Elevator System:

Time Limit 1.5 hours

Build an application that simulates an elevator system

Problem Statement

- User should be able to add elevators to the system.
- User should be able to see the status of an elevator
- User should be able to call an elevator to a certain floor

Bonus 1:

- Support for people getting in the elevators and requesting floors.

Bonus 2:

- Multiple elevators working in a coordinated manner to minimize waiting time for the passenger

Expectations:

Modelling, Definition of entities. Clean modular code for future extensibility.

An example input/output for demonstration.

- add-elevator <elevator-id> <max-weight-supported-in-kgs>
- status <elevator-id>
- call-elevator <elevator-id> <floor-name>
- enter-elevator <elevator-id> <total-weight-in-kgs-entering> <space-separated-floor-names-requested>
- exit-elevator <elevator-id> <weight-exiting>
- continue <elevator-id>

Please keep total number of floors and floor names hardcoded in code.

Feel free to choose your own commands and methods for demonstration. This is just for illustration purposes.
Detailed Examples:

- User should be able to add elevators

  Command: add-elevator <elevator-id> <max-weight-supported-in-kgs>
  Example: add-elevator E1 200
  This adds the elevator "E1" which can support a maximum weight of 200 kgs. All elevators by default start from the lowest floor.

  Response(s):
    success -> if the request is successful
    failure -> if the request fails

- User should be able to see the status of an elevator
  Command: status <elevator-id>
  Example: status E1
  This requests for real time status of elevator E1

  Response(s):

  moving-down 6 B1 0 - The elevator is moving down, is on the 6th floor, headed towards B1, currently has 0 weight on board

  moving-up L 1 23 - The elevator is moving up, is on floor L, headed towards the 1st floor, currently with 23kgs on board

  stationary B1 - The elevator is at rest on floor B1

- User should be able to call an elevator to a certain floor
  Command: call-elevator <elevator-id> <floor-name>
  Example: call-elevator E1 B1
  This command requests elevator E1 to come to the B1 floor.

  Response(s):

  success -> if the request is successful
  failure -> if the request fails

  Example Sequence of Commands:

  status E1 -> stationary 6 (The elevator is currently on floor 6 at rest)
  call-elevator E1 B1 -> success
  call-elevator E1 B2 -> success

status E1 -> moving-down 6 B1 0 (Moving down to floors B1 with 0 weight on board)

- Do next action for the elevator
  Command: continue <elevator-id>

  Example Sequence of Commands:

  status E1 -> stationary 6 (The elevator is currently on floor 6 at rest)
  continue E1
  status E1 -> stationary 6 (The elevator is currently on floor 6 at rest, Nothing changed cause no command has been issued to the elevator yet)

  call-elevator E1 2 -> success
  call-elevator E1 1 -> success
  call-elevator E1 UB -> success
  call-elevator E1 B1 -> success
  status E1 -> moving-down 6 2 0
  continue E1 (The current floor the elevator is at, should be 2 now)
  status E1 -> stationary 2 (current floor is 2)
  continue E1
  status E1 -> moving down 2 1 0
  continue E1 (The current floor the elevator is at, should be 1 now)
  status E1 -> stationary 1 (current floor is 1)
  continue E1
  status E1 -> moving down 1 UB 0
  continue E1 (The current floor the elevator is at, should be UB now)
  status E1 -> stationary UB (current floor is UB)
  continue E1
  status E1 -> moving down UB B1 0
  continue E1
  status E1 -> stationary B1

Bonus:

- Support for people getting in/out the elevators and requesting floors.
  Command: enter-elevator E1 200 6,3 (Total weight 200 kgs entering the elevator, and have requested floors 3 and 6)
  exit-elevator E1 100 (Weight of 100kgs exiting the elevator. This should be possible only when the elevator is at a stopped/stationary state on a particular floor)

  Response(s):

  success -> if the request is successful
  failure -> if the request fails

  Example Sequence of Commands:

  status E1 -> stationary 1 (The elevator is currently on floor 1 at rest)
  enter-elevator E1 60 6,3 -> success
  enter-elevator E1 100 -> failure (breaches weight limit)
  continue E1 (The current floor the elevator is at, should be 3 now)
  status E1 -> stationary 3 (current floor is 3)
  exit-elevator E1 25
  status E1 -> stationary 3
  continue E1
  status E1 -> moving-up 3 6 35 (Elevator is moving upwards from the 3rd floor to the 6th floor with 35 kgs on board)
  continue E1
  status E1 -> stationary 6
  exit-elevator E1 35