2.2.26

BALU-ai25btech11017

August 31, 2025

Question

Find the area of the triangle formed by the points $P(-1.5, 3)$, $Q(6, -2)$ and $R(-3, 4)$.

Theoretical Solution

Given three points

$$\mathbf{P} = \begin{pmatrix} -1.5 \\ 3 \end{pmatrix} \quad \mathbf{Q} = \begin{pmatrix} 6 \\ -2 \end{pmatrix} \quad \mathbf{R} = \begin{pmatrix} -3 \\ 4 \end{pmatrix} \tag{1}$$

$$\mathbf{Q} - \mathbf{P} = \begin{pmatrix} 7.5 \\ -5 \end{pmatrix} \tag{2}$$

$$\mathbf{R} - \mathbf{P} = \begin{pmatrix} -1.5 \\ 1 \end{pmatrix} \tag{3}$$

$$ar(PQR) = \frac{1}{2} \left\| (\mathbf{Q} - \mathbf{P}) \times (\mathbf{R} - \mathbf{P}) \right\| \tag{4}$$

$$ar(PQR) = \frac{1}{2} \left\| (\mathbf{Q} - \mathbf{P}) \times (\mathbf{R} - \mathbf{P}) \right\| = 0 \tag{5}$$

points are collinear

C Code

```c
#include <stdio.h>
#include <math.h>

// Function to calculate area of triangle using cross product
double triangle_area(double P[2], double Q[2], double R[2]) {
    double x1 = Q[0] - P[0];
    double y1 = Q[1] - P[1];
    double x2 = R[0] - P[0];
    double y2 = R[1] - P[1];

    // Cross product magnitude in 2D
    double cross = fabs(x1 * y2 - y1 * x2);

    return 0.5 * cross;
}
```

C Code - Resultant velocity

```c
  int main() {
    double P[2] = {-1.5, 3};
    double Q[2] = {6, -2};
    double R[2] = {-3, 4};

    double area = triangle_area(P, Q, R);

    printf("Area of triangle PQR = %.2f\n", area);

    return 0;
}
```

Python Code

```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D

# Points
P = np.array([-1.5, 3, 0])
Q = np.array([6, -2, 0])
R = np.array([-3, 4, 0])

# Function to compute area of triangle
def triangle_area(A, B, C):
    return 0.5 * np.linalg.norm(np.cross(B - A, C - A))

# Calculate area
area = triangle_area(P, Q, R)

# Plot the triangle in 3D
```

Python Code

```
fig = plt.figure(figsize=(6,6))
ax = fig.add_subplot(111, projection='3d')

# Plot points
ax.scatter(*P, color='r', s=50)
ax.scatter(*Q, color='g', s=50)
ax.scatter(*R, color='b', s=50)

# Plot triangle edges
ax.plot([P[0], Q[0]], [P[1], Q[1]], [P[2], Q[2]], 'k-')
ax.plot([Q[0], R[0]], [Q[1], R[1]], [Q[2], R[2]], 'k-')
ax.plot([R[0], P[0]], [R[1], P[1]], [R[2], P[2]], 'k-')
```

Python Code

```
# Labels for points
ax.text(*P, "P(-1.5,3)", color='r')
ax.text(*Q, "Q(6,-2)", color='g')
ax.text(*R, "R(-3,4)", color='b')

# Axis labels
ax.set_xlabel('X-axis')
ax.set_ylabel('Y-axis')
ax.set_zlabel('Z-axis')
ax.set_title(f"Area of Triangle PQR = {area:.2f}")

# Save and show
plt.savefig("triangle_area.png")
plt.show()
```

Plot