**Class Clock**

- minRange, maxRange, index: int  
- clock: Clock

-

+ Clock (int index)  
+ getMinRange, getMaxRange, getIndex: int  
+ setMinRange (int min), setMaxRange (int max): void  
+ setPreviousClock (Clock clock): void  
+ countUp, countDown: void

**Class Clock24**

+ Clock24

**Class Clock60**

+ Clock60

**Class CompleteClock**

- minHour, minMinute, minSecond, maxHour, maxMinute, maxSecond: int  
- type: String  
- time: Timer

-

+ setMinCompleteClock (int minHour, int minMinute, int minSecond): void  
+ setMaxCompleteClock (int maxHour, int maxMinute, int maxSecond): void  
+ setType (String type): void  
+ run: void  
- checkCompleteClock: void  
- checkSingleClock (int hour, int minute, int second): void  
- checkType: void  
- checkSettingTime: void  
- countUp: void  
- countDown: void  
- timeToString (int hour, int minute, int second): void

**Class RunClock**

+ runClock: static void

- Tạo object completeClock  
+) Truyền vào minTime  
+) Truyền vào maxTime  
+) Truyền vào type  
+) Gọi phương thức run()

- Kiểm tra tính hợp lệ của tham số truyền: phương thức checkCompleteClock => lần lượt gọi:  
+) checkSingleClock: kiểm tra giờ, phút, giây  
+) checkType: kiểm tra type (up/down)  
+) checkSettingTime: kiểm tra giờ max và min

Hiện thông báo

- Tạo 3 object: object hour từ Clock24, object minute, second từ object Clock60  
- Kết nối các object:  
+) Gọi phương thức setPreviousClock của second, truyền vào minute  
+) Gọi phương thức setPreviousClock của minute, truyền vào hour

- Khởi tạo đối tượng time

- Gọi time.schedule, chu kì 1000 ms, gọi phương thức countUp/countDown của second

- Kiểm tra thời gian: tới mốc giới hạn hay chưa?

Dừng đếm

**No**

**Yes**

**Yes**

**No**

**Class RunClock**

**Class CompleteClock**