```java
/**
 * {@code ToDoList} interface provides methods for managing to-do list events.
 * It extends {@code Comparable<ToDoList>} and {@code ToDoListKernel}.
 *
 * @initially The methods of this interface provide functionality to add and
 *            remove events, order events by importance or date, and display
 *            events that need to be done or completed.
 *
 * @author Jahnavi Acharya
 */
public interface ToDoList extends Comparable<ToDoList>, ToDoListKernel {

    /**
     * Adds a new event to the to-do list.
     *
     * @param event
     *            the event to be added
     * @requires event is not an uncompleted event already on the list
     * @ensures the specified event is added to the to-do list or added again if
     *          it was completed
     */
    void addEvent(String event);

    /**
     * Adds a new event with specified importance or date to the to-do list.
     *
     * @param event
     *            the event to be added
     * @param importance
     *            the importance of the event
     * @param date
     *            the date of the event
     * @requires importance is between -1 and 3 inclusive and date is a 6 digit
     *           integer in the format YearMonthDay and event is not an
     *           uncompleted event already on the list
     * @ensures the specified event with importance and date is added to the
     *          to-do list
     */
    void addEvent(String event, int importance, int date);

    /**
     * 'Removes' (by changing importance to -1) a specific event from the to-do
     * list based on its name and date.
     *
     * @param event
     *            the event to be 'removed'
     * @param dateRem
     *            the date of the event when completed
     * @requires date is a 6 digit integer in the format YearMonthDay and event
     *           is not an uncompleted event already on the list
     * @ensures the specified event is set as a -1 importance in the to-do list
     */
    void removeEvent(String event, int dateRem);

    /**
     * Orders the events in the to-do list by their importance.
     *
     * @ensures the events in the to-do list are ordered by importance
     */
    void orderByImportance();

    /**
     * Orders the events in the to-do list by their date.
     *
     * @ensures the events in the to-do list are ordered by date
     */
    void orderByDate();

    /**
     * Displays the events that need to be done.
     *
     * @return a string representing the events that need to be done
     */
    String displayEventsNeedToDo();

    /**
     * Displays the events that have been completed.
     *
     * @return a string representing the events that have been completed
     */
    String displayEventsCompleted();

}

```