|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ****Manager.java****  |  | | --- | | package com.thoughtworks.tdd; | |  |  |  | | --- | |  | |  |  |  | | --- | | import java.util.ArrayList; | |  |  |  | | --- | | import java.util.List; | |  |  |  | | --- | |  | |  |  |  | | --- | | public class Manager extends ParkingBoy { | |  |  |  | | --- | | private List<ParkingBoy> parkingBoys; | |  |  |  | | --- | |  | |  |  |  | | --- | | @Override | |  |  |  | | --- | | public Ticket parkCar(Car car) { | |  |  |  | | --- | | return super.parkCar(car); | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public Ticket parkCar(ParkingBoy parkingBoy,Car car){ | |  |  |  | | --- | | return parkingBoy.parkCar(car); | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public void manageParkingBoy(ParkingBoy parkingBoy) { | |  |  |  | | --- | | parkingBoys.add(parkingBoy); | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public void manage(ParkingBoy parkingBoy, ParkingLot parkingLot) { | |  |  |  | | --- | | if (!parkingBoy.getParkingLots().contains(parkingLot)) | |  |  |  | | --- | | parkingBoy.addParkingLot(parkingLot); | |  |  |  | | --- | | if(!parkingBoys.contains(parkingBoy)) | |  |  |  | | --- | | parkingBoys.add(parkingBoy); | |  |  |  | | --- | |  | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public void addParkingLot(ParkingLot parkingLot) { | |  |  |  | | --- | | parkingLots.add(parkingLot); | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public void setManager(){ | |  |  |  | | --- | | for(ParkingLot parkingLot:parkingLots){ | |  |  |  | | --- | | parkingLot.setManager(this); | |  |  |  | | --- | | } | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public Manager(List<ParkingLot> parkingLots) { | |  |  |  | | --- | | super(parkingLots); | |  |  |  | | --- | | parkingBoys = new ArrayList<>(); | |  |  |  | | --- | | setManager(); | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public Manager(List<ParkingLot> parkingLots, List<ParkingBoy> parkingBoys) { | |  |  |  | | --- | | super(parkingLots); | |  |  |  | | --- | | this.parkingBoys = parkingBoys; | |  |  |  | | --- | | setManager(); | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |   }   |  |  | | --- | --- | |  | [MostSmartParkingBoy.java](https://github.com/yjt-1997/ParkingLot-2019-7-11-13-35-0-378/blob/master/src/main/java/com/thoughtworks/tdd/MostSmartParkingBoy.java) | | package com.thoughtworks.tdd; | | | |  | | |  |  | | --- | |  | |  |  |  | | --- | | import java.util.List; | |  |  |  | | --- | |  | |  |  |  | | --- | | public class MostSmartParkingBoy extends ParkingBoy{ | |  |  |  | | --- | |  | |  |  |  | | --- | | @Override | |  |  |  | | --- | | public Ticket parkCar(Car car) { | |  |  |  | | --- | | double biggestRemainderRate = 0; | |  |  |  | | --- | | int index = 0; | |  |  |  | | --- | | for (int i = 0; i < parkingLots.size(); i++) { | |  |  |  | | --- | | ParkingLot parkingLot = parkingLots.get(i); | |  |  |  | | --- | | double remainderRate = parkingLot.getRemainderRate(); | |  |  |  | | --- | | if (remainderRate > biggestRemainderRate) { | |  |  |  | | --- | | biggestRemainderRate = remainderRate; | |  |  |  | | --- | | index = i; | |  |  |  | | --- | | } | |  |  |  | | --- | | } | |  |  |  | | --- | | if(biggestRemainderRate==0){ | |  |  |  | | --- | | System.out.print("位置不足\n"); | |  |  |  | | --- | | return null; | |  |  |  | | --- | | } | |  |  |  | | --- | | return parkingLots.get(index).parkCar(car); | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public MostSmartParkingBoy(List<ParkingLot> parkingLots) { | |  |  |  | | --- | | super(parkingLots); | |  |  |  | | --- | | } | |  |   }   |  |  |  | | --- | --- | --- | |  | [ParkingBoy.java](https://github.com/yjt-1997/ParkingLot-2019-7-11-13-35-0-378/blob/master/src/main/java/com/thoughtworks/tdd/ParkingBoy.java) | | |  | |  | | --- | | package com.thoughtworks.tdd; | |  |  |  | | --- | |  | |  |  |  | | --- | | import java.util.ArrayList; | |  |  |  | | --- | | import java.util.List; | |  |  |  | | --- | |  | |  |  |  | | --- | | public class ParkingBoy { | |  |  |  | | --- | | protected List<ParkingLot> parkingLots; | |  |  |  | | --- | | protected List<Ticket> tickets; | |  |  |  | | --- | |  | |  |  |  | | --- | | public Ticket parkCar(Car car) { | |  |  |  | | --- | | for (ParkingLot parkingLot : parkingLots) { | |  |  |  | | --- | | if (!parkingLot.isFull()) | |  |  |  | | --- | | return parkingLot.parkCar(car); | |  |  |  | | --- | | } | |  |  |  | | --- | | System.out.print("位置不足\n"); | |  |  |  | | --- | | return null; | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public Car fetchCar(Ticket ticket) { | |  |  |  | | --- | | if (ticket == null || ticket.isUsed()) { | |  |  |  | | --- | | System.out.print("未识别的停车单\n"); | |  |  |  | | --- | | return null; | |  |  |  | | --- | | } | |  |  |  | | --- | | Car fetchCar = null; | |  |  |  | | --- | | for (ParkingLot parkingLot : parkingLots) { | |  |  |  | | --- | | if (parkingLot.getStoreCars().containsKey(ticket)) { | |  |  |  | | --- | | fetchCar = parkingLot.fetchCar(ticket); | |  |  |  | | --- | | break; | |  |  |  | | --- | | } | |  |  |  | | --- | | } | |  |  |  | | --- | | return fetchCar; | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public ParkingBoy(List<ParkingLot> parkingLots) { | |  |  |  | | --- | | this.parkingLots = parkingLots; | |  |  |  | | --- | | tickets = new ArrayList<>(); | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public void addParkingLot(ParkingLot parkingLot){ | |  |  |  | | --- | | parkingLots.add(parkingLot); | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public List<ParkingLot> getParkingLots() { | |  |  |  | | --- | | return parkingLots; | |  |  |  | | --- | | } | |  |   }  [ParkingLot.java](https://github.com/yjt-1997/ParkingLot-2019-7-11-13-35-0-378/blob/master/src/main/java/com/thoughtworks/tdd/ParkingLot.java) | |  | |  | | --- | | package com.thoughtworks.tdd; | |  |  |  | | --- | |  | |  |  |  | | --- | | import java.text.DecimalFormat; | |  |  |  | | --- | | import java.util.HashMap; | |  |  |  | | --- | | import java.util.Map; | |  |  |  | | --- | |  | |  |  |  | | --- | | public class ParkingLot { | |  |  |  | | --- | | private final int capacity; | |  |  |  | | --- | | private Map<Ticket, Car> storeCars; | |  |  |  | | --- | | private Manager manager; | |  |  |  | | --- | |  | |  |  |  | | --- | | public boolean isFull() { | |  |  |  | | --- | | return storeCars.size() >= capacity; | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public Ticket parkCar(Car car) { | |  |  |  | | --- | | if (car == null) { | |  |  |  | | --- | | System.out.print("请提供您的停车票\n"); | |  |  |  | | --- | | return null; | |  |  |  | | --- | | } | |  |  |  | | --- | | for (Ticket ticket : storeCars.keySet()) { | |  |  |  | | --- | | if (storeCars.get(ticket).equals(car)) { | |  |  |  | | --- | | System.out.print("这辆车已经停过了\n"); | |  |  |  | | --- | | return null; | |  |  |  | | --- | | } | |  |  |  | | --- | | } | |  |  |  | | --- | | Ticket ticket = new Ticket(car); | |  |  |  | | --- | | storeCars.put(ticket, car); | |  |  |  | | --- | | return ticket; | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public Car fetchCar(Ticket ticket) { | |  |  |  | | --- | | Car fetchCar = null; | |  |  |  | | --- | | if (ticket == null) { | |  |  |  | | --- | | System.out.print("未识别的停车单\n"); | |  |  |  | | --- | | return null; | |  |  |  | | --- | | } | |  |  |  | | --- | | if (!ticket.isUsed()) { | |  |  |  | | --- | | ticket.useTicket(); | |  |  |  | | --- | | fetchCar = storeCars.get(ticket); | |  |  |  | | --- | | storeCars.remove(ticket); | |  |  |  | | --- | | } else { | |  |  |  | | --- | | System.out.print("未识别的停车单\n"); | |  |  |  | | --- | | } | |  |  |  | | --- | | return fetchCar; | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public int getRemainder() { | |  |  |  | | --- | | return capacity - storeCars.size(); | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public double getRemainderRate() { | |  |  |  | | --- | | double result = 1 - storeCars.size() / (capacity \* 1.0); | |  |  |  | | --- | | return result \* 100 / 100; | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public ParkingLot() { | |  |  |  | | --- | | this(10); | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public ParkingLot(int capacity) { | |  |  |  | | --- | | this.capacity = capacity; | |  |  |  | | --- | | storeCars = new HashMap<>(capacity); | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public Map<Ticket, Car> getStoreCars() { | |  |  |  | | --- | | return storeCars; | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public Manager getManager() { | |  |  |  | | --- | | return manager; | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public void setManager(Manager manager) { | |  |  |  | | --- | | this.manager = manager; | |  |  |  | | --- | | } | |  |   }  [SmartParkingBoy.java](https://github.com/yjt-1997/ParkingLot-2019-7-11-13-35-0-378/blob/master/src/main/java/com/thoughtworks/tdd/SmartParkingBoy.java)   |  | | --- | | package com.thoughtworks.tdd; | |  |  |  | | --- | |  | |  |  |  | | --- | | import java.util.List; | |  |  |  | | --- | |  | |  |  |  | | --- | | public class SmartParkingBoy extends ParkingBoy { | |  |  |  | | --- | |  | |  |  |  | | --- | | @Override | |  |  |  | | --- | | public Ticket parkCar(Car car) { | |  |  |  | | --- | | int biggestRemainder = 0; | |  |  |  | | --- | | int index = 0; | |  |  |  | | --- | | for (int i = 0; i < parkingLots.size(); i++) { | |  |  |  | | --- | | ParkingLot parkingLot = parkingLots.get(i); | |  |  |  | | --- | | if (parkingLot.getRemainder() > biggestRemainder) { | |  |  |  | | --- | | biggestRemainder = parkingLot.getRemainder(); | |  |  |  | | --- | | index = i; | |  |  |  | | --- | | } | |  |  |  | | --- | | } | |  |  |  | | --- | | if(biggestRemainder==0){ | |  |  |  | | --- | | System.out.print("位置不足\n"); | |  |  |  | | --- | | return null; | |  |  |  | | --- | | } | |  |  |  | | --- | | return parkingLots.get(index).parkCar(car); | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public SmartParkingBoy(List<ParkingLot> parkingLots) { | |  |  |  | | --- | | super(parkingLots); | |  |  |  | | --- | | } | |  |   } |   [Ticket.java](https://github.com/yjt-1997/ParkingLot-2019-7-11-13-35-0-378/blob/master/src/main/java/com/thoughtworks/tdd/Ticket.java) |  |  |
| |  | | --- | | package com.thoughtworks.tdd; | |  |  |  | | --- | |  | |  |  |  | | --- | | import java.util.Objects; | |  |  |  | | --- | |  | |  |  |  | | --- | | public class Ticket { | |  |  |  | | --- | |  | |  |  |  | | --- | | private boolean isUsed; | |  |  |  | | --- | |  | |  |  |  | | --- | | public void useTicket() { | |  |  |  | | --- | | isUsed = true; | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public boolean isUsed() { | |  |  |  | | --- | | return isUsed; | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | public Ticket(Car car) { | |  |  |  | | --- | | isUsed = false; | |  |  |  | | --- | | } | |  |   } |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Test

[ManagerTest.java](https://github.com/yjt-1997/ParkingLot-2019-7-11-13-35-0-378/blob/master/src/test/java/com/thoughtworks/tdd/ManagerTest.java)

|  |
| --- |
| package com.thoughtworks.tdd; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| import org.junit.Before; |
|  |

|  |
| --- |
| import org.junit.Test; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| import java.io.ByteArrayOutputStream; |
|  |

|  |
| --- |
| import java.io.PrintStream; |
|  |

|  |
| --- |
| import java.util.ArrayList; |
|  |

|  |
| --- |
| import java.util.List; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| import static org.hamcrest.CoreMatchers.is; |
|  |

|  |
| --- |
| import static org.hamcrest.MatcherAssert.assertThat; |
|  |

|  |
| --- |
| import static org.hamcrest.CoreMatchers.containsString; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| import static org.junit.jupiter.api.Assertions.\*; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| public class ManagerTest { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private ByteArrayOutputStream outContent = new ByteArrayOutputStream(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Before |
|  |

|  |
| --- |
| public void setup() { |
|  |

|  |
| --- |
| System.setOut(new PrintStream(outContent)); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private String systemOut() { |
|  |

|  |
| --- |
| return outContent.toString(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Test |
|  |

|  |
| --- |
| public void should\_return\_Ticket\_when\_parkCar\_given\_Car() { |
|  |

|  |
| --- |
| //given |
|  |

|  |
| --- |
| ParkingLot parkingLot1 = new ParkingLot(); |
|  |

|  |
| --- |
| ParkingLot parkingLot2 = new ParkingLot(4); |
|  |

|  |
| --- |
| List<ParkingLot> managerParkingLots = new ArrayList<>(); |
|  |

|  |
| --- |
| managerParkingLots.add(parkingLot1); |
|  |

|  |
| --- |
| managerParkingLots.add(parkingLot2); |
|  |

|  |
| --- |
| List<ParkingLot> boyParkingLots = new ArrayList<>(); |
|  |

|  |
| --- |
| boyParkingLots.add(parkingLot1); |
|  |

|  |
| --- |
| Manager manager = new Manager(managerParkingLots); |
|  |

|  |
| --- |
| ParkingBoy parkingBoy = new ParkingBoy(boyParkingLots); |
|  |

|  |
| --- |
| manager.manageParkingBoy(parkingBoy); |
|  |

|  |
| --- |
| //when |
|  |

|  |
| --- |
| Ticket ticket1 = manager.parkCar(parkingBoy, new Car()); |
|  |

|  |
| --- |
| //then |
|  |

|  |
| --- |
| assertNotNull(ticket1); |
|  |

|  |
| --- |
| assertThat(manager.getParkingLots().get(0).getStoreCars().size(), is(1)); |
|  |

|  |
| --- |
| assertThat(manager.getParkingLots().get(1).getStoreCars().size(), is(0)); |
|  |

|  |
| --- |
| assertEquals(manager.getParkingLots().get(0), parkingBoy.getParkingLots().get(0)); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Test |
|  |

|  |
| --- |
| public void should\_return\_Ticket\_when\_manager\_parkCar\_given\_Car() { |
|  |

|  |
| --- |
| //given |
|  |

|  |
| --- |
| ParkingLot parkingLot1 = new ParkingLot(); |
|  |

|  |
| --- |
| ParkingLot parkingLot2 = new ParkingLot(4); |
|  |

|  |
| --- |
| List<ParkingLot> managerParkingLots = new ArrayList<>(); |
|  |

|  |
| --- |
| managerParkingLots.add(parkingLot1); |
|  |

|  |
| --- |
| managerParkingLots.add(parkingLot2); |
|  |

|  |
| --- |
| ; |
|  |

|  |
| --- |
| Manager manager = new Manager(managerParkingLots); |
|  |

|  |
| --- |
| //when |
|  |

|  |
| --- |
| Ticket ticket1 = manager.parkCar(new Car()); |
|  |

|  |
| --- |
| //then |
|  |

|  |
| --- |
| assertNotNull(ticket1); |
|  |

|  |
| --- |
| assertThat(manager.getParkingLots().get(0).getStoreCars().size(), is(1)); |
|  |

|  |
| --- |
| assertThat(manager.getParkingLots().get(1).getStoreCars().size(), is(0)); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Test |
|  |

|  |
| --- |
| public void should\_return\_Message\_when\_parkCar\_given\_not\_action() { |
|  |

|  |
| --- |
| //given |
|  |

|  |
| --- |
| ParkingLot parkingLot1 = new ParkingLot(); |
|  |

|  |
| --- |
| ParkingLot parkingLot2 = new ParkingLot(4); |
|  |

|  |
| --- |
| List<ParkingLot> managerParkingLots = new ArrayList<>(); |
|  |

|  |
| --- |
| managerParkingLots.add(parkingLot1); |
|  |

|  |
| --- |
| managerParkingLots.add(parkingLot2); |
|  |

|  |
| --- |
| ; |
|  |

|  |
| --- |
| Manager manager = new Manager(managerParkingLots); |
|  |

|  |
| --- |
| List<ParkingLot> boyParkingLots = new ArrayList<>(); |
|  |

|  |
| --- |
| boyParkingLots.add(parkingLot1); |
|  |

|  |
| --- |
| ParkingBoy parkingBoy = new ParkingBoy(boyParkingLots); |
|  |

|  |
| --- |
| manager.manageParkingBoy(parkingBoy); |
|  |

|  |
| --- |
| //when |
|  |

|  |
| --- |
| Ticket ticket1 = manager.parkCar(parkingBoy, null); |
|  |

|  |
| --- |
| //then |
|  |

|  |
| --- |
| assertNull(ticket1); |
|  |

|  |
| --- |
| assertNotEquals(systemOut().length(), 0); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

}

|  |  |
| --- | --- |
|  | [MostSmartParkingBoyTest.java](https://github.com/yjt-1997/ParkingLot-2019-7-11-13-35-0-378/blob/master/src/test/java/com/thoughtworks/tdd/MostSmartParkingBoyTest.java) |
| |  | | --- | | package com.thoughtworks.tdd; | |  |  |  | | --- | |  | |  |  |  | | --- | | import org.junit.Before; | |  |  |  | | --- | | import org.junit.Test; | |  |  |  | | --- | |  | |  |  |  | | --- | | import java.io.ByteArrayOutputStream; | |  |  |  | | --- | | import java.io.PrintStream; | |  |  |  | | --- | | import java.util.ArrayList; | |  |  |  | | --- | | import java.util.List; | |  |  |  | | --- | |  | |  |  |  | | --- | | import static org.hamcrest.CoreMatchers.is; | |  |  |  | | --- | | import static org.hamcrest.MatcherAssert.assertThat; | |  |  |  | | --- | | import static org.hamcrest.CoreMatchers.containsString; | |  |  |  | | --- | |  | |  |  |  | | --- | | import static org.junit.jupiter.api.Assertions.\*; | |  |  |  | | --- | |  | |  |  |  | | --- | | public class MostSmartParkingBoyTest { | |  |  |  | | --- | |  | |  |  |  | | --- | | private ByteArrayOutputStream outContent = new ByteArrayOutputStream(); | |  |  |  | | --- | |  | |  |  |  | | --- | | @Before | |  |  |  | | --- | | public void setup() { | |  |  |  | | --- | | System.setOut(new PrintStream(outContent)); | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | private String systemOut() { | |  |  |  | | --- | | return outContent.toString(); | |  |  |  | | --- | | } | |  |  |  | | --- | |  | |  |  |  | | --- | | @Test | |  |  |  | | --- | | public void should\_parkToDifferent\_lot\_when\_parkCar\_given\_several\_Car() { | |  |  |  | | --- | | //given | |  |  |  | | --- | | List<ParkingLot> parkingLots = new ArrayList<>(); | |  |  |  | | --- | | parkingLots.add(new ParkingLot()); | |  |  |  | | --- | | parkingLots.add(new ParkingLot(4)); | |  |  |  | | --- | | MostSmartParkingBoy mostSmartParkingBoy = new MostSmartParkingBoy(parkingLots); | |  |  |  | | --- | | //when | |  |  |  | | --- | | Ticket ticket1 = mostSmartParkingBoy.parkCar(new Car()); | |  |  |  | | --- | | Ticket ticket2 = mostSmartParkingBoy.parkCar(new Car()); | |  |  |  | | --- | | Ticket ticket3 = mostSmartParkingBoy.parkCar(new Car()); | |  |  |  | | --- | | //then | |  |  |  | | --- | | assertNotNull(ticket1); | |  |  |  | | --- | | assertNotNull(ticket1); | |  |  |  | | --- | | assertNotNull(ticket3); | |  |  |  | | --- | | assertThat(mostSmartParkingBoy.getParkingLots().get(0).getStoreCars().size(),is(2)); | |  |  |  | | --- | | assertThat(mostSmartParkingBoy.getParkingLots().get(1).getStoreCars().size(),is(1)); | |  |  |  | | --- | | } | |  |   } |  |

[SmartParkingBoyTest.java](https://github.com/yjt-1997/ParkingLot-2019-7-11-13-35-0-378/blob/master/src/test/java/com/thoughtworks/tdd/SmartParkingBoyTest.java)

|  |
| --- |
| package com.thoughtworks.tdd; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| import org.junit.Before; |
|  |

|  |
| --- |
| import org.junit.Test; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| import java.io.ByteArrayOutputStream; |
|  |

|  |
| --- |
| import java.io.PrintStream; |
|  |

|  |
| --- |
| import java.util.ArrayList; |
|  |

|  |
| --- |
| import java.util.List; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| import static org.hamcrest.CoreMatchers.is; |
|  |

|  |
| --- |
| import static org.hamcrest.MatcherAssert.assertThat; |
|  |

|  |
| --- |
| import static org.hamcrest.CoreMatchers.containsString; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| import static org.junit.jupiter.api.Assertions.\*; |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| public class SmartParkingBoyTest { |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private ByteArrayOutputStream outContent = new ByteArrayOutputStream(); |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Before |
|  |

|  |
| --- |
| public void setup() { |
|  |

|  |
| --- |
| System.setOut(new PrintStream(outContent)); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| private String systemOut() { |
|  |

|  |
| --- |
| return outContent.toString(); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
|  |
|  |

|  |
| --- |
| @Test |
|  |

|  |
| --- |
| public void should\_parkToDifferent\_lot\_when\_parkCar\_given\_several\_Car() { |
|  |

|  |
| --- |
| //given |
|  |

|  |
| --- |
| List<ParkingLot> parkingLots = new ArrayList<>(); |
|  |

|  |
| --- |
| parkingLots.add(new ParkingLot()); |
|  |

|  |
| --- |
| parkingLots.add(new ParkingLot(4)); |
|  |

|  |
| --- |
| SmartParkingBoy smartParkingBoy = new SmartParkingBoy(parkingLots); |
|  |

|  |
| --- |
| //when |
|  |

|  |
| --- |
| Ticket ticket1 = smartParkingBoy.parkCar(new Car()); |
|  |

|  |
| --- |
| Ticket ticket2 = smartParkingBoy.parkCar(new Car()); |
|  |

|  |
| --- |
| //then |
|  |

|  |
| --- |
| assertNotNull(ticket1); |
|  |

|  |
| --- |
| assertNotNull(ticket1); |
|  |

|  |
| --- |
| assertThat(smartParkingBoy.getParkingLots().get(0).getStoreCars().size(),is(2)); |
|  |

|  |
| --- |
| assertThat(smartParkingBoy.getParkingLots().get(1).getStoreCars().size(),is(0)); |
|  |

|  |
| --- |
| } |
|  |

|  |
| --- |
| } |