**Appointment.java:**

import java.util.Date;

public class Appointment {

private String id;

private Date date;

private String details;

public Appointment(String id, Date date, String details) {

this.id = id;

this.date = date;

this.details = details;

}

// Getters

public String getId() {

return id;

}

public Date getDate() {

return date;

}

public String getDetails() {

return details;

}

}

**AppointmentService.java:**

import java.util.Date;

import java.util.HashMap;

import java.util.Map;

public class AppointmentService {

private Map<String, Appointment> appointments = new HashMap<>();

public void addAppointment(Appointment appointment) {

appointments.put(appointment.getId(), appointment);

}

public void deleteAppointment(String id) {

appointments.remove(id);

}

public void updateAppointment(Appointment updatedAppointment) {

appointments.put(updatedAppointment.getId(), updatedAppointment);

}

public Appointment getAppointmentById(String id) {

return appointments.get(id);

}

}

**AppointmentTest.java**

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

import org.junit.jupiter.api.Test;

import java.util.Date;

public class AppointmentTest {

@Test

public void testAppointmentCreation() {

Date currentDate = new Date();

Appointment appointment = new Appointment("A1", currentDate, "Meeting with client");

assertEquals("A1", appointment.getId());

assertEquals(currentDate, appointment.getDate());

assertEquals("Meeting with client", appointment.getDetails());

}

}

**AppointmentServiceTest.java**

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

import org.junit.jupiter.api.Test;

import java.util.Date;

public class AppointmentServiceTest {

@Test

public void testAddAppointment() {

AppointmentService appointmentService = new AppointmentService();

Date currentDate = new Date();

Appointment appointment = new Appointment("A1", currentDate, "Meeting with client");

appointmentService.addAppointment(appointment);

assertNotNull(appointmentService.getAppointmentById("A1"));

}

@Test

public void testDeleteAppointment() {

AppointmentService appointmentService = new AppointmentService();

Date currentDate = new Date();

Appointment appointment = new Appointment("A1", currentDate, "Meeting with client");

appointmentService.addAppointment(appointment);

appointmentService.deleteAppointment("A1");

assertNull(appointmentService.getAppointmentById("A1"));

}

@Test

public void testUpdateAppointment() {

AppointmentService appointmentService = new AppointmentService();

Date currentDate = new Date();

Appointment appointment = new Appointment("A1", currentDate, "Meeting with client");

appointmentService.addAppointment(appointment);

Appointment updatedAppointment = new Appointment("A1", currentDate, "Updated meeting");

appointmentService.updateAppointment(updatedAppointment);

assertEquals("Updated meeting", appointmentService.getAppointmentById("A1").getDetails());

}

@Test

public void testGetAppointmentById() {

AppointmentService appointmentService = new AppointmentService();

Date currentDate = new Date();

Appointment appointment = new Appointment("A1", currentDate, "Meeting with client");

appointmentService.addAppointment(appointment);

Appointment fetchedAppointment = appointmentService.getAppointmentById("A1");

assertNotNull(fetchedAppointment);

assertEquals("A1", fetchedAppointment.getId());

assertEquals(currentDate, fetchedAppointment.getDate());

assertEquals("Meeting with client", fetchedAppointment.getDetails());

}

}

**Contact.java**

public class Contact {

private String id;

private String firstName;

private String lastName;

private String phone;

private String address;

public Contact(String id, String firstName, String lastName, String phone, String address) {

this.id = id;

this.firstName = firstName;

this.lastName = lastName;

this.phone = phone;

this.address = address;

}

// Getters

public String getId() {

return id;

}

public String getFirstName() {

return firstName;

}

public String getLastName() {

return lastName;

}

public String getPhone() {

return phone;

}

public String getAddress() {

return address;

}

}

**ContactService.java**

import java.util.HashMap;

import java.util.Map;

public class ContactService {

private Map<String, Contact> contacts = new HashMap<>();

public void addContact(Contact contact) {

contacts.put(contact.getId(), contact);

}

public void deleteContact(String id) {

contacts.remove(id);

}

public void updateContact(Contact updatedContact) {

contacts.put(updatedContact.getId(), updatedContact);

}

public Contact getContactById(String id) {

return contacts.get(id);

}

}

**ContactTest.java**

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

import org.junit.jupiter.api.Test;

public class ContactTest {

@Test

public void testContactCreation() {

Contact contact = new Contact("C1", "John", "Doe", "1234567890", "123 Main St");

assertEquals("C1", contact.getId());

assertEquals("John", contact.getFirstName());

assertEquals("Doe", contact.getLastName());

assertEquals("1234567890", contact.getPhone());

assertEquals("123 Main St", contact.getAddress());

}

}

**ContactServiceTest.java**

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

import org.junit.jupiter.api.Test;

public class ContactServiceTest {

@Test

public void testAddContact() {

ContactService contactService = new ContactService();

Contact contact = new Contact("C1", "John", "Doe", "1234567890", "123 Main St");

contactService.addContact(contact);

assertNotNull(contactService.getContactById("C1"));

}

@Test

public void testDeleteContact() {

ContactService contactService = new ContactService();

Contact contact = new Contact("C1", "John", "Doe", "1234567890", "123 Main St");

contactService.addContact(contact);

contactService.deleteContact("C1");

assertNull(contactService.getContactById("C1"));

}

@Test

public void testUpdateContact() {

ContactService contactService = new ContactService();

Contact contact = new Contact("C1", "John", "Doe", "1234567890", "123 Main St");

contactService.addContact(contact);

Contact updatedContact = new Contact("C1", "John", "Doe", "0987654321", "456 Elm St");

contactService.updateContact(updatedContact);

Contact retrievedContact = contactService.getContactById("C1");

assertEquals("0987654321", retrievedContact.getPhone());

assertEquals("456 Elm St", retrievedContact.getAddress());

}

@Test

public void testGetContactById() {

ContactService contactService = new ContactService();

Contact contact = new Contact("C1", "John", "Doe", "1234567890", "123 Main St");

contactService.addContact(contact);

Contact retrievedContact = contactService.getContactById("C1");

assertNotNull(retrievedContact);

assertEquals("C1", retrievedContact.getId());

assertEquals("John", retrievedContact.getFirstName());

assertEquals("Doe", retrievedContact.getLastName());

assertEquals("1234567890", retrievedContact.getPhone());

assertEquals("123 Main St", retrievedContact.getAddress());

}

}

**Task.java**

public class Task {

private String id;

private String name;

private String description;

public Task(String id, String name, String description) {

this.id = id;

this.name = name;

this.description = description;

}

// Getters

public String getId() {

return id;

}

public String getName() {

return name;

}

public String getDescription() {

return description;

}

}

**TaskService.java**

import java.util.HashMap;

import java.util.Map;

public class TaskService {

private Map<String, Task> tasks = new HashMap<>();

public void addTask(Task task) {

tasks.put(task.getId(), task);

}

public void deleteTask(String id) {

tasks.remove(id);

}

public void updateTask(Task updatedTask) {

tasks.put(updatedTask.getId(), updatedTask);

}

public Task getTaskById(String id) {

return tasks.get(id);

}

}

**TaskTest.java**

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

import org.junit.jupiter.api.Test;

public class TaskTest {

@Test

public void testTaskCreation() {

Task task = new Task("T1", "Complete Assignment", "Finish coding and testing");

assertEquals("T1", task.getId());

assertEquals("Complete Assignment", task.getName());

assertEquals("Finish coding and testing", task.getDescription());

}

}

**TaskServiceTest.java**

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

import org.junit.jupiter.api.Test;

public class TaskServiceTest {

@Test

public void testAddTask() {

TaskService taskService = new TaskService();

Task task = new Task("T1", "Complete Assignment", "Finish coding and testing");

taskService.addTask(task);

assertNotNull(taskService.getTaskById("T1"));

}

@Test

public void testDeleteTask() {

TaskService taskService = new TaskService();

Task task = new Task("T1", "Complete Assignment", "Finish coding and testing");

taskService.addTask(task);

taskService.deleteTask("T1");

assertNull(taskService.getTaskById("T1"));

}

@Test

public void testUpdateTask() {

TaskService taskService = new TaskService();

Task task = new Task("T1", "Complete Assignment", "Finish coding and testing");

taskService.addTask(task);

Task updatedTask = new Task("T1", "Complete Assignment", "Finish coding, testing, and documentation");

taskService.updateTask(updatedTask);

Task retrievedTask = taskService.getTaskById("T1");

assertEquals("Finish coding, testing, and documentation", retrievedTask.getDescription());

}

@Test

public void testGetTaskById() {

TaskService taskService = new TaskService();

Task task = new Task("T1", "Complete Assignment", "Finish coding and testing");

taskService.addTask(task);

Task retrievedTask = taskService.getTaskById("T1");

assertNotNull(retrievedTask);

assertEquals("T1", retrievedTask.getId());

assertEquals("Complete Assignment", retrievedTask.getName());

assertEquals("Finish coding and testing", retrievedTask.getDescription());

}

}