```java
package caltool.model.schedule;

import caltool.model.caldb.*;
import mvp.*;

/****
 *
 * Class Schedule is the top-level model class in the schedule package.  It
 * provides methods to schedule the four types of calendar item.  It also
 * contains a Categories data field, which is the sub-model for editing
 * scheduled item categories.
 *
 * @author Gene Fisher (gfisher@calpoly.edu)
 * @version 13apr15
 *
 */
public class Schedule extends Model {

    /**
     * Construct this with the given companion view and the parent CalendarDB
     * model.  The CalendarDB is provided to access to its service methods that
     * store items in the current user calendar.
     */
    public Schedule(View view, CalendarDB calDB) {
        super(view);
        this.calDB = calDB;
    }

    /*-*
     * Derived methods
     */

    /**
     * ScheduleAppointment adds the given Appointment to the current Calendar
     * an appointment of the same time, duration, and title is not already
     * scheduled.
     */
    public void scheduleAppointment(Appointment appt) {
        System.out.println("In Schedule.scheduleAppointment.");
    }

    /**
     * ScheduleMeeting adds a Meeting to the current calendar, based on the the
     * given MeetingRequest.  The work is done by the three suboperations,
     * which determine a list of possible meetings times, set
     * meeting-scheduling options, and confirm the scheduling of a specific
     * meeting selected from the possibles list.
     */
    public void scheduleMeeting(MeetingRequest meeting_req) {
        System.out.println("In Schedule.scheduleMeeting.");
    }

    /**
     * Produce the list of possible meeting times that satisfy the given
     * MeetingRequest.
     */
    public PossibleMeetingTimes listMeetingTimes(MeetingRequest request) {
        System.out.println("In schedule.listMeetingTimes.");
        return null;
    }

    /**
     * Set the meeting options in the CalendarDB to those given.
     *
     */
    public void setMeetingOptions(MeetingSchedulingOptions options) {
        System.out.println("In schedule.setMeetingOptions.");
    }

    /**
     * ConfirmMeeting takes a CalendarDB, MeetingRequest, list of
     * PossibleMeetingTimes, and a selected time from the list.  It outputs a
     * new CalendarDB with the given request scheduled at the selected time.
     */
    public void confirmMeeting(MeetingRequest meeting_req,
            PossibleMeetingTimes possible_times, int selected_time) {
        System.out.println("In Schedule.confirmMeeting");
    }

    /**
     * ScheduleTask adds the given Task to the given CalendarDB, if a task of
     * the same start date, title, and priority is not already scheduled.
     */
    public void scheduleTask(Task task) {
        System.out.println("In Schedule.scheduleTask.");
    }

    /**
     * ScheduleEvent adds the given Event to the given CalendarDB, if an event
     * of the same start date and title is not already scheduled.
     */
    public void scheduleEvent(Event event) {
        System.out.println("In Schedule.scheduleEvent.");
    }

    /**
     * Change the given old appointment to the given new one in the
     * current calendar.
     */
    public void changeAppointment(Appointment oldAppt, Appointment newAppt) {
        System.out.println("In Schedule.changeAppointment.");
    }

    /**
     * Delete the given appointment from the current calendar.
     */
    public void deleteAppointment(Appointment appt) {
        System.out.println("In Schedule.deleteAppointment.");
    }


    /*-*
```

```
     * Access methods
     */

    /**
     * Return the categories component.
     */
    public Categories getCategories() {
        return categories;
    }


    /*-*
     * Derived data fields
     */

    /** Category list in which scheduled item categories are defined */
    protected Categories categories;


    /*-*
     * Process data fields
     */

    /** Calendar database in which scheduled items are stored */
    protected CalendarDB calDB;

}
```