UML coffee machine

Grind

- grindLevel : int

+ setGrindLevel(grindDegree : int) : void
+ getGrindLevel() : int
+grinding(grindLevel: int): void

CoffeeMachine

-water: WaterContainer
-beans: BeansContainer
-grind: Grind
-wastedTray: WastedTray
logger: Logger

+getWater(): WaterContainer
+setWater(water: WaterContainer)
+getBeans(): BeannsContainer
+setBeans(beans: BeansContainer): void
+getGrind(): Grind
+setGrind(grind: Grind): void
+getWastedTray(): WastedTray
+setWastedGray(wastedTray: WastdTray): void
+getLogger(): Logger
+setLogger(logger: Logger): void
+start(): void
+brewer(coffeeChoice: int): void

Has A

<<interface>>
Logger

+log(message String): void

WastedTray

-level: int
-capacity: int

+WastedTray()
+getLevel(): int
+setLevel(level: int): void
+getCapacity(): int
+setCapacity(capacity: int): void
+clean(): void

Has A

Has A

Has A

BeansContainer

- arabicaPercentage : double
- robustaPercentage : double

+ BeansContainer()
+BeansContainer(arabicaPercentage: double, robustaPercentage: double)
+ getArabicaPercentage() : double
+ getRobustaPercentage() : double
+ setArabicaPercentage(arabica: double) : void
+ setRobustaPercentage(robusta : double) : void
+ take(amount : int) : boolean
+ getInfo() : String
+fill(amount: int): void

Is A

<> Container

-private capcity: int
-private level: int

+Container()
+ getCapacity(): double
+ setCapacity(capacity: int): void
+getLevel(): int
+setLevel(level: int):void
+ take(amount: int): boolean//abstract
+getInfo(): String//abstract
+fill(amount: int): void//abstract

Is A

WaterContainer

+WaterContainer()
+take(amount: int): boolean
+getInfo(): String
+fill(amount: int): void