Garden

- clusters: HashMap<Direction, SprinklerCluster>
- date: SystemDate
- thermometer: Thermometer
- schedule: Schedule
- garden: Garden

- initializeSprinklersInClusters(int): void
+ getCluster(Direction): SprinklerCluster
+ getTotalClusterUsage(Direction): double
+ getTemperature(): double
+ incrementTemperature(): void
+ decrementTemperature(): void
+ getMinThreshold(): double
+ getMaxThreshold(): double
+ setMinThreshold(double): void
+ setMaxThreshold(double): void
+ getDate(): LocalDate
+ addDays(int): void
+ minusDays(int): void
+ getTime(): LocalTime
+ getSchedule(): Schedule
+ isActive(): boolean
+ activateSprinklers(): void
+ deactivateSprinklers(): void

DayUsage

- day: LocalDate
- usage: double

+ addUsage(double): void
+ getUsage(): double
+ getDay(): LocalDate

Schedule

- schedule: Map<Days, List<Interval>>

+ getSchedule(): Map<Days, List<Interval>>
+ getScheduleForDay(Days): List<Interval>
+ setSchedule(Map<days, List<Interval>>): void
+ mergeIntervals(List<Interval>): List<Interval>
+ removeAll(): void
+ removeDay(Days): void
+ toString(): String

<<Interface>>
ISprinkler

+ isActive(): boolean
+ deactive(): void
+ activate(): void
+ isFunctional(): boolean
+ setFunctional(boolean): void
+ getId(): String
+ getTotalUsage(): double
+ getUsageHistory(int): LinkedList<DayUsage>
+ getUsage(int): double
+ getWaterFlow(): double

Sprinkler

- WATER_FLOW: double
- id: String
- active: boolean
- functional: boolean

+ isActive(): boolean
+ deactive(): void
+ activate(): void
+ isFunctional(): boolean
+ setFunctional(boolean): void
+ getId(): String
+ getTotalUsage(): double
+ getUsageHistory(int): LinkedList<DayUsage>
+ getUsage(int): double
+ getWaterFlow(): double

SprinklerCluster

- sprinklerMap: HashMap<String, Sprinkler>
- clusterId: Direction
- nextId: HashMap<Direction, Integer>

+ isActive(): boolean
+ deactive(): void
+ activate(): void
+ isFunctional(): boolean
+ setFunctional(boolean): void
+ getId(): String
+ getTotalUsage(): double
+ getUsageHistory(int): LinkedList<DayUsage>
+ getUsage(int): double
+ getWaterFlow(): double
+ addSprinkler(): void
+ getCount(): int
+ getIterator(): Iterator<Sprinkler>
+ getSprinkler(String): Sprinkler

UpdateTimer

- UPDATE_INTERVAL: int
- activeSprinklers: HashMap<Sprinkler, Date>
- timer: Timer
- INCREMENTOR: int

- initialize(): void
- start(): void
- stop(): void
+ addSprinkler(Sprinkler): void
+ removeSprinkler(Sprinkler): void

SystemDate

- date: LocalDateTime

+ getDate(): LocalDate
+ addDays(int): void
+ minusDays(int): void
+ getTime(): LocalTime
- start(): void

Usage

- TOTAL_USAGES_FILE_PATH: String
- SPRINKLER_USAGES_FILE_PATH: String
- totalUsages: HashMap<String, Double>
- sprinklerUsages: HashMap<String, LinkedList<DayUsage>>

- update(Sprinkler, double): void
- getTotalUsage(String): double
- getSprinklerUsage(String, int): LinkedList<DayUsage>
+ readTotalUsages(): HashMap<String, Double>
+ readSprinklerUsages(): HashMap<String, LinkedList<DayUsage>>
+ writeTotalUsage(HashMap<String, Double>): void
- writeSprinklerUsage(HashMap<String, LinkedList<DayUsage>>)
+ resetUsages(): void

Thermometer

- degreesFarenheit: double
- minThreshold: double
- maxThreshold: double
- INCREMENTOR: int
- MIN_TEMP: int
- MAX_TEMP: int

+ incrementTemperature(): boolean
+ decrementTemperature(): boolean
+ shouldTurnOn(): boolean
+ shouldTurnOff(): boolean
+ toString(): String
+ setTemperature(double): boolean
+ setMinThreshold(double): boolean
+ setMaxThreshold(double): boolean
+ getTemperature(): double
+ getMinThreshold(): double
+ getMaxThreshold(): double