We have some important news to share: Dark Sky has joined Apple.

You can read more on our blog.

Dark Sky API

Dark Sky API — Overview

The Dark Sky API allows you to look up the weather anywhere on the globe, returning (where available):

- Current weather conditions
- Minute-by-minute forecasts out to one hour
- Hour-by-hour and day-by-day forecasts out to seven days
- Hour-by-hour and day-by-day observations going back decades
- Severe weather alerts in the US, Canada, European Union member nations, and Israel

Weather API Call Types

We provide two types of API requests to retrieve the weather anywhere in the world:

- The Forecast Request returns the current weather forecast for the next week.
- The Time Machine Request returns the observed or forecast weather conditions for a date in the past or future.

Weather Conditions

The Dark Sky API offers a full collection of meteorological conditions in 39 different languages, including:

- Apparent (feels-like) temperature
- Atmospheric pressure

- Cloud cover
- Dew point
- Humidity
- Liquid precipitation rate
- Moon phase
- Nearest storm distance
- Nearest storm direction
- Ozone
- Precipitation type
- Snowfall
- Sun rise/set
- Temperature
- Text summaries
- UV index
- Wind gust
- Wind speed
- Wind direction

Both forecast and time machine requests return the same weather conditions, in the same convenient JSON format. You can parse the response directly, or use one several community-contributed libraries to interact with our API in the programming language of your choice.

Pricing and Attribution

Pricing is dead-simple.

- The first 1,000 API requests you make every day are **free of charge**.
- Every API request over the free daily limit costs $0.0001.

You are required to display the message "Powered by Dark Sky" that links to `https://darksky.net/poweredby/` somewhere prominent in your app or service.

Further details can be found in the terms of service.

If you have any questions, please check the FAQ and, if you can't find what you're looking for there, send us an email.

API Request Types

Forecast Request

```
https://api.darksky.net/forecast/[key]/[latitude],[longitude]
```

A Forecast Request returns the current weather conditions, a minute-by-minute forecast for the next hour (where available), an hour-by-hour forecast for the next 48 hours, and a day-by-day forecast for the next week.

Example Request

```
GET https://api.darksky.net/forecast/0123456789abcdef9876543210fedcba/42.36
01,-71.0589


    {
        "latitude": 42.3601,
        "longitude": -71.0589,
        "timezone": "America/New_York",
        "currently": {
            "time": 1509993277,
            "summary": "Drizzle",
            "icon": "rain",
            "nearestStormDistance": 0,
            "precipIntensity": 0.0089,
            "precipIntensityError": 0.0046,
            "precipProbability": 0.9,
            "precipType": "rain",
            "temperature": 66.1,
            "apparentTemperature": 66.31,
            "dewPoint": 60.77,
            "humidity": 0.83,
            "pressure": 1010.34,
            "windSpeed": 5.59,
            "windGust": 12.03,
            "windBearing": 246,
```

```
            "cloudCover": 0.7,
            "uvIndex": 1,
            "visibility": 9.84,
            "ozone": 267.44
        },
        "minutely": {
            "summary": "Light rain stopping in 13 min., starting again 30
min. later.",
            "icon": "rain",
            "data": [{
                "time": 1509993240,
                "precipIntensity": 0.007,
                "precipIntensityError": 0.004,
                "precipProbability": 0.84,
                "precipType": "rain"
            },
          ...
          ]
        },
        "hourly": {
            "summary": "Rain starting later this afternoon, continuing
until this evening.",
            "icon": "rain",
            "data": [{
                "time": 1509991200,
                "summary": "Mostly Cloudy",
                "icon": "partly-cloudy-day",
                "precipIntensity": 0.0007,
                "precipProbability": 0.1,
                "precipType": "rain",
                "temperature": 65.76,
                "apparentTemperature": 66.01,
                "dewPoint": 60.99,
                "humidity": 0.85,
                "pressure": 1010.57,
                "windSpeed": 4.23,
                "windGust": 9.52,
                "windBearing": 230,
                "cloudCover": 0.62,
                "uvIndex": 1,
```

```
                    "visibility": 9.32,
                    "ozone": 268.95
                },
                ...
                ]
            },
        "daily": {
                "summary": "Mixed precipitation throughout the week, with
temperatures falling to 39°F on Saturday.",
                "icon": "rain",
                "data": [{
                    "time": 1509944400,
                    "summary": "Rain starting in the afternoon, continuing
until evening.",
                    "icon": "rain",
                    "sunriseTime": 1509967519,
                    "sunsetTime": 1510003982,
                    "moonPhase": 0.59,
                    "precipIntensity": 0.0088,
                    "precipIntensityMax": 0.0725,
                    "precipIntensityMaxTime": 1510002000,
                    "precipProbability": 0.73,
                    "precipType": "rain",
                    "temperatureHigh": 66.35,
                    "temperatureHighTime": 1509994800,
                    "temperatureLow": 41.28,
                    "temperatureLowTime": 1510056000,
                    "apparentTemperatureHigh": 66.53,
                    "apparentTemperatureHighTime": 1509994800,
                    "apparentTemperatureLow": 35.74,
                    "apparentTemperatureLowTime": 1510056000,
                    "dewPoint": 57.66,
                    "humidity": 0.86,
                    "pressure": 1012.93,
                    "windSpeed": 3.22,
                    "windGust": 26.32,
                    "windGustTime": 1510023600,
                    "windBearing": 270,
                    "cloudCover": 0.8,
                    "uvIndex": 2,
```

```json
              "uvIndexTime": 1509987600,
              "visibility": 10,
              "ozone": 269.45,
              "temperatureMin": 52.08,
              "temperatureMinTime": 1510027200,
              "temperatureMax": 66.35,
              "temperatureMaxTime": 1509994800,
              "apparentTemperatureMin": 52.08,
              "apparentTemperatureMinTime": 1510027200,
              "apparentTemperatureMax": 66.53,
              "apparentTemperatureMaxTime": 1509994800
            },
            ...
            ]
        },
        "alerts": [
        {
          "title": "Flood Watch for Mason, WA",
          "time": 1509993360,
          "expires": 1510036680,
          "description": "...FLOOD WATCH REMAINS IN EFFECT THROUGH LATE
MONDAY NIGHT...\nTHE FLOOD WATCH CONTINUES FOR\n* A PORTION OF NORTHWEST
WASHINGTON...INCLUDING THE FOLLOWING\nCOUNTY...MASON.\n* THROUGH LATE
FRIDAY NIGHT\n* A STRONG WARM FRONT WILL BRING HEAVY RAIN TO THE
OLYMPICS\nTONIGHT THROUGH THURSDAY NIGHT. THE HEAVY RAIN WILL PUSH
THE\nSKOKOMISH RIVER ABOVE FLOOD STAGE TODAY...AND MAJOR FLOODING
IS\nPOSSIBLE.\n* A FLOOD WARNING IS IN EFFECT FOR THE SKOKOMISH RIVER. THE
FLOOD\nWATCH REMAINS IN EFFECT FOR MASON COUNTY FOR THE POSSIBILITY
OF\nAREAL FLOODING ASSOCIATED WITH A MAJOR FLOOD.\n",
          "uri": "http://alerts.weather.gov/cap/wwacapget.php?
x=WA1255E4DB8494.FloodWatch.1255E4DCE35CWA.SEWFFASEW.38e78ec64613478bb70fc6
ed9c87f6e6"
        },
        ...
        ],
        {
          "flags": {
            "units": "us",
            ...
          }
```

```
    }
```

Request Parameters

Required parameters slot directly into the request URL. Optional parameters should be specified as HTTP query parameters.

key required
Your Dark Sky secret key. (Your secret key **must** be kept secret; in particular, **do not** embed it in JavaScript source code that you transmit to clients.)

latitude required
The latitude of a location (in decimal degrees). Positive is north, negative is south.

longitude required
The longitude of a location (in decimal degrees). Positive is east, negative is west.

exclude=[blocks] optional
Exclude some number of data blocks from the API response. This is useful for reducing latency and saving cache space. The value `blocks` should be a comma-delimeted list (without spaces) of any of the following:

- `currently`
- `minutely`
- `hourly`
- `daily`
- `alerts`
- `flags`

extend=hourly optional
When present, return hour-by-hour data for the next 168 hours, instead of the next 48. When using this option, we **strongly** recommend enabling HTTP compression.

lang=[language] optional

Return `summary` properties in the desired language. (Note that units in the summary will be set according to the `units` parameter, so be sure to set both parameters appropriately.) `language` may be:

- `ar` : Arabic
- `az` : Azerbaijani
- `be` : Belarusian
- `bg` : Bulgarian
- `bn` : Bengali
- `bs` : Bosnian
- `ca` : Catalan
- `cs` : Czech
- `da` : Danish
- `de` : German
- `el` : Greek
- `en` : English (which is the default)
- `eo` : Esperanto
- `es` : Spanish
- `et` : Estonian
- `fi` : Finnish
- `fr` : French
- `he` : Hebrew
- `hi` : Hindi
- `hr` : Croatian
- `hu` : Hungarian
- `id` : Indonesian
- `is` : Icelandic
- `it` : Italian
- `ja` : Japanese
- `ka` : Georgian
- `kn` : Kannada
- `ko` : Korean
- `kw` : Cornish

- `lv` : Latvian
- `ml` : Malayam
- `mr` : Marathi
- `nb` : Norwegian Bokmål
- `nl` : Dutch
- `no` : Norwegian Bokmål (alias for `nb`)
- `pa` : Punjabi
- `pl` : Polish
- `pt` : Portuguese
- `ro` : Romanian
- `ru` : Russian
- `sk` : Slovak
- `sl` : Slovenian
- `sr` : Serbian
- `sv` : Swedish
- `ta` : Tamil
- `te` : Telugu
- `tet` : Tetum
- `tr` : Turkish
- `uk` : Ukrainian
- `ur` : Urdu
- `x-pig-latin` : Igpay Atinlay
- `zh` : simplified Chinese
- `zh-tw` : traditional Chinese

If you require a language not listed here, please consider contributing to our API translation module on Github.

units=[units] optional

Return weather conditions in the requested units. `[units]` should be one of the following:

- `auto` : automatically select units based on geographic location
- `ca` : same as `si` , except that `windSpeed` and `windGust` are in kilometers per hour

- `uk2` : same as `si` , except that `nearestStormDistance` and `visibility` are in miles, and `windSpeed` and `windGust` in miles per hour
- `us` : Imperial units (the default)
- `si` : SI units

SI units are as follows:

- `summary` : Any summaries containing temperature or snow accumulation units will have their values in degrees Celsius or in centimeters (respectively).
- `nearestStormDistance` : Kilometers.
- `precipIntensity` : Millimeters per hour.
- `precipIntensityMax` : Millimeters per hour.
- `precipAccumulation` : Centimeters.
- `temperature` : Degrees Celsius.
- `temperatureMin` : Degrees Celsius.
- `temperatureMax` : Degrees Celsius.
- `apparentTemperature` : Degrees Celsius.
- `dewPoint` : Degrees Celsius.
- `windSpeed` : Meters per second.
- `windGust` : Meters per second.
- `pressure` : Hectopascals.
- `visibility` : Kilometers.

Time Machine Request

```
        https://api.darksky.net/forecast/[key]/[latitude],[longitude],
[time]
```

A Time Machine Request returns the observed (in the past) or forecasted (in the future) hour-by-hour weather and daily weather conditions for a particular date. A Time Machine request is identical in structure to a Forecast Request, except:

- The `currently` data point will refer to the time provided, rather than the current time.

- The `minutely` data block will be omitted, unless you are requesting a time within an hour of the present.
- The `hourly` data block will contain data points starting at midnight (local time) of the day requested, and continuing until midnight (local time) of the following day.
- The `daily` data block will contain a single data point referring to the requested date.
- The `alerts` data block will be omitted.

Example Request

```
GET https://api.darksky.net/forecast/0123456789abcdef9876543210fedcba/42.36
01,-71.0589,255657600?exclude=currently,flags
```

```json
    {
      "latitude": 42.3601,
      "longitude": -71.0589,
      "timezone": "America/New_York",
      "hourly": {
        "summary": "Snow (6–9 in.) and windy starting in the afternoon.",
        "icon": "snow",
        "data": [
          {
            "time": 255589200,
            "summary": "Mostly Cloudy",
            "icon": "partly-cloudy-night",
            "precipIntensity": 0,
            "precipProbability": 0,
            "temperature": 22.8,
            "apparentTemperature": 16.46,
            "dewPoint": 15.51,
            "humidity": 0.73,
            "pressure": 1026.78,
            "windSpeed": 4.83,
            "windBearing": 354,
            "cloudCover": 0.78,
            "uvIndex": 0,
            "visibility": 9.62
          },
          ...
```

```json
            ]
        },
        "daily": {
          "data": [
            {
              "time": 255589200,
              "summary": "Snow (9–14 in.) and windy starting in the
afternoon.",
              "icon": "snow",
              "sunriseTime": 255613996,
              "sunsetTime": 255650764,
              "moonPhase": 0.97,
              "precipIntensity": 0.0354,
              "precipIntensityMax": 0.1731,
              "precipIntensityMaxTime": 255657600,
              "precipProbability": 1,
              "precipAccumulation": 7.337,
              "precipType": "snow",
              "temperatureHigh": 31.84,
              "temperatureHighTime": 255632400,
              "temperatureLow": 28.63,
              "temperatureLowTime": 255697200,
              "apparentTemperatureHigh": 20.47,
              "apparentTemperatureHighTime": 255625200,
              "apparentTemperatureLow": 13.03,
              "apparentTemperatureLowTime": 255697200,
              "dewPoint": 24.72,
              "humidity": 0.86,
              "pressure": 1016.41,
              "windSpeed": 22.93,
              "windBearing": 56,
              "cloudCover": 0.95,
              "uvIndex": 1,
              "uvIndexTime": 255621600,
              "visibility": 4.83,
              "temperatureMin": 22.72,
              "temperatureMinTime": 255596400,
              "temperatureMax": 32.04,
              "temperatureMaxTime": 255672000,
              "apparentTemperatureMin": 11.13,
```

```
            "apparentTemperatureMinTime": 255650400,
            "apparentTemperatureMax": 20.47,
            "apparentTemperatureMaxTime": 255625200
        }
      ]
    },
    "offset": -5
  }
```

Request Parameters

Required parameters slot directly into the request URL. Optional parameters should be specified as HTTP query parameters.

key required

Your Dark Sky secret key. (Your secret key **must** be kept secret; in particular, **do not** embed it in JavaScript source code that you transmit to clients.)

latitude required

The latitude of a location (in decimal degrees). Positive is north, negative is south.

longitude required

The longitude of a location (in decimal degrees). Positive is east, negative is west.

time required

Either be a UNIX time (that is, seconds since midnight GMT on 1 Jan 1970) or a string formatted as follows: `[YYYY]-[MM]-[DD]T[HH]:[MM]:[SS][timezone]`. `timezone` should either be omitted (to refer to local time for the location being requested), `Z` (referring to GMT time), or `+[HH][MM]` or `-[HH][MM]` for an offset from GMT in hours and minutes. The timezone is only used for determining the time of the request; the response will always be relative to the local time zone.

exclude=[blocks] optional

Exclude some number of data blocks from the API response. This is useful for reducing latency and saving cache space. The value `blocks` should be a comma-delimeted list (without spaces) of any of the following:

- `currently`
- `minutely`
- `hourly`
- `daily`
- `alerts`
- `flags`

lang=[language] optional

Return `summary` properties in the desired language. (Note that units in the summary will be set according to the `units` parameter, so be sure to set both parameters appropriately.) `language` may be:

- `ar` : Arabic
- `az` : Azerbaijani
- `be` : Belarusian
- `bg` : Bulgarian
- `bn` : Bengali
- `bs` : Bosnian
- `ca` : Catalan
- `cs` : Czech
- `da` : Danish
- `de` : German
- `el` : Greek
- `en` : English (which is the default)
- `eo` : Esperanto
- `es` : Spanish
- `et` : Estonian
- `fi` : Finnish
- `fr` : French
- `he` : Hebrew
- `hi` : Hindi
- `hr` : Croatian
- `hu` : Hungarian
- `id` : Indonesian

- `is` : Icelandic
- `it` : Italian
- `ja` : Japanese
- `ka` : Georgian
- `kn` : Kannada
- `ko` : Korean
- `kw` : Cornish
- `lv` : Latvian
- `ml` : Malayam
- `mr` : Marathi
- `nb` : Norwegian Bokmål
- `nl` : Dutch
- `no` : Norwegian Bokmål (alias for `nb`)
- `pa` : Punjabi
- `pl` : Polish
- `pt` : Portuguese
- `ro` : Romanian
- `ru` : Russian
- `sk` : Slovak
- `sl` : Slovenian
- `sr` : Serbian
- `sv` : Swedish
- `ta` : Tamil
- `te` : Telugu
- `tet` : Tetum
- `tr` : Turkish
- `uk` : Ukrainian
- `ur` : Urdu
- `x-pig-latin` : Igpay Atinlay
- `zh` : simplified Chinese
- `zh-tw` : traditional Chinese

If you require a language not listed here, please consider contributing to our API translation module on Github.

units=[units] optional

Return weather conditions in the requested units. `[units]` should be one of the following:

- `auto` : automatically select units based on geographic location
- `ca` : same as `si` , except that `windSpeed` is in kilometers per hour
- `uk2` : same as `si` , except that `nearestStormDistance` and `visibility` are in miles and `windSpeed` is in miles per hour
- `us` : Imperial units (the default)
- `si` : SI units

SI units are as follows:

- `summary` : Any summaries containing temperature or snow accumulation units will have their values in degrees Celsius or in centimeters (respectively).
- `nearestStormDistance` : Kilometers.
- `precipIntensity` : Millimeters per hour.
- `precipIntensityMax` : Millimeters per hour.
- `precipAccumulation` : Centimeters.
- `temperature` : Degrees Celsius.
- `temperatureMin` : Degrees Celsius.
- `temperatureMax` : Degrees Celsius.
- `apparentTemperature` : Degrees Celsius.
- `dewPoint` : Degrees Celsius.
- `windSpeed` : Meters per second.
- `pressure` : Hectopascals.
- `visibility` : Kilometers.

Response Format

API responses consist of a UTF-8-encoded, JSON-formatted object with the following properties:

latitude required

The requested latitude.

longitude required
The requested longitude.

timezone (e.g. `America/New_York`) required
The IANA timezone name for the requested location. This is used for text summaries and for determining when `hourly` and `daily` data block objects begin.

offset deprecated
The current timezone offset in hours. (Use of this property will almost certainly result in Daylight Saving Time bugs. Please use `timezone`, instead.)

currently optional
A data point containing the current weather conditions at the requested location.

minutely optional
A data block containing the weather conditions minute-by-minute for the next hour.

hourly optional
A data block containing the weather conditions hour-by-hour for the next two days.

daily optional
A data block containing the weather conditions day-by-day for the next week.

alerts optional
An alerts array, which, if present, contains any severe weather alerts pertinent to the requested location.

flags optional
A flags object containing miscellaneous metadata about the request.

Data Point Object

A **data point** object contains various properties, each representing the average (unless otherwise specified) of a particular weather phenomenon occurring during a period of time:

an instant in the case of `currently`, a minute for `minutely`, an hour for `hourly`, and a day for `daily`. These properties are:

apparentTemperature optional, only on `hourly` and `currently`
The apparent (or "feels like") temperature in degrees Fahrenheit.

apparentTemperatureHigh optional, only on `daily`
The daytime high apparent temperature.

apparentTemperatureHighTime optional, only on `daily`
The UNIX time representing when the daytime high apparent temperature occurs.

apparentTemperatureLow optional, only on `daily`
The overnight low apparent temperature.

apparentTemperatureLowTime optional, only on `daily`
The UNIX time representing when the overnight low apparent temperature occurs.

apparentTemperatureMax optional, only on `daily`
The maximum apparent temperature during a given date.

apparentTemperatureMaxTime optional, only on `daily`
The UNIX time representing when the maximum apparent temperature during a given date occurs.

apparentTemperatureMin optional, only on `daily`
The minimum apparent temperature during a given date.

apparentTemperatureMinTime optional, only on `daily`
The UNIX time representing when the minimum apparent temperature during a given date occurs.

cloudCover optional
The percentage of sky occluded by clouds, between `0` and `1`, inclusive.

dewPoint optional

The dew point in degrees Fahrenheit.

humidity optional

The relative humidity, between `0` and `1`, inclusive.

icon optional

A machine-readable text summary of this data point, suitable for selecting an icon for display. If defined, this property will have one of the following values: `clear-day`, `clear-night`, `rain`, `snow`, `sleet`, `wind`, `fog`, `cloudy`, `partly-cloudy-day`, or `partly-cloudy-night`. (Developers should ensure that a sensible default is defined, as additional values, such as `hail`, `thunderstorm`, or `tornado`, may be defined in the future.)

moonPhase optional, only on `daily`

The fractional part of the lunation number during the given day: a value of `0` corresponds to a new moon, `0.25` to a first quarter moon, `0.5` to a full moon, and `0.75` to a last quarter moon. (The ranges in between these represent waxing crescent, waxing gibbous, waning gibbous, and waning crescent moons, respectively.)

nearestStormBearing optional, only on `currently`

The approximate direction of the nearest storm in degrees, with true north at 0° and progressing clockwise. (If `nearestStormDistance` is zero, then this value will not be defined.)

nearestStormDistance optional, only on `currently`

The approximate distance to the nearest storm in miles. (A storm distance of `0` doesn't necessarily refer to a storm at the requested location, but rather a storm in the vicinity of that location.)

ozone optional

The columnar density of total atmospheric ozone at the given time in Dobson units.

precipAccumulation optional, only on `hourly`, `currently` and `daily`

The amount of snowfall accumulation expected to occur (over the hour or day, respectively), in inches. (If no snowfall is expected, this property will not be defined.)

precipIntensity optional

The intensity (in inches of liquid water per hour) of precipitation occurring at the given time. This value is **conditional on probability** (that is, assuming any precipitation occurs at all).

precipIntensityError optional

The standard deviation of the distribution of `precipIntensity`. (We only return this property when the full distribution, and not merely the expected mean, can be estimated with accuracy.)

precipIntensityMax optional, only on `daily`

The maximum value of `precipIntensity` during a given day.

precipIntensityMaxTime optional, only on `daily`

The UNIX time of when `precipIntensityMax` occurs during a given day.

precipProbability optional

The probability of precipitation occurring, between `0` and `1`, inclusive.

precipType optional

The type of precipitation occurring at the given time. If defined, this property will have one of the following values: `"rain"`, `"snow"`, or `"sleet"` (which refers to each of freezing rain, ice pellets, and "wintery mix"). (If `precipIntensity` is zero, then this property will not be defined. Additionally, due to the lack of data in our sources, historical `precipType` information is usually estimated, rather than observed.)

pressure optional

The sea-level air pressure in millibars.

summary optional

A human-readable text summary of this data point. (This property has millions of possible values, so don't use it for automated purposes: use the `icon` property, instead!)

sunriseTime optional, only on `daily`

The UNIX time of when the sun will rise during a given day.

sunsetTime optional, only on `daily`

The UNIX time of when the sun will set during a given day.

temperature optional, only on `hourly` and `currently`

The air temperature in degrees Fahrenheit.

temperatureHigh optional, only on `daily`

The daytime high temperature.

temperatureHighTime optional, only on `daily`

The UNIX time representing when the daytime high temperature occurs.

temperatureLow optional, only on `daily`

The overnight low temperature.

temperatureLowTime optional, only on `daily`

The UNIX time representing when the overnight low temperature occurs.

temperatureMax optional, only on `daily`

The maximum temperature during a given date.

temperatureMaxTime optional, only on `daily`

The UNIX time representing when the maximum temperature during a given date occurs.

temperatureMin optional, only on `daily`

The minimum temperature during a given date.

temperatureMinTime optional, only on `daily`

The UNIX time representing when the minimum temperature during a given date occurs.

time required

The UNIX time at which this data point begins. `minutely` data point are always aligned to the top of the minute, `hourly` data point objects to the top of the hour, `daily` data point objects to midnight of the day, and `currently` data point object to the point of time provided all according to the local time zone.

uvIndex optional

The UV index.

uvIndexTime optional, only on `daily`

The UNIX time of when the maximum `uvIndex` occurs during a given day.

visibility optional

The average visibility in miles, capped at 10 miles.

windBearing optional

The direction that the wind is coming **from** in degrees, with true north at 0° and progressing clockwise. (If `windSpeed` is zero, then this value will not be defined.)

windGust optional

The wind gust speed in miles per hour.

windGustTime optional, only on `daily`

The time at which the maximum wind gust speed occurs during the day.

windSpeed optional

The wind speed in miles per hour.

Data Block Object

A **data block** object represents the various weather phenomena occurring over a period of time. Such objects contain the following properties:

data required

An array of data points, ordered by time, which together describe the weather conditions at the requested location over time.

summary optional

A human-readable summary of this data block.

icon optional

A machine-readable text summary of this data block. (May take on the same values as the `icon` property of data points.)

Alerts Array

The **alerts array** contains objects representing the severe weather warnings issued for the requested location by a governmental authority (please see our data sources page for a list of sources). Objects in the alerts array contain the following properties:

description required
A detailed description of the alert.

expires required
The UNIX time at which the alert will expire.

regions required
An array of strings representing the names of the regions covered by this weather alert.

severity required
The severity of the weather alert. Will take one of the following values: `"advisory"` (an individual should be aware of potentially severe weather), `"watch"` (an individual should prepare for potentially severe weather), or `"warning"` (an individual should take immediate action to protect themselves and others from potentially severe weather).

time required
The UNIX time at which the alert was issued.

title required
A brief description of the alert.

uri required
An HTTP(S) URI that one may refer to for detailed information about the alert.

Flags Object

The **flags object** contains various metadata information related to the request. This object may optionally contain any of the following properties:

darksky-unavailable optional

The presence of this property indicates that the Dark Sky data source supports the given location, but a temporary error (such as a radar station being down for maintenance) has made the data unavailable.

nearest-station required

The distance to the nearest weather station that contributed data to this response. Note, however, that many other stations may have also been used; this value is primarily for debugging purposes. This property's value is in miles (if US units are selected) or kilometers (if SI units are selected).

sources required

This property contains an array of IDs for each data source utilized in servicing this request.

units required

Indicates the units which were used for the data in this request.

Response Headers

The API will set the following HTTP response headers to values useful to developers:

Cache-Control optional

Set to a conservative value for data caching purposes based on the data present in the response body.

Expires deprecated

Set to a conservative value for data caching purposes based on the data present in the response body.

X-Forecast-API-Calls optional

The number of API requests made by the given API key for today.

X-Response-Time optional

The server-side response time of the request.

Notes

- Never make any assumptions about the presence of data or lengths of arrays. For example, a lack of data in our data sources may cause data to be missing; or Daylight Savings Time may cause a day to consist of 23 or 25 hours (instead of the usual 24); or, at high latitudes, a given day may not have a sunrise or sunset. **Always** check for the presence of data before trying to use it.
- Overnight low temperatures usually occur around dawn, and so will usually occur on a different date from the daytime high temperatures. (This is the primary difference between `temperatureLow` and `temperatureMin`.)
- Summaries on the `hourly` data block actually only cover up to a maximum of 24 hours, rather than the full time period in the data block. We found that covering the full 48 hours could be, in a number of circumstances, far too wordy to be practical.
- Summaries and icons on `daily` data point objects actually cover the period from 4AM to 4AM, rather than the stated time period of midnight to midnight. We found that the summaries so generated were less awkward.
- The Forecast Data API supports HTTP compression. We heartily recommend using it, as it will make responses much smaller over the wire. To enable it, simply add an `Accept-Encoding: gzip` header to your request. (Most HTTP client libraries wrap this functionality for you, please consult your library's documentation for details.)