Basic_SiderealLab

April 26, 2025

1 SiderealLab - Measure Your Local Earth's Rotational Speed by Using an Equatorial Mount (Basic Edition)

2 SiderealLab (Basic Edition)

2.1 I. Purpose of the Experiment

This experiment is designed to help users understand how the Earth's rotational velocity at the surface varies with latitude.

By inputting a latitude value, the program calculates and outputs the corresponding linear velocity based on a fundamental physical model of Earth's rotation.

It aims to deepen the understanding of Earth's dynamic properties while providing practical experience in applying physical formulas and computational techniques.

2.1.1 1. Background

- The Earth rotates at an approximately constant angular velocity.
- The linear velocity at the Earth's surface varies significantly with latitude.
- Theoretically, the linear velocity is maximal at the equator and approaches zero at the poles.
- This experiment simplifies the calculation by assuming a perfect spherical Earth and ignoring the effects of Earth's oblateness and tectonic motions.

  Note: This experiment assumes a perfect spherical Earth, akin to the idealized assumptions used in introductory physics when studying uniform circular motion and Newtonian gravitation. This simplification helps learners focus on fundamental concepts. In reality, the Earth is an oblate spheroid, with a slightly larger equatorial radius. Future professional editions will incorporate GPS data and the WGS84 standard model for greater accuracy.

2.1.2 2. Data Accuracy Considerations

2.1 Earth Radius Assumption

- A mean Earth radius of **6371 km** is used.
- Realistically, the Earth's equatorial radius is slightly larger than the polar radius.
- Advanced experiments should apply latitude-specific corrections.

2.2 Rotational Period

- The **sidereal day** (approximately **86164 seconds**) is used for calculations.
- Using the solar day (86400 seconds) introduces minor discrepancies.

2.3 Latitude Input Range

- Latitude values must be within **-90° to 90°**.
- Inputs outside this range should trigger warnings or corrections.

2.4 Consistency of Units

- Earth's radius is measured in kilometers (km).
- Rotational period is measured in seconds (s).
- All quantities must be consistent to avoid errors.

2.5 Simplified Physical Model

- Atmospheric motion and plate tectonics are not considered.
- A static, idealized spherical Earth model is adopted.

2.2 II. Experiment Objectives

1. Understand the basics of Earth's rotation.
2. Calculate Earth's rotational speed using observational data.
3. Analyze the impact of latitude on rotational speed.
4. Conduct hands-on data analysis and calculation exercises.

2.3 III. Message to Science Enthusiasts

This foundational version of the experiment is intended for astronomy enthusiasts, physics learners, high school students, and anyone passionate about science and the natural world.

> **Reminder:** Science is everywhere. Behind every observable phenomenon lie physical laws, and programming offers a powerful tool to explore these laws from new perspectives. Through simple experiments like this, learners can foster critical thinking, nurture curiosity, and build a deeper connection with the universe.

2.4 IV. Step-by-Step Instructions

2.4.1 Step 1 - Calibrate the Equatorial Mount

Ensure the equatorial mount is accurately aligned with the celestial pole to enable precise observations.

2.4.2 Step 2 - Find Sirius and Record Data

Identify a bright star (e.g., Sirius), record the current UTC time, and note the star's Right Ascension (RA) and Declination (Dec) from the mount's settings or connected device.

> RA is analogous to terrestrial longitude, and Dec is analogous to latitude.

2.4.3 Step 3 - Recording Time and Observation Duration

Keep the mount stationary for at least 24 hours. Monitor the stopwatch as it approaches the 23rd hour, recalibrate the mount, and record new measurements. This assists in determining the sidereal day length.

2.4.4 Step 4 - Data Analysis and Calculations

In the Basic Edition, users manually input their local latitude and calculate the Earth's rotation speed accordingly.

2.4.5 Step 5 - Visualization of Rotation Speed

Plot a graph to illustrate how Earth's rotational speed varies with latitude. The speed peaks at the equator and decreases towards the poles.

2.4.6 Step 6 - Final Summary and Explanation

Use a GPS app or website to find the local latitude. Record the necessary physical parameters:

Term	Meaning	Value
R_e	Earth's Equatorial Radius	6378.137 km
R_φ	Earth's Radius at Your Location	Calculated
φ	Your Local Latitude	Measured
ΔT	Time Difference (two nights)	Calculated
v	Earth's Rotational Speed	Final Result

2.5 V. Summary for the Basic Edition

This foundational exercise offers a simple but powerful introduction to studying Earth's rotation through accessible observations and basic data analysis. Users are encouraged to maintain detailed notes and reflect on their findings to deepen their scientific understanding.

2.6 VI. Data Analysis (Step by Step)

2.6.1 Import required or useful libraries

[17]:
```
pip install streamlit

pip install
```

```
Collecting streamlit
  Downloading streamlit-1.40.1-py2.py3-none-any.whl.metadata (8.5 kB)
Collecting altair<6,>=4.0 (from streamlit)
  Downloading altair-5.4.1-py3-none-any.whl.metadata (9.4 kB)
Collecting blinker<2,>=1.0.0 (from streamlit)
  Downloading blinker-1.8.2-py3-none-any.whl.metadata (1.6 kB)
Requirement already satisfied: cachetools<6,>=4.0 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from streamlit) (4.2.2)
```

```
Requirement already satisfied: click<9,>=7.0 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from streamlit) (8.1.7)
Requirement already satisfied: numpy<3,>=1.20 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from streamlit) (1.24.3)
Requirement already satisfied: packaging<25,>=20 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from streamlit) (23.2)
Requirement already satisfied: pandas<3,>=1.4.0 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from streamlit) (2.0.3)
Requirement already satisfied: pillow<12,>=7.1.0 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from streamlit) (10.2.0)
Requirement already satisfied: protobuf<6,>=3.20 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from streamlit) (3.20.3)
Collecting pyarrow>=7.0 (from streamlit)
  Downloading pyarrow-17.0.0-cp38-cp38-macosx_10_15_x86_64.whl.metadata (3.3 kB)
Requirement already satisfied: requests<3,>=2.27 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from streamlit) (2.31.0)
Requirement already satisfied: rich<14,>=10.14.0 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from streamlit) (13.3.5)
Requirement already satisfied: tenacity<10,>=8.1.0 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from streamlit) (8.2.2)
Requirement already satisfied: toml<2,>=0.10.1 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from streamlit) (0.10.2)
Requirement already satisfied: typing-extensions<5,>=4.3.0 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from streamlit) (4.9.0)
Collecting gitpython!=3.1.19,<4,>=3.0.7 (from streamlit)
  Downloading GitPython-3.1.44-py3-none-any.whl.metadata (13 kB)
Collecting pydeck<1,>=0.8.0b4 (from streamlit)
  Downloading pydeck-0.9.1-py2.py3-none-any.whl.metadata (4.1 kB)
Requirement already satisfied: tornado<7,>=6.0.3 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from streamlit) (6.3.3)
Requirement already satisfied: jinja2 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
altair<6,>=4.0->streamlit) (3.1.3)
Requirement already satisfied: jsonschema>=3.0 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
altair<6,>=4.0->streamlit) (4.19.2)
Collecting narwhals>=1.5.2 (from altair<6,>=4.0->streamlit)
  Downloading narwhals-1.33.0-py3-none-any.whl.metadata (9.2 kB)
Collecting typing-extensions<5,>=4.3.0 (from streamlit)
  Downloading typing_extensions-4.13.1-py3-none-any.whl.metadata (3.0 kB)
Collecting gitdb<5,>=4.0.1 (from gitpython!=3.1.19,<4,>=3.0.7->streamlit)
  Downloading gitdb-4.0.12-py3-none-any.whl.metadata (1.2 kB)
Requirement already satisfied: python-dateutil>=2.8.2 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
pandas<3,>=1.4.0->streamlit) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
pandas<3,>=1.4.0->streamlit) (2023.3.post1)
```

```
Requirement already satisfied: tzdata>=2022.1 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
pandas<3,>=1.4.0->streamlit) (2023.3)
Requirement already satisfied: charset-normalizer<4,>=2 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
requests<3,>=2.27->streamlit) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
requests<3,>=2.27->streamlit) (3.4)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
requests<3,>=2.27->streamlit) (1.26.18)
Requirement already satisfied: certifi>=2017.4.17 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
requests<3,>=2.27->streamlit) (2024.2.2)
Requirement already satisfied: markdown-it-py<3.0.0,>=2.2.0 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
rich<14,>=10.14.0->streamlit) (2.2.0)
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
rich<14,>=10.14.0->streamlit) (2.15.1)
Collecting smmap<6,>=3.0.1 (from
gitdb<5,>=4.0.1->gitpython!=3.1.19,<4,>=3.0.7->streamlit)
  Downloading smmap-5.0.2-py3-none-any.whl.metadata (4.3 kB)
Requirement already satisfied: MarkupSafe>=2.0 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
jinja2->altair<6,>=4.0->streamlit) (2.1.3)
Requirement already satisfied: attrs>=22.2.0 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
jsonschema>=3.0->altair<6,>=4.0->streamlit) (23.1.0)
Requirement already satisfied: importlib-resources>=1.4.0 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
jsonschema>=3.0->altair<6,>=4.0->streamlit) (6.1.1)
Requirement already satisfied: jsonschema-specifications>=2023.03.6 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
jsonschema>=3.0->altair<6,>=4.0->streamlit) (2023.7.1)
Requirement already satisfied: pkgutil-resolve-name>=1.3.10 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
jsonschema>=3.0->altair<6,>=4.0->streamlit) (1.3.10)
Requirement already satisfied: referencing>=0.28.4 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
jsonschema>=3.0->altair<6,>=4.0->streamlit) (0.30.2)
Requirement already satisfied: rpds-py>=0.7.1 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from
jsonschema>=3.0->altair<6,>=4.0->streamlit) (0.10.6)
Requirement already satisfied: mdurl~=0.1 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from markdown-it-
py<3.0.0,>=2.2.0->rich<14,>=10.14.0->streamlit) (0.1.0)
```

```
Requirement already satisfied: six>=1.5 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from python-
dateutil>=2.8.2->pandas<3,>=1.4.0->streamlit) (1.16.0)
Requirement already satisfied: zipp>=3.1.0 in
/Users/et137/anaconda3/lib/python3.8/site-packages (from importlib-
resources>=1.4.0->jsonschema>=3.0->altair<6,>=4.0->streamlit) (3.17.0)
Downloading streamlit-1.40.1-py2.py3-none-any.whl (8.6 MB)

8.6/8.6 MB 8.8 MB/s eta 0:00:00m eta
0:00:010:01:01m
Downloading altair-5.4.1-py3-none-any.whl (658 kB)

658.1/658.1 kB 9.4 MB/s eta 0:00:00[31m13.7 MB/s
eta 0:00:01
Downloading blinker-1.8.2-py3-none-any.whl (9.5 kB)
Downloading GitPython-3.1.44-py3-none-any.whl (207 kB)

207.6/207.6 kB 6.1 MB/s eta 0:00:00
Downloading pyarrow-17.0.0-cp38-cp38-macosx_10_15_x86_64.whl (29.0 MB)

29.0/29.0 MB 12.7 MB/s eta 0:00:00m eta
0:00:010:01:01
Downloading pydeck-0.9.1-py2.py3-none-any.whl (6.9 MB)

6.9/6.9 MB 12.7 MB/s eta 0:00:00m eta
0:00:010:01:01
Downloading typing_extensions-4.13.1-py3-none-any.whl (45 kB)

45.7/45.7 kB 1.5 MB/s eta 0:00:00
Downloading gitdb-4.0.12-py3-none-any.whl (62 kB)

62.8/62.8 kB 2.4 MB/s eta 0:00:00
Downloading narwhals-1.33.0-py3-none-any.whl (322 kB)

322.8/322.8 kB 6.7 MB/s eta 0:00:000:00:01
Downloading smmap-5.0.2-py3-none-any.whl (24 kB)
Installing collected packages: typing-extensions, smmap, pyarrow, narwhals,
blinker, pydeck, gitdb, gitpython, altair, streamlit
  Attempting uninstall: typing-extensions
    Found existing installation: typing_extensions 4.9.0
    Uninstalling typing_extensions-4.9.0:
      Successfully uninstalled typing_extensions-4.9.0
Successfully installed altair-5.4.1 blinker-1.8.2 gitdb-4.0.12 gitpython-3.1.44
narwhals-1.33.0 pyarrow-17.0.0 pydeck-0.9.1 smmap-5.0.2 streamlit-1.40.1 typing-
extensions-4.13.1

[notice] A new release of pip is
available: 23.0.1 -> 25.0.1
```

```
[notice] To update, run:
pip install --upgrade pip
Note: you may need to restart the kernel to use updated packages.
```

[19]:
```
pip install geopy
```

```
Collecting geopy
  Downloading geopy-2.4.1-py3-none-any.whl.metadata (6.8 kB)
Collecting geographiclib<3,>=1.52 (from geopy)
  Downloading geographiclib-2.0-py3-none-any.whl.metadata (1.4 kB)
Downloading geopy-2.4.1-py3-none-any.whl (125 kB)

125.4/125.4 kB 2.4 MB/s eta 0:00:00[36m0:00:01
Downloading geographiclib-2.0-py3-none-any.whl (40 kB)

40.3/40.3 kB 1.4 MB/s eta 0:00:00
Installing collected packages: geographiclib, geopy
Successfully installed geographiclib-2.0 geopy-2.4.1

[notice] A new release of pip is
available: 23.0.1 -> 25.0.1
[notice] To update, run:
pip install --upgrade pip
Note: you may need to restart the kernel to use updated packages.
```

[1]:
```
import numpy as np
import matplotlib.pyplot as plt
```

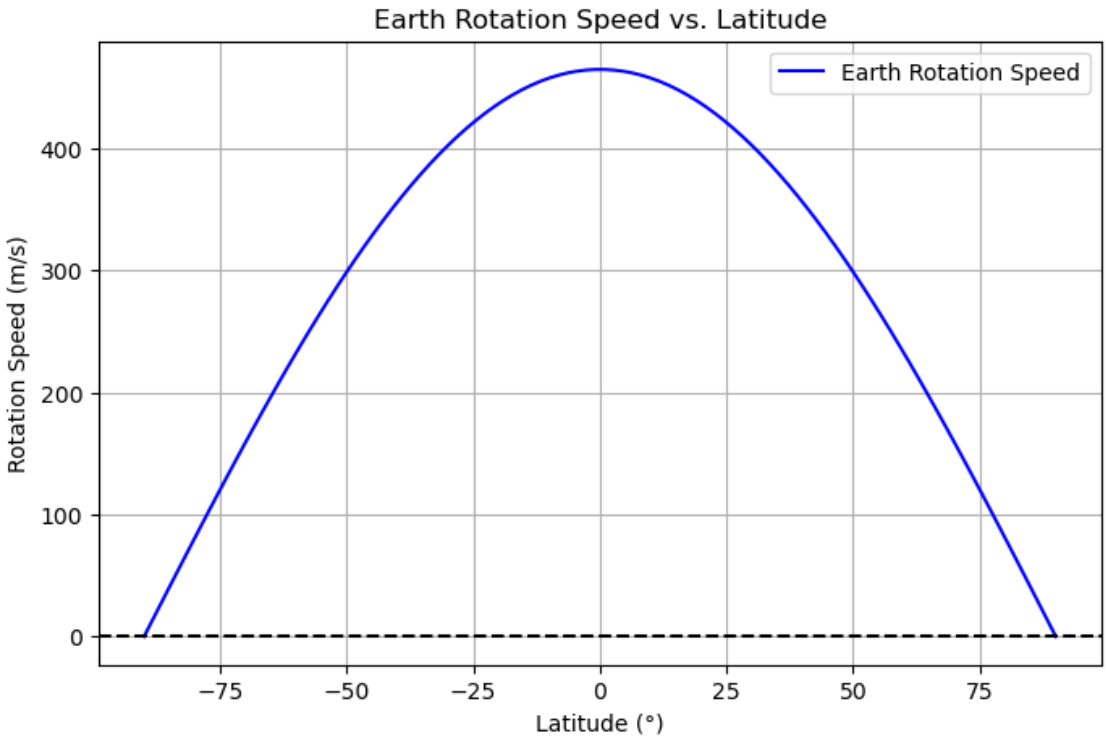
The code section below as the Standard version of codes, it needs users input all data and numbers manually

1. Calculate your local rotation speed with graph (Find and Enter data manually)

[2]:
```
def earth_rotation_speed(latitude):
    """
    calculate the Earth's rotational speed m/s

    parametre:
        latitude (float): latitude degrees

    return:
        float: the matched Earth's rotational speed m/s
    """
    omega = 7.292e-5  # Earth's angular velocity (rad/s)
    R = 6.371e6  # Earth's mean radius (m)
    phi = np.radians(latitude)  # degree convert to rad
    v = omega * R * np.cos(phi)
    return v
```

```
# Demo: Calculate the rotational speed at a specific and customised latitude
latitude = -30  # Enter you local latitude manually, such as mine is -30° in␣
 ↪latitude, it represents a location in the Southern Hemisphere. A positive␣
 ↪latitude represents a location in the Northern Hemisphere
speed = earth_rotation_speed(latitude)
print(f"At latitude {latitude}°, the Earth's rotation speed is {speed:.2f} m/s.
 ↪")

# Demo for Data Viz: Earth's rotational speed numbers at different latitude␣
 ↪points
latitudes = np.linspace(-90, 90, 100)  # latitude rangs from -90° to 90°
speeds = earth_rotation_speed(latitudes)

plt.figure(figsize=(8, 5))
plt.plot(latitudes, speeds, label="Earth Rotation Speed", color="b")
plt.xlabel("Latitude (°)")
plt.ylabel("Rotation Speed (m/s)")
plt.title("Earth Rotation Speed vs. Latitude")
plt.axhline(0, color='black', linestyle="--")
plt.legend()
plt.grid()
plt.show()
```

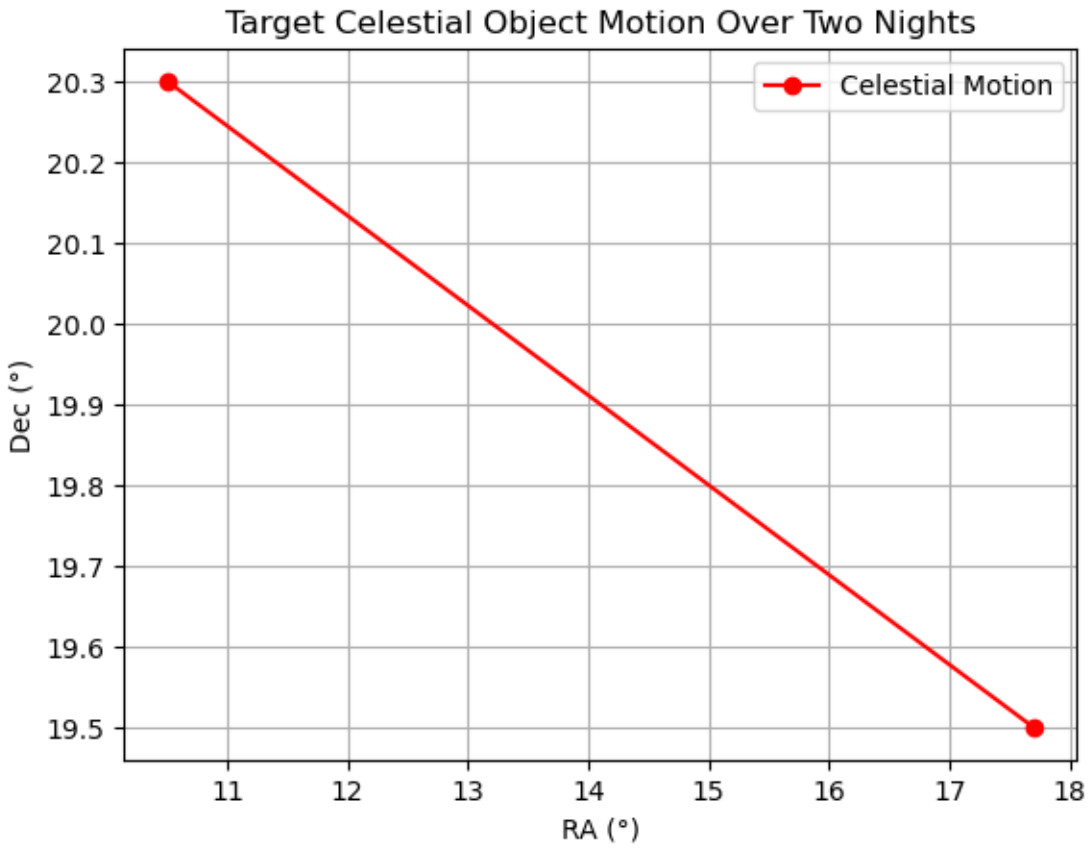
```
At latitude -30°, the Earth's rotation speed is 402.33 m/s.
```



2. Calculate the change of positions for a consecutive two nights manually

[3]:
```
# Calculate the change of positions for a consecutive two nights
def celestial_motion(ra1, dec1, ra2, dec2, time_diff):
    """
    Calculate the angular motion rate of the target star between two␣
 ↪consecutive nights (degrees/h)

    Parameters:
        ra1, dec1: Right Ascension & Declination on the 1st Night (in degrees)
        ra2, dec2: Right Ascension & Declination on the 2nd (next) Night (in␣
 ↪degrees)
        time_diff: The real time interval between the two observations (in␣
 ↪hours)

    Return:
        Angular displacement (degrees), motion rate (degrees/hour)
    """
    delta_ra = ra2 - ra1
    delta_dec = dec2 - dec1
    angular_shift = np.sqrt(delta_ra**2 + delta_dec**2)  # angular displacement
    motion_rate = angular_shift / time_diff  # angular speed
    return angular_shift, motion_rate

# Demo
ra1, dec1 = 10.5, 20.3  # Night 1's RA and Dec
ra2, dec2 = 17.7, 19.5  # Night 2's RA and Dec
time_diff = 23.8  # 24 hours

angular_shift, motion_rate = celestial_motion(ra1, dec1, ra2, dec2, time_diff)
print(f"angular shift: {angular_shift:.4f} degrees, motion rate: {motion_rate:.
 ↪4f} degrees/hour")

# Data Viz
plt.plot([ra1, ra2], [dec1, dec2], 'ro-', label="Celestial Motion")
plt.xlabel("RA (°)")
plt.ylabel("Dec (°)")
plt.title("Target Celestial Object Motion Over Two Nights")
plt.legend()
plt.grid()
plt.show()
```

```
angular shift: 7.2443 degrees, motion rate: 0.3044 degrees/hour
```



3. Enter Your local radius and latitude manually (Find and Enter your local radius manually)

[4]:
```
"""
Please search your latitude value and calculate your local Earth circle radius␣
 ↪by the formula: R_local = R_earth * cos(latitude),
and then get your number, input this number in the box
"""

import math

def calculate_rotation_speed(radius_km: float, latitude_deg: float) -> float:
    """
    Calculate the linear speed of Earth's rotation at a given latitude.

    Parameters:
    - radius_km: Earth's radius at your location in kilometers (default ~6371␣
 ↪km)
    - latitude_deg: Your geographical latitude in degrees (e.g., 30.0 for 30°N)
```

```
    Returns:
    - Speed in kilometers per hour (km/h)
    """
    # Convert latitude to radians
    latitude_rad = math.radians(latitude_deg)

    # Angular velocity of Earth: 2  radians / 24 hours
    omega = 2 * math.pi / 24

    # Linear speed formula: v = R *   * cos(latitude)
    speed = radius_km * omega * math.cos(latitude_rad)

    return speed

def main():
    print("  Earth Rotation Speed Estimator  \n")

    try:
        # Input from user
        radius = float(input("Enter the local Earth's radius in km [default =␣
 ↪6371]: ") or "6371")
        latitude = float(input("Enter your latitude in degrees (positive for N,␣
 ↪negative for S): "))

        # Calculate speed
        speed_kmh = calculate_rotation_speed(radius, latitude)
        print(f"\n  At latitude {latitude}°, the Earth's rotation speed is␣
 ↪approximately {speed_kmh:.2f} km/h.")

    except ValueError:
        print("  Invalid input. Please enter numerical values only.")

if __name__ == "__main__":
    main()
```

```
  Earth Rotation Speed Estimator

Enter the local Earth's radius in km [default = 6371]:  6300
Enter your latitude in degrees (positive for N, negative for S):  -33


  At latitude -33.0°, the Earth's rotation speed is approximately 1383.25 km/h.
```

2.7 VII. User's Personal Summary (Optional)

Users can keep this Section (V) for their own note or memo