ANSWER KEY

Answer Key: Day 2

Every page for this day is keyed, and nothing is left as "answers will vary." This file covers the bell ringer, the guided notes and the exit ticket. In this folder, the key for the Warm-Up Arcade sheet is in Activity_and_Lab_Keys.pdf.

Bell Ringer: The Snack Stand

```
1  bars = 11
2  bags = bars // 4
3  bars %= 4
4  print(f"{bags} bags, {bars} left")
```

Look for

- Question 1: `2 bags, 3 left`
- Question 2: line 3 changes `bars` to 3, so line 2 has to count the bags while `bars` still holds 11.

Model response

The program prints `2 bags, 3 left`. Line 2 must come first, because line 3 changes `bars` from 11 to 3, and 3 bars make 0 full bags.

Guided Notes: Day 2

Blanks are shown filled in, in order.

Whole groups and leftovers

```
1  bars = 11
2  bags = bars // 4
3  bars %= 4
4  print(f"{bags} bags, {bars} left")
```

Output

```
2 bags, 3 left
```

`bars // 4` counts the full bags of 4. It divides and rounds down to a whole number, so 11 bars make 2 bags.

The **remainder** is what's left over after a division. `bars %= 4` keeps it, so `bars` goes from 11 to 3.

Indents and auto-indent

```
1  seconds = 40
2  while seconds > 1:
3  seconds //= 2
4  print(seconds)
```

Error

Line 3: Python says: `IndentationError: expected an indented block after 'while' statement on line 2`

An **indent** is the spaces at the start of a line. The body of a loop or an `if` starts 4 spaces in.

A missing indent stops the program with an `IndentationError`.

With **auto-indent**, the editor adds the 4 spaces when you press Enter after a colon.

Fixing and testing

```
1  total = 12
2  # print("test:", total)
3  print(f"Total: {total}")
```

Output

```
Total: 12
```

The **Error panel** names the line where Python stopped. Fix that line first.

A **test line** prints or sets a value while you work.

To **comment out** a test line, type # in front of it. Delete the # to turn it back on.

Exit Ticket

The exit ticket is worth 3 points: 1 for each item.

1. Correct: (A) Type a # at the start of the line.

A # turns the line into a comment, so Python skips it. Deleting the # turns it back on.

2. Correct: (B)

```
1  bars = 17
2  bars %= 5
3  print(bars)
```

Output

```
2
```

17 bars make 3 bags of 5, which hold 15 bars. %= keeps the remainder, which is 2.

Item 3: written answer

Look for

- Any tool from the lesson, like the line number in the Error panel, auto-indent, the suggestion list or a # to comment out a line, with how the student used it.

Model response

The line number in the Error panel helped me most. It sent me straight to the `elif` line, and I added the missing colon.