

How to Use Interpolation

Using interpolation with configparser in python

WE'LL COVER THE FOLLOWING ^

- Wrapping Up

The configparser module also allows **interpolation**, which means you can use some options to build another option. We actually do this with the font_info option in that its value is based on the font and font_size options. We can change an interpolated value using a Python dictionary. Let's take a moment to demonstrate both of these facts.

```
import configparser
import os

def interpolationDemo(path):
    if not os.path.exists(path):
        createConfig(path)

    config = configparser.ConfigParser()
    config.read(path)

    print(config.get("Settings", "font_info"))

    print(config.get("Settings", "font_info",
                    vars={"font": "Arial", "font_size": "100"}))

if __name__ == "__main__":
    path = "settings.ini"
    interpolationDemo(path)
```



If you run this code, you should see output similar to the following:

```
You are using Courier at 12 pt
You are using Arial at 100 pt
```



Wrapping Up

At this point, you should know enough about the **configparser**'s capabilities that you can use it for your own projects. There's another project called **ConfigObj** that isn't a part of Python that you might also want to check out. **ConfigObj** is more flexible and has more features than **configparser**. But if you're in a pinch or your organization doesn't allow 3rd party packages, then **configparser** will probably fit the bill.