

```

1  /*Addition program, in class example
2 prints out the addition of two input numbers
3 Written by A Student for CS1301
4 On: Month, Day, Year*/
5 import CSCl.*;
6 public class Calculator //begin calculator class
7 {
8     public final static double ERROR = 0; //declare error constant
9     public final static String FORMAT = "%10.2f %2s %10.2f = %10.2f"; //f is float, or
10    decimal, 10 is max characters, 2 is number of displayed decimal places
11    public final static String FORMAT2 = "%10.2f %2s %10.2f = %-5b"; //f is float, or
12    decimal, 10 is max characters, 2 is number of displayed decimal places
13    public static String operator = "+"; //tells user what operation was completed
14    public static String operatorComparison = ">"; //operatorComparison is for
15    comparing the values
16
17    public static void main(String[] args) //begin main
18    {
19        double firstInput = CSClConvert.Parse(args[0], ERROR); //take input from
20        argument 0
21        double secondInput = CSClConvert.Parse(args[1], ERROR); //take input from
22        argument 1
23        double finalSum;
24
25        finalSum = add(firstInput,secondInput); //sum the inputs
26        printout(firstInput,operator,secondInput,finalSum); //print
27
28        finalSum = subtract(firstInput,secondInput); //subtract the inputs
29        printout(firstInput,operator,secondInput,finalSum); //print
30
31        finalSum = divide(firstInput,secondInput); //divide the inputs
32        if(secondInput == 0)
33        {
34            System.out.println("      Cannot divide by 0");
35        }
36        else
37        {
38            printout(firstInput,operator,secondInput,finalSum); //print
39        }
40        finalSum = multiply(firstInput,secondInput); //multiply the inputs
41        printout(firstInput,operator,secondInput,finalSum); //print
42
43        boolean booleanAnswer = greaterThan(firstInput,secondInput);
44        printout(firstInput,operatorComparison,secondInput,booleanAnswer); //return
45        whether first is greater than second
46
47        booleanAnswer = lesserThan(firstInput,secondInput);
48        printout(firstInput,operatorComparison,secondInput,booleanAnswer); //return
49        whether first is less than second
50
51        booleanAnswer = lessThanOrEqualTo(firstInput,secondInput);
52        printout(firstInput,operatorComparison,secondInput,booleanAnswer); //return
53        whether first is less than or equal to second
54
55        booleanAnswer = greatThanOrEqualTo(firstInput,secondInput);
56        printout(firstInput,operatorComparison,secondInput,booleanAnswer); //return
57        whether first is greater than or equal to second
58
59        booleanAnswer = doesEqual(firstInput,secondInput); //return whether first and
60        second are equal in value
61        printout(firstInput,operatorComparison,secondInput,booleanAnswer); //these
62        booleans all return true or false after
63                                         //making
64        the appropriate comparison
65    }
66
67    //operations methods below
68    public static double add(double firstInput, double secondInput) //addition method
69    {
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
589

```

```

58     double answer = firstInput + secondInput; //create answer by adding inputs
59     return answer; //return answer
60 }
61 public static double subtract(double firstInput, double secondInput) //subtraction
62 method
63 {
64     double answer = firstInput - secondInput; //create answer by adding inputs
65     operator = "-"; //change operator string so everything makes sense when
66     displayed; all methods do this
67     return answer; //return answer
68 }
69 public static double divide(double firstInput, double secondInput) //division method
70 {
71     if(secondInput == 0)
72     {
73         return firstInput;
74     }
75     else
76     {
77         double answer = firstInput/secondInput; //create answer by adding inputs
78         operator = "/";
79         return answer; //return answer
80     }
81 }
82 public static double multiply(double firstInput, double secondInput)
83 //multiplication method
84 {
85     double answer = firstInput*secondInput; //create answer by adding inputs
86     operator = "*";
87     return answer; //return answer
88 }
89 public static boolean greaterThan(double first, double second)
90 {
91     boolean answer = first > second; //compare to see if first is greater than
92     second input
93     return answer; //return boolean value with regards to above comparison
94 }
95 public static boolean lesserThan(double first, double second)
96 {
97     boolean answer = first < second; //compare to see if first is less than second
98     input
99     operatorComparison = "<";
100    return answer; //return boolean value with regards to above comparison
101 }
102 public static boolean lessThanEqual(double first, double second)
103 {
104     boolean answer = first <= second; //compare to see if first is less than or
105     equal to second input
106     operatorComparison = "<=";
107     return answer; //return boolean value with regards to above comparison
108 }
109 public static boolean greatThanEqual(double first, double second)
110 {
111     boolean answer = first >= second; //compare to see if first is greater than or
112     equal to second input
113     operatorComparison = ">=";
114     return answer; //return boolean value with regards to above comparison
115 }
116 public static void printout(double first, String operation, double second, double
117 answer)//printout method double
118 {
119     String line = String.format(FORMAT,first,operation,second,answer); //format

```

```
118     everything nice and tidy
119     System.out.println(line); //use java println to print out
120 }
120 public static void printout(double first, String operation, double second, boolean
121 answer)//printout method boolean
121 {
122     String line = String.format(FORMAT2,first,operation,second,answer); //format
123     everything nice and tidy
123     System.out.println(line); //use java println to print out
124 }
125 //end operations methods
126
127 }//end calculator
```