

# Plutarchus Documentation

Beta 0.1

August 30, 2019



# 1 Introduction

Plutarchus is an open source personal website building tool. It takes a JSON file and generates a website using the data in the JSON file. The JSON file is the dataset of the user's work. The websites that Plutarchus generates are static HTMLs, however Javascript and CSS are integrated to make them at par with modern web designs. Plutarchus v0.1 is designed to build resume websites. The examples and test cases are websites of academics and hence same are the target audience for v 0.1.

## 1.1 Why Plutarchus

There are a number of free website building tools available online. The key idea of the Plutarchus that separates it from other is that it generates a website from just a JSON file. Firstly, the user doesn't have to do any web design. Second, the user can maintain a JSON file of his work by simply adding entries to as new work is produced. The JSON file can be read as a dataset in various programming language for data analytics, etc.

The project that comes closest to Plutarchus is [JSON Resume](#). However, this project isn't aimed websites that are frequently updated.

## 1.2 Examples

The personal [website](#) of Renuka Sane is generated from her [JSON file](#)

The JSON file contains the a tag for theme, the same website is generated again, with different theme [here](#). 'Theme1a' has been changed to 'Theme2a' in the [JSON file](#) used to generate the second website.

A website can also be generated without a theme or with minimal theming. [This](#) is the same website without any theme. It doesn't contain any Javascripts and CSS files. This is called 'Theme 0' in the package.

# 2 Getting Started

There are broadly 2 steps.

1. Write a JSON file of your work
2. Build website using it by using the web app or the Command Line Interface.

## 2.1 Writing a JSON File

JSON, or JavaScript Object Notation, is a minimal, readable format for structuring data. It is used primarily to transmit data between a server and web application, as an alternative to XML.

Essentially, you follow some rules to write a text file and save the file using 'json' as the extension. These rules are fairly simple and can be understood by looking at an [example](#). It's easiest to do it by editing the [minimal JSON file](#) provided in the package. There is also a JSON editor provided in the Commandline Interface of plutarchus and there is a webapp under development for the same (will release in v0.2).

There are two sections that you need to be there in the JSON file, 'basics' and 'entries'. There are 7 items to be put in basic: theme, support, name, image, sort, email and pages. The theme tag tells the generator which theme to use. Example: 'theme':'Themela'. If you say yes to support, there will be link to plutarchus on your page. The pages tag will tell the generator the names of the pages and the order. If the value given to sort is decending, it will sort the entries in decending order or dates. The rest are probably self explanatory.

You add information about of your works in the entries section of the JSON file. It's list of elements each containing five items: page, date, section, subsection and entry. You may not put date, section and subsection, but page and entry are required. The page value tells the generator which page to put the entry in. The entry where text regarding the work is put. HTML tags can be put in the field for formatting, adding hyperlinks, images, etc. All hyperlinks must be put in single quotes.

example of an entry:

```
{
  "page":"Criminal Justice",
  "date":"10 May 2015",
  "section":"The Indian criminal justice system",
  "subsection":"Broad Work",
  "entry":"<a href='http://ajayshahblog.blogspot.in/'>
  Faulty tradeoffs in security</a>, by <h2>Ajay Shah</h2>,
  Ajay Shah's blog" }
```

Add such elements to the entries section of your JSON file and separate them with commas.

### 2.1.1 HJSON File

A **HJSON** file is essentially JSON file in which you can add comments. There are some additional features like 'do what I mean' so if you make minor mistakes in formatting, the program (plutarchus) reading the HJSON might be able to correct them.

It's recommended to use a HJSON file. The minimal HJSON file provided in the package has comments that can guide you.

### 2.1.2 Converting from HJSON to JSON

You may want to convert your HJSON file to a JSON file. If your HJSON file is called work.hjson, you can convert it by running.

```
pip install hjson
hjson -j work.hjson > work.json
```

The first line install the hjson converter and the second line does the conversion. It removes all the comments and also make corrections to all the mistakes in the formatting if any.

## 2.2 Examples

## 2.3 Website Generation

### 2.3.1 Examples

### 2.3.2 Command Line Interface

### 2.3.3 Graphics User Interface

## 3 Themes

### 3.1 Themes in V0.1

### 3.2 Theme0

## 4 Getting Help