1. The purpose of a loop is to repeat a set of instructions until a condition changes or is no longer true.

---

2. A while loop checks the condition before the loop runs, so it might not run at all.

A do-while loop runs the code first and then checks the condition, so it runs at least once.

---

3. The input validation loop was written in the part of the chapter where we practiced checking user input and repeating the loop until the user enters valid data. (The review on input validation.)

4.

a) An infinite loop is a loop that never stops running.

b) Two things that can cause infinite loops are:

- Forgetting to change the variable that controls the loop
- Using a condition that never becomes false

c) Overflow means that a number becomes too big to fit in the computer's memory space, so it "wraps around" to a very small or negative number.

---

5. The loop runs 60 times because it keeps adding 2 to x until x becomes 120.

---

6. If x starts at 120 or less, it will always go down by 3 and always stay less than 120, so the loop would never end. (Example: x = 0)

7.

- A counter counts how many times something happens. Examples: counting how many students passed a test, counting button clicks.

- An accumulator adds up numbers to get a total. Examples: adding up prices in a shopping cart, adding test scores to get a final grade.

Counters usually increase by 1. Accumulators increase by different values depending on what gets added.

8.

```
int sum = 0;

for (int i = 3; i <= 10; i++) {

    sum += i;

}
```

This adds all the numbers from 3 to 10.

---

9. Two things to think about when choosing a loop:

- Do I know exactly how many times the loop should run? (use a for loop)

- Does the loop need to run at least once? (use a do-while loop)

11.

```
String x = "my string.";
```

a) x.length() → 10

(There are 10 characters including the space and period.)

b) x.substring(0, 3) → “my “

(It takes characters from index 0 to 2.)