Overview

Copy

Moon API provides accurate and real-time lunar data through a RESTful API. It offers a range of endpoints to retrieve moon phases, lunar data, and astronomical calculations. Whether you're building a lunar-themed application, planning an event, or simply curious about the moon, Moon API is your go-to source for lunar information.

Getting Started

🌕 **Try Live Demo**
Interactive Astronomical Dashboard

🔑 **Get API Key**
Start Building

To get started with Moon API, follow these simple steps:

1. Sign up for an account on our platform.
2. Generate an API key from your account dashboard.
3. Use the API key in your API requests to authenticate.

Authentication

All API requests must include an API key. You can add the API key to your requests in the following ways:

- As a query parameter: `?api_key=your_api_key`
- As a header: `Authorization: Bearer your_api_key`

Endpoints

Moon API offers a variety of endpoints to cater to different needs and plans. Below are the available endpoints:

GET /basic

All Plans

Get basic moon data including phase name and stage.

Query Parameters:

Parameter	Type	Required	Description	Plan Availability
lat	number	optional	Latitude for location-specific data	Basic: ❌ Pro: ✅ (1 decimal) Ultra: ✅ (2 decimals) Mega: ✅ (3 decimals)
lon	number	optional	Longitude for location-specific data	Basic: ❌ Pro: ✅ (1 decimal) Ultra: ✅ (2 decimals) Mega: ✅ (3 decimals)

Response:

```
{
  "phase_name": "Waxing gibbous",
  "stage": "waxing",
  "illumination": "69%",
  "days_until_next_full_moon": 5,
  "days_until_next_new_moon": 21
}
```

Field Descriptions:

`phase_name` : (string) Name of the current moon phase
`stage` : (string) Stage of the moon's cycle (waxing/waning)
`days_until_next_full_moon` : (integer) Days until next full moon
`days_until_next_new_moon` : (integer) Days until next new moon

GET /advanced

All Plans

Get comprehensive moon data with detailed astronomical information.

Query Parameters:

Parameter	Type	Required	Description	Plan Availability
lat	number	optional	Latitude for location-specific data	Basic: ❌ Pro: ✅ (1 decimal) Ultra: ✅ (2 decimals) Mega: ✅ (3 decimals)
lon	number	optional	Longitude for location-specific data	Basic: ❌ Pro: ✅ (1 decimal) Ultra: ✅ (2 decimals) Mega: ✅ (3 decimals)
date	string	optional	Historical date (YYYY-MM-DD)	Basic: ❌ Pro: ❌ Ultra: ✅ (2 years) Mega: ✅ (10 years)

Response:


```
{
  "timestamp": 1738930410,
  "datestamp": "Fri, 07 Feb 2025 12:13:30 +0000",
  "plan": "BASIC",
  "sun": {
    "sunrise": 1738909680,
    "sunrise_timestamp": "07:28",
    "sunset": 1738944000,
    "sunset_timestamp": "17:00",
    "solar_noon": "12:14",
    "day_length": "09:32",
    "position": {
      "altitude": -51.4008,
      "azimuth": 331.0116,
      "distance": 147466465.745
    },
    "next_solar_eclipse": {
      "timestamp": 1743241716,
      "datestamp": "Sat, 29 Mar 2025 10:48:36 +0100",
      "type": "Partial Solar Eclipse",
      "visibility_regions": "nw Africa, Europe, n Russia"
    }
  },
  "moon": {
    "phase": 0.3118820182876104,
    "phase_name": "Waxing gibbous",
    "major_phase": "First Quarter",
    "stage": "waxing",
    "illumination": "69%",
    "age_days": 9,
    "lunar_cycle": "31.19%",
```

```
    "emoji": "🌒",
    "zodiac": {
      "sun_sign": "Aquarius",
      "moon_sign": "Gemini"
    },
    "moonrise": "10:28",
    "moonrise_timestamp": 1738920480,
    "moonset": "02:43",
    "moonset_timestamp": 1738892580,
    "detailed": {
      "position": {
        "altitude": -23.4953,
        "azimuth": 282.5668,
        "distance": 376978.4627,
        "parallactic_angle": 38.236,
        "phase_angle": 112.28
      },
      "visibility": {
        "visible_hours": 0,
        "best_viewing_time": "20:54",
        "visibility_rating": "Good",
        "illumination": "68.95%",
        "viewing_conditions": {
          "phase_quality": "Good for observing surface detail along terminator line",
          "recommended_equipment": {
            "filters": "No filters needed",
            "telescope": "4-inch or larger recommended",
            "best_magnification": "High magnification (100-200x) for crater detail"
          }
        }
      }
    },
    "events": {
      "moonrise_visible": false,
      "moonset_visible": true,
      "optimal_viewing_period": {
        "start_time": "18:00",
        "end_time": "22:00",
        "duration_hours": 4,
        "viewing_quality": "Good for crater observation along terminator",
        "recommendations": [
          "High in sky during evening hours",
          "Excellent time to observe crater shadows along terminator line",
          "Medium to high magnification recommended for detail"
        ]
      }
    }
  },
  "location": {
    "latitude": "51.00000",
    "longitude": "0.00000",
    "precision": 0,
    "using_default_location": true,
    "note": "Using default London coordinates. Upgrade from BASIC plan to use custom coordinates."
  }
}
```


Field Descriptions:

`timestamp` : (integer) UNIX timestamp of the moon phase calculation

`datestamp` : (string) Date and time in UTC format

`plan` : (string) Current API plan level (BASIC, PRO, ULTRA, MEGA)

`sun` : (object) Solar data including:
`sunrise` : (integer) UNIX timestamp for today's sunrise

`sunrise_timestamp` : (string) The time of sunrise (HH:mm)

`sunset` : (integer) UNIX timestamp for today's sunset

`sunset_timestamp` : (string) The time of sunset (HH:mm)

`solar_noon` : (string) The time when the sun reaches its highest point

`day_length` : (string) Duration of daylight (HH:mm)

`sun_altitude` : (float) The altitude of the sun above horizon in degrees

`sun_distance` : (float) The distance between Earth and sun in kilometers

`sun_azimuth` : (float) The azimuth of the sun's position in degrees

`next_solar_eclipse` : (object) Next solar eclipse details:
`timestamp` : (integer) UNIX timestamp of next solar eclipse

`datestamp` : (string) Date and time of next solar eclipse

`type` : (string) Type of solar eclipse (Total, Partial, Annular)

`visibility_regions` : (string) Regions where the eclipse will be visible

`moon` : (object) Lunar data including:
`phase` : (float) Moon phase as a number between 0 and 1

`phase_name` : (string) Name of the current moon phase

`major_phase` : (string) Major phase name (New Moon, First Quarter, etc)

`stage` : (string) Stage of the moon's cycle (waxing/waning)

`illumination` : (string) Percentage of moon's surface illuminated

`age_days` : (integer) Age of the moon in days since new moon

`lunar_cycle` : (string) Percentage through lunar cycle

`emoji` : (string) Moon phase emoji (🌑, 🌒, 🌓, etc)

`zodiac` : (object) Zodiac information:
`sun_sign` : (string) Current zodiac sign of the sun

`moon_sign` : (string) Current zodiac sign of the moon

`moonrise` : (string) Time of moonrise (HH:mm)

`moonrise_timestamp` : (integer) UNIX timestamp for moonrise

`moonset` : (string) Time of moonset (HH:mm)

`moonset_timestamp` : (integer) UNIX timestamp for moonset

`moon_altitude` : (float) The altitude of the moon above horizon in degrees

`moon_distance` : (float) Distance between Earth and moon in kilometers

`moon_azimuth` : (float) The azimuth of the moon's position in degrees

`moon_parallactic_angle` : (float) The parallactic angle of the moon in degrees

`next_lunar_eclipse` : (object) Next lunar eclipse details:
`timestamp` : (integer) UNIX timestamp of next lunar eclipse

`datestamp` : (string) Date and time of next lunar eclipse

`type` : (string) Type of lunar eclipse (Total, Partial, Penumbral)

`visibility_regions` : (string) Regions where the eclipse will be visible

`detailed` : (object) Detailed moon information:
`position` : (object) Current position details:
`altitude` : (float) Current altitude of the moon in degrees

`azimuth` : (float) Current azimuth of the moon in degrees

`phase_angle` : (float) Current phase angle in degrees

`illumination` : (float) Current illumination percentage

`visibility` : (object) Visibility information:
`visible_hours` : (integer) Number of hours the moon is visible

`best_viewing_time` : (string) Best time for moon observation (HH:mm)

`visibility_rating` : (string) Overall visibility rating

`illumination` : (string) Current illumination percentage

`viewing_conditions` : (object) Viewing conditions:
`phase_quality` : (string) Description of viewing quality for current phase

`recommended_equipment` : (object) Equipment recommendations:
`filters` : (string) Recommended filters for observation

`telescope` : (string) Recommended telescope specifications

`best_magnification` : (string) Recommended magnification range

`upcoming_phases` : (object) Upcoming moon phases:
`new_moon` : (object) New moon timings:
`last` : (object) Details of last new moon (timestamp, datestamp, days_ago)

`next` : (object) Details of next new moon (timestamp, datestamp, days_ahead)

`first_quarter` : (object) First quarter timings:
`last` : (object) Details of last first quarter (timestamp, datestamp, days_ago)

`next` : (object) Details of next first quarter (timestamp, datestamp, days_ahead)

`full_moon` : (object) Full moon timings:
`last` : (object) Details of last full moon including name and description

`next` : (object) Details of next full moon including name and description

`last_quarter` : (object) Last quarter timings:
`last` : (object) Details of last quarter (timestamp, datestamp, days_ago)

`next` : (object) Details of next last quarter (timestamp, datestamp, days_ahead)

`illumination_details` : (object) Detailed illumination data:
`percentage` : (float) Precise illumination percentage

`visible_fraction` : (float) Visible fraction of the moon's surface

`phase_angle` : (float) Current phase angle in degrees

`events` : (object) Moon event information:
`moonrise_visible` : (boolean) Whether moonrise is visible from location

`moonset_visible` : (boolean) Whether moonset is visible from location

`optimal_viewing_period` : (object) Best viewing details:
`start_time` : (string) Start of optimal viewing period (HH:mm)

`end_time` : (string) End of optimal viewing period (HH:mm)

`duration_hours` : (float) Duration of optimal viewing period in hours

`viewing_quality` : (string) Description of viewing conditions

`recommendations` : (array) List of viewing recommendations

`moon_phases` : (object) Moon phase timings:
`new_moon` : (object) New moon timings:
`last` : (object) Details of last new moon with timestamp and days ago

`next` : (object) Details of next new moon with timestamp and days ahead

`first_quarter` : (object) First quarter timings:
`last` : (object) Details of last first quarter with timestamp and days ago

`next` : (object) Details of next first quarter with timestamp and days ahead

`full_moon` : (object) Full moon timings:
`last` : (object) Details of last full moon including name and description

`next` : (object) Details of next full moon including name and description

`last_quarter` : (object) Last quarter timings:
`last` : (object) Details of last quarter with timestamp and days ago

`next` : (object) Details of next last quarter with timestamp and days ahead

`location` : (object) Location information:
`latitude` : (string) Latitude of the location used for calculations

`longitude` : (string) Longitude of the location used for calculations

`precision` : (integer) Precision level of coordinates

`using_default_location` : (boolean) Whether default location is being used

`note` : (string) Additional information about location usage

Historical Data Access

Access historical data by adding the `date` parameter:

- Ultra plan: Access up to 2 years of historical data
- Mega plan: Access up to 10 years of historical data

  Precise calculations based on specific coordinates worldwide

GET /calendar

All Plans

Get a beautiful moon phase calendar in HTML or Markdown format.

Query Parameters:

Parameter	Type	Required	Description	Plan Availability
format	string	optional	Response format (html/markdown)	All Plans ✅
month	integer	optional	Calendar month (1-12)	All Plans ✅
year	integer	optional	Calendar year (YYYY)	All Plans ✅

Response Formats:

HTML Format

Returns an HTML table with moon phase emojis, perfect for web integration.

Markdown Format

Returns a markdown-formatted table, ideal for documentation.

Example Response (Markdown):

```
| Month | 1 | 2 | ... | 31 |
|---|---|---|---|---|
| January | 🌒 | 🌓 | ... | 🌕 |
```

GET /phase

All Plans

Get current moon phase as plain text, perfect for simple integrations.

Query Parameters:

Parameter	Type	Required	Description	Plan Availability
lat	number	optional	Latitude for location-specific data	Same as /advanced endpoint
lon	number	optional	Longitude for location-specific data	Same as /advanced endpoint

Response:

```
Waxing gibbous
```

GET /emoji

All Plans

Get current moon phase as an emoji, perfect for social media integration.

Query Parameters:

Parameter	Type	Required	Description	Plan Availability
lat	number	optional	Latitude for location-specific data	Same as /advanced endpoint
lon	number	optional	Longitude for location-specific data	Same as /advanced endpoint

Response:

Code Examples

Here are examples of how to use Moon API in various programming languages:

JavaScript | Python | PHP | Arduino | cURL

```
const options = {
  method: 'GET',
  url: 'https://moon-phase.p.rapidapi.com/advanced',
  headers: {
    'X-RapidAPI-Key': 'your-api-key',
    'X-RapidAPI-Host': 'moon-phase.p.rapidapi.com'
  }
};

try {
  const response = await axios.request(options);
  console.log(response.data);
} catch (error) {
  console.error(error);
}
```

Features

Feature	Basic	Pro	Ultra	Mega
All Endpoints	✅	✅	✅	✅
Requests/Month	100	10,000	100,000	Custom
Requests/Second	1	10	100	Custom
Custom Coordinates	❌	✅ (1 decimal)	✅ (2 decimals)	✅ (3 decimals)
Historical Data	❌	❌	✅ (2 years)	✅ (10 years)
Response Time	Standard	Fast	Faster	Fastest
Support	Community	Email	Priority Email	24/7 Priority

Best Practices

🚀 Performance

- Cache responses when possible
- Use the simplest endpoint for your needs
- Implement rate limiting on your side
- Handle API responses asynchronously

🛡️ Security

- Keep your API key secure
- Use HTTPS for all requests
- Implement proper error handling
- Validate user inputs before sending

📈 Optimization

- Use appropriate coordinate precision
- Batch requests when possible
- Implement proper error handling
- Monitor your API usage

Troubleshooting

Having issues with the API? Check out our FAQ page for common questions and answers.

Common Issues

- **Authentication Errors:** Make sure your API key is valid and properly included in the request.
- **Rate Limiting:** Check your current plan's rate limits and monitor your usage.
- **Invalid Parameters:** Verify that all required parameters are included and properly formatted.

Error Codes

- `401` - Invalid or missing API key
- `403` - Rate limit exceeded
- `404` - Endpoint not found
- `422` - Invalid parameters
- `500` - Server error

If you're still experiencing issues, please contact our support team.