Goal: Place Blu, Hopper, and the expert in order by height.

The `Array` type comes with a set of methods that let you add or remove items from your array.

> Array methods
>
> **`remove(at: Int)`**. Removes an item at an index.
> **`append(newElement: Element)`**. Adds an item to the end of the array.
> **`insert(newElement: Element, at: Int)`**. Inserts an item at a specific index.

Use dot notation to call a method on an array.

> Example
>
> `var favoriteFoods = [🌮, 🍓, 🥓, 🍳, 🧀]`
>
> `favoriteFoods.remove(at: 2)`
> Calling `remove(at: 2)` removes 🥓 from the array.
> `// [🌮, 🍓, 🍳, 🧀]`
>
> `favoriteFoods.insert(🍝, at: 1)`
> Calling `insert(🍝, at: 1)` adds 🍝 at index `1`.
> `// [🌮, 🍝, 🍓, 🍳, 🧀]`

1 In the `characters` array below, remove the portal and gem.

2 Insert an instance of type `Expert` so that the characters are arranged from *shortest* in the front (row `0`) to *tallest* in the back.

```
characters = [
    Character(name: .blu),
    Portal(color: 🟥),
    Character(name: .hopper),
    Gem()
]
```

```
// Remove the portal.
characters.remove(at: 1)
// Remove the gem.
characters.remove(at: 2)
// Insert the expert.
```

```
characters.insert(Expert(), at: 1)
var rowPlacement = 0
for character in characters {
    world.place(character, at: Coordinate(column: 1, row:
     rowPlacement))
    rowPlacement += 1
}
```