

## ***SIMULAZIONE***

Dentro a package model classe SIMULATORE e classe EVENTO.

### **EVENTO**

Implementa (implements) Comparable<Evento> per essere inserita nella coda Queue (su cosa confrontare due eventi, tipo su LocalTime) --- return this.tempo.compareTo(other.tempo);

- Variabile TEMPO

Tempo è continuo o discreto (predefinito) ?  
Se continuo: LocalTime (o Intero che definisca minuti)  
private LocalTime tempo;

- Variabile TIPO DI EVENTO

```
public enum TipoEvento {  
    ffff → tipo uno  
    oooo → tipo due  
}  
  
private TipoEvento tipo;
```

- Metodo COSTRUTTORE  
- Aggiungere solo getter

### **SIMULATORE**

- Variabile CODA EVENTI

```
private PriorityQueue<Evento> queue= new PriorityQueue<>();
```

- Variabile STATO DEL MONDO (insieme minimo di informazioni che mi servono per descrivere il problema) private

iniziali: possono variare o rimanere fissi in base alla simulazione

- Variabile PARAMETRI DI SIMULAZIONE (con quale modalità generare eventi)

dati dal testo e fissi! (private dati di inizio e fine simulazione ad esempio con LocalTime o intervalli con Duration )  
scelta random di valore: lista di durate  
NB da inizializzare lì preferibilmente!

- Variabile STATISTICHE RACCOLTE (esito)

ciò che il testo richiede come risultato

- Variabili INTERNE

```
private Random ran= new Random();
```

- Metodo INIT (preparare dei dati inizializzandoli)

```
inizializzare Stato del Mondo, statistiche azzerate, resetto coda eventi ( this.queue.clear(); )  
caricare eventi iniziali (for( inizio; fine ; somma intervallo;))  
    creazione evento  
    aggiunta alla queue  queue.add(new Evento(x, TipoEvento.ffff));  
inizializzazione lista parametro random con valori possibili
```

- Metodo RUN (grosso loop finchè condizione di uscita si verifica)

```
-può essere finchè coda non è vuota  
while(!queue.isEmpty())  
{  
    Evento ev= queue.poll(); (poll prende e toglie, pick lo guarda e basta)  
    switch (sulla base di tipo di evento  ev.getTipo())  
        case ffff:  
            break;  *  
        case oooo:  
            break;  
    }  
-può essere fino a un certo LocalTime/valore statistico predefinito
```

- \* aggiornare stato del mondo, statistiche, schedulare nuovo evento  
 estrazione random rand.nextInt( x) (tra 0 e x valore specificato escluso)  
 per sommare LocalTime uso .plus()

genero getters e setters dove servono