

NAME

DATE

PERIOD

UNIT 2 · TOPIC 2.8 · WORKSHEET

Start, Stop, Step

Count the passes, rebuild the loops, out-think the compiler, and fix a countdown that never ends.

About 35 minutes

10 questions

Cumulative through 2.8

TOPIC

2.8



DIRECTIONS

Work each puzzle in pencil. When a question asks for output, write exactly what prints, including spaces between values. Some puzzles include pieces you won't need.

PART A · 2 QUESTIONS

A Warm up your brain

1 PENCIL CHECK

Each row is a for loop header with a body that does not change `i`. Write how many times the body runs and the last value of `i` the body sees. If the body never runs, write *none*.

Header	Iterations	Last <code>i</code> the body sees
<code>for (int i = 0; i < 6; i++)</code>		
<code>for (int i = 1; i <= 6; i++)</code>		
<code>for (int i = 0; i <= 6; i++)</code>		
<code>for (int i = 6; i > 0; i--)</code>		
<code>for (int i = 1; i <= 10; i += 3)</code>		
<code>for (int i = 10; i < 5; i++)</code>		
<code>for (int i = 2; i < 40; i *= 2)</code>		

**THINK FIRST**

In a for loop header, the update is written *before* the body. Does Java run it before the body, or after? Guess in the margin; question 2 settles it.

2 YOU'RE THE JVM

You are the Java Virtual Machine. Run this code in your head. What prints?

```
1  int total = 0;
2  for (int n = 2; n <= 8; n += 3) {
3      total += n;
4      System.out.print(n + " ");
5  }
6  System.out.println(total);
```

A 2 5 8 15**B** 2 5 7**C** 5 8 11 24**D** 2 5 8 11 26**WATCH OUT**

Let the header do the counting. This loop also changes its loop control variable inside the body, so *i* goes up by 2 each pass and it prints only 0 2 4:

```
for (int i = 0; i < 6; i++) {
    System.out.print(i + " ");
    i++;
}
```

If a loop needs to step by 2, say so in the update (*i* += 2), where every reader will look.

PART B · 5 PUZZLES

B Puzzles

3 CODE TILES

Hopper's game scores 10 points in round 1, 20 in round 2, and 30 in round 3, and keeps a running total. The loop fell off the fridge. Put the tiles back in order, below the line already written, so the program prints the target output. One tile is a decoy.

```
int score = 0;
```

```
System.out.println("Final score: " + score);
```

```
for (int round = 1; round <= 3; round++) {
```

```
score += round * 10;
```

```
}
```

```
System.out.println("Round " + round + ": " + score);
```

```
for (int round = 1; round < 3; round++) {
```

Target output

Round 1: 10

Round 2: 30

Round 3: 60

Final score: 60

YOUR PROGRAM

4 SNIPPET POOL

Fill the holes from the pool so the program prints 12 8 4 go!. Not every snippet is needed.

```
for (int t = (1); (2); (3)) {
    System.out.print(t + " ");
}
System.out.println("go!");
```

THE POOL

Each snippet can be used only once, and not every snippet is needed.

12	0	t > 0	t >= 0
t < 12	t -= 4	t --	t += 4

5 YOU'RE THE COMPILER

For each snippet: does it compile? If it does, write exactly what prints. If not, say why.

Snippet	Compiles?	Output, or why not
1 for (int k = 0, k < 3, k++) { System.out.println(k); }		
2 for (int k = 10; k > 0; k /= 2) { System.out.print(k + " "); }		
3 for (int k = 0; k < 3; k++) { System.out.print(k); } System.out.println(k);		
4 for (int k = 1; k <= 3; k++); { System.out.println("hop"); }		

6 MATCH-UP

Someone tried to rewrite `for (int x = 1; x <= 4; x++)` as a while loop, four different ways. Each loop has the same print statement. Match each loop to its output from the list below. Then name the loop that does exactly what the for loop does.

Loop	Code	Output
1	<pre>int x = 1; while (x <= 4) { x++; System.out.print(x * 2 + " "); }</pre>	
2	<pre>int x = 1; while (x < 4) { System.out.print(x * 2 + " "); x++; }</pre>	
3	<pre>int x = 1; while (x <= 4) { System.out.print(x * 2 + " "); x++; }</pre>	
4	<pre>int x = 0; while (x <= 4) { System.out.print(x * 2 + " "); x++; }</pre>	

- **P** 2 4 6
- **Q** 0 2 4 6 8
- **R** 4 6 8 10
- **S** 2 4 6 8

The loop that matches the for loop: _____



FAIR QUESTION

If a while loop can do everything a for loop can, why have both?

Readability. When a counter starts, stops, and steps in a way you know ahead of time, a for loop puts the whole plan on one line. With no counter ("keep asking until the user types quit"), a while loop says what it means.

7 FACE-OFF

Tonight: the for loop and the while loop argue about who's tidier. Fill in the blanks.

FOR LOOP	I keep my initialization, my Boolean expression, and my update together in my _____, separated by _____.
WHILE LOOP	I only keep the test. The setup goes _____ me, and the update goes _____ in my body.
FOR LOOP	My initialization runs just _____. My update runs _____ the whole body, every pass.
WHILE LOOP	Here's one thing we share: if the test is _____ at the very start, neither of us runs the body. That's _____ iterations.
FOR LOOP	Fine. But there's nothing you can do that I can't. Any while loop can be _____ as a for loop.
WHILE LOOP	And the reverse is true too. We're the same loop in different outfits.

PART C · 3 QUESTIONS

C Catch the mistake

8 ONE-TOKEN FIX

This countdown should print 5 4 3 2 1 done, but it prints 5 6 7 8 9 10 ... and never stops. Change exactly one token to fix it.

```

1  for (int c = 5; c > 0; c++) {
2      System.out.print(c + " ");
3  }
4  System.out.println("done");

```

WHY IT NEVER STOPS, AND YOUR FIX (LINE, OLD TOKEN, NEW TOKEN)

9 WHO AM I?

The pieces of a for loop are playing a party game. Each gives a clue about itself. Name the piece.

Clue	I am...
I run exactly once, before anything else in the loop.	
I run after every full pass of the body, and never in the middle of one.	
I'm checked one more time than the body runs. My last answer is false.	
I'm set up by the initialization, checked by the test, and changed by the update.	
I'm one run of the loop body.	
There are exactly two of me in every for loop header.	

10 WRAP-UP



REFLECTION

In one or two sentences: a loop's test runs one more time than its body. Why?

YOUR ANSWER