

Pfad des Endpoints	Typ	Consumes	Produces	Response-Code	Beschreibung
/kunden	POST	JSON	JSON	201 (Created), 400 (Bad Request)	Für das gesendete Kunden-Objekt soll der Server eine eindeutige Id vergeben, das Objekt anschließend speichern und dem Client mit dem Code 201 zurückschicken. Beachten Sie Hinweis Nr. 1
/ablesungen	POST	JSON	JSON	201 (Created), 400 (Bad Request), 404 (Not Found)	Der Server soll für das gesendete Objekt eine eindeutige Id vergeben, dieses abspeichern und das Objekt mit der Id mit dem Code 201 zurückschicken. Existiert jedoch der Kunde des gesendeten Ablesungsobjekt nicht, so soll mit dem Code 404 eine entsprechende String-Antwort gesendet werden. Beachten Sie Hinweis Nr. 1 und 2.
/kunden	PUT	JSON	TEXT_PLAIN	200 (Ok), 400 (Bad Request), 404 (Not Found)	Der Server soll die Werte des gespeicherten Kundenobjekts mit den Werten des gesendeten Objekts überschreiben und mit dem Code 200 eine entsprechende String-Antwort senden. Beachten Sie Hinweis 1 und 2.
/ablesungen	PUT	JSON	TEXT_PLAIN	200 (Ok), 400 (Bad Request), 404 (Not Found)	Der Server soll die Werte des gespeicherten Ablesungsobjekt mit den Werten des gesendeten Objekts überschreiben. Beachten Sie Hinweis Nr. 1 und 2.
/ablesungen/{id}	DELETE	-	JSON	200 (Ok), 404 (Not Found)	Die Ablesung mit der übergebenen {id} im Pfad soll gelöscht werden. Dabei soll das gelöschte Ablesungsobjekt mit dem Code 200 zurückgeschickt werden. Beachten Sie Hinweis Nr. 2.
/kunden/{id}	DELETE	-	JSON	200 (Ok), 404 (Not Found)	Der Kunde mit der übergebenen id im Pfad soll gelöscht werden. Die Ablesungen dieses Kunden sollen aber noch auf dem Server gespeichert bleiben, der Attributwert von kunde soll dabei auf null gesetzt werden. Das gelöschte Kunden-Objekt soll zusammen mit der Liste der Ablesungen dieses Kunden als Map mit dem Code 200 zurückgeschickt werden. Beachten Sie Hinweis Nr. 2.

Pfad des Endpoints	Typ	Consumes	Produces	Response-Code	Beschreibung
/kunden	GET	-	JSON	200 (Ok)	Der Server antwortet mit Code 200 und einer Liste aller gespeicherten Kunden. Falls kein Kunde existiert, soll die Liste leer sein.
/kunden/{id}	GET	-	JSON	200 (Ok), 404 (Not Found)	Der Server sendet mit Code 200 das entsprechende Kunden-Objekt zurück. Beachten Sie Hinweis Nr. 2.
/ablesungenVorZweiJahrenHeute	GET	-	JSON	200 (Ok)	Der Server antwortet mit Code 200 und einer Liste aller gespeicherten Ablesungen, die aus Netzwerklast-Gründen nicht älter sind als der 01.01. des Vor-Vorjahres. Falls keine Ablesung existiert, soll die Liste leer sein.
/ablesungen/{id}	GET	-	JSON	200 (Ok), 404 (Not Found)	Der Server sendet mit Code 200 das Ablesungs-Objekt mit der übergebenen {id}. Beachten Sie Hinweis Nr. 2.
/ablesungen?kunde={id}& beginn={datum}& ende={datum}	GET	-	JSON	200 (Ok), 400 (Bad Request), 404 (Not Found)	Der Server antwortet mit einer Liste aller Ablesungen, die zu den übergebenen Parametern passen. Mit beginn und ende kann der Zeitraum der angeforderten Ablesungen eingegrenzt werden. Wird kein Anfangsdatum übergeben, sollen alle Ablesungen bis zum Enddatum gesendet werden. Wird kein Enddatum übergeben, so sollen alle Ablesungen bis zum heutigen Datum gesendet werden. Wird keine Kunden-Id übergeben, so sollen alle Ablesungen, die zum übergebenem Bereich passen gesendet werden. Wird ein Datum nicht im Format yyyy-MM-dd übermittelt, so soll mit dem Code 400 eine entsprechende Fehlermeldung gesendet werden. Beachten Sie außerdem Hinweis 2.

Hinweise:

1. Wird kein Objekt der Klasse Kunde bzw. Ablesung gesendet, soll mit dem Code 400 eine entsprechende Fehlermeldung gesendet werden.
2. Existiert das gesendete Kunden- bzw. Ablesungsobjekt nicht, soll mit dem Code 404 eine entsprechende Fehlermeldung gesendet werden.