```
Problem : Order a hotel online before a trip
Identify Objects and Behaviors:
Thing(Object)/Entity:
        Information(Data)/(State):
        Services(Behaviors)/Actions
Thing:
        OnlineBookService:
                Data: hotels(list of hotel)
                Behaviors: searchHotels(), book(), cancel(), update()
        Hotel:
                Data: Name, location, rooms(list of room)
                Behaviors: orderRoom(), releaseRoom(),
hasAvailableRoom()
        Trip:
                Data: From, Destination, beginTime, endTime
                Behaviors:
        traveller:
                Data: Name, id,
                Behaviors:
Sequence of invoking behaviors on objects:
        Traveller traveller,
        OnlineBookService bookService,
        Trip trip,
        Hotel hotel

        if(trip.isReady()) {

                bookService.searchHotels -> trip.from, trip.beginTime,
trip.destination, trip.endTime: Collection of qualified hotels(ordered
by match score);
                Loop hotel collection:
                        if hotel.hasAvailableRoom():
                                bookService.book()-> traveller.name,
room.id, timeRange

                                hotel.orderRoom(room.id)
                                break
                        end

                if (trip.needUpdate()) {
                        bookService.getRoomByTravellerName() ->
traveller.name: booked room
                        bookService.cancel(booked room)
                        hotel.releaseRoom(booked room)
                }
        }
```