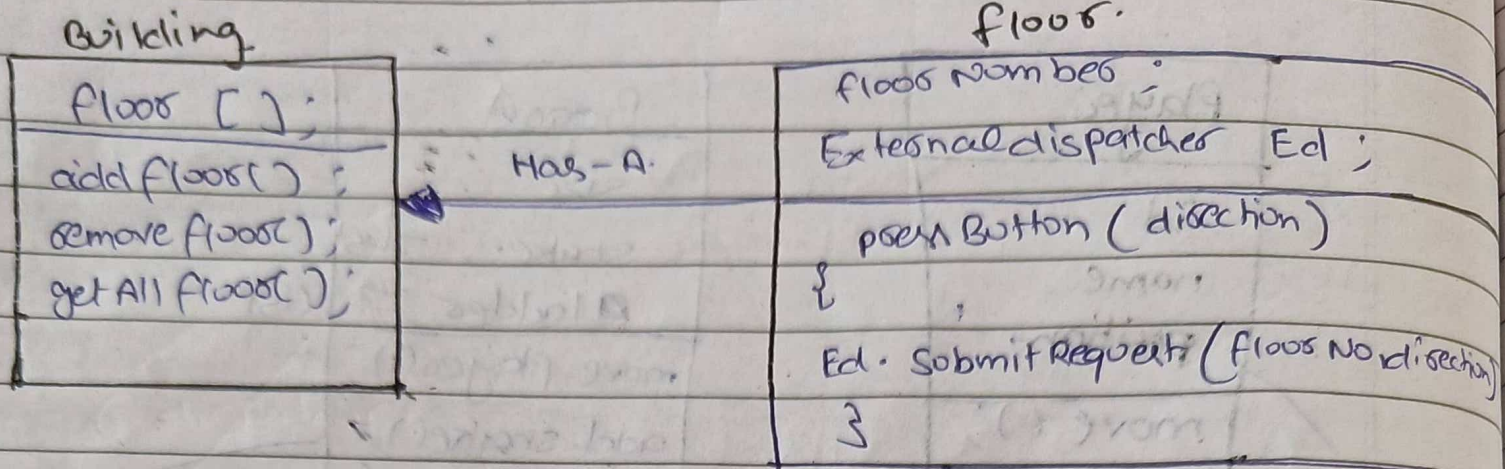


UO Elevator System



ENUM direction { 'UP' , 'DOWN' }

ENUM Elevator State { MOVING , IDLE }

Elevator Car

- 1) int Id ;
- 2) ElevatorDisplay elevatordisplay
- 3) Internal Buttons
- 4) ElevatorState
- 5) int Current floor
- 6) direction
- 7) Elevator door
- 8) Show Display () ⇒ display . showdisplay ()
- 9) set Display () ⇒ display . set Display (current floor , direction)
- 10) press Button () ⇒ internal Button . pressButton (floor No , direction)

Elevator Display

int floor;

Direction;

ShowDisplay();

setdisplay();

Elevator Door

isopen bool

Boolean isopen

openDoor();

closeDoor();

Elevator Controller

minHeap UPPE;

maxHeap DOWNPE;

Elevator Car

Submit External Request (floor, direction);

Submit Internal Request (floor, direction)

Elevator CreatorStatic list of Elevator
Controller

Static {

create controller

add in list

Dispatcher
Submit Request()

External dispatcher.

```
list[] Controller;
@Override
Submit Request() {
    call external Request();
}
```

Internal dispatcher

```
list[] Controller;
Submit Request (" ")
{
    call internal Request();
}
```

Button

Dispatcher D.

Button (Dispatcher D)

```
{
```

This.D = D;

```
}
```

press Button (floor Number, dis)

```
{
```

D.SubmitRequest(fN, dis)

```
}
```

is-a

is-a

External Button

```
value [] :  
constructor() {  
    super(new External  
           Dispatcher())  
}
```

Internal Button

```
value [] :  
constructor() {  
    super(new Internal  
           Dispatcher())  
}
```