Problem recap skipped due to length of problem.

Three Step Analysis:

A. Take any <u>character</u> as input. Return the entered character and its ASCII value.
B. Determine if the entered character is upper case, lower case, 0-9, or other. Store that info.
C. Return the next two characters
D. Loop until # is entered. Then return a table for the count of each character type.

INPUT	OUTPUT	EQUATIONS
Any char	The same char	**do…while** (Ch != '#')
# to quit	Char's ASCII value	**if** (Ch != '#')
	Next two characters (from input char)	**if** (Character.isUpperCase(Ch)) Uppercase++ **else if** (Character.isLowerCase(Ch)) Lowercase++ **else if** (Character.isDigit(Ch)) Digit++ **else if** (Ch != '#') Other++ **end if**
	Table of counts - display quantity of each type of char entered	

E. Limits / Constraints:
  a. Can only take one character at a time
  b. Cannot handle whitespace or null values.
      i. Only functions for ASCII characters within typeable range, excluding the space bar.
      ii. 33 - 126 (!, ~)

```
char Ch, NextCh, NextCh2
int Ch_Value, Uppercase = 0, Lowercase = 0, Digit = 0, Other = 0

do
        write "Please enter your character: "
        read Ch
        Ch_Value = Ch

        if (Ch != '#')
                write (Ch + " " + Ch_Value)
                NextCh = Ch
                NextCh2 = ++NextCh
                write ((NextCh++) + " " + (++NextCh2))
        end if

        if (Character.isUpperCase(Ch))
                Uppercase++
        else if (Character.isLowerCase(Ch))
                Lowercase++
        else if (Character.isDigit(Ch))
                Digit++
        else if (Ch != '#')
                Other++
        end if

while (Ch != '#')
end do-while

write ("Number of uppercase: " + Uppercase + "Number of lowercase: " + Lowercase + "Numbers: " +
Digit + "Number of other characters: " + Other)
```

src\Problem1.java

```
 1   /*
 2   Name: Hunter Poole
 3   Date: 2/24/25
 4   HW #: 5
 5   Problem #: 1
 6   Source Code: Problem1.java
 7   Action: Takes a single char, returns char, ASCII value,
 8           and next two characters. Runs in a loop.
 9           Counts # of uppercase, lowercase, digits, and
10           other characters entered. '#' to exit.
11           Displays a table for counts of char types at exit.
12    */
13
14   import java.util.Scanner;
15
16   public class Problem1
17   {
18       public static void main(String[] args)
19       {
20           char Ch, NextCh, NextCh2;
21           int Ch_Value, Uppercase = 0, Lowercase = 0, Digit = 0, Other = 0;
22
23           Scanner Input = new Scanner(System.in);
24
25           do
26           {
27               System.out.print("Please enter your character ---> ");
28               Ch = Input.next().charAt(0);
29               Ch_Value = Ch;
30
31               if (Ch != '#')
32               {
33               System.out.printf("%n%s %c %n%s %c %s %d %n","You entered", Ch, "The ASCII value
     of",
34               Ch, "is", Ch_Value);
35
36               NextCh = Ch;
37               NextCh2 = ++NextCh;
38
39               System.out.printf("%s %c %s %c %n%n", "The next two characters are:", NextCh++,
40                           "and", ++NextCh2);
41               }
42
43               if (Character.isUpperCase(Ch))
44               {
45                   Uppercase++;
46               }
```

```java
47              else if (Character.isLowerCase(Ch))
48              {
49                  Lowercase++;
50              }
51              else if (Character.isDigit(Ch))
52              {
53                  Digit++;
54              }
55              else if (Ch != '#')
56              {
57                  Other++;
58              }
59
60          }while (Ch != '#');
61
62          System.out.printf("%n%s %d %n%s %d %n%s %d %n%s %d", "Number of uppercase--------->",
    Uppercase,
63                              "Number of lowercase--------->", Lowercase, "Number of
    numbers----------->", Digit,
64                              "Number of other characters-->", Other);
65      }
66 }
67
68 /*
69 Please enter your character ---> A
70
71 You entered A
72 The ASCII value of A is 65
73 The next two characters are: B and C
74
75 Please enter your character ---> v
76
77 You entered v
78 The ASCII value of v is 118
79 The next two characters are: w and x
80
81 Please enter your character ---> 5
82
83 You entered 5
84 The ASCII value of 5 is 53
85 The next two characters are: 6 and 7
86
87 Please enter your character ---> @
88
89 You entered @
90 The ASCII value of @ is 64
91 The next two characters are: A and B
92
93 Please enter your character ---> (
```

```
 94
 95   You entered (
 96   The ASCII value of ( is 40
 97   The next two characters are: ) and *
 98
 99   Please enter your character ---> #
100
101   Number of uppercase---------> 1
102   Number of lowercase---------> 1
103   Number of numbers-----------> 1
104   Number of other characters--> 2
105
106       ////// Extremes \\\\\\
107
108   Please enter your character ---> !
109
110   You entered !
111   The ASCII value of ! is 33
112   The next two characters are: " and #
113
114   Please enter your character ---> ~
115
116   You entered ~
117   The ASCII value of ~ is 126
118   The next two characters are:  and ?
119
120   Please enter your character ---> #
121
122   Number of uppercase---------> 0
123   Number of lowercase---------> 0
124   Number of numbers-----------> 0
125   Number of other characters--> 2
126    */
```

2) Write a program that displays the first 40 Fibonacci numbers. A Fibonacci number is created by add the previous two, with the first two always being 0 and 1. A partial sequence is as follows: 0, 1, 1, 2, 3, 5, 8, 13, 21,.... Your table must display 6 numbers per row and use a spacing of 10 for each number. Don't forget to look at handout on formatting output and probably use "printf()".

Three Step Analysis:

    A. Initialize two int variables at 0 & 1.
    B. Add them, display result.
    C. Write variable 2 into variable 1.
    D. Write result into variable 2.
    E. Add them, display result.
    F. Repeat B-E for a total of 40 outputs.

INPUT	OUTPUT	EQUATIONS
	Fibonacci sequence, first 40 numbers.	**for** (i = 39; i > 0; i--)
	0, 1, 1, 2, 3, 5, 8, 13, 21,...	Num3 = Num1 + Num2
		if ((i + 1) % 6 == 1)

    G. Limits / Constraints:
        a. Must have six numbers on a line
        b. Numbers must have spacing of 10
        c. Must go only to the 40th fibonacci number
            i. 102334155
           ii. Counted 0 as 0, 1 as 1, 2 as 1.
                1. https://planetmath.org/listoffibonaccinumbers

int i, Num1 = 0, Num2 = 1, Num3

write Num1, Num2

for (i = 39; i > 0; i--)
      Num3 = Num1 + Num2
      write Num3

      Num1 = Num2
      Num2 = Num3

      if ((i + 1) % 6 == 1)
            write (%n)
      end if
end for

src\Problem2.java

```
1    /*
2    Name: Hunter Poole
3    Date: 2/24/25
4    HW #: 5
5    Problem #: 2
6    Source Code: Problem2.java
7    Action: Displays the first 40 fibonacci numbers in a table.
8             Table to have 6 columns.
9             Numbers in table have width of 10.
10    */
11
12   public class Problem2
13   {
14       public static void main(String[] args)
15       {
16           int i, Num1 = 0, Num2 = 1, Num3;
17
18           System.out.printf("%-10d%-10d", Num1, Num2);
19
20           for (i = 39; i > 0; i--)
21           {
22               Num3 = Num1 + Num2;
23               System.out.printf("%-10d",Num3);
24
25               Num1 = Num2;
26               Num2 = Num3;
27
28               if ((i + 1) % 6 == 1)
29               {
30                   System.out.printf("%n");
31               }
32           }
33       }
34   }
35
36   /*
37   0          1          1          2          3          5
38   8          13         21         34         55         89
39   144        233        377        610        987        1597
40   2584       4181       6765       10946      17711      28657
41   46368      75025      121393     196418     317811     514229
42   832040     1346269    2178309    3524578    5702887    9227465
43   14930352   24157817   39088169   63245986   102334155
44    */
45
46
47   /*
```

```
48   Chose to count Fibonacci number 0 as 0, 1 as 1, 2 as 1, 3 as 2, etc
49   per https://planetmath.org/listoffibonaccinumbers
50   So, 102334155 is the 40th Fibonacci number with 0 and 1 counting as 0 and 1.
51
52   If wrong, update i = 39 --> i = 40, change if statement:
53       if (i % 6 == 1)
54   Will print 165580141 (#41) as the last number, satisfying 6 num per row requirement.
55   */
```

src\Problem3.java

```
 1  /*
 2  Name: Hunter Poole
 3  Date: 2/24/25
 4  HW #: 5
 5  Problem #: 3
 6  Source Code: Problem3.java
 7  Action: Given a series of numbers, return how many of them are even.
 8          Exit program when "0" is entered.
 9  */
10
11  import java.util.Scanner;
12
13  public class Problem3
14  {
15
16      public static void main(String[] args)
17      {
18          int Even_Count = 0;
19
20          Scanner Input = new Scanner(System.in);
21          System.out.print("Please provide your numbers one at a time. Enter 0 to exit: ");
22          int Number = Input.nextInt();
23
24          while (Number != 0)
25          {
26              if (Number % 2 == 0)
27              {
28                  Even_Count++;
29              }
30
31              System.out.print("Next number: ");
32              Number = Input.nextInt();
33          }
34
35          if (Number == 0)
36          {
37              System.out.print("You have entered " + Even_Count + " even numbers.");
38          }
39
40      }
41  }
42
43  /*
44  Please provide your numbers one at a time. Enter 0 to exit: 3
45  Next number: 56
46  Next number: 4
47  Next number: 13
```

```
48   Next number: 779
49   Next number: 46
50   Next number: 0
51   You have entered 3 even numbers.
52
53   Please provide your numbers one at a time. Enter 0 to exit: 905775
54   Next number: 13
55   Next number: 777
56   Next number: 346
57   Next number: 42
58   Next number: 12
59   Next number: 0
60   You have entered 3 even numbers.
61   */
```

src\Problem4.java

```java
1   /*
2   Name: Hunter Poole
3   Date: 2/24/25
4   HW #: 5
5   Problem #: 4
6   Source Code: Problem4
7   Action: Given a whole number, displays multiples of 3 down to 3.
8   */
9
10  import java.util.Scanner;
11
12  public class Problem4
13  {
14      public static void main(String[] args)
15      {
16          Scanner Input = new Scanner(System.in);
17          System.out.print("Provide your number: ");
18          int Num = Input.nextInt();
19
20          for (int Multiples = Num; Multiples >= 3; Multiples--)
21          {
22              if (Multiples % 3 == 0)
23              {
24                  System.out.print(Multiples + "  ");
25              }
26          }
27      }
28  }
29
30  /*
31  Provide your number: 16
32  15  12  9  6  3
33
34  Provide your number: 25
35  24  21  18  15  12  9  6  3
36
37  Provide your number: 70
38  69  66  63  60  57  54  51  48  45  42  39  36  33  30  27  24  21  18  15  12  9  6  3
39   */
```