

```
github.com/ishika22love/oop-java/blob/master/week6_extra_programs/ActorMain.java

1 import java.util.Scanner;
2
3 class Actor
4 {
5     private String id,name;
6     private int no_of_movies,no_years_of_exp;
7     public double avg_exp;
8
9     Scanner ss=new Scanner(System.in);
10
11     void Accept()
12     {
13         System.out.println("Enter the Details of the Actor: ");
14         System.out.println("ID: ");id=ss.next();
15         System.out.println("Name: ");name=ss.next();
16         System.out.println("Number of Movies: ");no_of_movies=ss.nextInt();
17         System.out.println("Number of years of Experience: ");no_years_of_exp=ss.nextInt();
18         this.calc_avg_exp();
19     }
20
21     void calc_avg_exp()
22     {
23         avg_exp=no_of_movies/no_years_of_exp;
24     }
25
26     public double getExp()
27     {return avg_exp;}
28
29     public String toString()
30     {
31         return ("The Actor Details are:\nID: "+id+"\nName: "+name+"\nNumber of Movies: "+no_of_movies+"\nNumber of Years of Experience: "+no_years_of_exp+"\nAverage Experience: "+avg_exp);
32     }
33 }
34
35 public class ActorMain
```

```

33 }
34
35 class ActorMain
36 {
37     public static void main(String args[])
38     {
39         Scanner ss=new Scanner(System.in);
40         int n,ind=0;
41         double max=0;
42         System.out.println("Enter the number of Actors: ");
43         n=ss.nextInt();
44         Actor ac[]=new Actor[n];
45         for(int i=0;i<n;i++)
46         {
47             ac[i]=new Actor();
48             ac[i].Accept();
49
50         }
51         for(int i=0;i<n;i++)
52         {
53             if(max<ac[i].getExp())
54             {
55                 max=ac[i].getExp();
56                 ind=i;
57             }
58         }
59         System.out.println("The actor with max experience: ");
60         System.out.println(ac[ind]);
61
62     }
63 }

```

23 lines (22 sloc) 597 Bytes

Raw Blame   

```
1 import java.util.Scanner;
2 import java.util.Arrays;
3
4 class ArrayMain
5 {
6     public static void main(String args[])
7     {
8         Scanner ss=new Scanner(System.in);
9         double arr[]=new double[args.length];
10        int len=args.length;
11        System.out.println("The Commandline Arguments are:");
12        for(int i=0;i<len;i++)
13        {
14            System.out.printf(" %d ",args[i]);
15            arr[i]=Double.parseDouble(args[i]);
16        }
17        Arrays.sort(arr);
18        for(int i=0;i<len;i++)
19        {
20            System.out.printf(" %d ",arr[i]);
21        }
22    }
23 }
```

© 2020 GitHub, Inc.

[Terms](#)

[Privacy](#)

[Security](#)

[Status](#)

[Help](#)



[Contact GitHub](#)

[Pricing](#)

[API](#)

[Training](#)

[Blog](#)

[About](#)

Untitled - Notepad

File Edit Format View Help

```
import java.util.Scanner;
```

```
class CircleDemo
```

```
{
```

```
    private double radius;
    private double area;
    private double perimeter;
```

```
    Scanner ss=new Scanner(System.in);
```

```
    void accept()
```

```
    {
        System.out.println("Enter the Radius of the circle: ");
        radius=ss.nextDouble();
        this.Calculate();
    }
```

```
    void Calculate()
```

```
    {
        area=3.14*radius*radius;
        perimeter=2*3.14*radius;
    }
```

```
    void display()
```

```
    {
        System.out.printf("The Area of the Circle is: %.2f",area);
        System.out.printf("\nThe perimeter of the Circle is: %.2f",perimeter);
    }
```

```
class CircleMain
```

```
{
```

```
    public static void main(String args[])
```

```
    {
        CircleDemo cd = new CircleDemo();
    }
```

Untitled - Notepad

File Edit Format View Help

```
private double perimeter;

Scanner ss=new Scanner(System.in);

void accept()
{
    System.out.println("Enter the Radius of the circle: ");
    radius=ss.nextDouble();
    this.Calculate();
}

void Calculate()
{
    area=3.14*radius*radius;
    perimeter=2*3.14*radius;
}

void display()
{
    System.out.printf("The Area of the Circle is: %.2f",area);
    System.out.printf("\n\nThe perimeter of the Circle is: %.2f",perimeter);
}

}

class CircleMain
{
    public static void main(String args[])
    {
        CircleDemo c1=new CircleDemo();
        c1.accept();
        c1.display();
    }
}

}
```

github.com/ishika22love/oop-java/blob/master/week6\_extra\_programs/MatrixMain.java

```
1 import java.util.Scanner;
2
3 class Matrix
4 {
5     private int i,j,m,n;
6
7     int matr[][]=new int[10][10];
8     int matr_tr[][]=new int[10][10];
9
10    Scanner ss=new Scanner(System.in);
11
12    void Accept()
13    {
14        System.out.println("Enter the Details of the Matrix: ");
15        System.out.println("Enter the number of rows: ");
16        m=ss.nextInt();
17        System.out.println("Enter the number of columns: ");
18        n=ss.nextInt();
19        for(i=0;i<m;i++)
20        {
21            System.out.println("Enter the elements of row "+i+": ");
22            for(j=0;j<n;j++)
23                matr[i][j]=ss.nextInt();
24        }
25    }
26
27    void Transpose()
28    {
29        for(i=0;i<n;i++)
30        {
31            for(j=0;j<m;j++)
32                matr_tr[i][j]=matr[j][i];
33        }
34    }
35 }
```

```

32     matr_tr[i][j] = matr[j][i];
33 }
34 System.out.println("*****");
35 System.out.println("The Transpose of Matrix is:");
36 for(i=0;i<n;i++)
37 {
38     for(j=0;j<m;j++)
39         System.out.printf(" %matr_tr[i][j] ", "");
40     System.out.printf("\n");
41 }
42 }
43 void Display()
44 {
45     System.out.println("*****");
46     System.out.println("The Matrix is:");
47     for(i=0;i<n;i++)
48     {
49         for(j=0;j<m;j++)
50             System.out.printf(" %matr[i][j] ", "");
51         System.out.printf("\n");
52     }
53 }
54 }
55 }
56 }
57 }
58 }
59 }
60 }
61 }
62 }
63 }
64 }
65 }
66 }
67 }
68 }
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 }
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 }

```

```
1  /*class StaticDemo
2  {
3      static int x; //class variable
4
5      static void setd(int x1)
6      { x=x1;}
7
8      static
9      { x=10;
10         System.out.println("Inside Static Block x="+x);
11     }
12     public static void main(String ss[])
13     {
14         System.out.println("Inside main Block");
15         setd(100);
16         System.out.println("Inside main method:"+x);
17     }
18 }*/
19
20 class StaticDemo
21 {
22     static int x;
23     int y=100;
24
25     static
26     {
27         System.out.println("Hello Good Morning");
28     }
29
30     static void setd(int x1)
31     {
32         x=x1;
33         System.out.println("Inside setd x="+x);
34         //System.out.println("y="+y);
```



```
37 void nonstatic()
38 {
39     System.out.println("Inside nonstatic x="+x);
40     System.out.println("Inside nonstatic y="+y);
41 }
42
43 static
44 {
45     System.out.println("Inside Static Block");
46 }
47 }
48
49 class StaticDemo1
50 {
51     static int m=0,n=0;
52     public static void main(String ss[])
53     {
54         int m=10,x=20;
55         {
56             int n=30;
57             System.out.println("m+n= "+(m+n));
58
59         }
60         x=m+n;
61         System.out.println(n);
62         System.out.println("x= "+x);
63
64
65         /*System.out.println("Jai Shree Ram!!");
66         System.out.println("Inside main Block");
67         StaticDemo.setd(100);
68         System.out.println("x:"+StaticDemo.x);
69         StaticDemo sd=new StaticDemo();
70         System.out.println("y:"+sd.y);
71         ... ..
```

```

1 import java.util.Scanner;
2
3 class StringCalc
4 {
5     private String str;
6     private int vow_count, con_count, spa_count;
7     private char ch;
8
9     Scanner ss= new Scanner(System.in);
10
11     void Accept()
12     {
13         System.out.println("Enter the String: ");
14         str=ss.nextLine();
15         calculate();
16     }
17
18     void calculate()
19     {
20         for(int i=0;i<str.length();i++)
21         {
22             ch=str.charAt(i);
23             if((ch=='a' || ch=='e' || ch=='o' || ch=='u' || ch=='i'))
24                 vow_count++;
25             else if(ch==' ')
26                 spa_count++;
27             else
28                 con_count++;
29         }
30     }
31
32     void display()
33     {
34         System.out.println("The number of vowels are: "+vow_count);
35     }
36 }

```

```
22:         ch= str.charAt(i);
23:         if(ch=='a' || ch=='e' || ch=='o' || ch=='u' || ch=='i')
24:             vow_count++;
25:         else if(ch==' ')
26:             spa_count++;
27:         else
28:             con_count++;
29:     }
30: }
31:
32: void display()
33: {
34:     System.out.println("The number of vowels are: "+vow_count);
35:     System.out.println("The number of consonants are: "+con_count);
36:     System.out.println("The number of Spaces are: "+spa_count);
37: }
38: }
39:
40: class StringMain
41: {
42:     public static void main(String args[])
43:     {
44:         StringCalc c1= new StringCalc();
45:         c1.Accept();
46:         c1.display();
47:     }
48: }
```

