Goal: Append to an empty array based on coordinate properties.

Adding each item to an array individually is *really* repetitive. What if you could create a set of rules for the coordinates to include in your array?

First, start with `allCoordinates`, an array of all the coordinates in the puzzle world.

Next, you'll need an empty array to append your coordinates to. And because you're declaring an array with no stored values, you'll need to specify the type of items it should hold.

> Creating an empty array
>
> Use `:` after your variable name to declare its type, then assign it an empty array.
>
> `var newLocations: [Coordinate] = []`

Finally, iterate over `allCoordinates` and check the `column` and `row` properties of each coordinate. If the `column` property of a coordinate is greater than `5`, **or** its `row` property is less than `4`, append it to your empty array. Then place six blocks on each coordinate in the array.

```
let allCoordinates = world.allPossibleCoordinates
var blockSet: [Coordinate] = []
```

```
for coordinate in allCoordinates {
    // Check for coordinates with a column > 5 OR a row < 4.
    if coordinate.column > 5 || coordinate.row < 4 {
        // Append coordinate to blockSet.
        blockSet.append(coordinate)
}
}

// For each coordinate in blockSet, place six blocks.
for coordinate in blockSet {
    for i in 1...6 {
        world.place(Block(), at: coordinate)
    }
}
```