');
63:
64:     f3 = fraction.multiply(f1, f2);
65:     process.stdout.write('f1 * f2: ' + fraction.toString(f3) + '\n');

```

fractionlclient.js (Page 2 of 2)

```

66:
67:     f3 = fraction.divide(f1, f2);
68:     process.stdout.write('f1 / f2: ' + fraction.toString(f3) + '\n');
69:   }
70:   catch (e) {
71:     process.stderr.write(String(e) + '\n');
72:   }
73: }
74:
75: if (require.main === module)
76:   main();

```

fraction2.js (Page 1 of 2)

```

1: //-----
2: // fraction2.js
3: // Author: Bob Dondero
4: //-----
5:
6: 'use strict';
7:
8: const euclid = require('./euclid.js');
9:
10: function createFraction(num=0, den=1)
11: {
12:   if (arguments.length > 2)
13:     throw new Error('Too many arguments');
14:
15:   if (den === 0)
16:     throw new Error('Denominator cannot be zero');
17:
18:   let f = {};
19:
20:   f._num = num;
21:   f._den = den;
22:
23:   if (f._den < 0) {
24:     f._num *= -1;
25:     f._den *= -1;
26:   }
27:   if (f._num === 0)
28:     f._den = 1;
29:   else {
30:     let gcden = euclid.gcd(f._num, f._den);
31:     f._num /= gcden;
32:     f._den /= gcden;
33:   }
34:
35:   f.toString = function() {
36:     if (this._den === 1)
37:       return String(this._num);
38:     return String(this._num) + '/' + String(this._den);
39:   };
40:
41:   f.compareTo = function(other) {
42:     if ((this._num * other._den) < (other._num * this._den))
43:       return -1;
44:     if ((this._num * other._den) > (other._num * this._den))
45:       return 1;
46:     return 0;
47:   };
48:
49:   f.negate = function() {
50:     return createFraction(-this._num, this._den);
51:   };
52:
53:   f.add = function(other) {
54:     let newNum = (this._num * other._den) + (other._num * this._den);
55:     let newDen = this._den * other._den;
56:     return createFraction(newNum, newDen);
57:   };
58:
59:   f.subtract = function(other) {
60:     let newNum = (this._num * other._den) - (other._num * this._den);
61:     let newDen = this._den * other._den;
62:     return createFraction(newNum, newDen);
63:   };
64:
65:   f.multiply = function(other) {

```

fraction2.js (Page 2 of 2)

```

66:     let newNum = this._num * other._num;
67:     let newDen = this._den * other._den;
68:     return createFraction(newNum, newDen);
69:   };
70:
71:   f.divide = function(other) {
72:     let newNum = this._num * other._den;
73:     let newDen = this._den * other._num;
74:     return createFraction(newNum, newDen);
75:   };
76:
77:   return f;
78: }
79:
80: module.exports = { createFraction };

```

fraction2client.js (Page 1 of 2)

```

1: //-----
2: // fraction2client.js
3: // Author: Bob Dondero
4: //-----
5:
6: 'use strict';
7:
8: const readlineSync = require('readline-sync');
9: const fraction = require('./fraction2.js');
10:
11: //-----
12:
13: function readInt(prompt) {
14:   let line = readlineSync.question(prompt);
15:   if (line === '')
16:     throw new Error('Missing integer');
17:   if (isNaN(line))
18:     throw new Error('Not a number');
19:   let n = Number(line);
20:   if (!Number.isInteger(n))
21:     throw new Error('Not an integer');
22:   return n;
23: }
24:
25: //-----
26:
27: function main() {
28:   try {
29:     let n1 = readInt('Numerator 1: ');
30:     let d1 = readInt('Denominator 1: ');
31:     let n2 = readInt('Numerator 2: ');
32:     let d2 = readInt('Denominator 2: ');
33:
34:     let f1 = fraction.createFraction(n1, d1);
35:     let f2 = fraction.createFraction(n2, d2);
36:
37:     process.stdout.write('f1: ' + f1.toString() + '\n');
38:     //process.stdout.write('f1: ' + String(f1) + '\n');
39:     //process.stdout.write('f1: ' + f1 + '\n');
40:     process.stdout.write('f2: ' + f2.toString() + '\n');
41:
42:
43:     if (f1 === f2)
44:       process.stdout.write('f1 is identical to f2\n');
45:     else
46:       process.stdout.write('f1 is not identical to f2\n');
47:
48:     let compare = f1.compareTo(f2);
49:     if (compare < 0)
50:       process.stdout.write('f1 is less than f2\n');
51:     if (compare > 0)
52:       process.stdout.write('f1 is greater than f2\n');
53:     if (compare === 0)
54:       process.stdout.write('f1 is equal to f2\n');
55:
56:     let f3;
57:
58:     f3 = f1.negate();
59:     process.stdout.write('-f1: ' + f3.toString() + '\n');
60:
61:     f3 = f1.add(f2);
62:     process.stdout.write('f1 + f2: ' + f3.toString() + '\n');
63:
64:     f3 = f1.subtract(f2);
65:     process.stdout.write('f1 - f2: ' + f3.toString() + '\n');

```

fraction2client.js (Page 2 of 2)

```

66:
67:     f3 = f1.multiply(f2);
68:     process.stdout.write('f1 * f2: ' + f3.toString() + '\n');
69:
70:     f3 = f1.divide(f2);
71:     process.stdout.write('f1 / f2: ' + f3.toString() + '\n');
72:   }
73:   catch (e) {
74:     process.stderr.write(String(e) + '\n');
75:   }
76: }
77:
78: if (require.main === module)
79:   main();

```