

**Gebze Technical University
Computer Engineering**

CSE 222 - 2018 Spring

HOMEWORK 1 REPORT

**BURHAN ELGÜN
151044060**

Course Assistant : Fatma Nur ESİRCİ

1 INTRODUCTION

1.1 Problem Definition

Bir otelde odalar , kullanıcılar ve check-in, check-out, rezervasyon gibi birtakım operasyonlar vardır. Bunların hepsinin bir arada ve birbirleriyle ilişki içerisinde çalışması gerekmektedir. Bu işlerin özellikle büyük otellerde bir otel yönetim sistemi yardımı olmadan yapılması büyük bir vakit kaybına neden olur.Ayrıca fazla maliyetlere de neden olabilmektedir.

1.2 System Requirements

boolean bookRoom(int roomNumber) throws Exception = bookRoom methodunda roomNumber girdisine ihtiyaç var. roomNumber değişkeni oda numarasını tutmalı.Fonksiyon sonuç olarak boolean döndürür. Eğer book edildiyse true, edilmediyse false döndürür.Geçersiz değerde exception fırlatır.Hem receptionist hemde hotel guest bu methodu çağırabilir.

boolean cancelReservation(int roomNumber) throws Exception =cancelReservation methodunda roomNumber girdisine ihtiyaç var. roomNumber değişkeni oda numarasını tutmalı.Fonksiyon sonuç olarak boolean döndürür. Eğer cancel edildiyse true, edilmediyse false döndürür.Geçersiz değerde exception fırlatır.Hem receptionist hemde hotel guest bu methou çağırabilir.

void printConfirmation() = printConfirmation methodu void tir ve herhangi bir girdiye ihtiyaç duymaz. Hotel classının bir methodur.Hotel classı üzerinden çağırılabilir.

void printExitInfo()= printExitInfo methodu void tir ve herhangi bir girdiye ihtiyaç duymaz. Hotel classının bir methodur.Hotel classı üzerinden çağırılabilir.

public boolean checkIn(int roomNumber) throws Exception = checkIn methodunda roomNumber girdisine ihtiyaç var. roomNumber değişkeni oda numarasını tutmalı.Fonksiyon sonuç olarak boolean döndürür. Eğer checkIn edildiyse true, edilmediyse false döndürür.Geçersiz değerde exception fırlatır.Yalnızca receptionist objesi üzerinden çağırılabilir.

public boolean checkOut(int roomNumber) throws Exception = checkOut methodunda roomNumber girdisine ihtiyaç var. roomNumber değişkeni oda numarasını tutmalı.Fonksiyon sonuç olarak boolean döndürür. Eğer checkOut edildiyse true,

edilmediyse false döndürür.Geçersiz değerde exception fırlatır.Yalnızca receptionist objesi üzerinden çağırılabilir.

public void printRoomStatus() = printRoomStatus methodu herhangi bir sistem gereksinimine ihtiyaç duymaz.Void tir , girdi almaz.Hotel objesi üzerinden çağırılmalıdır.

public void writeFileRoomStatus() throws

FileNotFoundException,UnsupportedEncodingException = writeFileRoomStatus methodu herhangi bir sitem gereksinimine ihtiyaç duymaz.Void tir, girdi almaz. Dosya işleminde hata olursa Exception fırlatır.Hotel objesi üzerinden çağırılmalıdır.

public void beginARecordFile() throws

FileNotFoundException,UnsupportedEncodingException =beginARecordFile methodu herhangi bir sitem gereksinimine ihtiyaç duymaz.Void tir, girdi almaz. Dosya işleminde hata olursa Exception fırlatır.Hotel objesi üzerinden çağırılmalıdır.

public void readRoomStatusFromFile()= readRoomStatusFromFile methodu herhangi bir sistem gereksinimine ihtiyaç duymaz.Void tir, girdi almaz. Hotel objesi üzerinden çağırılmalıdır.

public int askUserType() throws InputMismatchException = askUserType methodu herhangi bir sistem gereksinimine ihtiyaç duymaz.Void tir, girdi almaz. Hotel objesi üzerinden çağırılmalıdır.

public int askReceptionistOperationType() throws InputMismatchException= askReceptionistOperationType methodu herhangi bir sitem gereksinimine ihtiyaç duymaz.Void tir, girdi almaz.Hotel objesi üzerinden çağırılmalıdır.

public int askUserRoomID() throws InputMismatchException= askUserRoomID methodu herhangi bir sitem gereksinimine ihtiyaç duymaz.Void tir, girdi almaz.Hotel objesi üzerinden çağırılmalıdır.

protected class Room= Room classı , Hotel classının inner class ıdır. Ayrıca Room classı data field olarak Hotel Classının içerisinde kullanılmıştır.Has-A relation yapılmıştır.

public class Hotel = Hotel classı, hotel ile alakalı bilgileri ekrana print eden, dosyaya yazan , dosyadan okuyan methodları içerir.

public class HotelGuest = HotelGuest classı , bir oteldeki bir guestin yapabileceği operasyonların methodlarını içerir. Yani book ve cancel.

public class Receptionist = Receptionist classı, bir oteldeki bir receptionistin yapabileceği operasyonların methodlarını içerir.Yani book, cancel, check in , check out.

public abstrac class SystemUsers = SystemUsers classı hem recepitonistin hemde guestin ortak olarak aynı şekilde yapabilgiği tek bir method olan cancelReservationu implement eder.

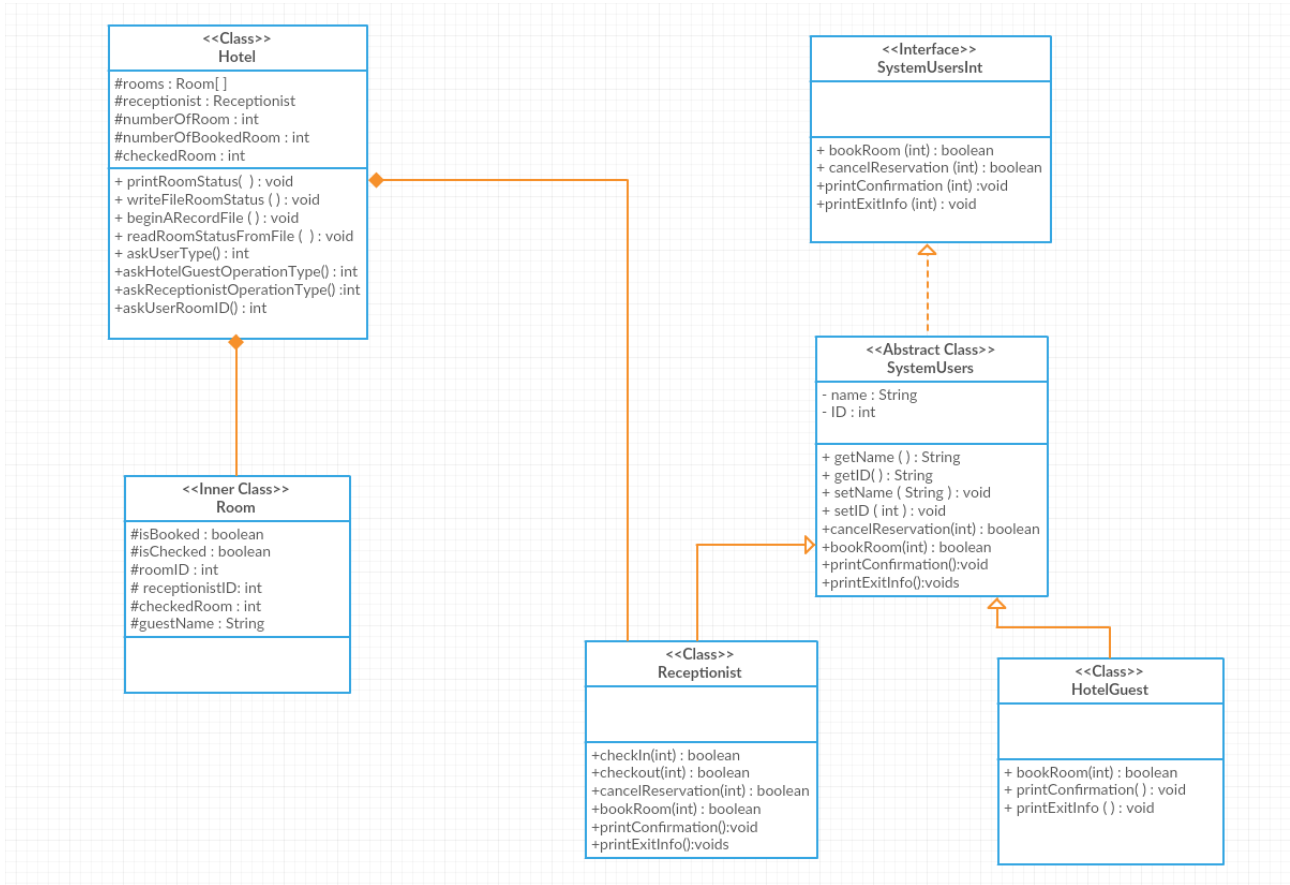
public interface SystemUsersInt = SystemUsersInt interfacesi ise book , cancel, print gibi methodları içerir.

Genel olarak sistem kullanım hiyerarşisi şu şekilde dir:

Sistem kullanıcısı sisteme girdiğinde eğer bir kayıt dosyası oluşmamışsa oluşturur ve içine default room değerleri atar.Eğer oluşmuşsa onu kullanarak , kullanılan rooms arrayine gerekli atamalar yapılır. Artık sistem kapanana kadar o array içindeki odalarla işlemler yapılır. Sonrasında sistem kullanıcısına kim olduğu sorulur . Kullanıcı ya receptionist yada hotel guest i seçebilir. Receptionist seçilirse , hangi operasyonu yapacağı sorulur bunlar (book , cancel, check in , check out tur) bunlardan birini seçer.Ardından bu operasyonu hangi oda için yapacağı sorulur , receptionist ekrana print edilen odalar arasından bir oda numarasını seçer ve operasyon tamamlanmış olur.Aynı şekilde eğer hotelGuest seçilirse hangi operasyonu yapacağı sorulur bunlar (book , cancel, check in , check out tur) bunlardan birini seçer.Ardından bu operasyonu hangi oda için yapacağı sorulur , receptionist ekrana print edilen odalar arasından bir oda numarasını seçer ve operasyon tamamlanmış olur.

2 METHOD

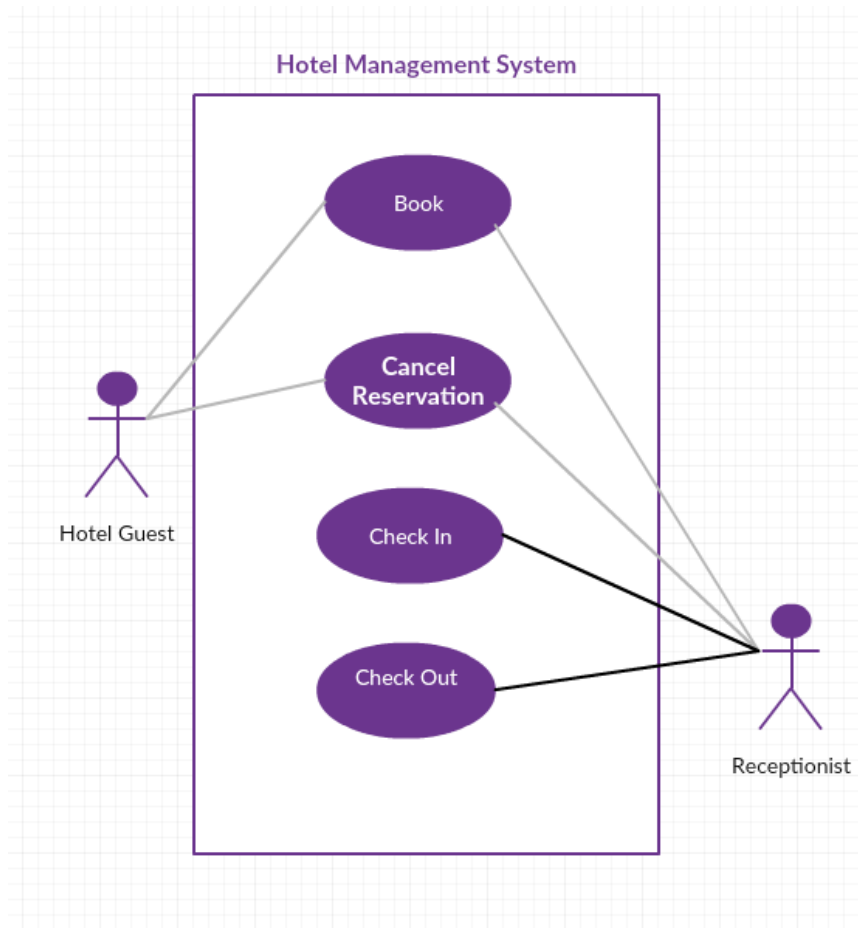
2.1 Class Diagrams



Görüldüğü üzere 1 SystemUsersInt adında interface, 1 SystemUsers adında abstract class, 4 tane de Hotel, Room, Receptionist, HotelGuest adında normal class bulunmaktadır. Hotel classından başlarsak. Hotel classı, içerisinde rooms adında Room arrayi tutmaktadır. Burda has – A relation yani composition ilkesi sağlanmıştır. Çünkü Hotel has a room cümlesi doğrudur. Hotel aynı zamanda Receptioniste de sahiptir bu yüzden burdada has – A kullanıldı. Hotel classının içerisinde ise inner olarak protected Room classı bulunmaktadır. Ve bir odanın book, checkIn, checkOut olup olmadığını ayrıca o odayı hangi receptionist check in yaptıysa onun ID si yada hangi hotel guest book yaptıysa onun adını tutar. SystemUserInt interfacesi ise bir Sistem kullanıcısının temel olarak yapması gereken operasyonların methodlarını içerir. Bu interfaceyi kısmen implement eden Abstract class SystemUsers ise, sistem kullanıcısının ID sini veya name i ni tutmaktadır. Getter ve setterleri vardır. Fakat yalnızca cancelReservation methodu implement edilmiştir bu yüzden abstract olmuştur. Diğer 3 methodun implement edilememesi ise kendisinden türeyecek olan classa göre implementasyonun değişecek

olmasıdır. Receptionist classında ve HotelGuest classında, bir receptionistin ve hotel guestin yapabildiği operasyonların methodları implemet edilmiştir.

2.2 Use Case Diagrams



Use case diagramında görüldüğü üzere bir Hotel Management System'de Hotel Guest kullanıcısı book ve cancel reservation yapabilir.Receptionist kullanıcısı ise book,cancel reservation,check in, check out yapabilir.

2.3 Other Diagrams (optional)

Add other diagrams if required.

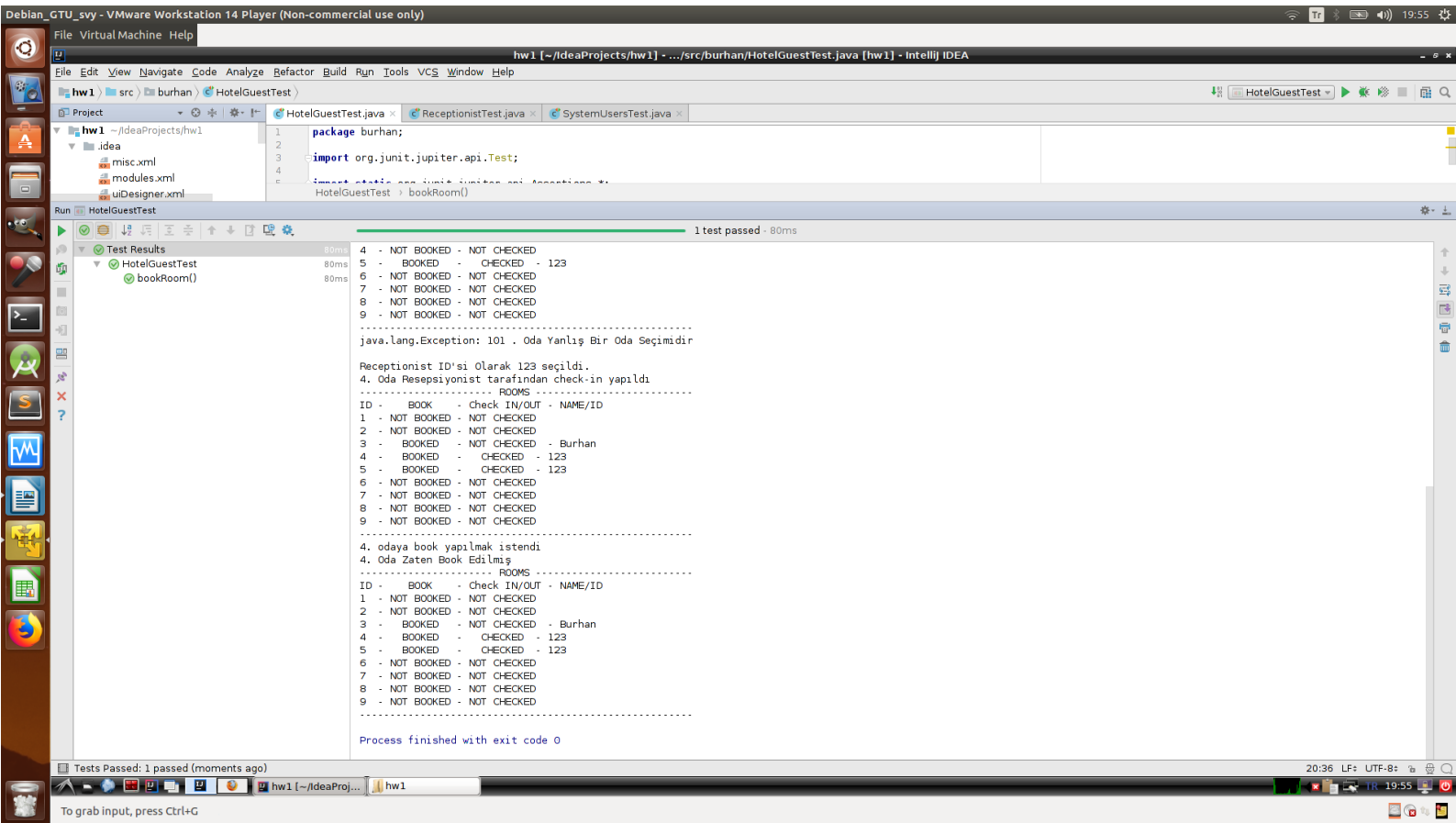
2.4 Problem Solution Approach

Problemi çözmek için Hotel classı oluşturdum ve bu Hotel classında Room classını composition olarak kullandım.Yani bir hotelde bir çok Room tipinde oda olabilir.Her bir odanında kendi özellikleri vardır.Bu bilgileri tutmak için Hotel classının içinde Room arrayi kullandım.Hoteldeki room sayısı kadar array elemanı oluşturdum.Her bir indexte bir oda bulunuyor ve bu Room tipindeki objelerin data fieldlerinde check in olup olmadığı , book olup olmadığı bilgileri de tutuluyor. Ardından bu odalar üzerinde operasyonları gerçekleştiren iki farklı sistem kullanıcısı olması gerekiyordu. Bir sistem kullanıcısında olması gereken ortak özellikleri bir SystemUsersInt adlı bir interface de topladım. Ardından onu implement eden bir SystemUsers adlı abstract bir class yaptım.Sadece cancelReservation adlı methodu implement edebildim çünkü diğer bookRoom , printExitInfo,printConfirmation methodlarının implementasyonları extend olan classa göre değişiklik gösterecekti.Ardından bu abstract classtan , Receptionist ve HotelGuest adlı iki tane class extend ettim. Abstract classtaki abstract methodları overriding yaptım.Polimorfizmi ise şu şeklide kullandım. SystemUsers referansı gösteren bir Receptionist objesi oluşturun ve bu referans üzerinden printExitInfo yada printConfirmation methodlarını çağırdığımda Receptionist implement ettiği methodlar çalışıyor. Ama eğer systemUsers referansı gösteren bir HotelGuest objesi oluşturursam ve bu referans üzerinden printExitInfo yada printConfirmation methodlarını çağırdığımda HotelGuest in implement ettiği methodlar çalışıyor. Yani bir fonksiyona birden çok mana verilmiş oluyor.Run time da karar verilmiş oluyor hangi classın methodunun çalıştırılacağına. Ayrıca mainde testArr arrayinin 0. indexi üzerinde HotelGuest in operasyonlarının test edildiği odanın numarasıdır, 1.index üzerinde Receptionist in operasyonlarının test edildiği odanın numarasıdır.

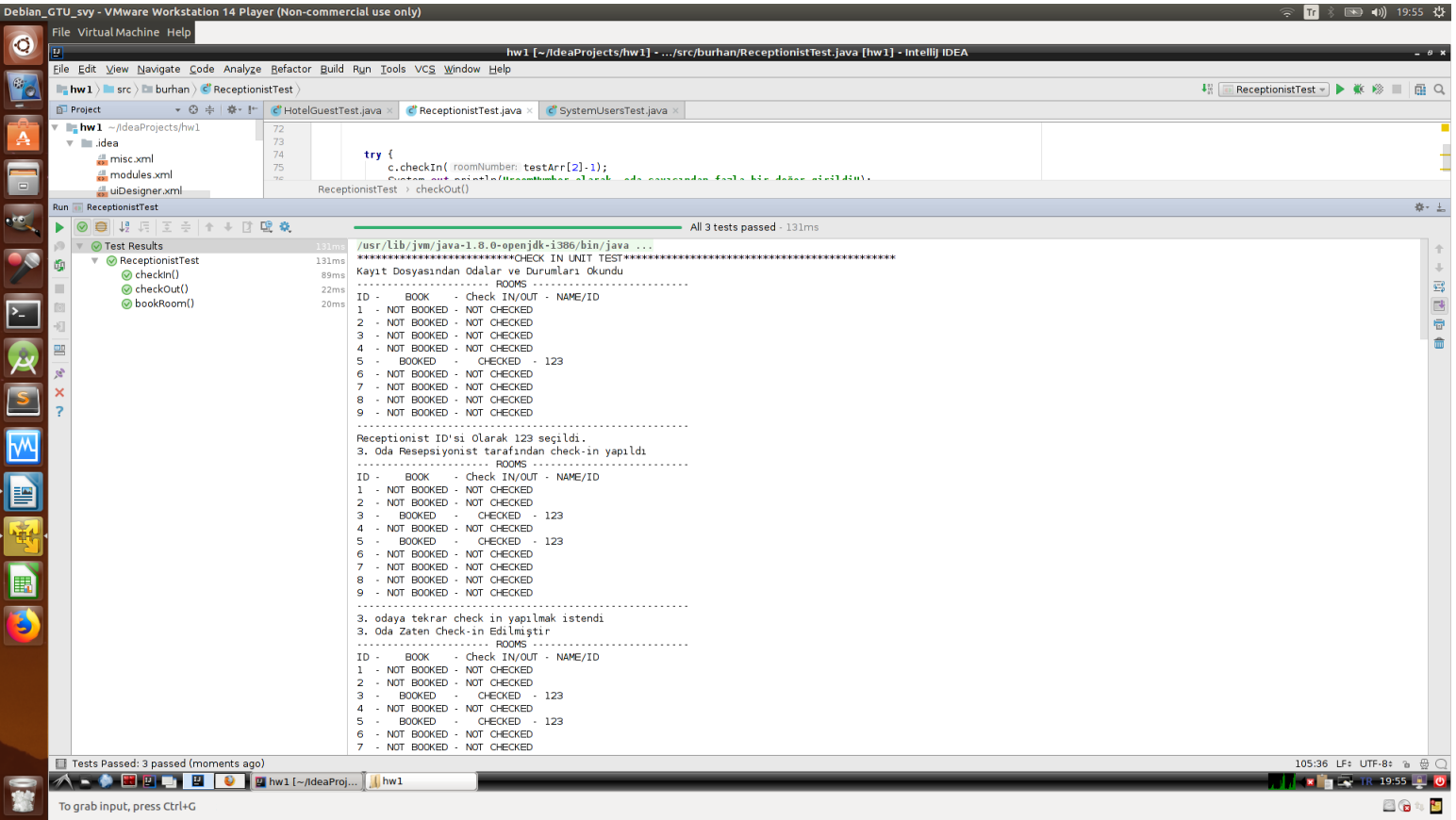
3 RESULT

3.1 Test Cases

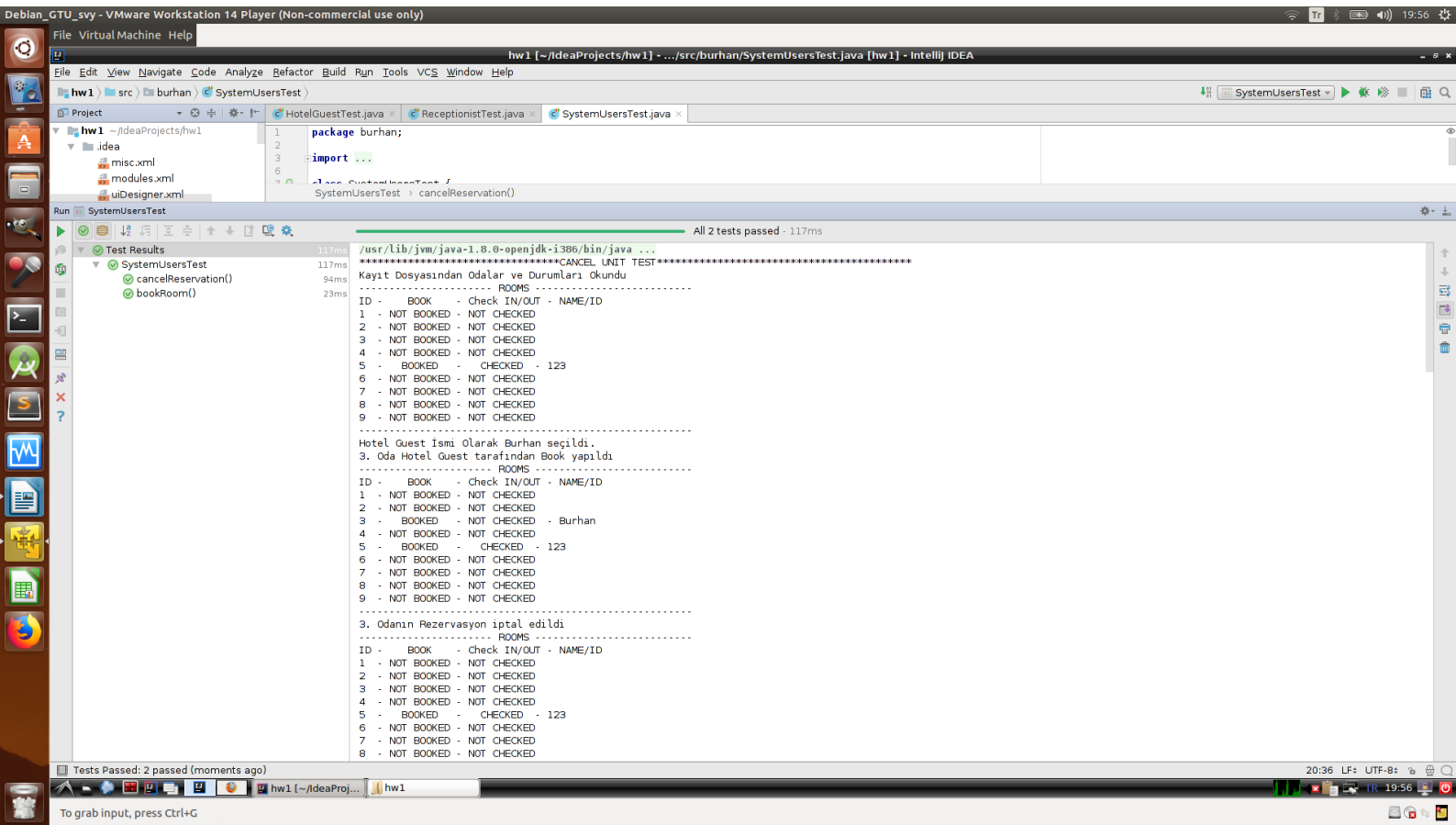
İlk screen shot HotelGuesti ile alakalı testleri içerir.HotelGuesti nin yaptığı booking operasyonu test edilmiştir. Book yapılmış bir yere book yapılmaya çalışılmıştır.Check in yapılmış bir yere book yapılmaya çalışılmıştır. Olmayan bir odaya book yapılmaya çalışılmıştır.Not booked olan bir oda book edilmeye çalışılmıştır.Methodlara parametre ile testArr adındaki arraydan gönderme yapıyorum. Arrayin hangi elemanı hangi işlevi gördüğü array initialize edilirken yanına yazılmıştır.



İkinci test cases receptionist ile alakalı testleri içermektedir. Check in , check out, bookRoom senaryolarını test etmektedir Check inde , zaten checkin olan bir yer check in edilmeye çalışılmıştır.Sonra normal boş olan bir yer check in edilmeye çalışılmıştır. Sonra book edilen bir yer check in edilmeye çalışılmıştır.Check methodunda , checkin edilmiş bir yer check out yapılmaya çalışılmıştır, ardından aynı yer tekrar denenmiştir. Hem check out hemde checkin methodlarına olmayan oda numaraları gönderilmiştir.Receptionistin yaptığı booking operasyonu test edilmiştir. Book yapılmış bir yere book yapılmaya çalışılmıştır.Check in yapılmış bir yere book yapılmaya çalışılmıştır. Olmayan bir odaya book yapılmaya çalışılmıştır.Not booked olan bir oda book edilmeye çalışılmıştır.Methodlara parametre ile testArr adındaki arraydan gönderme yapıyorum. Arrayin hangi elemanı hangi işlevi gördüğü array initialize edilirken yanına yazılmıştır.

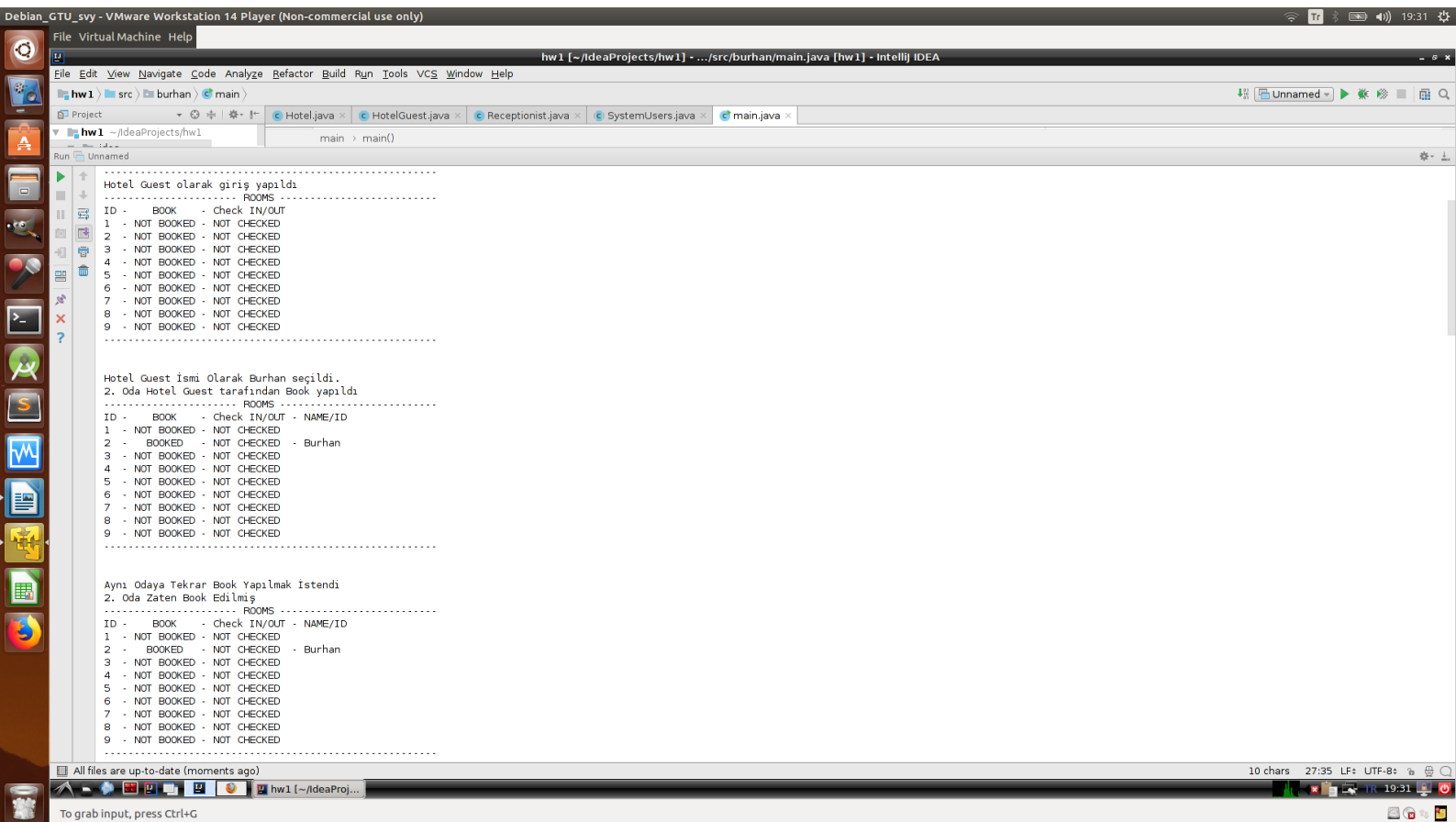
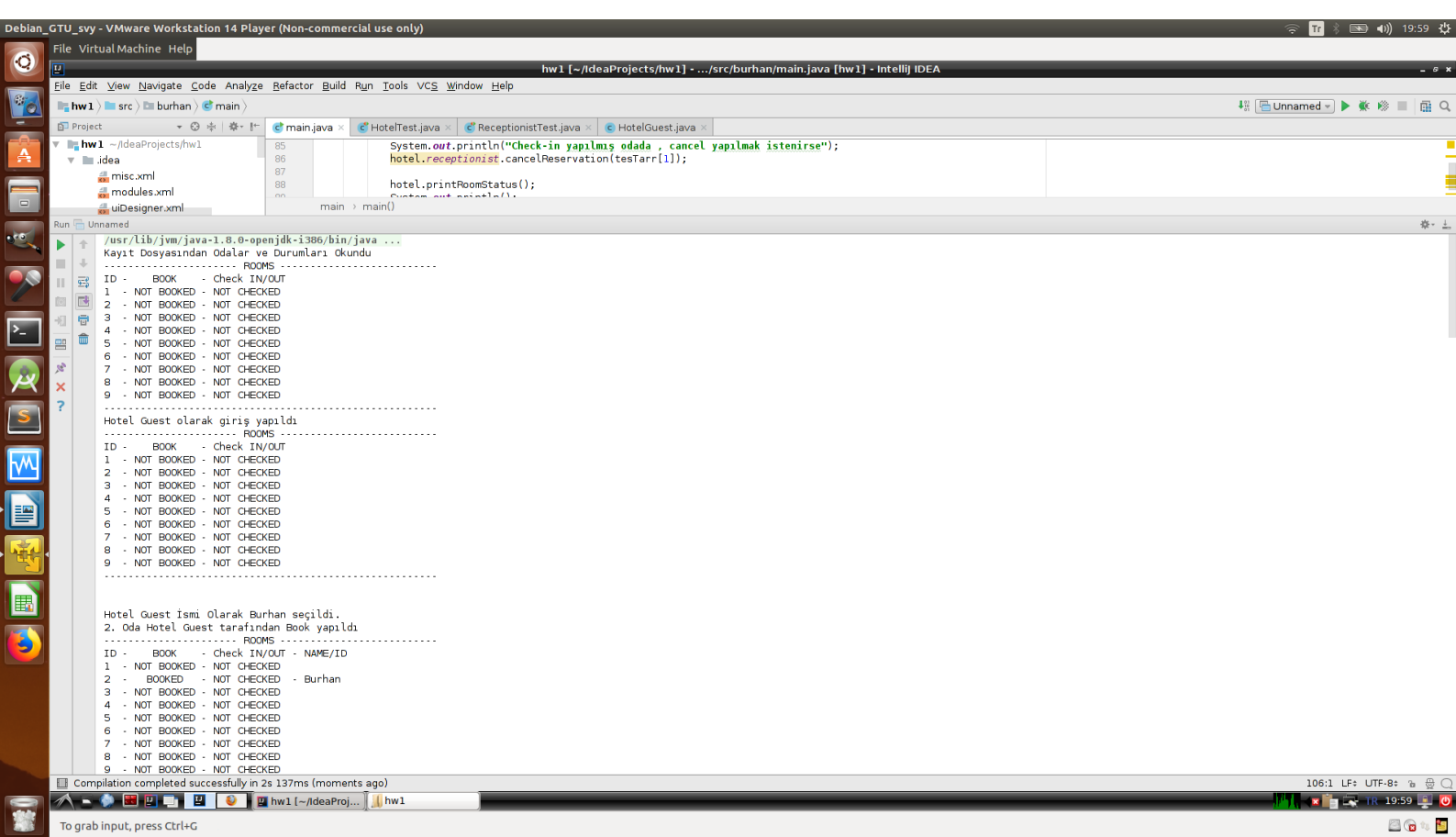


Üçüncü test cases SystemUsers ile alakalı testleri içermektedir. Cancel methodu çeşitli senaryolarla test edilmiştir. Book edilen bir yer cancel edilmeye çalışılmıştır, ardından check in edilen bir yer book edilmeye çalışılmıştır ardından , olamayan bir oda numarası cancelReservaion methoduna gönderilmiştir. Methodlara parametre ile testArr adındaki arraydan gönderme yapıyorum. Arrayin hangi elemanı hangi işlevi gördüğü array initialize edilirken yanına yazılmıştır.



3.2 Running Results

Main test screen shotları içermektedir, birden çok senaryo denenmiştir. Dosyadan okuma dosyaya yazma yapılmıştır. Book edilen room tekrar book edilmeye çalışılmıştır. Check in edilen bir yere cancel yapılmaya çalışılmıştır. Check out yapılan bir yere tekrar check out yapılmaya çalışılmıştır. Oda sayısından fazla bir oda sayısı girilmiştir. Aynı zamanda normal bir şekilde hata yapmadan canceli book, check out, check in işlemleri gerçekleştirilmiştir. Aşağıdaki bu testlerin çıktıları bulunmaktadır. Methodlara parametre ile testArr adındaki arraydan gönderme yapıyorum. Arrayın hangi elemanı hangi işlevi gördüğü array initialize edilirken yanına yazılmıştır.



Debian_GTU_svy - VMware Workstation 14 Player (Non-commercial use only)

File Virtual Machine Help

hw1 [-/IdeaProjects/hw1] - .../src/burhan/main.java [hw1] - IntelliJ IDEA

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

Project hw1 [-/IdeaProjects/hw1] rmain - main()

Run Unnamed

```
.....
ID - BOOK - Check IN/OUT - NAME/ID
1 - NOT BOOKED - NOT CHECKED
2 - BOOKED - NOT CHECKED - Burhan
3 - NOT BOOKED - NOT CHECKED
4 - NOT BOOKED - NOT CHECKED
5 - NOT BOOKED - NOT CHECKED
6 - NOT BOOKED - NOT CHECKED
7 - NOT BOOKED - NOT CHECKED
8 - NOT BOOKED - NOT CHECKED
9 - NOT BOOKED - NOT CHECKED
.....

2. Odanın Rezervasyon iptal edildi
.....
ID - BOOK - Check IN/OUT
1 - NOT BOOKED - NOT CHECKED
2 - NOT BOOKED - NOT CHECKED
3 - NOT BOOKED - NOT CHECKED
4 - NOT BOOKED - NOT CHECKED
5 - NOT BOOKED - NOT CHECKED
6 - NOT BOOKED - NOT CHECKED
7 - NOT BOOKED - NOT CHECKED
8 - NOT BOOKED - NOT CHECKED
9 - NOT BOOKED - NOT CHECKED
.....

Aynı oda tekrar cancel yapılmak istendi
2 - Oda Zaten Not Booked tir
Hotel Guest olarak çıkış yapıldı
.....
Receptionist olarak giriş yapıldı
.....
ID - BOOK - Check IN/OUT
1 - NOT BOOKED - NOT CHECKED
2 - NOT BOOKED - NOT CHECKED
3 - NOT BOOKED - NOT CHECKED
4 - NOT BOOKED - NOT CHECKED
5 - NOT BOOKED - NOT CHECKED
6 - NOT BOOKED - NOT CHECKED
7 - NOT BOOKED - NOT CHECKED
8 - NOT BOOKED - NOT CHECKED
9 - NOT BOOKED - NOT CHECKED
.....
```

All files are up-to-date (a minute ago)

hw1 [-/IdeaProj...

To grab input, press Ctrl+G

Debian_GTU_svy - VMware Workstation 14 Player (Non-commercial use only)

File Virtual Machine Help

hw1 [-/IdeaProjects/hw1] - .../src/burhan/main.java [hw1] - IntelliJ IDEA

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

Project hw1 [-/IdeaProjects/hw1] rmain - main()

Run Unnamed

```
Receptionist ID'si olarak 123 seçildi.
5. Oda Resepsiyonist tarafından check-in yapıldı
.....
ID - BOOK - Check IN/OUT - NAME/ID
1 - NOT BOOKED - NOT CHECKED
2 - NOT BOOKED - NOT CHECKED
3 - NOT BOOKED - NOT CHECKED
4 - NOT BOOKED - NOT CHECKED
5 - BOOKED - CHECKED - 123
6 - NOT BOOKED - NOT CHECKED
7 - NOT BOOKED - NOT CHECKED
8 - NOT BOOKED - NOT CHECKED
9 - NOT BOOKED - NOT CHECKED
.....

Aynı oda tekrar check-in yapılmak istendi
5. Oda Zaten Check-in Edilmiştir
.....
ID - BOOK - Check IN/OUT - NAME/ID
1 - NOT BOOKED - NOT CHECKED
2 - NOT BOOKED - NOT CHECKED
3 - NOT BOOKED - NOT CHECKED
4 - NOT BOOKED - NOT CHECKED
5 - BOOKED - CHECKED - 123
6 - NOT BOOKED - NOT CHECKED
7 - NOT BOOKED - NOT CHECKED
8 - NOT BOOKED - NOT CHECKED
9 - NOT BOOKED - NOT CHECKED
.....

Check-in yapılmış odada , cancel yapılmak istenirse
5. Oda zaten check-in olduğu için rezervasyonunuzu iptal edemezsiniz.
.....
ID - BOOK - Check IN/OUT - NAME/ID
1 - NOT BOOKED - NOT CHECKED
2 - NOT BOOKED - NOT CHECKED
3 - NOT BOOKED - NOT CHECKED
4 - NOT BOOKED - NOT CHECKED
5 - BOOKED - CHECKED - 123
6 - NOT BOOKED - NOT CHECKED
7 - NOT BOOKED - NOT CHECKED
8 - NOT BOOKED - NOT CHECKED
9 - NOT BOOKED - NOT CHECKED
.....
```

All files are up-to-date (a minute ago)

hw1 [-/IdeaProj...

To grab input, press Ctrl+G

Debian_GTU_svy - VMware Workstation 14 Player (Non-commercial use only)

File Virtual Machine Help

hw1 [~/IdeaProjects/hw1] - .../src/burhan/main.java [hw1] - IntelliJ IDEA

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

Project hw1 src burhan main

Project hw1 src burhan main

Run Unnamed

```
5. Oda Resepsiyonist tarafından check-out yapıldı
-----
ID - BOOK - Check IN/OUT
1 - NOT BOOKED - NOT CHECKED
2 - NOT BOOKED - NOT CHECKED
3 - NOT BOOKED - NOT CHECKED
4 - NOT BOOKED - NOT CHECKED
5 - NOT BOOKED - NOT CHECKED
6 - NOT BOOKED - NOT CHECKED
7 - NOT BOOKED - NOT CHECKED
8 - NOT BOOKED - NOT CHECKED
9 - NOT BOOKED - NOT CHECKED
-----

Receptionist ID'si Olarak 123 seçildi.
5. Oda Resepsiyonist tarafından Book yapıldı
-----
ID - BOOK - Check IN/OUT - NAME/ID
1 - NOT BOOKED - NOT CHECKED
2 - NOT BOOKED - NOT CHECKED
3 - NOT BOOKED - NOT CHECKED
4 - NOT BOOKED - NOT CHECKED
5 - BOOKED - NOT CHECKED - 123
6 - NOT BOOKED - NOT CHECKED
7 - NOT BOOKED - NOT CHECKED
8 - NOT BOOKED - NOT CHECKED
9 - NOT BOOKED - NOT CHECKED
-----

5. Odanın Rezervasyon iptal edildi
-----
ID - BOOK - Check IN/OUT
1 - NOT BOOKED - NOT CHECKED
2 - NOT BOOKED - NOT CHECKED
3 - NOT BOOKED - NOT CHECKED
4 - NOT BOOKED - NOT CHECKED
5 - NOT BOOKED - NOT CHECKED
6 - NOT BOOKED - NOT CHECKED
7 - NOT BOOKED - NOT CHECKED
8 - NOT BOOKED - NOT CHECKED
9 - NOT BOOKED - NOT CHECKED
-----
ID - BOOK - Check IN/OUT
```

All files are up-to-date (2 minutes ago)

hw1 [~/IdeaProjects/hw1] - .../src/burhan/main.java [hw1] - IntelliJ IDEA

To grab input, press Ctrl+G

Debian_GTU_svy - VMware Workstation 14 Player (Non-commercial use only)

File Virtual Machine Help

hw1 [~/IdeaProjects/hw1] - .../src/burhan/main.java [hw1] - IntelliJ IDEA

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

Project hw1 src burhan main

Project hw1 src burhan main

Run Unnamed

```
5. Oda Resepsiyonist tarafından Book yapıldı
-----
ID - BOOK - Check IN/OUT - NAME/ID
1 - NOT BOOKED - NOT CHECKED
2 - NOT BOOKED - NOT CHECKED
3 - NOT BOOKED - NOT CHECKED
4 - NOT BOOKED - NOT CHECKED
5 - BOOKED - NOT CHECKED - 123
6 - NOT BOOKED - NOT CHECKED
7 - NOT BOOKED - NOT CHECKED
8 - NOT BOOKED - NOT CHECKED
9 - NOT BOOKED - NOT CHECKED
-----

5. Odanın Rezervasyon iptal edildi
-----
ID - BOOK - Check IN/OUT
1 - NOT BOOKED - NOT CHECKED
2 - NOT BOOKED - NOT CHECKED
3 - NOT BOOKED - NOT CHECKED
4 - NOT BOOKED - NOT CHECKED
5 - NOT BOOKED - NOT CHECKED
6 - NOT BOOKED - NOT CHECKED
7 - NOT BOOKED - NOT CHECKED
8 - NOT BOOKED - NOT CHECKED
9 - NOT BOOKED - NOT CHECKED
-----

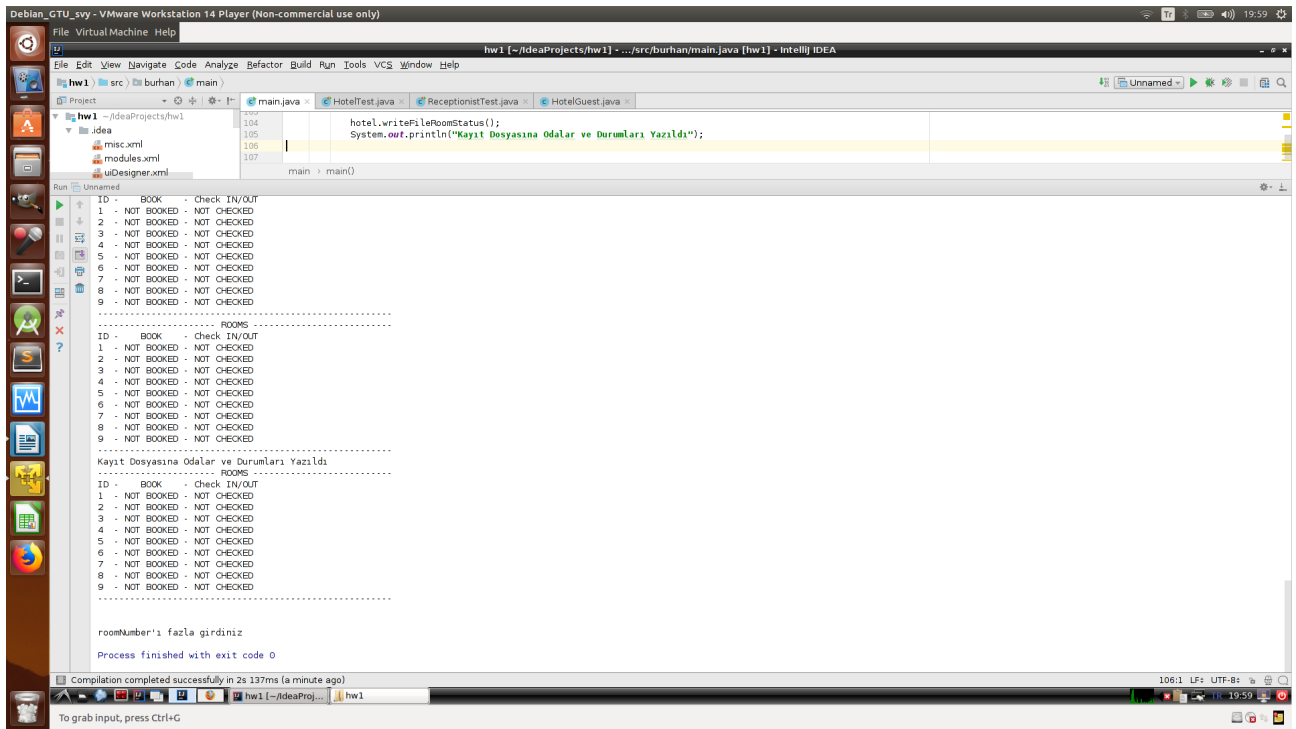
roomNumber'1 fazla girdiniz

Process finished with exit code 0
```

All files are up-to-date (2 minutes ago)

hw1 [~/IdeaProjects/hw1] - .../src/burhan/main.java [hw1] - IntelliJ IDEA

To grab input, press Ctrl+G



- Main titles -> 16pt , 2 line break
- Subtitles -> 14pt, 1.5 line break
- Paragraph -> 12pt, 1.5 line break