

Philip Pesic

Week 6

February 26 2023

Week 6 Prog 5

Write program using two(2) nested 'for' loops to print out the values of a Multiplication

Table 12 By 12.

```
package Prog5;
import java.util.Formatter;

class prog5 {
    public static void main(String[] args) {

        Formatter f = new Formatter();

        for (int mult = 1; mult <= 12; mult++) {
            for (int count = 1; count <= 12; count++) {
                System.out.print(f.format("%5d", mult * count));
                f = new Formatter();
            }
            System.out.println();
        }
        f.close();
        System.out.println("Philip Pesic 2/26/23");
    }
}
```

Philip Pesic

Week 6

February 26 2023

Week 6 Prog 5

The screenshot shows an IDE with a Java file named `prog5.java` open. The code defines a class `prog5` with a `main` method. Inside `main`, a `Formatter` object `f` is created. Two nested `for` loops are used to calculate and print a multiplication table. The outer loop iterates over `mult` from 1 to 12, and the inner loop iterates over `count` from 1 to 12. The output of the program is displayed in the `TERMINAL` panel, showing a 12x12 grid of numbers. The status bar at the bottom indicates the current line and column: `Ln 11, Col 46`.

```
1 package Prog5;
2 import java.util.Formatter;
3
4 class prog5 {
5     public static void main(String[] args) {
6         Formatter f = new Formatter();
7
8         for (int mult = 1; mult <= 12; mult++) {
9             for (int count = 1; count <= 12; count++) {
10                 System.out.print(f.format("{} ", mult * count));
11                 f = new Formatter();
12             }
13             System.out.println();
14         }
15         f.close();
16         System.out.println("Philip Pesic 2/26/23");
17     }
18 }
19
20
```

	1	2	3	4	5	6	7	8	9	10	11	12
1	1	2	3	4	5	6	7	8	9	10	11	12
2	2	4	6	8	10	12	14	16	18	20	22	24
3	3	6	9	12	15	18	21	24	27	30	33	36
4	4	8	12	16	20	24	28	32	36	40	44	48
5	5	10	15	20	25	30	35	40	45	50	55	60
6	6	12	18	24	30	36	42	48	54	60	66	72
7	7	14	21	28	35	42	49	56	63	70	77	84
8	8	16	24	32	40	48	56	64	72	80	88	96
9	9	18	27	36	45	54	63	72	81	90	99	108
10	10 <td>20<td>30<td>40<td>50<td>60<td>70<td>80<td>90<td>100<td>110<td>120</td></td></td></td></td></td></td></td></td></td></td>	20 <td>30<td>40<td>50<td>60<td>70<td>80<td>90<td>100<td>110<td>120</td></td></td></td></td></td></td></td></td></td>	30 <td>40<td>50<td>60<td>70<td>80<td>90<td>100<td>110<td>120</td></td></td></td></td></td></td></td></td>	40 <td>50<td>60<td>70<td>80<td>90<td>100<td>110<td>120</td></td></td></td></td></td></td></td>	50 <td>60<td>70<td>80<td>90<td>100<td>110<td>120</td></td></td></td></td></td></td>	60 <td>70<td>80<td>90<td>100<td>110<td>120</td></td></td></td></td></td>	70 <td>80<td>90<td>100<td>110<td>120</td></td></td></td></td>	80 <td>90<td>100<td>110<td>120</td></td></td></td>	90 <td>100<td>110<td>120</td></td></td>	100 <td>110<td>120</td></td>	110 <td>120</td>	120
11	11 <td>22<td>33<td>44<td>55<td>66<td>77<td>88<td>99<td>110<td>121<td>132</td></td></td></td></td></td></td></td></td></td></td>	22 <td>33<td>44<td>55<td>66<td>77<td>88<td>99<td>110<td>121<td>132</td></td></td></td></td></td></td></td></td></td>	33 <td>44<td>55<td>66<td>77<td>88<td>99<td>110<td>121<td>132</td></td></td></td></td></td></td></td></td>	44 <td>55<td>66<td>77<td>88<td>99<td>110<td>121<td>132</td></td></td></td></td></td></td></td>	55 <td>66<td>77<td>88<td>99<td>110<td>121<td>132</td></td></td></td></td></td></td>	66 <td>77<td>88<td>99<td>110<td>121<td>132</td></td></td></td></td></td>	77 <td>88<td>99<td>110<td>121<td>132</td></td></td></td></td>	88 <td>99<td>110<td>121<td>132</td></td></td></td>	99 <td>110<td>121<td>132</td></td></td>	110 <td>121<td>132</td></td>	121 <td>132</td>	132
12	12 <td>24<td>36<td>48<td>60<td>72<td>84<td>96<td>108<td>120<td>132<td>144</td></td></td></td></td></td></td></td></td></td></td>	24 <td>36<td>48<td>60<td>72<td>84<td>96<td>108<td>120<td>132<td>144</td></td></td></td></td></td></td></td></td></td>	36 <td>48<td>60<td>72<td>84<td>96<td>108<td>120<td>132<td>144</td></td></td></td></td></td></td></td></td>	48 <td>60<td>72<td>84<td>96<td>108<td>120<td>132<td>144</td></td></td></td></td></td></td></td>	60 <td>72<td>84<td>96<td>108<td>120<td>132<td>144</td></td></td></td></td></td></td>	72 <td>84<td>96<td>108<td>120<td>132<td>144</td></td></td></td></td></td>	84 <td>96<td>108<td>120<td>132<td>144</td></td></td></td></td>	96 <td>108<td>120<td>132<td>144</td></td></td></td>	108 <td>120<td>132<td>144</td></td></td>	120 <td>132<td>144</td></td>	132 <td>144</td>	144

Philip Pesic 2/26/23

I learned: how to nest loops