```java
/**
 * @author Caroline Ta
 * @since 05.19.2020
 */
package com.company.data.person;

import com.company.data.course.Course;
import com.company.data.course.Session;

import java.io.File;
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Scanner;

/**
 * The type Student.
 */
public class Student extends Person {

    /**
     * The Student count.
     */
    private int studentCount = 0;
    /**
     * The Student date of birth.
     */
    private String dob;
    /**
     * The Student GPA.
     */
    private String gpa;
    /**
     * The Student start date to attend.
     */
    private String dateStart;
    /**
     * The Student id.
     */
    private String id;
    /**
     * The Student status of scheduled or unscheduled.
     */
    private boolean scheduledStatus;
    /**
     * The Student's list of courses scheduled.
     */
    private ArrayList<Course> courseScheduledList;
    /**
     * The Student's list of sessions scheduled.
     */
    private ArrayList<Session> sessionScheduledList;
    /**
     * The Student's course wish list
     */
    private ArrayList<String> courseWishList;

    // -----------------------------------------------------------------------
    // CONSTRUCTORS
    // -----------------------------------------------------------------------

    /**
     * Instantiates a new Student.
     */
    public Student() {
    }

    /**
     * Instantiates a new Student.
     *
     * @param nameO     the name o
     * @param email     the email
     * @param tel       the tel
     * @param addressO  the address o
     * @param id        the id
     * @param dob       the dob
     * @param gpa       the gpa
     * @param dateStart the date start
     * @throws Exception the exception
     */
    public Student(PersonName nameO,
                   String email,
                   String tel,
                   PersonAddress addressO,
                   String id,
                   String dob,
                   String gpa,
                   String dateStart
    ) throws Exception {
        super(nameO, email, tel, addressO);
        setDob(dob);
        setGpa(gpa);
        setDateStart(dateStart);
        setId(id);

        courseScheduledList = new ArrayList<>();
        sessionScheduledList = new ArrayList<>();
        courseWishList = new ArrayList<>();
    }

    // -------------------------------------------------------------------------
    // ACCESSORS - GETTER METHODS
    // -------------------------------------------------------------------------

    /**
     * Gets dob.
     *
     * @return the dob
     */
    public String getDob() {
        return dob;
    }

    /**
     * Gets gpa.
     *
     * @return the gpa
     */
    public String getGpa() {
        return gpa;
    }

    /**
     * Gets date start.
     *
     * @return the date start
     */
    public String getDateStart() {
        return dateStart;
    }

    /**
     * Gets id.
     *
     * @return the id
     */
    public String getId() {
        return id;
    }

    /**
```

```java
     * Gets scheduled status.
     *
     * @return the scheduled status
     */
    public boolean getScheduledStatus() {
        updateScheduledStatus();
        return scheduledStatus;
    }

    /**
     * Gets course scheduled list.
     *
     * @return the course scheduled list
     */
    public ArrayList<Course> getCourseScheduledList() {
        return courseScheduledList;
    }

    // ------------------------------------------------------------------------
    // MUTATORS - SETTER METHODS
    // ------------------------------------------------------------------------

    /**
     * Sets dob.
     *
     * @param dob the dob
     * @throws Exception the exception
     */
    public void setDob(String dob) throws Exception {
        checkStringValue(dob, "Dob");
        this.dob = dob;
    }

    /**
     * Sets gpa.
     *
     * @param gpa the gpa
     * @throws Exception the exception
     */
    public void setGpa(String gpa) throws Exception {
        checkStringValue(gpa, "Gpa");
        this.gpa = gpa;
    }

    /**
     * Sets date start.
     *
     * @param dateStart the date start
     * @throws Exception the exception
     */
    public void setDateStart(String dateStart) throws Exception {
        checkStringValue(dateStart, "Datestart");
        this.dateStart = dateStart;
    }

    /**
     * Sets id.
     *
     * @param id the id
     * @throws Exception the exception
     */
    public void setId(String id) throws Exception {
        checkStringValue(id, "Id");
        this.id = id;
    }

    /**
     * Sets scheduled status.
     *
     * @param scheduledStatus the scheduled status
     */
    public void setScheduledStatus(boolean scheduledStatus) {
        this.scheduledStatus = scheduledStatus;
    }

    // ------------------------------------------------------------------------
    // FUNCTIONALITY METHODS
    // ------------------------------------------------------------------------

    /**
     * Generate course wish list.
     *
     * @param pathName the path name
     */
    public void generateCourseWishList(String pathName)
    // Path name: "src\\com\\company\\inputfiles\\S001_wish.txt"
    {
        String line;

        courseWishList = new ArrayList<>();

        try {
            File wishListFile = new File(pathName);
            Scanner input = new Scanner(wishListFile);

            while (input.hasNext()) {

                line = input.nextLine();

                String[] lineAr = line.split("_");

                courseWishList.addAll(Arrays.asList(lineAr));
            }

        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    /**
     * Generate id.
     */
    public void generateId() {
        if (studentCount < 10) {
            this.id = "S00" + studentCount;
        } else if (studentCount < 100) {
            this.id = "S0" + studentCount;
        } else {
            this.id = "S" + studentCount;
        }
    }

    /**
     * Update scheduled status.
     */
    public void updateScheduledStatus() {
        if (sessionScheduledList.size() > 0 || courseScheduledList.size() > 0) {
            scheduledStatus = true;
        } else {
            scheduledStatus = false;
        }

    }

    /**
     * Add course.
     *
     * @param courseToAdd the course to add
     */
    public void addCourse(Course courseToAdd) {
        if (!courseScheduledList.contains(courseToAdd)) {
            courseScheduledList.add(courseToAdd);
        }
        updateScheduledStatus();
    }

    /**
     * Remove course boolean.
     *
     * @param courseToRemove the course to remove
     * @return the boolean
     */
    public boolean removeCourse(Course courseToRemove) {
        courseScheduledList.remove(courseToRemove);
        updateScheduledStatus();
        return courseScheduledList.remove(courseToRemove);
    }

    /**
     * Add session.
     *
     * @param sessionToAdd the session to add
     */
    public void addSession(Session sessionToAdd) {
        if (!sessionScheduledList.contains(sessionToAdd)) {
            sessionScheduledList.add(sessionToAdd);
        }
        sessionScheduledList.add(sessionToAdd);
        updateScheduledStatus();
    }

    /**
     * Remove session boolean.
     *
     * @param sessionToRemove the session to remove
     * @return the boolean
     */
    public boolean removeSession(Session sessionToRemove) {
        sessionScheduledList.remove(sessionToRemove);
        updateScheduledStatus();
        return sessionScheduledList.remove(sessionToRemove);
    }

    // -------------------------------------------------------------------------
    // OVERRIDDEN METHODS
    // -------------------------------------------------------------------------

    /**
     * Override toString method.
     *
     * @return a string description of the Student class.
     */
    @Override
    public String toString() {

        updateScheduledStatus();

        StringBuilder builder = new StringBuilder();
        builder.append(super.toString());

        builder.append("\nDate of birth:  ");
        builder.append(dob);

        builder.append("\nGPA:            ");
        builder.append(gpa);

        builder.append("\nDate start:     ");
        builder.append(dateStart);

        builder.append("\nID:             ");
        builder.append(id);

        builder.append("\nScheduled:      ");
        builder.append(scheduledStatus);

        return builder.toString();
    }
}

```