Design rationale for Requirement 3: Magical Fountains

Fountains, PowerFountain and HealthFountain classes

An abstract class Fountains was created and extended from Ground. This is because fountains has a lot of similar functionalities to Ground. Fountains was made into an abstract class so that new types of fountains can be created in the future, and it allows scalability. PowerFountain and HealthFountain both extend Fountains as they all share similar functionalities.

Bottle, HealthWater and PowerWater classes

A class Bottle that extends Item was created. A static variable/stack bottle was created; static is used because the variable/stack bottle can be accessed without instantiating the class Bottle, which prevents the use of 'instanceof' and casting. HealthWater and PowerWater both extend Item as they share similar methods and features.

FillBottleAction

FillBottleAction is just like any similar implemented action, and it extends Action. It uses the player's location and ground to check whether the player is on a PowerFountain or HealthFountain. It creates an appropriate instance of either a PowerWater or a HealthWater and adds it to the player's stack bottle.