```java
import java.util.Scanner;

public class ForLoops1 {
   public static void main(String[] args) {
      Scanner input = new Scanner(System.in);
      int i;

      for(i = 1; i <= 100; i++)
      {
         if(i%5 ==0)
            System.out.print(i + " ");
      }
      System.out.println();
      for(i = 5; i <= 100; i+=5)
      {
         System.out.print(i  + " ");
      }
      System.out.println();
      // there is no difference between them
      // the 2nd one will run faster because it has fewer statements to check and doesn't check every number for 1 - 100

      for(i = 1; i <= 50; i+=2)
      {
         System.out.print(i  + " ");
      }
      System.out.println();
      for(i = 3; i <= 30; i+=3)
      {
         System.out.print(i  + " ");
      }
      System.out.println();
      for(i = 2; i <= 100; i+=2)
      {
         System.out.print(i  + " ");
      }
      System.out.println();

      int sum = 0;
      for(i = 1; i <= 10; i++)
      {
         System.out.print(i  + " ");
         sum += i;
      }
      System.out.println();
      System.out.println(sum);

      sum = 0;
      for(i = 30; i <= 50; i++)
```

```
49          {
50             System.out.print(i  + " ");
51             sum += i;
52          }
53          System.out.println();
54          System.out.println(sum);
55
56
57          for(i = 1; i <= 50; i++)
58          {
59             if (i%8 ==0)
60                System.out.print(i  + " nice ");
61             else System.out.print(i + " ");
62          }
63          System.out.println();
64
65          System.out.print("Enter num1 ->");
66          int num1 = input.nextInt();
67          System.out.print("Enter num2 ->");
68          int num2 = input.nextInt();
69          sum = 0;
70          int count = 0;
71          for(i = Math.min(num1, num2); i <= Math.max(num1, num2); i++)
72          {
73             count++;
74             sum += i;
75          }
76          System.out.println("there were " + count + " numbers");
77          System.out.println("sum = "+ sum);
78
79          for(i = 12; i < 100; i +=3)
80          {
81             System.out.print(i  + " ");
82          }
83          System.out.println();
84          for(i = 10; i < 100; i ++)
85          {
86             if (Math.sqrt(i)%1 == 0)
87                System.out.print(i + " ");
88          }
89          System.out.println();
90          for(i = 12; i < 100; i +=3)
91          {
92             if (Math.sqrt(i)%1 == 0)
93                System.out.print(i + " ");
94          }
95          System.out.println();
96          for(i = 10; i < 100; i ++)
97          {
```

```
98              if (Math.sqrt(i)%1 == 0)
99                 System.out.print("has whole squrt: " + i);
100             else if (i%3==0)
101             {
102                System.out.println("deviadeable by 3: " + i);
103             }
104          }
105
106          System.out.println();
107          for(i = 10; i < 100; i ++)
108          {
109             if (i%7 == 0 || i%10 ==7 || i/10 == 7)
110                System.out.print(i + " ");
111
112          }
113       }
114    }
115    /*
116    5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100
117    5 10 15 20 25 30 35 40 45 50 55 60 65 70 75 80 85 90 95 100
118    1 3 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45 47 49
119    3 6 9 12 15 18 21 24 27 30
120    2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50 52 54
        56 58 60 62 64 66 68 70 72 74 76 78 80 82 84 86 88 90 92 94 96 98 100
121    1 2 3 4 5 6 7 8 9 10
122    55
123    30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50
124    840
125    1 2 3 4 5 6 7 8 nice 9 10 11 12 13 14 15 16 nice 17 18 19 20 21 22 23 24 nice
       25 26 27 28 29 30 31 32 nice 33 34 35 36 37 38 39 40 nice 41 42 43 44 45 46
       47 48 nice 49 50
126    Enter num1 ->1
127    Enter num2 ->15
128    there were 15 numbers
129    sum = 120
130    12 15 18 21 24 27 30 33 36 39 42 45 48 51 54 57 60 63 66 69 72 75 78 81 84
       87 90 93 96 99
131    16 25 36 49 64 81
132    36 81
133    deviadeable by 3: 12
134    deviadeable by 3: 15
135    has whole squrt: 16deviadeable by 3: 18
136    deviadeable by 3: 21
137    deviadeable by 3: 24
138    has whole squrt: 25deviadeable by 3: 27
139    deviadeable by 3: 30
140    deviadeable by 3: 33
141    has whole squrt: 36deviadeable by 3: 39
142    deviadeable by 3: 42
```

```
143 deviadeable by 3: 45
144 deviadeable by 3: 48
145 has whole squrt: 49deviadeable by 3: 51
146 deviadeable by 3: 54
147 deviadeable by 3: 57
148 deviadeable by 3: 60
149 deviadeable by 3: 63
150 has whole squrt: 64deviadeable by 3: 66
151 deviadeable by 3: 69
152 deviadeable by 3: 72
153 deviadeable by 3: 75
154 deviadeable by 3: 78
155 has whole squrt: 81deviadeable by 3: 84
156 deviadeable by 3: 87
157 deviadeable by 3: 90
158 deviadeable by 3: 93
159 deviadeable by 3: 96
160 deviadeable by 3: 99
161
162 14 17 21 27 28 35 37 42 47 49 56 57 63 67 70 71 72 73 74 75 76 77 78 79 84
    87 91 97 98
163 Process finished with exit code 0
164
165  */
166
```