

```

1 package caltool.model.schedule;
2
3 import caltool.model.caldb.*;
4 import mvp.Model;
5
6 /****
7 *
8 * A ScheduledItem is the generic definition for the types of items stored in a
9 * calendar. The Title component is a brief description of what the item is
10 * for. The StartOrDueDate and EndDate components indicate when the item is
11 * scheduled. The Category component is used to organize items into related
12 * color-coded categories.
13 *
14 * There are four specializations of ScheduledItem. They are Appointment,
15 * Meeting, Event, and Task, q.q.v.
16 *
17 * @author Gene Fisher (gfisher@calpoly.edu)
18 * @version 13apr15
19 *
20 */
21
22 public abstract class ScheduledItem extends Model {
23
24     /**
25     * Construct an empty scheduled item.
26     */
27     public ScheduledItem() {
28         super();
29     }
30
31
32     /*-*
33     * Access methods
34     */
35     /**
36     * Return the title
37     */
38     public String getTitle() {
39         return title;
40     }
41
42     /**
43     * Return the .
44     */
45     public Date getDate() {
46         return startOrDueDate;
47     }
48
49     /**
50     * Return the end date.
51     */
52     public Date getEndDate() {
53         return endDate;
54     }
55
56     /**
57     * Return the category.
58     */
59     public Category getCategory() {
60         return category;
61     }
62
63
64     /*-*
65     * Process methods
66     */
67
68     /**
69     * Return the unique lookup key for this. This method is specialized in
70     * each subclass per the unique key requirements described in class
71     * <a href= UserCalendar.html> UserCalendar. </a>
72     */
73     public abstract ItemKey getKey();
74
75
76
77     /*-*
78     * Derived data fields
79     */
80
81     /** Brief description of the scheduled item */
82     protected String title;
83
84     /** Date on which item is scheduled or due. The startOrDueDate is a
85     multi-purpose field of ScheduledItem. Its purpose depends on whether
86     an item is a Task and whether it is recurring (Events cannot recur).
87     For non-recurring appointments and meetings, StartOrDueDate is used as
88     the single date on which the item is scheduled. If the item is
89     recurring, StartOrDueDate is the first date on which it occurs. For a
90     non-recurring Task, StartOrDueDate is the single date the task is due.
91     If the task is recurring, StartOrDueDate is the first date it is due.
92     */
93     protected Date startOrDueDate;
94
95     /** Last date on which item is scheduled or due. In recurring appointments,
96     meetings, and tasks, the end date defines the last date on which the
97     item will recur. In events, the end date defines the last date of a
98     multi-day event. When the value of end date is empty, the
99     StartOrDueDate component is interpreted as the single date on which the
100     item occurs.
101     */
102     protected Date endDate;
103
104     /** Used to organize items into related color-coded categories */
105     protected Category category;
106
107 }

```