


Java Import and export [HOME](#)  [GUIDES](#)  [JAVA](#)  [IMPORTING EXPORTING](#)  [INSTANT JSON](#)

Import and export annotations from Instant JSON in Java



COPY PAGE



Annotations can be imported and exported with the Nutrient Instant JSON format.

Instant JSON is a serializable representation of the current changes to a document, i.e. a diff between the document's saved and unsaved changes. This can be used to transfer a set of changes across devices without having to send the entire PDF, which could potentially be large.

In Nutrient Java SDK, we use the terms Instant JSON and Document JSON interchangeably.

Refer to the [Instant JSON](#) guide for details on how the format works.

Exporting to Instant JSON

To generate Instant Document JSON, use `PdfDocument.exportDocumentJson`, which will take a `WritableDataProvider` to write the JSON to.

Exporting to a file



ASK AI

This is an example using a `FileDataProvider`:

```
File file = new File("instant.json");
document.exportDocumentJson(new FileDataProvider(filename));
```

Exporting to an in-memory data provider

By extending `WritableDataProvider` to hold an array of data bytes (see the [save to a custom data provider](#) guide for details on implementing this), you can export the document Instant JSON to in-memory data:

```
CustomDataProvider customDataProvider = new CustomDataProvider();
document.exportDocumentJson(customDataProvider);
byte[] data = customDataProvider.read(customDataProvider.getSize(), 0);
// ... Do something with `data`.
```

Other reasons for implementing a custom `WritableDataProvider` could be for encrypting the exported JSON, for writing into a database, or for streaming the export to a remote server.

Importing from Instant JSON

The generated JSON in the previous example can be applied to a document using the `PdfDocument.importDocumentJson` method, which takes a `DataProvider`. This is an example using a `FileDataProvider`:

```
File file = new File("Assets/instant.json");
document.importDocumentJson(new FileDataProvider(file));
```

If the imported data is malformed or has no data, an exception will be thrown.

The `DataProvider` can be inherited from in order to implement custom readers — for example, for decrypting the imported JSON, for reading from a database, or for streaming from a remote server.

Modifying and generating Instant JSON

You can use the [Instant JSON format](#) to generate or alter the JSON prior to importing it with the `PdfDocument.importDocumentJson` method described above.

Was this helpful?

✓ YES

✗ NO



Copyright 2025 Nutrient. All rights reserved.