The GardenLayout object represents a customizable garden bed that users can design. It stores data such as the layout's name, height, and width, as well as the available plant types and their matching images (emojis). These properties allow the site to display a visual table mimicking a garden layout where users can preview different garden sizes and arrangements. The object's methods are displayLayout() which builds the garden's html and displays the garden grid based on the chosen dimensions, getPlantEmoji() retrieves the correct emoji for a specific plant, and fillGardenWithPlant() which automatically fills the layout with plant icons by updating the html. Upon a user inputting a name, dimensions and clicking generate garden, a new object is created and displayed to the user in response to their request.

Examples:

Object Type: GardenLayout	
Data layoutName height width	**Value** "My First Garden" 1 12
Task displayLayout() getPlantEmoji() fillGardenWithPlant()	**Purpose** Takes the inputted height and width and builds an html table Based on the index, gets the specified plant emoji For each cell in the table, adds the appropriate plant emoji

Object Type: GardenLayout	
Data layoutName height width	**Value** "Vegetable Patch" 5 5
Task displayLayout() getPlantEmoji() fillGardenWithPlant()	**Purpose** Takes the inputted height and width and builds an html table Based on the index, gets the specified plant emoji For each cell in the table, adds the appropriate plant emoji