

# Input/Output writeup v.1

**Alexander Archer**

## Files

io.py, \*config.json, settings.json

## Use Case

Manage all input and output operations to the local OS

## Use Case Explained

Handles numerous disk operations: loading configuration files, user settings, song files; creation of music library directory; extracting song meta data

## Introduction

- Most programs have some form of input and output
- Python has built-in functions for working with files, namely **open()**, **read()**, **write()**, **append()**, and **close()**
- Python also has built-in modules for working with directories, namely **os.path** and **shutil**

## Intended File Behavior

**io.py**: The python file that handles disk operations

- Load the various configuration files
- Load the settings file
- Manage song importation and local music library
- Extract song meta-data

**\*config.json**: any number of files that contain configuration data in a specific format

**settings.json**: user-specific program settings

## I/O Event Handling

On program start:

- Load configuration file
- If user settings doesn't exist
  - create (paths defined in config)
- else
  - apply the settings

On program close:

- Handle file closure gracefully

On settings update:

- Correctly load settings file from disk, edit, and write back to disk

On song add:

- Load file from disk into memory and write to correct internal folder
- Extract any file meta-data (and populate db table)

On song load:

- Load song from internal folder into memory (and metadata from db)

On song remove:

- Delete correct song from disk (and remove from db tables)