

Autoload INI File

Version: 1.1

Author: [Remi Bodin \(https://www.remibodin.fr\)](https://www.remibodin.fr)

Unity AssetStore: [Autoload INI File \(https://assetstore.unity.com/packages/slug/130860\)](https://assetstore.unity.com/packages/slug/130860)

Usage

Autoload INI File package allows you to store configuration value in INI file and change it after the export with minimum of code.

You must add **File attribute** to your class, **Section attribute** to your field and that's it.
All static value with a Section attribute can be override in a INI file even after compilation.

Some of built-in C# types are support :

int
float
double
bool
string
char

But you can add more types with **TypeParser attribute**

Api

IniConfiguration.Attributes.File(string filePath)

string filePath: Path of ini file. Must be relative to binary file (or project folder in Editor)

Class attribute

```
[File("configuration.ini")]
public class Configuration
{
    ...
}
```

IniConfiguration.Attributes.Section(string sectionName)

string sectionName: Section name in INI file where variable is.

public static field attribute

```
[Section("SectionName")]
public static float MyValue;
```

IniConfiguration.Attributes.TypeParser(System.Type type)

System.Type type: type parse by this delegate.

public static method attribute

```
[TypeParser(typeof(MyType))]  
public static void MyTypeParser(System.Reflection.FieldInfo field, string  
value)  
{  
    MyType parsedValue;  
    ...  
    ReflectionUtils.SetField(field, parsedValue);  
}
```

Example

You can find an example in Assets/Nrtx/INIFileExample.

Add **ApplyGlobalConfiguration.cs** in first scene and build your project.

Now you can create configuration.ini file like

```
[Display]  
Fullscreen = False  
  
[Sound]  
MasterVolume = 0.8
```

to enable/disable fullscreen mode and change master volume.

Don't forget to remove Assets/Nrtx/INIFileExample directory before release your project.