

## Do...While Loop

Similar to the `while` loop, a `do...while` loop will continue running until a stopping condition is met. One key difference is that no matter what, a `do...while` loop will *always run once*.

```
do
{
    statement;
} while (condition);
```

Instead of checking the condition before the code block executes, the program in the block runs once and *then* checks the conditional statement. It will either stop or continue to execute until the condition is no longer true. `do...while` loops are good for when a program should execute at least once and then depending on the circumstances, continue to execute or stop.

In your video game, you want to show the start screen for at least one frame. If the user does not hit next, you want the start screen to continue to appear until they do so.

```
bool startGame = false;

do
{
    ShowStartScreen();
} while (!startGame);
```

In this example, we initialize a Boolean value `startGame` to the state `false`. The program runs once, showing the start screen for one frame, then checks to see if the player has started the game. If the player has yet to start the game, it will call `ShowStartScreen()` again and will then continue to show the start screen until the player presses start.

### ☒ Instructions

1.

You've decided to try making another productivity tool. This time you build a [pomodoro timer](#). When the timer finishes, an alarm will ring once. If you don't click the button to turn it off, the alarm will repeat until it is turned off.

Write a program that plays the alarm once, and then if the button isn't clicked (`!buttonClick`), it will repeat the alarm. To simulate an alarm, you can print something like "BLARRRRRRRRR" to the console.

Hint



A `do...while` loop will execute a statement once, then check a condition. If the condition is true, it will repeat the statement and continue doing so until the condition is false.

```
// game hasn't started
bool startGame = false;

do
{
    // do this at least once
    // as long as the player hasn't started the game
    ShowStartScreen();
} while (!startGame);
```

After the loop finishes, the program should print a message telling you to take a break, like:  
"Time for a five minute break."