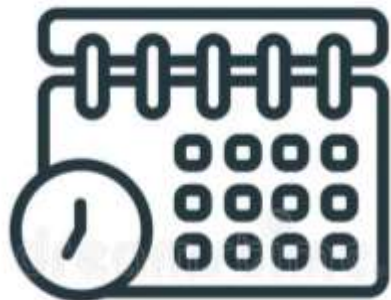


Prenotazione camere

Aggregazione, ArrayList o array, ordinamenti



Camere e prenotazioni


descrizione

- ▶ Realizzare le classi `Room` e `Reservation`, che rappresentano una camera d'albergo e una prenotazione per la camera.
- ▶ Una camera ha un numero da 100 a 599.
- ▶ Il metodo `reserve` di `Room` riceve il nome cliente, la data di inizio e di fine prenotazione, e restituisce un oggetto di tipo `Reservation`.
 - ▶ Se la camera è occupata in una delle giornate richieste, il metodo non esegue la prenotazione lancia un'errore
 - ▶ Viceversa registra la prenotazione nell'elenco di prenotazioni della camera.
- ▶ Per semplicità, una data è rappresentata da un numero intero tra 1 a 365.
- ▶ Il metodo `reservations` di `Room` restituisce l'elenco delle prenotazioni, eventualmente in ordine cronologico di data di check in.
- ▶ L'implementazione deve rispettare il seguente esempio d'uso →

Camere e prenotazioni

esempio d'uso

```
Room r = new Room(103);  
Reservation p1 = r.reserve("mario rossi", 105, 120);  
Reservation p2 = r.reserve("anna bianchi", 5, 20);  
Reservation p3 = r.reserve("piro neri", 20, 22);  
Reservation p4 = r.reserve("gianna gialli", 200, 222);  
for (Reservation p: r. reservations ())  
    System.out.println(p.getName());
```



OUTPUT:
anna bianchi
piro neri
mario rossi
gianna gialli

Eeguire anche il test con la chiamata:

```
Reservation p5 = r.reserve("tony blu", 21, 23);
```

Che dovrebbe andare in errore poiché la camera è occupata !