

## Admin Test:

The screenshot shows an IDE with the `AdminTest.java` file open. The code is as follows:

```
1 package org.medicare.test;
2
3 import static org.assertj.core.api.Assertions.assertThat;
4
5 @SpringBootTest
6 class AdminTest {
7     @Autowired
8     AdminRepository adminRepository;
9
10    @Test
11    public void insertAdmin() {
12        Admin A = new Admin();
13        A.setAdmin("abc1");
14        A.setPassword("1234@");
15        adminRepository.save(A);
16        assertThat(adminRepository.findById(3).get());
17    }
18
19    @Test
20    public void adminAuth() {
21        Admin A = adminRepository.findByAdminAndPassword("abc1", "1234@");
22        assertEquals("abc1", A.getAdmin());
23        assertEquals("1234@", A.getPassword());
24    }
25 }
```

The left sidebar shows the test results for `AdminTest` (Runner: JUnit 5) [0.359 s]. The tests `insertAdmin()` [0.249 s] and `adminAuth()` [0.109 s] both passed. The top status bar indicates "Finished after 4.445 seconds", "Runs: 2/2", "Errors: 0", and "Failures: 0".

## Cart Test:

The screenshot shows an IDE with the `CartTest.java` file open. The code is as follows:

```
1 package org.medicare.test;
2
3 import static org.assertj.core.api.Assertions.assertThat;
4
5 @SpringBootTest
6 class CartTest {
7     @Autowired
8     CartRepository cartRepository;
9
10    @Test
11    public void insertMedicines() {
12        Cart C = new Cart();
13        C.setMedicineId(1);
14        C.setMedicineName("dolo");
15        C.setPrice(27);
16        C.setDescription("headache");
17        cartRepository.save(C);
18        assertThat(cartRepository.findById(1).get());
19    }
20
21    @Test
22    public void showAllMedicines() {
23        List<Cart> list = cartRepository.findAll();
24        assertThat(list.size().isGreaterThan(0));
25    }
26
27    @Test
28    public void deleteMedicine() {
29        cartRepository.deleteByid(1);
30        assertThat(cartRepository.existsByid(1)).isFalse();
31    }
32 }
```

The left sidebar shows the test results for `CartTest` (Runner: JUnit 5) [0.507 s]. The tests `insertMedicines()` [0.295 s], `deleteMedicine()` [0.180 s], and `showAllMedicines()` [0.030 s] all passed. The top status bar indicates "Finished after 4.815 seconds", "Runs: 3/3", "Errors: 0", and "Failures: 0".

## Contact UsTest:

The screenshot shows an IDE with the `ContactUsTest.java` file open. The test class is annotated with `@SpringBootTest` and `@Autowired`. It contains a single test method `insertContactUs()` that creates a `ContactUs` object, sets its properties, saves it to the repository, and asserts that it can be found by ID. The test runner shows that the test passed successfully after 0.263 seconds.

```
1 package org.medicare.test;
2
3 import static org.junit.jupiter.api.Assertions.*;
4
5
6
7
8
9
10
11 @SpringBootTest
12 class ContactUsTest {
13     @Autowired
14     ContactUsRepository contactusrepository;
15
16     @Test
17     public void insertContactUs() {
18         ContactUs CU = new ContactUs();
19         //CU.setCid(1);
20         CU.setName("harinihsc");
21         CU.setEmailId("@gmail.com");
22         CU.setContactNo(1234567891);
23         CU.setDescription("NO Description");
24         contactusrepository.save(CU);
25         assertNotNull(contactusrepository.findById(2).get());
26     }
27
28
29
30 }
```

Finished after 4.631 seconds  
Runs: 1/1 Errors: 0 Failures: 0

contactUsTest (Runner: JUnit 5) (0.263 s)  
insertContactUs() (0.263 s)

## Medicine Test:

The screenshot shows an IDE with the `MedicineTest.java` file open. The test class contains several test methods: `contextLoads()`, `insertMedicines()`, `showAllMedicines()`, `getMedicineById()`, `deleteMedicine()`, and `updateMedicine()`. Each method tests a specific functionality of the `MedicineRepository`. The test runner shows that all tests passed successfully after 4.839 seconds.

```
19 /*@Test
20 void contextLoads() {
21 }*/
22
23 @Test
24 public void insertMedicines() {
25     Medicines M = new Medicines();
26     //M.setMedicineId(99);
27     M.setMedicineName("asclio");
28     M.setManufactureDate("22-02-2022");
29     M.setType("tablet");
30     M.setPrice(27);
31     M.setDescription("For headache and body pain");
32     M.setExpiryDate("25-02-2024");
33     M.setStatus("available");
34     M.setSeller("medicare");
35     medicinerespository.save(M);
36     assertNotNull(medicinerespository.findById(4).get());
37 }
38
39 @Test
40 public void showAllMedicines() {
41     List<Medicines> list = medicinerespository.findAll();
42     assertTrue(list.size() > 0);
43 }
44
45 @Test
46 public void getMedicineById() {
47     Medicines M = medicinerespository.findById(4);
48     assertEquals(4, M.getMedicineId());
49 }
50
51
52 @Test
53 public void deleteMedicine() {
54     medicinerespository.deleteById(3);
55     assertTrue(medicinerespository.existsById(3)).isFalse();
56 }
57
58 @Test
59 public void updateMedicine() {
60     Medicines M = medicinerespository.findById(5);
61     M.setPrice(20);
62     medicinerespository.save(M);
63     assertEquals(500, medicinerespository.findById(5).getPrice());
64 }
65 }
```

Finished after 4.839 seconds  
Runs: 5/5 Errors: 0 Failures: 0

MedicineTest (Runner: JUnit 5) (0.567 s)  
updateMedicine() (0.424 s)  
insertMedicines() (0.038 s)  
getMedicineById() (0.010 s)  
deleteMedicine() (0.067 s)  
showAllMedicines() (0.025 s)

## UserTest:

The screenshot shows the Eclipse IDE with the `UserTest.java` file open. The file is located at `Backend/src/test/java/org/medicare/test/UserTest.java`. The code defines a `UserTest` class with two test methods: `insertUser` and `userAuth`. The `insertUser` method creates a `User` object, sets its attributes, saves it to the `userRepository`, and asserts that it was found. The `userAuth` method finds a user by username and password and asserts that the retrieved user's details match the input.

```
1 package org.medicare.test;
2
3 import static org.junit.jupiter.api.Assertions.*;
4
5
6
7
8
9
10
11 @SpringBootTest
12 class UserTest {
13     @Autowired
14     UserRepository userRepository;
15
16     @Test
17     public void insertUser() {
18         User U = new User();
19         //U.setUserId(1);
20         U.setFirst_name("dhoni");
21         U.setLast_name("MS");
22         U.setMobile_no("741852963");
23         U.setAge(40);
24         U.setUsername("Dhoni");
25         U.setPassword("#123");
26         U.setGender("Male");
27         userRepository.save(U);
28         assertNotNull(userRepository.findById(1).get());
29     }
30
31     @Test
32     public void userAuth() {
33         User U = userRepository.findByUsernameAndPassword("Haru", "Haru@123");
34         assertEquals("Haru", U.getUsername());
35         assertEquals("Haru@123", U.getPassword());
36     }
37 }
38
```

The left sidebar shows the Package Explorer with the `UserTest` class selected. The bottom status bar indicates that the tests passed successfully: `Runs: 2/2`, `Errors: 0`, and `Failures: 0`. The `Failure Trace` panel is empty.