

Forum:[Hack Forums](#)

Thread: [\[Release\] Installer.cs | Hidden/Normal startup | Supports environment variables+++](#)

gigajew - November 13, 2017, 5:21 pm

Here's a new installation class I wrote. It supports installing as hidden or normal, and a really great thing about it, is that you can natively pass string paths with environment variables, such as: %appdata%\file.exe which is a thing a lot of crypter-developers may find useful. Example usage is in the paste, but here it is too:

Code // normal, passed with app title

```
var settings = new InstallerSettings("Windows Update", "%AppData%\MyFile.exe");
if (!Installer.IsInstalled(settings)) {
    Installer.Install(settings);
}
```

// hidden, no app title necessary

```
var settings = new InstallerSettings("%AppData%\MyFile.exe") { Hidden = true };
if (!Installer.IsInstalled(settings)) {
    Installer.Install(settings);
}
```

cocaine_monkey - November 14, 2017, 4:20 pm

As im learning i would like to hear your opinion on get set properties, in installersettings why you choose to use them like that? do you use them always like that? i tend to use them only if i need some checks in there or i want to limit it to just get only. Also why use private then public property on that class? wouldnt public get set work just fine? Is it stupid to create get set functions like in c++? i think c++ way looks more sexy but is it stupid?

gigajew - November 14, 2017, 4:22 pm

(Nov 14, 2017 05:20 PM)cocaine_monkey Wrote: [\[CITE\]](#) As im learning i would like to hear your opinion on get set properties, in installersettings why you choose to use them like that? do you use them always like that? i tend to use them only if i need some checks in there or i want to limit it to just get only

You used to always have to do it that way. I keep doing it for backwards compability and because of habit

cocaine_monkey - November 14, 2017, 4:27 pm

(Nov 14, 2017 05:22 PM)gigajew Wrote: [\[CITE\]](#) You used to always have to do it that way. I keep doing it for backwards compability and because of habitOk yes its new feature also i edited my post if you could share your opinion

gigajew - November 14, 2017, 5:11 pm

(Nov 14, 2017 05:27 PM)cocaine_monkey Wrote: [\[CITE\]](#) Ok yes its new feature also i edited my post if you could share your opinion

The things you see that are private are fields and hold the actual values. The properties are like methods that expose them.

cocaine_monkey - November 15, 2017, 3:08 am

(Nov 14, 2017 06:11 PM)gigajew Wrote: [\[CITE\]](#) The things you see that are private are fields and hold the actual values. The properties are like methods that expose them.

Yes i know that

Just so you can learn something new too in c++

Code private int value;

```
public void SetValue(int value)
{
    this.value = value;
}
```

same with get and why _ prefix instead of this? this. Is feature?

gigajew - November 15, 2017, 5:17 am

(Nov 15, 2017 04:08 AM)cocaine_monkey Wrote: [\[CITE\]](#) Yes i know that

Just so you can learn something new too in c++

Code private int value;

```
public void SetValue(int value)
{
    this.value = value;
}
```

same with get and why _ prefix instead of this? this. Is feature?

Oh I didn't know you meant that part, that's just preference

cocaine_monkey - November 15, 2017, 2:55 pm

(Nov 15, 2017 06:17 AM)gigajew Wrote: [\[CITE\]](#) Oh I didn't know you meant that part, that's just preference

Yes im trying to learn how to write "clean" code and saw your properties usage so was interested

Ninja - November 15, 2017, 3:39 pm

Why not use auto properties? They are one of the things that make C# great.

WinA - May 17, 2018, 5:23 am

How could we use this with programs that require admin permissions? Like we run the program initially as admin but once its copied to startup would it fail to run properly without running as admin?