```java
public class Room {

 private String code;
 private int capacity;
 private static Room[] rooms = new Room[100];
 private static int numRooms = 0;
 /**
 * The constructor for the Room class.
 * This is a class used to represent the rooms used for COVID tests.
      *
      * @param code this is the code of the room to identify it.
      * @param capacity this is the number of assistants that can be allocated to a
     room.
      */
      public Room(String code, int capacity){
      this.code = checkCode(code);
      this.capacity = checkCapacity(capacity);
      addRooms(this);
      iterateNumRooms();
      }
      /**
      * The getter method for the room code
      *
      * @return code this is the code of the room to identify it.
      */
      public String getCode(){
      return code;
      }
      /**
      * The getter method for the capacity of the room.
      *
      * @return capacity this is the number of assistants that can be allocated to a
     room.
      */
      public int getCapacity(){
      return capacity;
      }
      /**
      * The getter method for the array of all rooms.
      *
      * @return the array of all rooms.
      */
      public static Room[] getRooms(){
      return rooms;
      }
      /**
      * The getter method for then number of rooms.
      *
      * @return the number of rooms.
```

```
*/
public static int getnumRooms(){
return numRooms;
}
/**
* The setter method for the code of the room.
*
* @param code this is the code of the room to identify it.
*/
public void setCode(String code){
this.code = checkCode(code);
}
/**
* This is the setter method for the capacity of the room.
*
* @param capacity this is the number of assistants that can be allocated to a
room.
*/
public void setCapacity(int capacity){
this.capacity = checkCapacity(capacity);
}
/**
* This method increases the number of rooms by one when called.
*/
private static void iterateNumRooms(){
numRooms += 1;
}
/**
* This method is used to add new rooms to the array of rooms.
*
* @param room this is a instance of a room.
*/
private static void addRooms(Room room){
rooms[numRooms] = room;
}
/**
* This method will check if the code is unique.
*
* @param code this is the code being tested.
* @return this code is returned only if the code is unique.
*/
private String checkCode(String code){
for (int i=0;i<numRooms;i++){
if (rooms[i].getCode().equals(code)){
throw new IllegalArgumentException("Code is not unique.");
}
}
return code;
}
/**
* This method will check the capacity is greater than zero.
*
* @param capacity this is the number of assistants that can be allocated to a
room.
* @return capacity this is the number of assistants that can be allocated to a
room.
*/
private int checkCapacity(int capacity){
if (capacity < 0){
throw new IllegalArgumentException("Capacity should be greater than
zero.");
}
return capacity;
}
/**
* This method will return a string of all the room's parameters.
*
* @return a string of all the room's parameters
*/
public static String toStringAll(){
String allRooms = "Rooms-\n";
for (int i=0;i<numRooms;i++){
allRooms = allRooms.concat((i+11)+". "+rooms[i].toString()+"\n");
}
return allRooms;
}
/**
* This method will return the string of a room's parameters.
*
* @return a string of a room's parameters.
*/
public String toString(){
return "| "+code+" | capacity: "+capacity+" |";
}
}
```