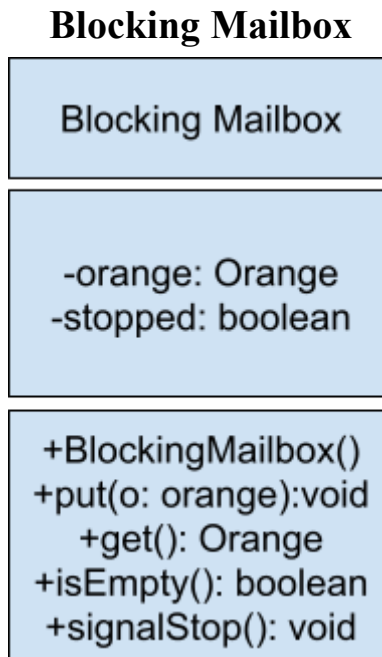
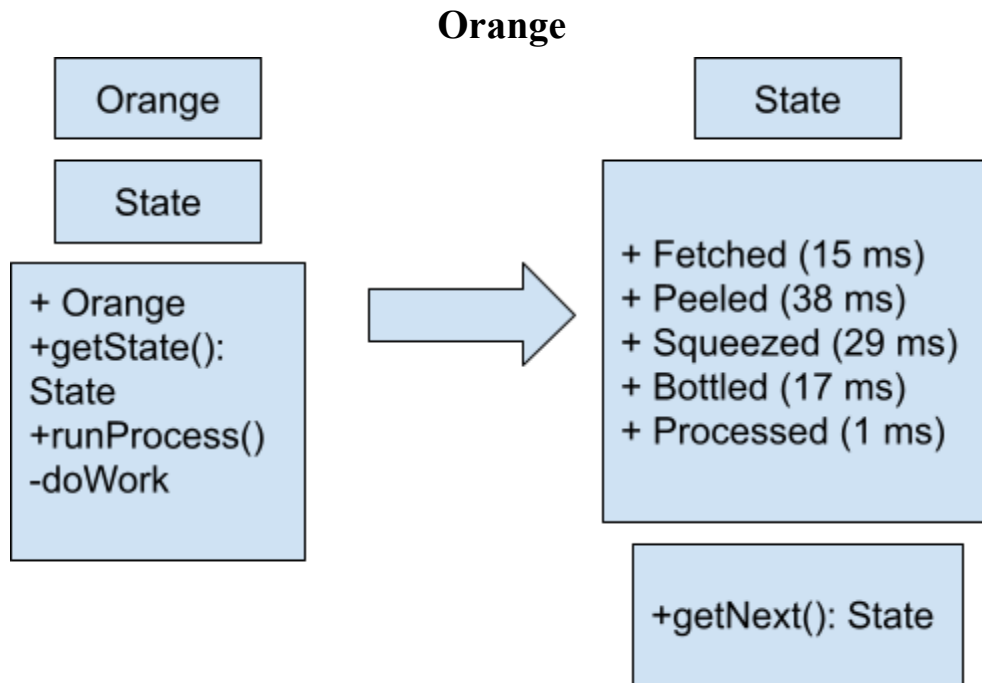


Juice Bottler UML Diagrams



Plant

Plant

```
+ PROCESSING_TIME: long
+ NUM_PLANT: int
+ WORKERS_PER_PLANT: int
- thread: Thread
- plantID: int
- timeToWork: volatile boolean
- inputMailboxes: BlockingMailbox[]
- orangesProvided: int
- orangesProcessed: int
```

```
+ Plant(plantID: int)
+ startPlant(): void
+ stopPlant(): void
+ waitToStop(): void
+ run(): void
- processOranges(workerID: int) void
+ getProvidedOranges(): int
+ getProcessedOranges(): int
+ getBottles(): int
+ getWaste(): int
+ main(args: String[]): void
```

Attributes:

- PROCESSING_TIME: The time the plant runs before stopping.
- NUM_PLANTS: Number of plant instances.
- WORKERS_PER_PLANT: Number of workers per plant.
- thread: The plant's main thread.
- plantId: Unique identifier for the plant.
- timeToWork: Flag to control plant operation.
- inputMailboxes: Array of mailboxes for worker communication.
- orangesProvided: Count of oranges sent for processing.
- orangesProcessed: Count of successfully processed oranges.

Methods:

- Plant(int plantId): Constructor initializes a plant.
- startPlant(): Starts the plant's processing thread.
- stopPlant(): Stops the plant and signals workers to halt.
- waitToStop(): Waits for the plant's thread to finish execution.
- run(): Runs the plant's main logic, managing workers and oranges.
- processOranges(workerId): Handles orange processing per worker.
- getProvidedOranges(): Returns total oranges provided.
- getProcessedOranges(): Returns total oranges processed.
- getBottles(): Calculates the number of bottled oranges.
- getWaste(): Computes the number of wasted oranges.
- main(args): Starts and manages multiple plant instances.