ARI FULLAH

B2 2023

POSTTEST 4

Main.java

```java
import java.util.Scanner;

public class Main {
    Run | Debug
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        UserAuth userAuth = new UserAuth();

        while (true) {
            System.out.println(x:"\n=== Selamat Datang di Gym Fitness Jaya ===");
            System.out.println(x:"1. Login");
            System.out.println(x:"2. Registrasi");
            System.out.println(x:"3. Keluar");
            System.out.print(s:"Pilih menu: ");

            int choice = scanner.nextInt();
            scanner.nextLine();

            if (choice == 1) {
                System.out.print(s:"Username: ");
                String username = scanner.nextLine();
                System.out.print(s:"Password: ");
                String password = scanner.nextLine();

                User user = userAuth.login(username, password);
                if (user == null) {
                    System.out.println(x:"Login gagal! Username atau password salah.");
                } else {
                    if (user.getRole().equals(anObject:"admin")) {
                        adminMenu(scanner, user);
                    } else {
                        memberMenu(scanner, user);
                    }
                }
            } else if (choice == 2) {
                System.out.print(s:"Masukkan username baru: ");
                String username = scanner.nextLine();
                System.out.print(s:"Masukkan password baru: ");
                String password = scanner.nextLine();
```

```
                if (userAuth.register(username, password)) {
                    System.out.println(x:"Registrasi berhasil! Silakan login.");
                } else {
                    System.out.println(x:"Username sudah digunakan. Pilih username lain.");
                }
            } else if (choice == 3) {
                System.out.println(x:"Terima kasih telah menggunakan sistem.");
                scanner.close();
                return;
            } else {
                System.out.println(x:"Pilihan tidak valid.");
            }
        }
    }

    public static void adminMenu(Scanner scanner, User user) {
        Admin admin = (Admin) user;
        while (true) {
            System.out.println(x:"\n=== Menu Admin ===");
            System.out.println(x:"1. Tambah Jadwal Latihan");
            System.out.println(x:"2. Lihat Anggota");
            System.out.println(x:"3. Lihat Semua Jadwal");
            System.out.println(x:"4. Logout");
            System.out.print(s:"Pilih menu: ");

            int choice = scanner.nextInt();
            scanner.nextLine();

            if (choice == 1) {
                System.out.print(s:"Masukkan nama member: ");
                String memberName = scanner.nextLine();
                System.out.print(s:"Jenis Latihan: ");
                String workoutType = scanner.nextLine();
                System.out.print(s:"Tanggal Latihan (YYYY-MM-DD): ");
                String scheduleDate = scanner.nextLine();
```

```
                // Panggil metode overload (bisa pilih salah satu)
                admin.addSchedule(memberName, workoutType, scheduleDate);
                // admin.addSchedule(new Schedule(memberName, workoutType, scheduleDate));
            } else if (choice == 2) {
                admin.viewMembers();
            } else if (choice == 3) {
                admin.viewSchedule(); // override
            } else if (choice == 4) {
                return;
            } else {
                System.out.println(x:"Pilihan tidak valid.");
            }
        }
    }

    public static void memberMenu(Scanner scanner, User user) {
        user.viewSchedule(); // polymorphism (override akan otomatis jalan)
    }
}
```

Admin.java

```java
public class Admin extends User {

    public Admin(String fullName, String address, String phoneNumber, String username, String password) {
        super(fullName, address, phoneNumber, username, password, role:"admin");
    }

    // Overloading
    public void addSchedule(String memberName, String workoutType, String scheduleDate) {
        System.out.println("Admin " + getUsername() + " menambahkan jadwal latihan untuk " + memberName);
    }

    public void addSchedule(Schedule schedule) {
        System.out.println("Admin " + getUsername() + " menambahkan jadwal: " +
            schedule.getWorkoutType() + " untuk " + schedule.getMemberName() +
            " pada " + schedule.getScheduleDate());
    }

    @Override
    public void viewSchedule() {
        System.out.println("Admin " + getUsername() + " melihat semua jadwal anggota.");
    }

    public void viewMembers() {
        System.out.println(x:"Viewing all members...");
    }
}
```

User.java

```java
public class User extends Person {
    private String username;
    private String password;
    private String role; // "admin" or "member"

    public User(String fullName, String address, String phoneNumber, String username, String password, String role) {
        super(fullName, address, phoneNumber);
        this.username = username;
        this.password = password;
        this.role = role;
    }

    // Getter and Setter
    public String getUsername() {
        return username;
    }

    public void setUsername(String username) {
        this.username = username;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }

    public String getRole() {
        return role;
    }

    public void setRole(String role) {
        this.role = role;
    }

    // Polymorphism (Override)
    public void viewSchedule() {
        System.out.println("User " + getUsername() + " belum memiliki jadwal spesifik.");
    }
}
```

Member.java

```java
public class Member extends User {

    public Member(String fullName, String address, String phoneNumber, String username, String password) {
        super(fullName, address, phoneNumber, username, password, role:"member");
    }

    @Override
    public void viewSchedule() {
        System.out.println("Member " + getUsername() + " melihat jadwal latihan pribadinya.");
    }
}
```

MemberCRUD.java

```java
import java.util.ArrayList;

public class MemberCRUD {
    private ArrayList<Member> members = new ArrayList<>();

    // Menambahkan anggota baru
    public void addMember(String username, String fullName, String address, String phoneNumber) {
        members.add(new Member(fullName, address, phoneNumber, username, password:"defaultPassword"));
        System.out.println("Anggota " + username + " berhasil ditambahkan.");
    }

    // Menampilkan semua anggota
    public void showMembers() {
        if (members.isEmpty()) {
            System.out.println(x:"Belum ada anggota yang terdaftar.");
            return;
        }
        for (Member member : members) {
            System.out.println(member);
        }
    }

    // Getter untuk mengambil daftar anggota
    public ArrayList<Member> getMembers() {
        return members;
    }
}
```

Person.java

```java
public class Person {
    private String fullName;
    private String address;
    private String phoneNumber;

    public Person(String fullName, String address, String phoneNumber) {
        this.fullName = fullName;
        this.address = address;
        this.phoneNumber = phoneNumber;
    }

    // Getter and Setter
    public String getFullName() {
        return fullName;
    }

    public void setFullName(String fullName) {
        this.fullName = fullName;
    }

    public String getAddress() {
        return address;
    }

    public void setAddress(String address) {
        this.address = address;
    }

    public String getPhoneNumber() {
        return phoneNumber;
    }

    public void setPhoneNumber(String phoneNumber) {
        this.phoneNumber = phoneNumber;
    }
}
```

Schedule.java

```java
public class Schedule {
    private String memberName;
    private String workoutType;
    private String scheduleDate;

    public Schedule(String memberName, String workoutType, String scheduleDate) {
        this.memberName = memberName;
        this.workoutType = workoutType;
        this.scheduleDate = scheduleDate;
    }

    // Getter and Setter
    public String getMemberName() {
        return memberName;
    }

    public void setMemberName(String memberName) {
        this.memberName = memberName;
    }

    public String getWorkoutType() {
        return workoutType;
    }

    public void setWorkoutType(String workoutType) {
        this.workoutType = workoutType;
    }

    public String getScheduleDate() {
        return scheduleDate;
    }

    public void setScheduleDate(String scheduleDate) {
        this.scheduleDate = scheduleDate;
    }

    @Override
    public String toString() {
        return "Member: " + memberName + " | Latihan: " + workoutType + " | Tanggal: " + scheduleDate;
    }
}
```

ScheduleCRUD.java

```java
import java.util.ArrayList;

public class ScheduleCRUD {
    private ArrayList<Schedule> schedules = new ArrayList<>();

    public void addSchedule(String memberName, String workoutType, String scheduleDate) {
        schedules.add(new Schedule(memberName, workoutType, scheduleDate));
        System.out.println("Jadwal latihan untuk " + memberName + " berhasil ditambahkan!");
    }

    public void showSchedules() {
        if (schedules.isEmpty()) {
            System.out.println(x:"Belum ada jadwal latihan yang terdaftar.");
            return;
        }
        for (Schedule schedule : schedules) {
            System.out.println(schedule);
        }
    }

    public void deleteSchedule(int index) {
        if (index >= 0 && index < schedules.size()) {
            System.out.println("Jadwal latihan untuk " + schedules.get(index).getMemberName() + " berhasil dihapus.");
            schedules.remove(index);
        } else {
            System.out.println(x:"Indeks tidak valid.");
        }
    }

    // Getter untuk mengambil daftar jadwal
    public ArrayList<Schedule> getSchedules() {
        return schedules;
    }
}
```

UseraAuth.java

```java
import java.util.ArrayList;

public class UserAuth {
    protected ArrayList<User> users = new ArrayList<>(); // Protected agar bisa diwarisi

    public UserAuth() {
        users.add(new Admin(fullName:"Admin", address:"Admin Address", phoneNumber:"0000", username:"admin", passw…"123"));
    }

    public boolean register(String username, String password) {
        for (User user : users) {
            if (user.getUsername().equals(username)) {
                return false; // Username sudah ada
            }
        }
        users.add(new Member(fullName:"Member", address:"Member Address", phoneNumber:"1111", username, password));
        return true;
    }

    public User login(String username, String password) {
        for (User user : users) {
            if (user.getUsername().equals(username) && user.getPassword().equals(password)) {
                return user;
            }
        }
        return null; // Login gagal
    }
}
```