Test Cases

Booking Class Tests

- **Test 1:** Ensures that a booking is correctly created with valid guest and room details, including ID, check-in, and check-out dates.

- **Test 2:** Verifies that the check-out date of an existing booking can be updated successfully.

```
@HamedAlashwal →/workspaces/Royal-Stay-Hotel-Management-System (main) $ python test_booking.py

===== Test 1 ======
1
Ali Hassan
101
2025-03-28
2025-03-30

===== Test 2 ======
2025-04-01
@HamedAlashwal →/workspaces/Royal-Stay-Hotel-Management-System (main) $
```

Guest Class Tests

- **Test 1:** Checks if guest objects store correct details, including guest ID, name, email, phone number, and loyalty status.

- **Test 2:** Confirms that a guest's loyalty status can be modified and reflects the correct updated value.

```
@HamedAlashwal →/workspaces/Royal-Stay-Hotel-Management-System (main) $ python test_guest.py

===== Test 1 ======
1
Ali Hassan
ali.hassan@gmail.com
+971501234567
LoyaltyStatus.REGULAR
2
Fatima Noor
fatima.noor@gmail.com
+971502345678
LoyaltyStatus.GOLD

===== Test 2 ======
LoyaltyStatus.SILVER
```

Room Class Tests

- **Test 1:** Ensures room objects store correct details such as room number, type, amenities, price, and availability.

- **Test 2:** Verifies that room availability can be changed and reflects the updated status.

```
● @HamedAlashwal →/workspaces/Royal-Stay-Hotel-Management-System (main) $ python test_room.py

===== Test 1 ======
101
RoomType.SINGLE
300
['Wi-Fi', 'TV']
True
202
RoomType.SUITE
800
['Wi-Fi', 'Mini-Bar', 'TV']
False

===== Test 2 ======
True
```

Invoice Class Tests

- **Test 1:** Validates that an invoice correctly links to a booking and stores the correct amount and payment status.

- **Test 2:** Checks if the payment status of an invoice can be updated properly.

```
● @HamedAlashwal →/workspaces/Royal-Stay-Hotel-Management-System (main) $ python test_invoice.py

===== Test 1 ======
1001
10
1400
Pending
Paid

===== Test 2 ======
1002
11
2700
Pending
Failed
```

Review Class Tests

- **Test 1:** Ensures a review object stores correct details, including guest information, rating, and review text.

- **Test 2:** Verifies that different guests can submit reviews with various ratings and feedback.

```
@HamedAlashwal →/workspaces/Royal-Stay-Hotel-Management-System (main) $ python test_review.py

===== Test 1 ======
Review ID: 301, Guest: Ayesha Al Ketbi, Rating: ★★★★, Comment: Great stay, very comfortable.

===== Test 2 ======
Review ID: 302, Guest: Hassan Al Zarooni, Rating: ★★★, Comment: Good service but room was small.
```

Reservation History and Cancellation

- **Test 1** - Checks if a guest's reservation history is stored correctly. The test creates two bookings for the same guest and prints them to ensure they are recorded properly.

- **Test 2** - Verifies that a booking can be canceled and that the room becomes available again after cancellation.

```
@HamedAlashwal →/workspaces/Royal-Stay-Hotel-Management-System (main) $ python test_history_cancellation.py

===== Test 1 ======
Booking ID: 101, Guest: Khalid Al Mansoori, Room: 201, Check-in: 2025-04-01, Check-out: 2025-04-05, Status: Completed
Booking ID: 102, Guest: Khalid Al Mansoori, Room: 305, Check-in: 2025-05-10, Check-out: 2025-05-15, Status: Confirmed

===== Test 2 ======
Booking ID: 102, Guest: Khalid Al Mansoori, Room: 305, Check-in: 2025-05-10, Check-out: 2025-05-15, Status: Canceled
Room Availability: True
```