

```
reviewers: {
    "1": ["reviewerId"],
}
}, {
    autoActivate: true
});
```

The option `autoActivate` replaces the *Activate the task at start date automatically* option in the user interface.

### 5.34.14 Using the Status Change APIs for any Calendar Composite Task

You can use the `submit()`, `decline()`, `approve()`, or `reject()` script API methods for any composite task, given the task ID.

*Example:*

With the following sample code snippet, you can submit a Composite Task with a given task ID:

```
var task = CalendarIntegration_1.getCalendarTaskById("taskId");
if (task.getType() === CalendarTaskType.CompositeTask) {
    var compositeTask = cast(Type.CalendarCompositeTask, task);
    compositeTask.submit();
}
```

### 5.34.15 Using the canUserSubmit API

You can use the `canUserSubmit()` script API method to check if a calendar task is ready to be submitted. If you use this method when the task status is *Open*, then the task status is changed to *In Progress*.

*Example:*

With the following sample code snippet, you can submit a Composite Task with the ID "`myTaskId`" that has the task status *Open*:

```
var task = CalendarIntegration_1.getCalendarTaskById("myTaskId");
var compositeTask = cast(Type.CalendarCompositeTask, task);
var canSubmit = compositeTask.canUserSubmit();
if (canSubmit) {
    var isSubmitted = compositeTask.submit();
    if (isSubmitted) {
        console.log("You've successfully submitted your task.");
    } else {
        console.log("Something went wrong when submitting your task.");
    }
} else {
    console.log("Sorry, you can't submit this composite task.");
}
```