class VehicleFactory

+createVehicle(enum VehicleType) : Vehicle*

class Simulation

-currTime : long long -endTime : long long

-getCurrentTime() : long long -getTimeRemaining() : long long -printRemainingTime() : void +startSimulation(vector<Vehicle*>&, vector<Charger*>&, long ms) : void

class Charger

-chargeQueue : deque<Vehicle*> -vehicleCharging : Vehicle*

+enqueueVehicleWaiting(Vehicle*): void

- +dequeVehicleWaiting(): Vehicle*
- +isChargeQueueEmpty(): bool
- +increaseTimeWaiting(long ms): void
- +faultCheckQueue(): void
- +getVehicleCharging(): Vehicle*
- +setVehicleCharging(Vehicle*): void

class Vehicle

-Battery: Battery*
-cruiseSpeedMph: int
-passengerCount: int
-faultProbability: double

#getBattery(): Battery*
#getCruiseSpeed(): int
#getPassengerCount(): int
#getFaultProbability(): double

Has - - - →

class Battery

-capacityKwh: int

-currentChargeKwh : double -energyUseKwh : double -stateOfCharge : double -timeToChargeHr : double

-charging : bool

-setStateOfCharge() : void +getCapacity() : int

+getStateOfCharge : double +getEnergyUse() : double +getTimeToChargeHr() : double +drainBattery(long ms) : void

+chargeBattery(): void +isCharging(): bool

+setCharging(int EnOrDi): void

Is

class Alpha

class Beta

class Charlie

class Delta

class Echo

(class template)

-name : string

-vehicleCount: int

-timeInFlightHr: double

-timeChargingHr: double

-timeAwaitingChargeHr: double

-faultCount: int

-distanceAllPassengersMi : double

+getName(): string

+getVehicleCount(): int

+getTimeInFlight(): double

+getTimeCharging(): double

+getTimeAwaitingCharge(): double

+getFaultCount(): int

+getDistanceAllPassengers(): double

+addTimeInFlight(long ms): void

+addTimeCharging(long ms): void

+addTimeAwaitingCharge(long ms): void

+triggerFault(): void

+printData(): void

Main Notes

// Create vehicle factory

// Create vehicles in vector

// Create chargers in vector

// Get current time and store in main stack variable

// Start simulation for 3 minutes (equivalent to 3 hours)

// End simulation

// Clean up dynamically created objects