

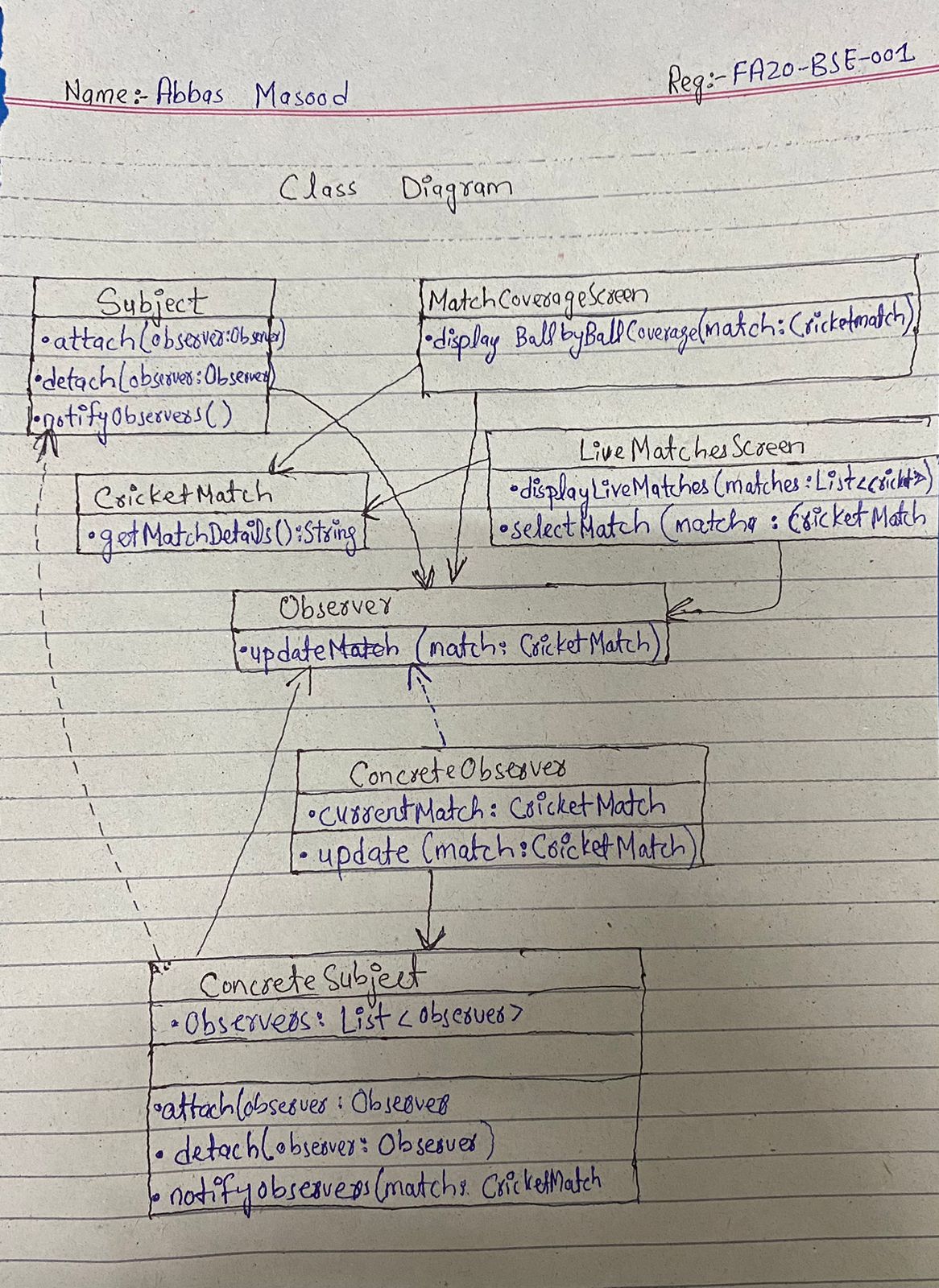
**NAME: - ABBAS MASOOD**

**REG: - FA20-BSE-001**

**LAB: - MID LAB**

**SUBJECT: - Design Patterns**

**Class Diagram: -**



**CODE: -**

import java.util.ArrayList;

import java.util.List;

interface Subject {

void attach(Observer observer);

void detach(Observer observer);

void notifyObservers();

}

class CricketMatch {

private String matchDetails;

CricketMatch(String matchDetails) {

this.matchDetails = matchDetails;

}

String getMatchDetails() {

return matchDetails;

}

}

interface Observer {

void update(CricketMatch match);

}

class ConcreteSubject implements Subject {

private List<Observer> observers = new ArrayList<>();

private CricketMatch currentMatch;

@Override

public void attach(Observer observer) {

observers.add(observer);

}

@Override

public void detach(Observer observer) {

observers.remove(observer);

}

@Override

public void notifyObservers() {

for (Observer observer : observers) {

observer.update(currentMatch);

}

}

public void setMatch(CricketMatch match) {

this.currentMatch = match;

notifyObservers();

}

}

class ConcreteObserver implements Observer {

private CricketMatch currentMatch;

@Override

public void update(CricketMatch match) {

this.currentMatch = match;

System.out.println("Observer updated with match details: " + match.getMatchDetails());

}

}

class MainScreen implements Observer {

private ConcreteSubject subject;

MainScreen(ConcreteSubject subject) {

this.subject = subject;

subject.attach(this);

}

void displayLiveMatches(List<CricketMatch> matches) {

System.out.println("Main Screen: Live Matches - " + matches);

}

void selectMatch(CricketMatch match) {

BallByBallCoverageScreen coverageScreen = new BallByBallCoverageScreen(subject);

subject.setMatch(match);

}

@Override

public void update(CricketMatch match) {

System.out.println("Main Screen: Received match update - " + match.getMatchDetails());

}

}

class BallByBallCoverageScreen implements Observer {

private ConcreteSubject subject;

BallByBallCoverageScreen(ConcreteSubject subject) {

this.subject = subject;

subject.attach(this);

}

@Override

public void update(CricketMatch match) {

System.out.println("BallByBallCoverageScreen: Displaying ball-by-ball coverage for " + match.getMatchDetails());

}

}

public class Main {

public static void main(String[] args) {

ConcreteSubject subject = new ConcreteSubject();

MainScreen mainScreen = new MainScreen(subject);

List<CricketMatch> liveMatches = new ArrayList<>();

liveMatches.add(new CricketMatch("Pakistan vs India"));

liveMatches.add(new CricketMatch("Australia vs England"));

mainScreen.displayLiveMatches(liveMatches);

mainScreen.selectMatch(liveMatches.get(0));

CricketMatch newMatch = new CricketMatch("Australia vs England");

subject.setMatch(newMatch);

}

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_