



Java 2: Variables and Data Types

Storing values, doing math, and joining text

Variables and Data Types

A **variable** is a named box that stores a value. Its **data type** says what kind of value it holds: `int` (whole numbers like `7` or `-2`), `double` (decimal numbers like `3.5`), `String` (text in double quotes like `"Hi"`), or `boolean` (only `true` or `false`). An `int` cannot hold a decimal.

To **declare** a variable, write its type and name. To **assign** it, use `=`, which means "store the right side in the variable on the left": `int score = 10;`. To **reassign**, store a new value; the old one is replaced. `score = score + 5;` makes `score` 15.

Parentheses go first, then `*` `/` `%`, then `+` `-`. **Integer division**: if both numbers are `int`, `/` drops the remainder (`7 / 2` is `3`). If either is a `double`, the decimal stays (`7.0 / 2` is `3.5`). **Modulus** `%` gives the remainder (`7 % 2` is `1`).

With a `String`, `+` means **concatenation** (joining). Java does `+` **left to right**; once text is involved, `+` joins instead of adding: `"Score: " + 2 + 3` prints `Score: 23`, but `"Score: " + (2 + 3)` prints `Score: 5`.

1 Do the Math

Write the value Java computes for each expression. Watch for integer division!

- a. `7 / 2 =` _____ d. `20 / 6 =` _____ g. `2 + 3 * 4 =` _____
b. `7 % 2 =` _____ e. `20 % 6 =` _____ h. `(2 + 3) * 4 =` _____
c. `7.0 / 2 =` _____ f. `3 / 4 =` _____ i. `9 % 3 =` _____

2 Trace the Variables

Fill in the value of each variable **after** each line runs. If a variable does not exist yet, write a dash (-).

```
1 int coins = 5;
2 int gems = 2;
3 coins = coins + 10;
4 gems = gems * 3;
5 coins = coins - gems;
6 gems = coins % 4;
7 coins = coins / 2;
```

Tip: read `coins = coins + 10;` as "take the value in `coins`, add 10, and store the answer back in `coins`." Only the variable on the left of `=` changes.

After line	coins	gems
1		
2		
3		
4		
5		
6		
7		



Java 2: Variables and Data Types (page 2)

3 Pick the Data Type

Write the best data type under each value: `int`, `double`, `String` or `boolean`.

42	3.14	"hello"	true	"42"	-7	false	0.5

4 Predict the Output

```
1 String item = "pencil";
2 int price = 2;
3 int count = 3;
4 boolean onSale = false;
5 System.out.println(item + "s");
6 System.out.println("Total: " + price * count);
7 System.out.println("Oops: " + price + count);
8 System.out.println(price + count + " items");
9 System.out.println("On sale? " + onSale);
```

a. Write the exact output, one row per line.

b. The programmer wanted line 7 to print `Oops: 5`. Rewrite line 7 so it does.

5 Find the Bug

Each line has one bug. Lines a and b will not compile. Line c runs, but `half` becomes `3.0`, not `3.5`.

	Buggy line	What is wrong?	Fixed line
a.	<code>int age = 12.5;</code>		
b.	<code>String name = Sam;</code>		
c.	<code>double half = 7 / 2;</code>		

6 Write Your Own Code

23 students sit at tables of 4. Store `23` and `4` in two `int` variables, then print `Full tables: 5` and `Left over: 3`. Use `/` and `%` on your variables (do not type the 5 or the 3 yourself).