

```

1  import java.util.Scanner;
2  import java.util.InputMismatchException;
3  class Calculator
4  {
5
6      public void add(float a, float b, float c)
7      {
8          System.out.println(a+"+"+b+"+"+c+"="+ (a+b+c));
9      }
10     public void add(float a, float b)
11     {
12         System.out.println(a+"+"+b+"="+ (a+b));
13     }
14
15     public void subtract(float a, float b, float c)
16     {
17         System.out.println(a+"-"+b+"-"+c+"="+ (a-b-c));

```

⋮ File info ⓘ



```
18     System.out.println(a+"-"+b+"-"+c+"="+ (a-b-c));
19 }
20 public void subtract(float a, float b)
21 {
22     System.out.println(a+"-"+b+"="+ (a-b));
23 }
24
25
26 public void product(float a, float b)
27 {
28     System.out.println(a+"*"+b+"="+ (a*b));
29 }
30
31
32 public void division(float a, float b)
33 {
34     System.out.println(a+"/"+b+"="+ (a/b));
35 }
```

⋮ File info ⓘ



```

33     {
34         System.out.println(a+"/"+b+"="+a/b);
35     }
36 }
37 public class Main
38 {
39     public static void main (String[] args) {
40         Calculator cal=new Calculator();
41         Scanner sc=new Scanner(System.in);
42         System.out.println("Author: v. pavan kumar\nSAP ID:51834509");
43         try
44         {
45             System.out.println("1. ADD\n2. SUBTRACT\n3. MULTIPLICATION\n4. DIVISION\n5. EXIT\nEnter
46             int op=sc.nextInt();
47             switch(op)
48             {
49                 case 0:
50                     System.out.println("Exit...");
51                     System.exit(0);

```

⋮ File info ⓘ



```

51         System.exit(0);
52         break;
53     case 1:
54         System.out.print("Enter operand 1: ");
55         float add1=sc.nextFloat();
56         System.out.print("Enter operand 2: ");
57         float add2=sc.nextFloat();
58         System.out.print("Enter operand 3(if you want. else enter 0): ");
59         float add3=sc.nextFloat();
60         if(add3==0)
61         {
62             cal.add(add1, add2);
63         }
64         else
65         {
66             cal.add(add1, add2, add3);
67         }
68         break;
69     case 2:

```

⋮ File info ⓘ



```

68         break;
69     case 2:
70         System.out.print("Enter operand 1: ");
71         float sub1=sc.nextFloat();
72         System.out.print("Enter operand 2: ");
73         float sub2=sc.nextFloat();
74         System.out.print("Enter operand 3(if you want. else enter 0): ");
75         float sub3=sc.nextFloat();
76         if(sub3==0)
77         {
78             cal.subtract(sub1, sub2);
79         }
80         else
81         {
82             cal.subtract(sub1, sub2, sub3);
83         }
84         break;
85     case 3:
86         System.out.print("Enter operand 1: ");
87         float sub1=sc.nextFloat();
88         System.out.print("Enter operand 2: ");
89         float sub2=sc.nextFloat();
90         System.out.print("Enter operand 3(if you want. else enter 0): ");
91         float sub3=sc.nextFloat();
92         if(sub3==0)
93         {
94             cal.subtract(sub1, sub2);
95         }
96         else
97         {
98             cal.subtract(sub1, sub2, sub3);
99         }
100        break;
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    }
590    }
591    }
592    }
593    }
594    }
595    }
596    }
597    }
598    }
599    }
600    }
601    }
602    }
603    }
604    }
605    }
606    }
607    }
608    }
609    }
610    }
611    }
612    }
613    }
614    }
615    }
616    }
617    }
618    }
619    }
620    }
621    }
622    }
623    }
624    }
625    }
626    }
627    }
628    }
629    }
630    }
631    }
632    }
633    }
634    }
635    }
636    }
637    }
638    }
639    }
640    }
641    }
642    }
643    }
644    }
645    }
646    }
647    }
648    }
649    }
650    }
651    }
652    }
653    }
654    }
655    }
656    }
657    }
658    }
659    }
660    }
661    }
662    }
663    }
664    }
665    }
666    }
667    }
668    }
669    }
670    }
671    }
672    }
673    }
674    }
675    }
676    }
677    }
678    }
679    }
680    }
681    }
682    }
683    }
684    }
685    }
686    }
687    }
688    }
689    }
690    }
691    }
692    }
693    }
694    }
695    }
696    }
697    }
698    }
699    }
700    }
701    }
702    }
703    }
704    }
705    }
706    }
707    }
708    }
709    }
710    }
711    }
712    }
713    }
714    }
715    }
716    }
717    }
718    }
719    }
720    }
721    }
722    }
723    }
724    }
725    }
726    }
727    }
728    }
729    }
730    }
731    }
732    }
733    }
734    }
735    }
736    }
737    }
738    }
739    }
740    }
741    }
742    }
743    }
744    }
745    }
746    }
747    }
748    }
749    }
750    }
751    }
752    }
753    }
754    }
755    }
756    }
757    }
758    }
759    }
760    }
761    }
762    }
763    }
764    }
765    }
766    }
767    }
768    }
769    }
770    }
771    }
772    }
773    }
774    }
775    }
776    }
777    }
778    }
779    }
780    }
781    }
782    }
783    }
784    }
785    }
786    }
787    }
788    }
789    }
790    }
791    }
792    }
793    }
794    }
795    }
796    }
797    }
798    }
799    }
800    }
801    }
802    }
803    }
804    }
805    }
806    }
807    }
808    }
809    }
810    }
811    }
812    }
813    }
814    }
815    }
816    }
817    }
818    }
819    }
820    }
821    }
822    }
823    }
824    }
825    }
826    }
827    }
828    }
829    }
830    }
831    }
832    }
833    }
834    }
835    }
836    }
837    }
838    }
839    }
840    }
841    }
842    }
843    }
844    }
845    }
846    }
847    }
848    }
849    }
850    }
851    }
852    }
853    }
854    }
855    }
856    }
857    }
858    }
859    }
860    }
861    }
862    }
863    }
864    }
865    }
866    }
867    }
868    }
869    }
870    }
871    }
872    }
873    }
874    }
875    }
876    }
877    }
878    }
879    }
880    }
881    }
882    }
883    }
884    }
885    }
886    }
887    }
888    }
889    }
890    }
891    }
892    }
893    }
894    }
895    }
896    }
897    }
898    }
899    }
900    }
901    }
902    }
903    }
904    }
905    }
906    }
907    }
908    }
909    }
910    }
911    }
912    }
913    }
914    }
915    }
916    }
917    }
918    }
919    }
920    }
921    }
922    }
923    }
924    }
925    }
926    }
927    }
928    }
929    }
930    }
931    }
932    }
933    }
934    }
935    }
936    }
937    }
938    }
939    }
940    }
941    }
942    }
943    }
944    }
945    }
946    }
947    }
948    }
949    }
950    }
951    }
952    }
953    }
954    }
955    }
956    }
957    }
958    }
959    }
960    }
961    }
962    }
963    }
964    }
965    }
966    }
967    }
968    }
969    }
970    }
971    }
972    }
973    }
974    }
975    }
976    }
977    }
978    }
979    }
980    }
981    }
982    }
983    }
984    }
985    }
986    }
987    }
988    }
989    }
990    }
991    }
992    }
993    }
994    }
995    }
996    }
997    }
998    }
999    }
1000   }

```

File info ⓘ



```

86     System.out.print("Enter operand 1: ");
87     float mul1=sc.nextFloat();
88     System.out.print("Enter operand 2: ");
89     float mul2=sc.nextFloat();
90     cal.product(mul1,mul2);
91     break;
92 case 4:
93     System.out.print("Enter operand 1: ");
94     float div1=sc.nextFloat();
95     System.out.print("Enter operand 2: ");
96     float div2=sc.nextFloat();
97     if(div2==0)
98     {
99         throw new ArithmeticException("Number cannot be divided by zero!!");
100     }
101     cal.division(div1,div2);
    break;
default:
    System.out.println("Invalid choice.");

```

File info ⓘ



```

101         calcDivision(div1,div2),
102         break;
103     default:
104         System.out.println("Invalid choice: ");
105     }
106 }
107 catch(InputMismatchException ime)
108 {
109     System.out.println("You have entered input of wrong datatype!!");
110 }
111 catch(ArithmeticException ae)
112 {
113     System.out.println(ae.getMessage());
114 }
115 }
116 }

```

⋮ File info ⓘ



Author: v. pavan kumar

SAP ID:51834509

1. ADD

2. SUBTRACT

3. MULTIPLICATION

4. DIVISION

5. EXIT

Enter your choice:

1

Enter operand 1: 10

Enter operand 2: 10

Enter operand 3(if you want. else enter 0): 0

10.0+10.0=20.0

Process finished.



```
1 public class Demo {
2     public static void main(String []args) {
3         String str[] = { "s", "k", "r", "v", "n"};
4         String temp;
5         System.out.println("Sorted string...");
6         for (int j = 0; j < str.length; j++) {
7             for (int i = j + 1; i < str.length; i++) {
8                 // comparing strings
9                 if (str[i].compareTo(str[j]) < 0) {
10                     temp = str[j];
11                     str[j] = str[i];
12                     str[i] = temp;
13                 }
14             }
15             System.out.println(str[j]);
16         }
17     }
18 }
```

⋮ File info ⓘ



× Terminal



Sorted string...

k  
n  
r  
s  
v

Process finished.