

# Third-Party Libraries

Python is supported by a vast ecosystem of third-party libraries that enhance its functionality for specific tasks. These can be installed using the command

```
pip install <library name>
```

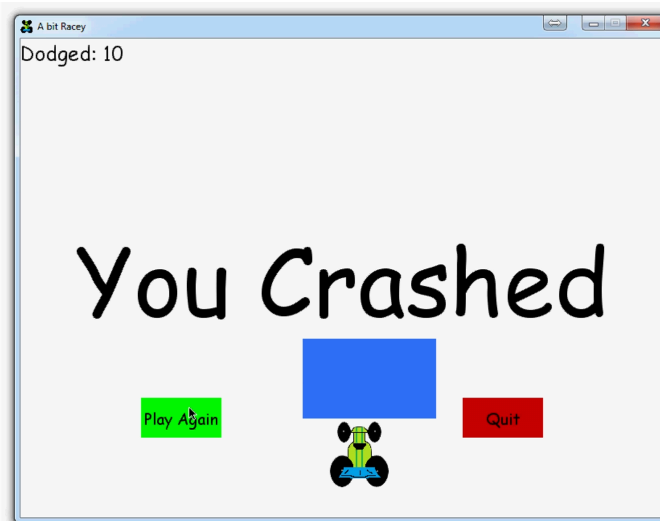
in your terminal or command prompt. Here are some useful ones:

## Pytest

Pytest is a flexible testing framework that is suitable for both simple and complex tests. It provides features like the `pytest.mark.parametrize` function, which allows testing a function with multiple input values, eliminating the need for writing repetitive test cases.

## Pygame

A collection of Python modules designed for creating and displaying simple video games. It is built on top of the Simple DirectMedia Layer (SDL) library, which provides low-level multimedia functionality.



## NumPy

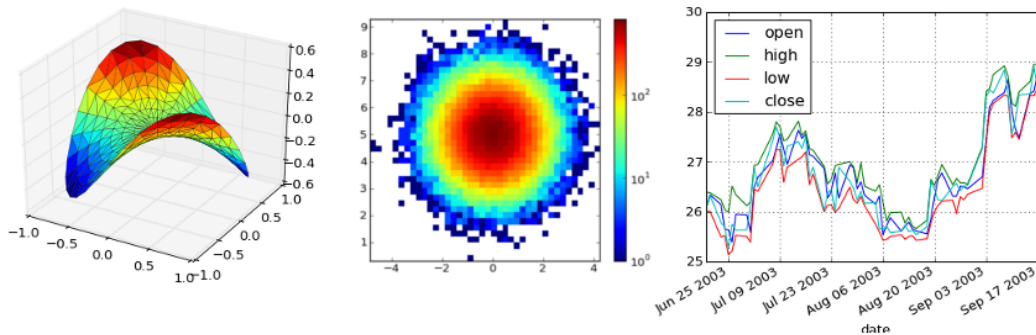
The fundamental package for numerical computation in Python, offering features for handling arrays, matrices, and advanced math functions. Its `numpy.array()` function is used for creating arrays, while `numpy.dot()` can be used for matrix multiplication.

## SciPy

Focuses on scientific and technical computing and serves as a complement to NumPy. It offers additional modules for various scientific computations. For example, the `scipy.integrate.quad()` function can be used to compute definite integrals.

## Matplotlib

A versatile library for creating a wide range of visualizations in Python, spanning from basic line plots to interactive charts. The `matplotlib.pyplot.plot()` function is used for line plots, while `matplotlib.pyplot.bar()` is employed for bar charts.



## Pandas

A popular library for data manipulation and analysis. It excels at working with numerical tables and time-series data. Functions like `pandas.DataFrame()` allow for the creation of data frames, while `pandas.read_csv()` is used to read data from CSV files.

## Requests

Simplifies HTTP requests by hiding complexities behind a clean API. Functions like `requests.get()` and `requests.post()` allow you to send HTTP GET and POST requests respectively.

## **Flask**

A simple, lightweight framework that simplifies the development of web applications. It provides a straightforward and minimalist approach for creating server-side applications.

## **Django**

A high-level web framework that enables rapid and secure website development. It follows the model-view-controller (MVC) architectural pattern, providing a comprehensive set of tools and libraries for building robust web applications.

## **TensorFlow**

Popular for machine learning tasks, particularly for constructing and training deep learning models. It offers a rich set of tools and resources to facilitate machine learning experimentation and deployment.