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 (err) {
74:       process.stderr.write(err.message + '\n');
75:    }
76: }
77:
78: if (require.main === module)
79:    main();
```

fraction3.js (Page 1 of 2)

```
 1: //-----------------------------------------------------------------------
 2: // fraction3.js
 3: // Author: Bob Dondero
 4: //-----------------------------------------------------------------------
 5:
 6: 'use strict';
 7:
 8: const euclid = require('./euclid.js');
 9:
10: function Fraction(num=0, den=1) {
11:    if (arguments.length > 2)
12:       throw new Error('Too many arguments');
13:
14:    if (den === 0)
15:       throw new Error('Denominator cannot be zero');
16:
17:    this._num = num;
18:    this._den = den;
19:
20:    if (this._den < 0) {
21:       this._num *= -1;
22:       this._den *= -1;
23:    }
24:    if (this._num === 0)
25:       this._den = 1;
26:    else {
27:       let gcden = euclid.gcd(this._num, this._den);
28:       this._num /= gcden;
29:       this._den /= gcden;
30:    }
31:
32:    this.toString = function() {
33:       if (this._den === 1)
34:          return String(this._num);
35:       return String(this._num) + '/' + String(this._den);
36:    };
37:
38:    this.compareTo = function(other) {
39:       if ((this._num * other._den) < (other._num * this._den))
40:          return -1;
41:       if ((this._num * other._den) > (other._num * this._den))
42:          return 1;
43:       return 0;
44:    };
45:
46:    this.negate = function() {
47:       return new Fraction(-this._num, this._den);
48:    };
49:
50:    this.add = function(other) {
51:       let newNum = (this._num * other._den) + (other._num * this._den);
52:       let newDen = this._den * other._den;
53:       return new Fraction(newNum, newDen);
54:    };
55:
56:    this.subtract = function(other) {
57:       let newNum = (this._num * other._den) - (other._num * this._den);
58:       let newDen = this._den * other._den;
59:       return new Fraction(newNum, newDen);
60:    };
61:
62:    this.multiply = function(other) {
63:       let newNum = this._num * other._num;
64:       let newDen = this._den * other._den;
65:       return new Fraction(newNum, newDen);
```

fraction3.js (Page 2 of 2)

```
66:    };
67:
68:    this.divide = function(other) {
69:       let newNum = this._num * other._den;
70:       let newDen = this._den * other._num;
71:       return new Fraction(newNum, newDen);
72:    };
73: }
74:
75: module.exports = { Fraction };
```

fraction3client.js (Page 1 of 2)

```
 1: //-----------------------------------------------------------------------
 2: // fraction3client.js
 3: // Author: Bob Dondero
 4: //-----------------------------------------------------------------------
 5:
 6: 'use strict';
 7:
 8: const readlineSync = require('readline-sync');
 9: const fraction = require('./fraction3.js');
10:
11: //-----------------------------------------------------------------------
12:
13: function readInt(prompt) {
14:    let line = readlineSync.question(prompt);
15:    if (line === '')
16:       throw new Error('Missing integer');
17:
18:    let n = Number(line);
19:    if (! Number.isInteger(n))
20:       throw new Error('Not an integer');
21:
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 = new fraction.Fraction(n1, d1);
35:       let f2 = new fraction.Fraction(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:       if (f1 === f2)
43:          process.stdout.write('f1 is identical to f2\n');
44:       else
45:          process.stdout.write('f1 is not identical to f2\n');
46:
47:       let compare = f1.compareTo(f2);
48:       if (compare < 0)
49:          process.stdout.write('f1 is less than f2\n');
50:       if (compare > 0)
51:          process.stdout.write('f1 is greater than f2\n');
52:       if (compare === 0)
53:          process.stdout.write('f1 is equal to f2\n');
54:
55:       let f3;
56:
57:       f3 = f1.negate();
58:       process.stdout.write('-f1: ' + f3.toString() + '\n');
59:
60:       f3 = f1.add(f2);
61:       process.stdout.write('f1 + f2: ' + f3.toString() + '\n');
62:
63:       f3 = f1.subtract(f2);
64:       process.stdout.write('f1 - f2: ' + f3.toString() + '\n');
65:
```

fraction3client.js (Page 2 of 2)

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

fraction4.js (Page 1 of 2)

```
 1: //-----------------------------------------------------------------------
 2: // fraction4.js
 3: // Author: Bob Dondero
 4: //-----------------------------------------------------------------------
 5:
 6: 'use strict';
 7:
 8: const euclid = require('./euclid.js');
 9:
10: function Fraction(num=0, den=1) {
11:    if (arguments.length > 2)
12:       throw new Error('Too many arguments');
13:
14:    if (den === 0)
15:       throw new Error('Denominator cannot be zero');
16:
17:    this._num = num;
18:    this._den = den;
19:
20:    if (this._den < 0) {
21:       this._num *= -1;
22:       this._den *= -1;
23:    }
24:    if (this._num === 0)
25:       this._den = 1;
26:    else {
27:       let gcden = euclid.gcd(this._num, this._den);
28:       this._num /= gcden;
29:       this._den /= gcden;
30:    }
31: }
32:
33: Fraction.prototype.toString = function() {
34:    if (this._den === 1)
35:       return String(this._num);
36:    return String(this._num) + '/' + String(this._den);
37: };
38:
39: Fraction.prototype.compareTo = function(other) {
40:    if ((this._num * other._den) < (other._num * this._den)) return -1;
41:    if ((this._num * other._den) > (other._num * this._den)) return 1;
42:    return 0;
43: };
44:
45: Fraction.prototype.negate = function() {
46:    return new Fraction(-this._num, this._den);
47: };
48:
49: Fraction.prototype.add = function(other) {
50:    let newNum = (this._num * other._den) + (other._num * this._den);
51:    let newDen = this._den * other._den;
52:    return new Fraction(newNum, newDen);
53: };
54:
55: Fraction.prototype.subtract = function(other) {
56:    let newNum = (this._num * other._den) - (other._num * this._den);
57:    let newDen = this._den * other._den;
58:    return new Fraction(newNum, newDen);
59: };
60:
61: Fraction.prototype.multiply = function(other) {
62:    let newNum = this._num * other._num;
63:    let newDen = this._den * other._den;
64:    return new Fraction(newNum, newDen);
65: };
```

fraction4.js (Page 2 of 2)

```
66:
67: Fraction.prototype.divide = function(other) {
68:    let newNum = this._num * other._den;
69:    let newDen = this._den * other._num;
70:    return new Fraction(newNum, newDen);
71: };
72:
73: module.exports = { Fraction };
```

fraction4client.js (Page 1 of 2)

```
//-----------------------------------------------------------------------
// fraction4client.js
// Author: Bob Dondero
//-----------------------------------------------------------------------

'use strict';

const readline = require('readline-sync');
const fraction = require('./fraction4.js');

//-----------------------------------------------------------------------

function readInt(prompt) {
   let line = readline.question(prompt);
   if (line === '')
      throw new Error('Missing integer');

   let n = Number(line);
   if (! Number.isInteger(n))
      throw new Error('Not an integer');

   return n;
}

//-----------------------------------------------------------------------

function main() {
   try    {
      let n1 = readInt('Numerator 1: ');
      let d1 = readInt('Denominator 1: ');
      let n2 = readInt('Numerator 2: ');
      let d2 = readInt('Denominator 2: ');

      let f1 = new fraction.Fraction(n1, d1);
      let f2 = new fraction.Fraction(n2, d2);

      process.stdout.write('f1: ' + f1.toString() + '\n');
      //process.stdout.write('f1: ' + String(f1) + '\n');
      //process.stdout.write('f1: ' + f1 + '\n');
      process.stdout.write('f2: ' + f2.toString() + '\n');

      if (f1 === f2)
         process.stdout.write('f1 is identical to f2\n');
      else
         process.stdout.write('f1 is not identical to f2\n');

      let compare = f1.compareTo(f2);
      if (compare < 0)
         process.stdout.write('f1 is less than f2\n');
      if (compare > 0)
         process.stdout.write('f1 is greater than f2\n');
      if (compare === 0)
         process.stdout.write('f1 is equal to f2\n');

      let f3;

      f3 = f1.negate();
      process.stdout.write('-f1: ' + f3.toString() + '\n');

      f3 = f1.add(f2);
      process.stdout.write('f1 + f2: ' + f3.toString() + '\n');

      f3 = f1.subtract(f2);
      process.stdout.write('f1 - f2: ' + f3.toString() + '\n');

      f3 = f1.multiply(f2);
      process.stdout.write('f1 * f2: ' + f3.toString() + '\n');

      f3 = f1.divide(f2);
      process.stdout.write('f1 / f2: ' + f3.toString() + '\n');
   }
   catch (err) {
      process.stderr.write(err.message + '\n');
   }
}

if (require.main === module)
   main();
```

fraction4client.js (Page 2 of 2)


```
      f3 = f1.multiply(f2);
      process.stdout.write('f1 * f2: ' + f3.toString() + '\n');

      f3 = f1.divide(f2);
      process.stdout.write('f1 / f2: ' + f3.toString() + '\n');
   }
   catch (err) {
      process.stderr.write(err.message + '\n');
   }
}

if (require.main === module)
   main();
```

fraction5.js (Page 1 of 2)

```
 1: //-----------------------------------------------------------------------
 2: // fraction5.js
 3: // Author: Bob Dondero
 4: //-----------------------------------------------------------------------
 5:
 6: 'use strict';
 7:
 8: const euclid = require('./euclid.js');
 9:
10: class Fraction {
11:    constructor(num=0, den=1) {
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:       this._num = num;
19:       this._den = den;
20:
21:       if (this._den < 0) {
22:          this._num *= -1;
23:          this._den *= -1;
24:       }
25:       if (this._num === 0)
26:          this._den = 1;
27:       else {
28:          let gcden = euclid.gcd(this._num, this._den);
29:          this._num /= gcden;
30:          this._den /= gcden;
31:       }
32:    }
33:
34:    toString() {
35:       if (this._den === 1)
36:          return String(this._num);
37:       return String(this._num) + '/' + String(this._den);
38:    }
39:
40:    compareTo(other) {
41:       if ((this._num * other._den) < (other._num * this._den))
42:          return -1;
43:       if ((this._num * other._den) > (other._num * this._den))
44:          return 1;
45:       return 0;
46:    }
47:
48:    negate() {
49:       return new Fraction(-this._num, this._den);
50:    }
51:
52:    add(other) {
53:       let newNum = (this._num * other._den) + (other._num * this._den);
54:       let newDen = this._den * other._den;
55:       return new Fraction(newNum, newDen);
56:    }
57:
58:    subtract(other) {
59:       let newNum = (this._num * other._den) - (other._num * this._den);
60:       let newDen = this._den * other._den;
61:       return new Fraction(newNum, newDen);
62:    }
63:
64:    multiply(other) {
65:       let newNum = this._num * other._num;
```

fraction5.js (Page 2 of 2)

```
66:       let newDen = this._den * other._den;
67:       return new Fraction(newNum, newDen);
68:    }
69:
70:    divide(other) {
71:       let newNum = this._num * other._den;
72:       let newDen = this._den * other._num;
73:       return new Fraction(newNum, newDen);
74:    }
75: }
76:
77: module.exports = { Fraction };
```

fraction5client.js (Page 1 of 2)

```
//-----------------------------------------------------------------------
// fraction5client.js
// Author: Bob Dondero
//-----------------------------------------------------------------------

'use strict';

const readlineSync = require('readline-sync');
const fraction = require('./fraction5.js');

//-----------------------------------------------------------------------

function readInt(prompt) {
   let line = readlineSync.question(prompt);
   if (line === '')
      throw new Error('Missing integer');

   let n = Number(line);
   if (! Number.isInteger(n))
      throw new Error('Not an integer');

   return n;
}

//-----------------------------------------------------------------------

function main() {
   try {
      let n1 = readInt('Numerator 1: ');
      let d1 = readInt('Denominator 1: ');
      let n2 = readInt('Numerator 2: ');
      let d2 = readInt('Denominator 2: ');

      let f1 = new fraction.Fraction(n1, d1);
      let f2 = new fraction.Fraction(n2, d2);

      process.stdout.write('f1: ' + f1.toString() + '\n');
      //process.stdout.write('f1: ' + String(f1) + '\n');
      //process.stdout.write('f1: ' + f1 + '\n');
      process.stdout.write('f2: ' + f2.toString() + '\n');

      if (f1 === f2)
         process.stdout.write('f1 is identical to f2\n');
      else
         process.stdout.write('f1 is not identical to f2\n');

      let compare = f1.compareTo(f2);
      if (compare < 0)
         process.stdout.write('f1 is less than f2\n');
      if (compare > 0)
         process.stdout.write('f1 is greater than f2\n');
      if (compare === 0)
         process.stdout.write('f1 is equal to f2\n');

      let f3;

      f3 = f1.negate();
      process.stdout.write('-f1: ' + f3.toString() + '\n');

      f3 = f1.add(f2);
      process.stdout.write('f1 + f2: ' + f3.toString() + '\n');

      f3 = f1.subtract(f2);
      process.stdout.write('f1 - f2: ' + f3.toString() + '\n');

      f3 = f1.multiply(f2);
      process.stdout.write('f1 * f2: ' + f3.toString() + '\n');

      f3 = f1.divide(f2);
      process.stdout.write('f1 / f2: ' + f3.toString() + '\n');
   }
   catch (err) {
      process.stderr.write(err.message + '\n');
   }
}

if (require.main === module)
   main();
```

fraction5client.js (Page 2 of 2)


```
      f3 = f1.multiply(f2);
      process.stdout.write('f1 * f2: ' + f3.toString() + '\n');

      f3 = f1.divide(f2);
      process.stdout.write('f1 / f2: ' + f3.toString() + '\n');
   }
   catch (err) {
      process.stderr.write(err.message + '\n');
   }
}

if (require.main === module)
   main();
```