

## src/Problem1.java

```
1  /*
2  Name: Hunter Poole
3  Date: 3/5/25
4  HW #: 6
5  Problem #: 1
6  Source Code: Problem1.java
7  Action: Provides proof that a penny doubled every day for
8          thirty days would make you a millionaire.
9  */
10
11 public class Problem1
12 {
13     public static void main(String[] args)
14     {
15         double Balance = 0.01;
16
17         System.out.printf("%s %10s %n%s %13s %n%d %7s%.2f%n", "Day", "Amount",
18             "----", "-----", 1, "$", Balance);
19
20         for (int i = 2; i <= 30; i++)
21         {
22             Balance = Balance * 2.0;
23
24             if (i < 10)
25             {
26                 System.out.printf("%d %7s%.2f %n", i, "$", Balance);
27             }
28             else
29             {
30                 System.out.printf("%d %6s%.2f %n", i, "$", Balance);
31             }
32         }
33     }
34 }
35
36
37 /*
38 Day      Amount
39 ----      -----
40 1         $0.01
41 2         $0.02
42 3         $0.04
43 4         $0.08
44 5         $0.16
45 6         $0.32
46 7         $0.64
47 8         $1.28
```

48	9	\$2.56
49	10	\$5.12
50	11	\$10.24
51	12	\$20.48
52	13	\$40.96
53	14	\$81.92
54	15	\$163.84
55	16	\$327.68
56	17	\$655.36
57	18	\$1310.72
58	19	\$2621.44
59	20	\$5242.88
60	21	\$10485.76
61	22	\$20971.52
62	23	\$41943.04
63	24	\$83886.08
64	25	\$167772.16
65	26	\$335544.32
66	27	\$671088.64
67	28	\$1342177.28
68	29	\$2684354.56
69	30	\$5368709.12
70	*/	
71		

## src/Problem2.java

```
1  /*
2  Name: Hunter Poole
3  Date: 3/5/25
4  HW #: 6
5  Problem #: 2
6  Source Code: Problem2.java
7  Action: Takes two user integers and a user math operand.
8          Performs the specified operation between the two
9          numbers. Provides the formula used. Loops until
10         N is entered.
11  */
12
13  import java.util.Scanner;
14
15  public class Problem2
16  {
17      public static void main(String[] args)
18      {
19          char Operand;
20          int X, Y;
21
22          do
23          {
24              Scanner Input = new Scanner(System.in);
25
26              System.out.print("Enter your first number: ");
27              X = Input.nextInt();
28
29              System.out.print("Enter your second number: ");
30              Y = Input.nextInt();
31
32              System.out.print("Enter your math operand. N to quit: ");
33              Operand = Input.next().charAt(0);
34
35
36              switch (Operand)
37              {
38                  case '+':
39                      System.out.printf("%d %s %d %s %d %n%n", X, "+", Y, "=", (X
40 + Y));
41                      break;
42                  case '-':
43                      System.out.printf("%d %s %d %s %d %n%n", X, "-", Y, "=", (X
44 - Y));
45                      break;
46                  case '*':
```

```
45         System.out.printf("%d %s %d %s %d %n%n", X, "*", Y, "=", (X
    * Y));
46         break;
47     case '/':
48         System.out.printf("%d %s %d %s %d %n%n", X, "/", Y, "=", (X
    / Y));
49         break;
50     case '%':
51         System.out.printf("%d %s %d %s %d %n%n", X, "%", Y, "=", (X
    % Y));
52         break;
53     default:
54         if (Operand != 'N')
55         {
56             System.out.printf("%s %n%n", "Error! Please enter a valid
    math operand (+ - * / %)");
57         }
58         break;
59     }
60     } while (Operand != 'N');
61 }
62 }
63
64 /*
65 Enter your first number: 4
66 Enter your second number: 6
67 Enter your math operand. N to quit: +
68 4 + 6 = 10
69
70 Enter your first number: 18
71 Enter your second number: 6
72 Enter your math operand. N to quit: %
73 18 % 6 = 0
74
75 Enter your first number: 8
76 Enter your second number: 32
77 Enter your math operand. N to quit: *
78 8 * 32 = 256
79
80 Enter your first number: 512
81 Enter your second number: 300
82 Enter your math operand. N to quit: -
83 512 - 300 = 212
84
85 Enter your first number: 44
86 Enter your second number: 11
87 Enter your math operand. N to quit: /
88 44 / 11 = 4
89
90 Enter your first number: 1
```

```
91 Enter your second number: 1
92 Enter your math operand. N to quit: D
93 Error! Please enter a valid math operand (+ - * / %)
94
95 Enter your first number: 1
96 Enter your second number: 1
97 Enter your math operand. N to quit: .
98 Error! Please enter a valid math operand (+ - * / %)
99
100 Enter your first number: 1
101 Enter your second number: 1
102 Enter your math operand. N to quit: +
103 1 + 1 = 2
104
105 Enter your first number: 1
106 Enter your second number: 1
107 Enter your math operand. N to quit: N
108 */
```

## src/Problem3.java

```
1  /*
2  Name: Hunter Poole
3  Date: 3/5/25
4  HW #: 6
5  Problem #: 3
6  Source Code: Problem3.java
7  Action: Takes two strings, outputs them in double quotes with their length.
8          Combines strings, outputs combo and combo length.
9          Outputs combo string in all caps.
10         Finds # of capital letters in combo string and returns that number.
11  */
12
13  import java.util.Scanner;
14
15  public class Problem3
16  {
17      public static void main(String[] args)
18      {
19          String FirstString, SecondString, ComboString;
20          int UppercaseCount = 0;
21
22          Scanner Input = new Scanner(System.in);
23          System.out.print("Enter first string --> ");
24          FirstString = Input.nextLine();
25
26          System.out.print("Enter second string --> ");
27          SecondString = Input.nextLine();
28
29          ComboString = FirstString + " " + SecondString;
30
31          System.out.printf("%n%s %8s \"%s\" %s %d %n", "String one", "-->",
32          FirstString, "Length, ", FirstString.length());
33          System.out.printf("%s %8s \"%s\" %s %d %n", "String two", "-->",
34          SecondString, "Length, ", SecondString.length());
35          System.out.printf("%s \"%s\" %s %d %n", "Strings combined-->",
36          ComboString, "Length, ", ComboString.length());
37
38          System.out.printf("%s \"%s\" %n", "String Upper Case --> ",
39          ComboString.toUpperCase());
40
41          for (int i = 0; i < ComboString.length(); i++)
42          {
43              if (Character.isUpperCase(ComboString.charAt(i)))
44              {
45                  UppercaseCount++;
46              }
47          }
48      }
49  }
```

```
44
45         System.out.printf("%s %d %s\n", "The final string has", UppercaseCount,
"upper case letters");
46     }
47 }
48
49 /*
50 Enter first string --> This
51 Enter second string --> is Fun!
52
53 String one      --> "This" Length,  4
54 String two      --> "is Fun!" Length,  7
55 Strings combined--> "This is Fun!" Length,  12
56
57 String Upper Case --> "THIS IS FUN!"
58 The final string has 2 upper case letters
59
60 Enter first string --> Perfect
61 Enter second string --> teST C453_*
62
63 String one      --> "Perfect" Length,  7
64 String two      --> "teST C453_*" Length,  11
65 Strings combined--> "Perfect teST C453_*" Length,  19
66
67 String Upper Case --> "PERFECT TEST C453_*"
68 The final string has 4 upper case letters
69
70 Enter first string --> STILL fun
71 Enter second string --> but not as much FUN
72
73 String one      --> "STILL fun" Length,  9
74 String two      --> "but not as much FUN" Length,  19
75 Strings combined--> "STILL fun but not as much FUN" Length,  29
76
77 String Upper Case --> "STILL FUN BUT NOT AS MUCH FUN"
78 The final string has 8 upper case letters
79 */
```