

Zählerdaten

getCounters getCounterData

- [getCounters](#)
- [getCounterData](#)

getCounters

Beschreibung:

Ruft eine Liste aller Zähler und deren Informationen für ein bestimmtes PowerDog-Gerät ab. Diese Funktion gibt detaillierte Informationen zu den angeschlossenen Energiezählern zurück.

Endpunkt: `/api`

Methode: `POST`

Parameter:

- `apikey` (string) – API-Schlüssel des Benutzers, aus `getApiKey()`
- `PowerDogID` (number) – Eindeutige Kennung des PowerDog-Geräts, aus `getPowerDogs()`

Beispielanfrage:

```
{
  "action": "getCounters",
  "apikey": "6641d282073d76b625987af5141d3e2a",
  "PowerDogID": "36789"
}
```

Beispielantwort:

```
{
  "valid": 1,
  "counters": {
    "pv_global_1499321681": {
      "KEY": "pv_global_1499321681",
      "Type": "Energy",
      "Max": 5000,
      "Name": "PV",
      "Hardware": "pv_global",
      "Medium": "ELECTRIC",
      "id": "551904"
    },
    "buscounter_1579943663": {
      "KEY": "buscounter_1579943663",
```

```

    "Type": "Energy",
    "Max": "5000",
    "Name": "GRIDIN",
    "Hardware": "buscounter",
    "Medium": "ELECTRIC",
    "id": "551911"
  }
},
"id": 36789
}

```

Beschreibung der Antwortparameter:

- **valid** (number):

Gibt an, ob die Anfrage erfolgreich war (1 = Erfolg, nicht vorhanden = Fehler).

- **counters** (object):

Ein Objekt, das eine Liste aller Zähler enthält. Jeder Zähler hat eine eindeutige **KEY**-ID.

- **KEY** (string):

Eindeutige Kennung des Zählers.

- **Type** (string):

Typ des Zählers, z. B. `Energy`.

- **Max** (number):

Maximale Messkapazität des Zählers.

- **Name** (string):

Name des Zählers.

- **Hardware** (string):

Bezeichnung der Zählerhardware.

- **Medium** (string):

Medium des Zählers (z. B. `ELECTRIC`, `MBUS_HEATMETER`).

- **id** (string):

Interne ID des Zählers.

- **id** (number):

Eindeutige Kennung des PowerDog-Geräts, zu dem die Zähler gehören.

getCounterData

Beschreibung:

Ruft Zählerdaten (Counter Data) oder Sensordaten für einen bestimmten Zeitraum ab. Die Werte werden in 5-Minuten-Intervallen bereitgestellt.

Endpunkt: `/api`

Methode: `POST`

Parameter:

- `apikey` (string) – API-Schlüssel des Benutzers, aus `getApiKey()`
- `SensorID` (number) – Eindeutige Kennung des Sensors/Zählers, aus `getSensors()` oder `getCounters()`
- `UTC_TIMESTAMP_FROM` (number) – Startzeitpunkt im UTC-Timestamp-Format
- `UTC_TIMESTAMP_TO` (number) – Endzeitpunkt im UTC-Timestamp-Format

Beispielanfrage:

```
{
  "action": "getCounterData",
  "apikey": "6641d282073d76b625987af5141d3e2a",
  "SensorID": "551907",
  "UTC_TIMESTAMP_FROM": "1739867939",
  "UTC_TIMESTAMP_TO": "1739877939"
}
```

Beispielantwort:

```
{
  "sensor_id": 551907,
  "valid": 1,
  "datasets": {
    "1739868002": {
      "DATA": "1165",
      "TIMESTAMP_UTC": 1739868002,
      "TIMESTAMP_LOCAL": 1739871602
    },
    "1739868302": {
```

```
"DATA": "1428",  
"TIMESTAMP.UTC": 1739868302,  
"TIMESTAMP.LOCAL": 1739871902  
},  
}  
}
```

Beschreibung der Antwortparameter:

- **sensor_id** (number):
Eindeutige Kennung des Zählers oder Sensors, für den die Daten abgefragt wurden.
- **valid** (number):
Gibt an, ob die Anfrage erfolgreich war (**1** = Erfolg, **0** = Fehler).
- **datasets** (object):
Enthält die aufgezeichneten Messwerte im 5-Minuten-Intervall.
Jeder Schlüssel stellt einen UTC-Timestamp dar.
 - **DATA** (string):
Erfasster Messwert des Zählers oder Sensors.
 - **TIMESTAMP.UTC** (int):
Zeitstempel der Messung im UTC-Format.
 - **TIMESTAMP.LOCAL** (int):
Zeitstempel der Messung in der lokalen Zeitzone.

Nähere Informationen zu Antwortformat und Fehlermeldungen:

[Antwortformat und Fehler](#)