


```

    }
}
if(oSVersion.Version.Major == 10)
{
    switch (oSVersion.Version.Minor)
    {
        case 0:
            osName = "W10";
            break;
    }
}
}
}

```

I am trying to get the correct major and minor numbers so it will read the correct operating system. I'm sure i am making a mistake somewhere but can't figure it out. It reads windows 10 as windows 8

HFDone - December 14, 2019, 10:00 am

<https://stackoverflow.com/questions/2819934/detect-windows-version-in-net>

__Digital__ - December 14, 2019, 10:22 am

(Dec 14, 2019 11:00 AM)HFDone Wrote: [\[CITE\]](#)

<https://stackoverflow.com/questions/2819934/detect-windows-version-in-net>

Yes i have looked at all that and even with correct changes it still reads as windows 8 when its a windows 10 machine.

NYANxCAT - December 14, 2019, 10:39 am

<https://docs.microsoft.com/en-us/dotnet/api/microsoft.visualbasic.devices.computerinfo.osfullname?view=netframework-4.8>

PHP Code

```

<?php
new ComputerInfo().OSFullName.ToString()

```

HFDone - December 14, 2019, 1:58 pm

<https://www.prugg.at/2019/09/09/properly-detect-windows-version-in-c-net-even-windows-10>

itcorp - December 14, 2019, 2:03 pm

(Dec 14, 2019 11:22 AM)__Digital__ Wrote: [\[CITE\]](#) Yes i have looked at all that and even with correct changes it still reads as windows 8 when its a windows 10 machine.

if you really want to use this helper class what prevents you to modify it ,because generally you have more or less everything there and changes that you will make there is very easy following an already existing concept.

Bunnyslippers69 - December 14, 2019, 3:07 pm

You could try something similar to this. Returns the version number of the OS. Can be used to find the name of the OS using this [page](#).

My bad if a better method exists. This is just what came to mind.