

NAME _____

DATE _____

PERIOD _____

UNIT 3 · TOPIC 3.2 · WORKSHEET

Beyond the Screen

Sort the impacts, catch the crash, test the edges, and decide whose code you can use.

About 35 minutes

10 questions

Cumulative through 3.2

TOPIC

3.2

**DIRECTIONS**

Work each puzzle in pencil. Every app, company, and person here is made up. For Java code, write exactly what prints. Read past `public static` in method headers; it is explained in a later topic.

**THINK FIRST**

Think of an app you use every day. Who does it help? Is there anyone it makes things worse for? Jot one of each in the margin.

PART A · 2 QUESTIONS

A Programs affect people

1 PENCIL CHECK

A city launches **ScootGo**, an app for renting electric scooters by the minute. For each effect, write the area it mostly affects (*society, economy, or culture*) and whether it *helps* or *harms*.

Effect of ScootGo	Area	Helps or harms?
Students without a car can get to their after-school jobs.		
Taxi drivers lose short trips and earn less money.		
Scooters left on sidewalks block people who use wheelchairs.		
The city collects a fee on every ride and fixes more roads with it.		
Weekend group scooter tours become a new local tradition.		

2 THINK IT THROUGH

PetPal lets dog owners post a photo of their dog at the park, tagged with the park's location, so that other owners can set up play dates. Nobody on the team meant for anything bad to happen.

WHAT IS THE INTENDED USE?

ONE UNINTENDED HARMFUL EFFECT, AND WHO IT HARMS



WATCH OUT

"My program helps people, so it can't be harmful" stops after the first question. Always ask the second one: who might be worse off because this exists, even if no one meant it?

PART B · 5 PUZZLES

B Reliable and tested

3 YOU'RE THE JVM

StepUp promises a weekly report "for every user." A brand-new user has walked 0 steps on 0 days. Run this code in your head. What happens?

```
int totalSteps = 0;
int days = 0;
System.out.println("Weekly report");
System.out.println("Steps per day: " + totalSteps / days);
```

- A It prints Weekly report, then Steps per day: 0.
- B It prints Weekly report, then stops with an `ArithmeticException`.
- C Nothing prints; it stops with an `ArithmeticException` before line 1 runs.
- D It doesn't compile, because dividing by zero is a syntax error.

4 PENCIL CHECK

CakeBox gives a discount on orders of **10 or more** cupcakes. For each test value, write *typical*, *boundary*, or *special*.

Cupcakes ordered	Kind of test case
3	
10	
0	
25	
9	

5 TWO TRUTHS AND A LIE

StepUp's description says: *Counts steps while you walk or run, with the phone in your pocket.* Two claims are true and one is a lie. Circle the lie, then rewrite it so it's true.

- 1 If StepUp miscounts steps while someone walks with the phone in a pocket, it is not reliable.
- 2 If StepUp can't count steps while the phone sits on a desk, it has not broken its promise.
- 3 StepUp counted correctly the one time its programmer tried it, so it is reliable.

THE LIE, REWRITTEN

6 YOU'RE THE TESTER

FlixRent's rule: rentals of **3 days or fewer** cost \$2; longer rentals cost \$5. Here is the method. For each test, write the expected cost (from the rule), the actual cost (from the code), and whether the test passes.

```
public static int rentalCost(int days) {
    int cost;
    if (days < 3) {
        cost = 2;
    } else {
        cost = 5;
    }
    return cost;
}
```

Call	Expected	Actual	Pass?
rentalCost(1)			
rentalCost(3)			
rentalCost(4)			
rentalCost(7)			



FAIR QUESTION

Why test both 64 and 65 for a rule that changes at 65?

Because a rule can be off by one in either direction. Testing the last value of one group *and* the first value of the next shows the rule changes in exactly the right place, not one value early or one value late.

7

ONE-TOKEN FIX

ReviewIt shows a product's average star rating. With 0 stars from 0 reviews, it should print Ratings and then No reviews yet, but it crashes after Ratings. Change exactly **one token** so it works for 0 reviews and still prints Average: 3 for 9 stars from 3 reviews.

```
1  int totalStars = 0;
2  int reviews = 0;
3  System.out.println("Ratings");
4  if (reviews >= 0) {
5      System.out.println("Average: " + totalStars / reviews);
6  } else {
7      System.out.println("No reviews yet");
8  }
```

YOUR FIX, AND WHY IT CRASHED

PART C · 3 QUESTIONS

C Whose code is it?

8 FACE-OFF

Tonight: Open-Source Code and Sold Code argue about who's easier to work with. Fill in the blanks.

OPEN-SOURCE CODE

My owner published me so anyone can use me for _____. No need to ask first.

SOLD CODE

Lucky you. Before anyone puts me in a program, they need my owner's _____, and they usually have to _____.

OPEN-SOURCE CODE

I still have rules, though. My _____ says what users may do, like keeping my author's name in the code.

SOLD CODE

And either way, we both belong to someone. We're _____, even when anyone can see us online.

9 CAN I USE IT?

For each situation, write **OK** (fine as is), **license** (fine if you follow the license), or **ask** (get the owner's permission, and maybe pay, first).

Situation	OK, license, or ask?
You call <code>Math.sqrt</code> and <code>String</code> methods from Java's own library.	
You find a game's scoring code shown on a company's website, with no statement that you may use it.	
You use a chart library published as open source, which asks users to keep the authors' names.	
A classmate shares a file copied out of a code package that a company sells.	
You wrote the code yourself, from scratch.	

10 WHO AM I?

Some ideas from this topic are playing a party game. Each gives a clue about itself. Name the idea.

Clue	I am...
I'm a program doing its tasks as expected, under its stated conditions, without failing.	
I'm a test value sitting exactly where a rule changes.	
I'm a result the designers never planned, and sometimes I hurt people.	
I'm code whose owner published it free for anyone to use.	
I'm the short statement that tells you what you may do with someone's code.	