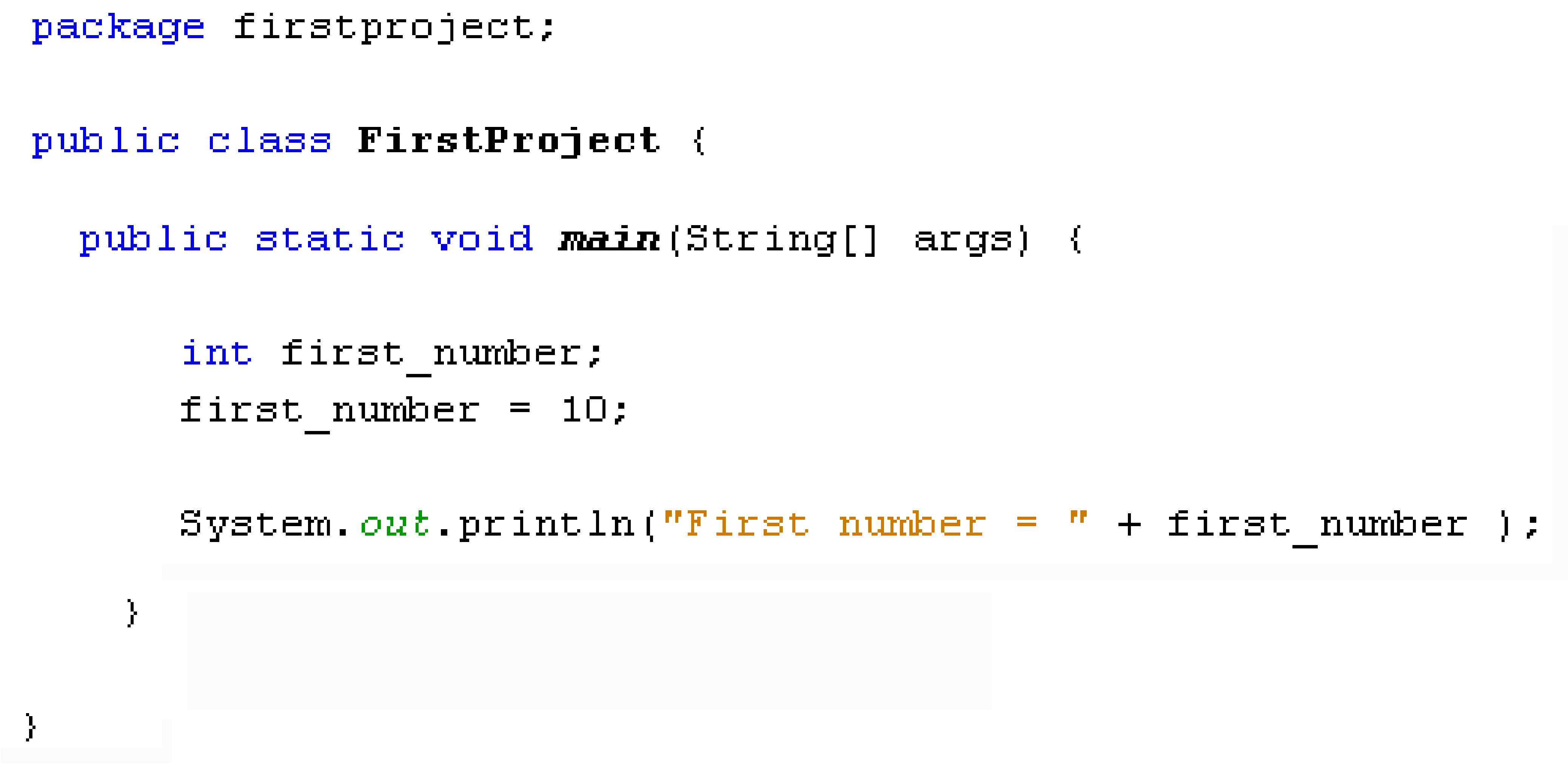
Java for Beginners

SIMPLE ARITHMETIC

ERIC Y. CHOU, PH.D. IEEE SENIOR MEMBER



```
package firstproject;

public class FirstProject {

  public static void main(String[] args) {

      int first_number;
      first_number = 10;

      System.out.println("First number = " + first_number );
    }
}
```

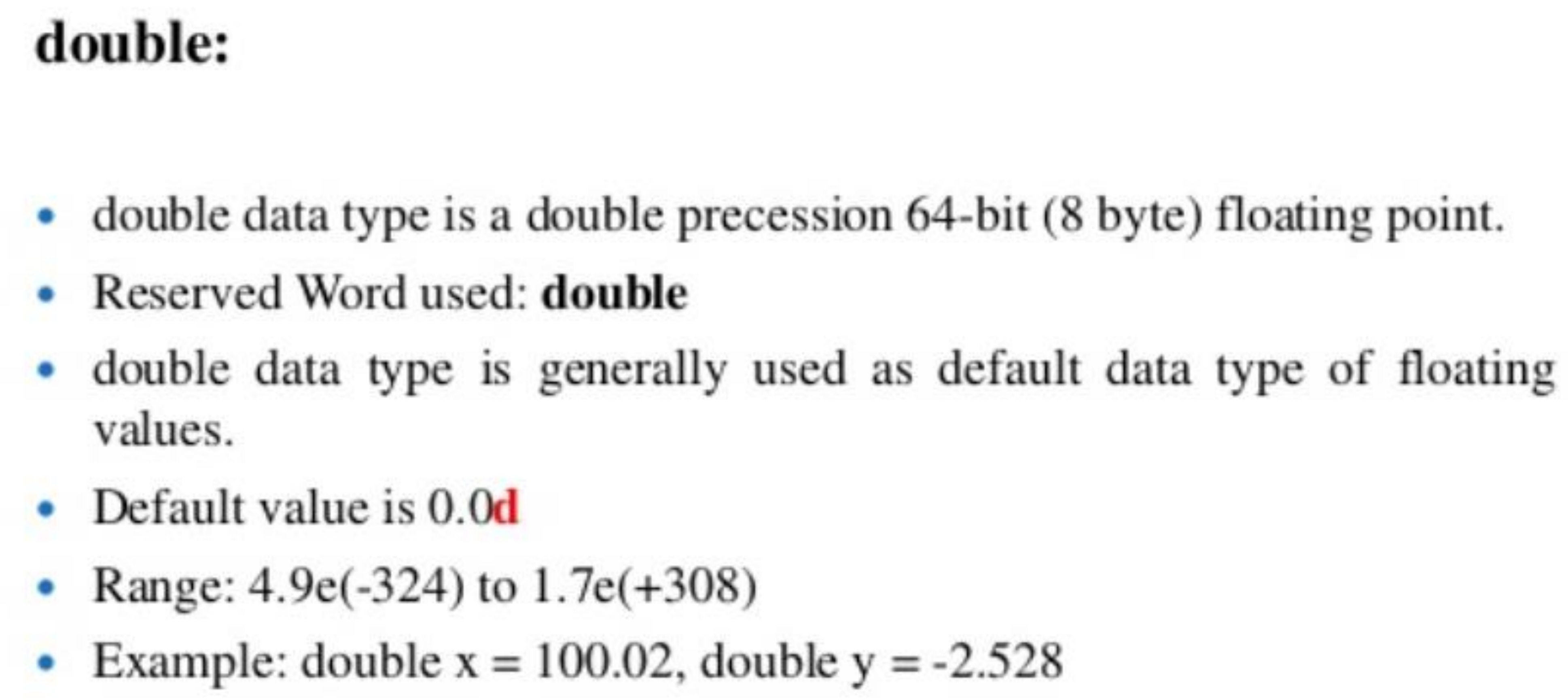
Integer or Real Number?

- Categorize each of the following quantities by whether an `int` or `double` variable would best to store it:

integer (`int`)	real number (`double`)

1. Temperature in degrees Celsius
2. The population of lemmings
3. Your grade point average
4. A person's age in years
5. A person's weight in pounds
6. A person's height in meters
7. Number of miles traveled today
8. Number of dry days in the past month
9. The number of games the volleyball team wins this season
10. Number of seconds left in a game
11. The sum of a group of integers
12. The average of a group of integers

Primitives Data Types



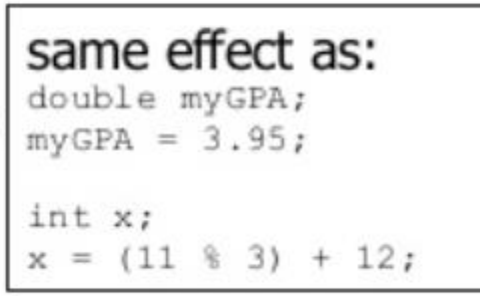
double:

- double data type is a double precession 64-bit (8 byte) floating point.
- Reserved Word used: **double**
- double data type is generally used as default data type of floating values.
- Default value is 0.0**d**
- Range: 4.9e(-324) to 1.7e(+308)
- Example: double x = 100.02, double y = -2.528

- Declaration and initialization statement syntax:

  ***<type> <name>* = *<value>* ;**

  - Example: `double myGPA = 3.95;`
  - Example: `int x = (11 % 3) + 12;`



same effect as:

```
double myGPA;
myGPA = 3.95;

int x;
x = (11 % 3) + 12;
```

Assignment Statement:

- **Assignment Statement**: A Java statement that stores a value into a variable's memory location.
    - Variables must be declared before they can be assigned a value.

- Assignment statement **syntax**:

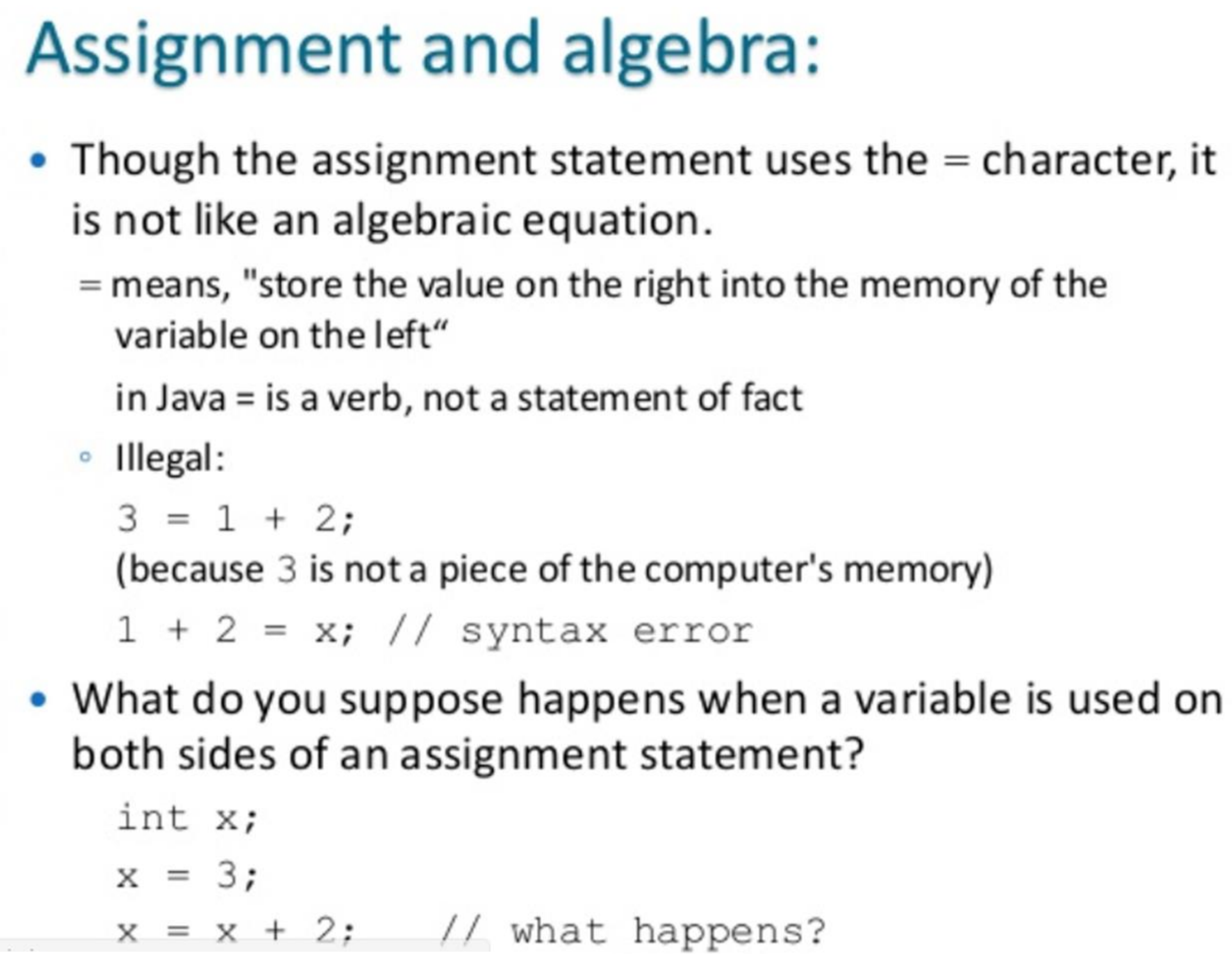
    ***<name>* = *<value>* ;**

    - Example: `x = 3;`
    - Example: `myGPA = 3.95;`

    - Another (crude) diagram of part of the computer's memory:

```
       +---+             +------+
  x    | 3 |     myGPA   | 3.95 |
       +---+             +------+
```

- Technically, = is an operator like + or *, called the ***assignment operator***, with very low precedence (it is carried out last).



Assignment and algebra:

- Though the assignment statement uses the = character, it is not like an algebraic equation.

  = means, "store the value on the right into the memory of the variable on the left“

  in Java = is a verb, not a statement of fact

  - Illegal:

  ```
  3 = 1 + 2;
  ```

  (because 3 is not a piece of the computer's memory)

  ```
  1 + 2 = x; // syntax error
  ```

- What do you suppose happens when a variable is used on both sides of an assignment statement?

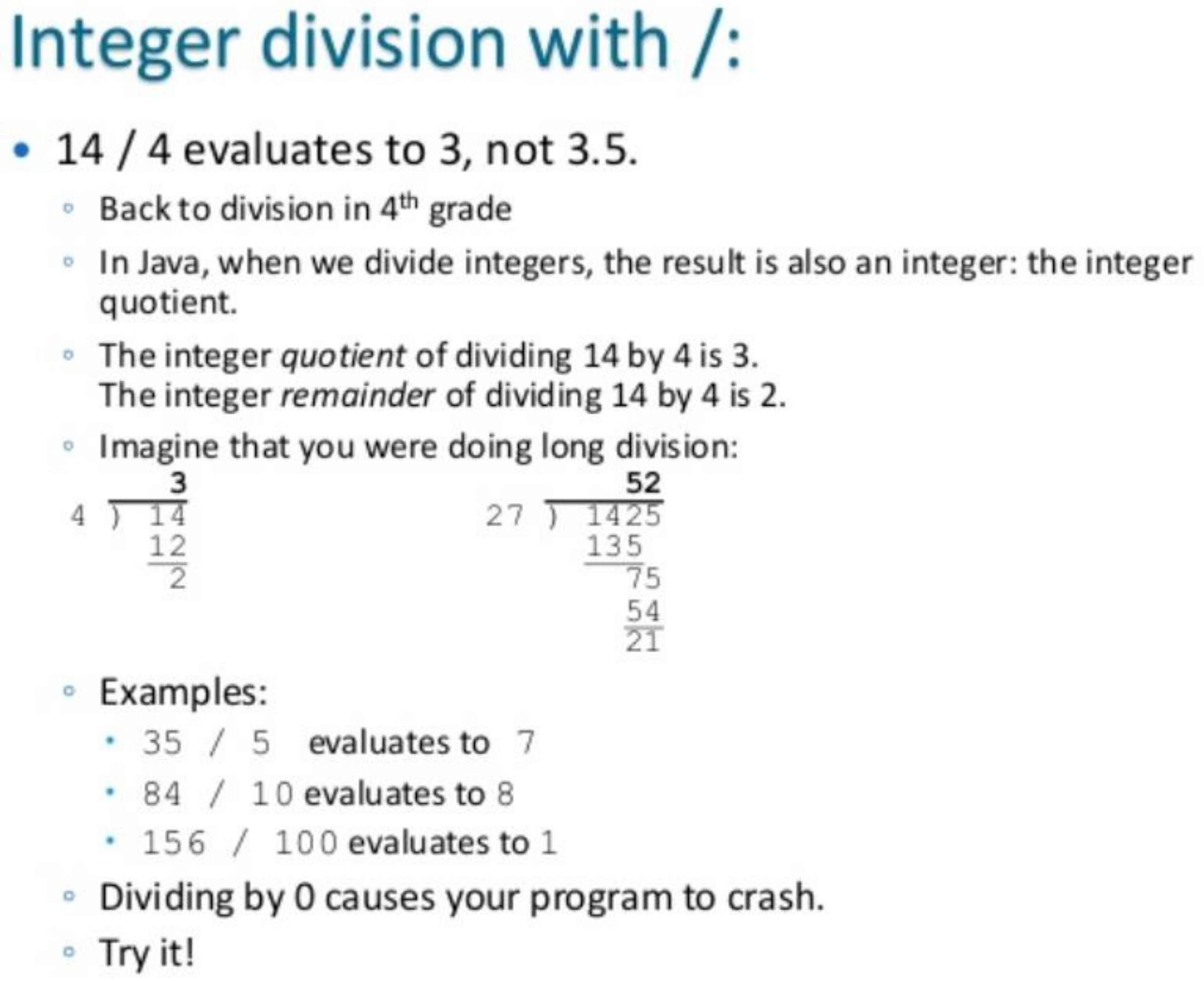
  ```
  int x;
  x = 3;
  x = x + 2;   // what happens?
  ```

Expressions:

- **Expression**: A data value, or a set of operations that compute a data value.
  - Example: `1 + 4 * 3`
  - The simplest expression is a *literal value*.
  - A more complex expression can have *operators* and/or parentheses.
    - The values that an operator applies to are called *operands*.

- 5 common arithmetic operators we will use:

`+`	(addition)
`-`	(subtraction or negation)
`*`	(multiplication)
`/`	(division)
`%`	(modulus, a.k.a. remainder)



Integer division with /:

- 14 / 4 evaluates to 3, not 3.5.
  - Back to division in 4th grade
  - In Java, when we divide integers, the result is also an integer: the integer quotient.
  - The integer *quotient* of dividing 14 by 4 is 3.
    The integer *remainder* of dividing 14 by 4 is 2.
  - Imagine that you were doing long division:

```
       3                     52
4  ) 14               27 ) 1425
     12                    135
      2                      75
                             54
                             21
```

  - Examples:
    - 35 / 5 evaluates to 7
    - 84 / 10 evaluates to 8
    - 156 / 100 evaluates to 1
  - Dividing by 0 causes your program to crash.
  - Try it!

Integer remainder with %:

- The % operator computes the remainder from a division of integers.
  - Example: `14 % 4` is `2`
  - Example: `218 % 5` is `3`

```
        3                      43
     ____                   _____
4 )  14                5 )  218
     12                     20
     --                     --
      2                      18
                             15
                             --
                              3
```

- What do the following expressions evaluate to?
  - `45 % 6`
  - `2 % 2`
  - `8 % 20`
  - `11 % 0`