

# INI cheatsheet

This is a quick reference cheat sheet for understanding and writing INI-format configuration files.

## # Getting Started

### Introduction

- INI is a configuration file with a fixed standard format
- Base elements are keys or properties
- Each key consists of a name and a value, separated by an equal sign (=)
- key name is displayed to the left side of the equals sign
- equal sign and semicolon are reserved characters
- INI configuration method comes from the MS-DOS operating system

Now an informal standard for many configurations, other operating systems may use .conf or .cfg as a suffix

### Example

```
; Here are the comments
[owner]
name=John Doe
organization=Acme Products

[database]
; Here are the comments
server=192.0.2.42
port=143
file="acme payroll.dat"

[section.subsection]
foo = bar
```

### Comments

```
comment (;)

; This is the comment text and will be ignored

comment (#)

# Here is the comment text, ^ Some compilers ignore this

Comments after a line (;,#) (not standard)

var = a ; this is an inline comment
foo = bar # this is another inline comment

Comments must appear alone on lines in some cases
```

### Sections

- The name appears on a line by itself
- Names are enclosed in square brackets [ and ]
- No explicit section end delimiter
- End at the next section declaration or at the end of the file
- Section and attribute names are case insensitive

```
[section]
key1 = a
key2 = b
```

The same as JSON below 📌

```
{
  "section": {
    "key1": "a",
    "key2": "b"
  }
}
```

### Nesting (supported by some parsers)

```
[section]
domain = quickref.me
[section.subsection]
foo = bar

The same as JSON below 📌

{
  "section": {
    "domain": "quickref.me",
    "subsection": {
      "foo": "bar"
    }
  }
}
```

Nest to previous section (shorthand)

```
[section]
domain = quickref.me
[.subsection]
foo = bar
```

### Escape character

\\	\ (single backslash, escape escape character)
\'	apostrophe
\"	double quotes
\0	null character
\a	ringtone/alert/sound
\b	Backspace, [Bell character] for some applications ( <a href="https://en.wikipedia.org/wiki/Bell_character">https://en.wikipedia.org/wiki/Bell_character</a> )
\t	tab character
\r	carriage return
\n	newline
\;	semicolon
\#	number sign
\=	equal sign
\:	colon
\x????	Unicode character for the hexadecimal code point corresponding to ????

### Array

```
[section]
domain = quickref.me
array[]=first value
array[]=second value
```

The same as JSON below 📌

```
{
  "section": {
    "domain": "quickref.me",
    "array": [
      "first value",
      "second value"
    ]
  }
}
```

### Interpreter

- @go-ini/ini (golang)
- @npm/ini (nodejs)
- @zonyitoo/rust-ini (rust)
- @rx/ini (c)
- @pulzed/miNI (c++)
- @rickyah/ini-parser (c#)
- @Enichan/ini (c#)

## # See also

[INI file configuration](#)(wikipedia.org)

### Top Cheatsheet

**Python Cheatsheet**  
Quick Reference

**Vim Cheatsheet**  
Quick Reference

**Google Search Cheatsheet**  
Quick Reference

**Kubernetes Cheatsheet**  
Quick Reference

**JavaScript Cheatsheet**  
Quick Reference

**Bash Cheatsheet**  
Quick Reference

**ES6 Cheatsheet**  
Quick Reference

**ASCII Code Cheatsheet**  
Quick Reference

### Recent Cheatsheet



**QuickRef.ME**

Share quick reference and cheat sheet for developers.

中文版 #Notes



Build & scale your NFT apps on QuickNode without the hassle of gas fees and infra maintenance

ADS VIA CARBON