Test fixture:

package hotelreservation.entities;

import hotelreservation.controllers.\*;

import org.junit.Before;

import org.junit.After;

import static org.junit.Assert.\*;

public class HotelTest {

private HotelControllerMock hotelController;

@Before

public void setUp() {

hotelController = HotelControllerMock.getInstance();

}

@After

public void tearDown() {

hotelController = null;

}

@Test

public void testGetId() {

assertEquals((Integer) 1, hotelController.getHotelByID(1).getId());

assertEquals((Integer) 2, hotelController.getHotelByID(2).getId());

assertEquals((Integer) 3, hotelController.getHotelByID(3).getId());

}

@Test

public void testGetName() {

assertEquals("Rangá", hotelController.getHotelByID(1).getName());

assertEquals("Hilton", hotelController.getHotelByID(2).getName());

assertEquals("Ölfus", hotelController.getHotelByID(3).getName());

}

@Test

public void testGetPriceRating() {

assertEquals((Integer) 3, hotelController.getHotelByID(1).getPriceRating());

assertEquals((Integer) 4, hotelController.getHotelByID(2).getPriceRating());

assertEquals((Integer) 5, hotelController.getHotelByID(3).getPriceRating());

}

}

Result of test fixture:

A screenshot of a computer

Description automatically generated with medium confidence

Mock object:

Mock SQL connection:

package hotelreservation.data;

import java.util.HashMap;

import hotelreservation.entities.\*;

/\*\*

\* Simulates having a connection to our future SQL database

\*/

public class MockConnection implements HotelDataConnection {

private static MockConnection instance = null;

private HashMap<Integer, Hotel> hotels = new HashMap<Integer, Hotel>();

/\*\*

\* private constructor that creates three mock hotels

\*/

private MockConnection(){

hotels.put(1, new Hotel(1, "Rangá", 2, 3));

hotels.put(2, new Hotel(2, "Hilton", 3, 4));

hotels.put(3, new Hotel(3, "Ölfus", 4, 5));

}

/\*\*

\* Returns the instance of the MockConnection

\* @return the MockConnection instance

\*/

public static MockConnection getInstance(){

if(instance == null){

instance = new MockConnection();

}

return instance;

}

/\*\*

\* Returns the Hotel associated with the given ID

\* @param ID: the id

\* @return the MockConnection instance

\*/

public Hotel getHotelByID(Integer ID) {

return hotels.get(ID);

}

/\*\*

\* Implemented later

\* @return null

\*/

public Room getRoomByID(Integer ID) {

return null;

}

}

Hotel object:

package hotelreservation.entities;

import java.util.ArrayList;

public class Hotel {

private Integer id;

private String name;

private Integer starRating;

private Integer priceRating;

private ArrayList<Room> rooms;

public Hotel(Integer id, String name, Integer starRating, Integer priceRating) {

this.id = id;

this.name = name;

this.starRating = starRating;

this.priceRating = priceRating;

this.rooms = null;

}

public void addRoom(Room newRoom) {

rooms.add(newRoom);

}

public Integer getId() {

return this.id;

}

public void setId(Integer id) {

this.id = id;

}

public String getName() {

return this.name;

}

public void setName(String name) {

this.name = name;

}

public Integer getStarRating() {

return this.starRating;

}

public void setStarRating(Integer starRating) {

this.starRating = starRating;

}

public Integer getPriceRating() {

return this.priceRating;

}

public void setPriceRating(Integer priceRating) {

this.priceRating = priceRating;

}

public ArrayList<Room> getRooms() {

return this.rooms;

}

}

For additional code please refer to github repository: <https://github.com/Bobbi11/HotelReservation>

or contact us by email.