Testy Klasy Reservations

# Klasa Reservations

package test2;

import java.util.ArrayList;

import java.util.Iterator;

import test1.User;

public class Reservations{

    private ArrayList<Reservation> reservationList;

    public ArrayList<Reservation> getReservationListByUser(User user){

        ArrayList<Reservation> reservationsOfUser = new ArrayList<Reservation>();

        Iterator<Reservation> itr = null;

        itr = this.reservationList.listIterator(); //stworzenie iteratora listy reservationList

        //przesuzkiwanie przez iteratra

        while (itr.hasNext()) {

            Reservation reservation = itr.next(); //pobieranie następnego elementu z listy

            if (reservation.getUserName().equals(user.getName())) {

                reservationsOfUser.add(reservation);

            }

        }

        return reservationsOfUser;

    }

    public void setReservationList(ArrayList<Reservation> reservationsOfUser) {

        this.reservationList = reservationsOfUser;

    }

}

# Klasa Reservation

package test2;

public class Reservation {

    String userName;

    public String getUserName() {

        return this.userName;

    }

    public void setUserName(String name) {

        this.userName = name;

    }

}

# Test nr 1

package test2;

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

import java.util.ArrayList;

import org.junit.jupiter.api.Test;

import test1.User;

class getReservationListOfUser {

    @Test

    void testGetReservationListByUser() {

        Reservations reservations = new Reservations();

        ArrayList<Reservation> reservationsOfUser = new ArrayList<Reservation>();

        ArrayList<Reservation> allReservations = new ArrayList<Reservation>();

        //tworzymy przykładowe rezerwacje

        Reservation reservation1  = new Reservation();

        reservation1.setUserName("name1");

        Reservation reservation  = new Reservation();

        reservation.setUserName("name3");

        Reservation reservation2  = new Reservation();

        reservation2.setUserName("name2");

        reservationsOfUser.add(reservation);

        allReservations.add(reservation);

        allReservations.add(reservation2);

        allReservations.add(reservation2);

        reservations.setReservationList(allReservations);

        User user = new User();

        user.setName("name");

        ArrayList<Reservation> result = reservations.getReservationListByUser(user);

        assertEquals(reservationsOfUser, result);

    }

}

# Test nr 2

Sprawdzamy działanie gdy użytkownik nie posiada rezerwacji;

    @Test

    void testGetReservationListByUser1() {

        Reservations reservations = new Reservations();

        ArrayList<Reservation> reservationsOfUser = new ArrayList<Reservation>();

        ArrayList<Reservation> allReservations = new ArrayList<Reservation>();

        //tworzymy przykładowe rezerwacje

        Reservation reservation1  = new Reservation();

        reservation1.setUserName("name1");

        Reservation reservation  = new Reservation();

        reservation.setUserName("name3");

        Reservation reservation2  = new Reservation();

        reservation2.setUserName("name2");

        allReservations.add(reservation);

        allReservations.add(reservation2);

        allReservations.add(reservation2);

        reservations.setReservationList(allReservations);

        User user = new User();

        user.setName("name");

        ArrayList<Reservation> result = reservations.getReservationListByUser(user);

        assertEquals(reservationsOfUser, result);

    }

# Test 3

Gdy nie ma rezerwacji w rezervationList

    @Test

    void testGetReservationListByUser2() {

        Reservations reservations = new Reservations();

        ArrayList<Reservation> reservationsOfUser = new ArrayList<Reservation>();

        ArrayList<Reservation> allReservations = new ArrayList<Reservation>();

        //tworzymy przykładowe rezerwacje

        Reservation reservation1  = new Reservation();

        reservation1.setUserName("name1");

        Reservation reservation  = new Reservation();

        reservation.setUserName("name3");

        Reservation reservation2  = new Reservation();

        reservation2.setUserName("name2");

        reservations.setReservationList(allReservations);

        User user = new User();

        user.setName("name");

        ArrayList<Reservation> result = reservations.getReservationListByUser(user);

        assertEquals(reservationsOfUser, result);

    }

Testy Klasy User

# Klasa User

package test1;

import java.awt.List;

public class User{

    private List reservationList;

    private String name;

    private String surname;

    public List getReservationList() {

        return this.reservationList;

    }

    public void setReservationList(List reservationList){

        this.reservationList = reservationList;

    }

    public String getName() {  return this.name; }

    public void setName(String name) {

        this.name = name;

    }

    public String getSurname() {

        return this.surname;

    }

    public void setSurname(String surname) {

        this.surname = surname;

    }

}

# Klasa Testująca metodę

package test1;

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

import java.awt.List;

import org.junit.jupiter.api.Test;

import org.junit.Assert;;

class GetReservationListTest {

    @Test

    void testGetReservationList() {

        User test = new User();

        List reservationList = new List();

        reservationList.add("Rezerwacja");

        test.setReservationList(reservationList);

        List result = test.getReservationList();

        assertEquals(reservationList, result);

    }

}