

	<p>“resources” subobjects (list of resource objects used in reservations), and “users” subobject (list of users objects used in reservations)</p>
<p>PUT /reservations/<ID></p>	<p>Updates a reservation with the supplied ID.</p> <p>Common errors: Trying to extend reservation, trying to add new resources</p> <p>Returns a JSON object with is_error (boolean), error_msg (possibly empty string). If is_error is false, also returns a reservation subobject with all the fields that were actually committed (ideally, the same as the values that were passed in)</p>
<p>DELETE /reservations/<ID></p>	<p>Deletes the reservation with the given ID, if it exists</p> <p>Common error: no reservation with specified ID</p> <p>Returns a JSON object with is_error (boolean) and error_msg (possibly empty string)</p>
<p>GET /reservations/approvableReservations</p>	<p>Returns all incomplete reservations which you can partially or fully approve. In addition, returns a data structure which specifies which resources have been approved and which resources you can approve, for each reservation listed.</p> <p>Returns an object with top level object “reservations” (a list of reservations), and top level object “approvableResources”, which is a list of objects, each of which lists a “reservation_id”, a list of resources “approvable_resources”, and a list of resources “approved_resources”. Thus, it shows, for each reservation, which resources are approvable by user, and which resources have already been approved (a resource that the user CAN approve but has already been approved will ONLY SHOW under “approved_resources”</p>