

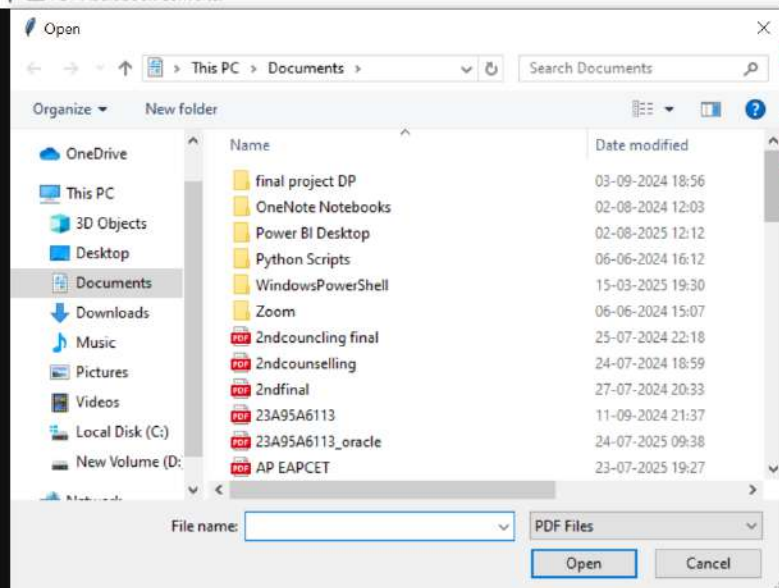
PDF Audiobook Generator

📄 Open PDF & Read

⏸ Stop Reading

▶ Continue Reading

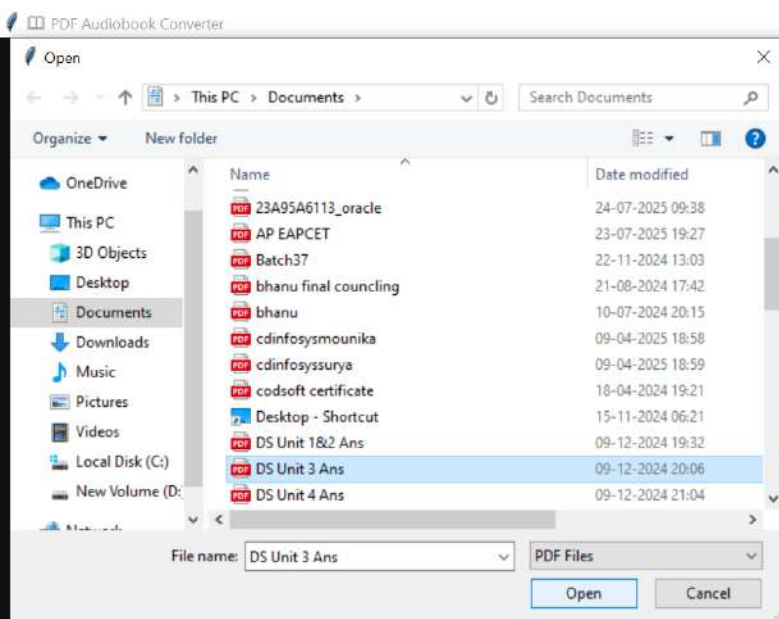




Book Generator

Reading

Continue Reading



Book Generator

Reading

Continue Reading

PDF Audiobook Generator

[📄 Open PDF & Read](#)[⏸ Stop Reading](#)[▶ Continue Reading](#)

1. What is JSON, and why is it a popular format for storing and exchanging structured data? Provide an example of loading JSON data into a pandas DataFrame .

Ans:

1. JSON: A Popular Data Format for Storing and Exchanging Structured Data

Overview

JSON (JavaScript Object Notation) is a lightweight data interchange format. It's text-based, human-readable, and often used to transmit data between a server and a web application. JSON's structure consists of key-value pairs, arrays, and objects, making it highly versatile for representing various data structures. JSON has become popular because it is easy to read and write for humans, and easy to parse and generate for machines.

Why JSON is Popular:

- Readability and Simplicity : JSON's structure is easy to understand, using straightforward key-value pairs and array syntax.
- Interoperability : JSON is language-independent, meaning most programming languages support JSON parsing and serialization.
- Lightweight : Compared to XML, JSON has a less verbose structure, reducing data transfer sizes, making it efficient for web applications.
- Flexibility : JSON can handle nested data structures, making it suitable for complex data and object representations.

Example: Loading JSON Data into a Pandas DataFrame

Let's say we have JSON data that looks like this:

PDF Audiobook Generator

[Open PDF & Read](#)[Stop Reading](#)[Continue Reading](#)

JSON's structure consists of key-value pairs, arrays, and objects, making it highly versatile for representing various data structures. JSON has become popular because it is easy to read and write for humans, and easy to parse and generate for machines.

Why JSON is Popular:

- Readability and Simplicity : JSON's structure is easy to understand, using straightforward key-value pairs and array syntax.
- Interoperability : JSON is language-independent, meaning many programming languages support JSON parsing and serialization.
- Lightweight : Compared to XML, JSON has a less verbose syntax, resulting in smaller transfer sizes, making it efficient for web applications.
- Flexibility : JSON can handle nested data structures, making it suitable for complex data and object representations.

Example: Loading JSON Data into a Pandas DataFrame
Let's say we have JSON data that looks like this:

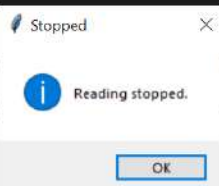
```
[
  {"name": "Alice", "age": 30, "city": "New York"},
  {"name": "Bob", "age": 25, "city": "Los Angeles"},
  {"name": "Charlie", "age": 35, "city": "Chicago"}
]
```

Here's how you can load this JSON data into a Pandas DataFrame:

```
import pandas as pd
import json
```

```
# Sample JSON data
```

```
json_data = '''
```



PDF Audiobook Generator

[Open PDF & Read](#)[Stop Reading](#)[Continue Reading](#)

JSON's structure consists of key-value pairs, arrays, and objects, making it highly versatile for representing various data structures. JSON has become popular because it is easy to read and write for humans, and easy to parse and generate for machines.

Why JSON is Popular:

- ↳ Readability and Simplicity : JSON's structure is easy to understand, using straightforward key-value pairs and array syntax.
- ↳ Interoperability : JSON is language-independent, meaning many programming languages support JSON parsing and serialization.
- ↳ Lightweight : Compared to XML, JSON has a less verbose syntax, resulting in smaller transfer sizes, making it efficient for web applications.
- ↳ Flexibility : JSON can handle nested data structures, arrays, and objects, making it suitable for complex data and object representations.

Example: Loading JSON Data into a Pandas DataFrame

Let's say we have JSON data that looks like this:

```
[
  {"name": "Alice", "age": 30, "city": "New York"},
  {"name": "Bob", "age": 25, "city": "Los Angeles"},
  {"name": "Charlie", "age": 35, "city": "Chicago"}
]
```

Here's how you can load this JSON data into a Pandas DataFrame:

```
import pandas as pd
import json
```

```
# Sample JSON data
```

```
json_data = '''
```

