Solution: Print a parallelogram pattern

Here is a solution to the coding challenge given in the previous lesson.

WE'LL COVER THE FOLLOWING

- Solution
- Solution explanation

Solution

Here is a code that will print a parallelogram pattern of # characters using two `for` loops:

```
import std.stdio;

void printParallelogram() {
    int rows = 5;
    int cols = 8;

    for (int line = 0; line < rows; ++line) {
        for (int i = 0; i < line; ++i) {
            write(' ');
        }

        writeln("########");
    }

}
```

Code to print parallelogram pattern

Solution explanation

- **Line 7:**

```
for (int line = 0; line < rows; ++line) {
```

  As explained in the for loop lesson, we do the preparation, condition check and the iteration on the same line. Condition check makes sure that the loop exits when the `line` variable reaches 5.

- **Line 8 and 9:**

  ```
  for (int i = 0; i < line; ++i) {
      write(' ');
  ```

  These two lines maintain the spaces to be printed on the screen to form the parallelogram pattern. As you can see, with the `line` number increment, the amount of space increases before the first *.

- **Line 12:**

  ```
  writeln("########");
  ```

  This line simply prints the * pattern once we have accounted for the spaces in that line to form the parallelogram pattern.

In the next lesson, you will learn another variant of the `while` loop, i.e. the `do-while` loop.