Using Java's REMAINDER (%) or MOD operation

MOD / REMAINDER
NUMBER % DIVISOR = REMAINDER

The second section of the **Logic-1 Basic** problems (from **isOddNumber**() onwards)
use the **MOD** or **REMAINDER** operation, which uses the percent sign **%**.

There are actually TWO answers when you divide one number (a dividend)
by another (a divisor): a QUOTIENT and a REMAINDER.

For example: 52 divided by 10 gives 5 with a remainder of 2.

In Java, we have TWO different operations for INTEGERS (not floats or doubles):
    Quotient division, which uses a **FORWARD SLASH (/)**, and
    Remainder division, which used the **PERCENT SIGN (%)**.
Therefore:
    52 / 10 gives 5 (the QUOTIENT)
    52 % 10 give 2 (the REMAINDER)

The REMAINDER operation has many uses.

1. You can test for an **EVEN number** by checking that
the remainder is 0 when you divide by 2, e.g. 8 % 2 == 0.
Therefore, given a number n, **if (n % 2 == 0), then the number is even.**

2. You can test for an **ODD number** by checking that the
remainder is NOT 0 when you divide by 2, e.g. 7 % 2 != 0
Therefore, given a number n, **if (n % 2 != 0), then the number is odd.**
NOTE: n % 2 == 1 DOES NOT WORK WITH NEGATIVE NUMBERS!

3. You can also test whether one number is a **FACTOR** of the other.
If you divide a number by a divisor and the remainder is zero,
then the divisor is a factor of that number.
Therefore, if n % 6 == 0, then 6 is a factor of n,
and n is a multiple of 6.
Conversely if n % 6 != 0, then 6 is NOT a factor of n,
and n is NOT a multiple of 6.

4. Given ANY integer, you can **extract the rightmost digit**
(the digit in the 1's column) using MOD 10.
For example:

Using Java's REMAINDER (%) or MOD operation

```
 7186 % 10 give 6
  593 % 10 gives 3
   14 % 10 gives 4
    8 % 10 gives 8
   n % 10 will extract the digit in the 1's column.

5. Given any integer, you can extract the 2nd-to-rightmost digit
(the digit in the 10's column) as folows:
 7186 / 10 % 10 gives 8
  593 / 10 % 10 gives 9
   25 / 10 % 10 gives 2
    8 / 10 % 10 gives 0
   (n / 10) % 10 will extract the digit in the 10's column.

6. Given any integer, you can extract the digit in the 100's column as follows:
 7186 / 100 % 10 gives 1
  593 / 100 % 10 gives 5
   25 / 100 % 10 gives 0
    8 / 100 % 10 gives 0
  (n / 100) % 10 will extract the digit in the 100's column
```