Configurando o Internet Explorer via registro - FEATURE\_BFCACHE.

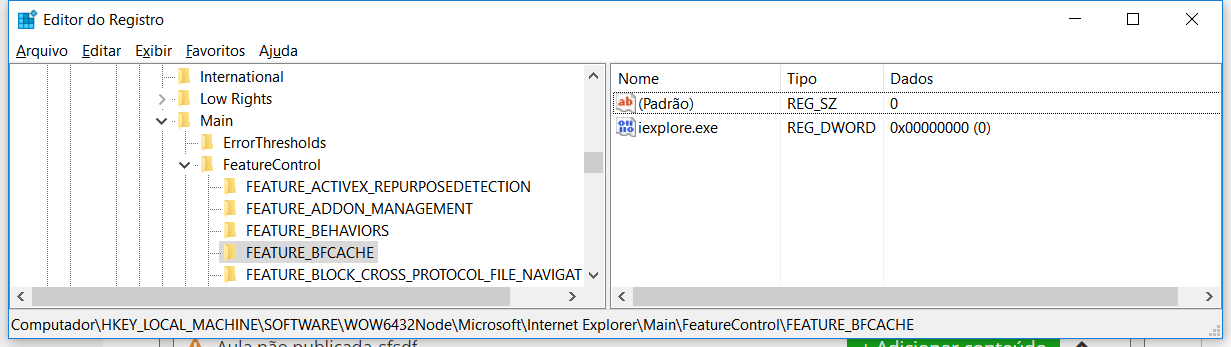
Muitas da vezes precisamos habilitar uma certa configuração no Internet Explorer ou numa versão especifica do IE, por isso criei uma classe RegistryWin no nosso Projeto, um problema que temos é adicionar no *FEATURE\_BFCACHE*  a chave *iexplore.exe* com valor 0, que é do tipo *REG\_DWORD, temos que adicionar essa chave no IE11 como está descrito no site do driver do IE, segue a citação abaixo*.

*For IE 11 only, you will need to set a registry entry on the target computer so that the driver can maintain a connection to the instance of Internet Explorer it creates. For 32-bit Windows installations, the key you must examine in the registry editor is HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Internet Explorer\Main\FeatureControl\FEATURE\_BFCACHE. For 64-bit Windows installations, the key is HKEY\_LOCAL\_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Internet Explorer\Main\FeatureControl\FEATURE\_BFCACHE. Please note that the FEATURE\_BFCACHE subkey may or may not be present, and should be created if it is not present. Important: Inside this key, create a DWORD value named iexplore.exe with the value of 0.*

1. # importando a classe.
2. from registry import RegistryWin
4. # Instanciando a classe.
5. reg = RegistryWin()

Nosso teste fica assim:

1. # Verificar se o sistema operacional eh 64 ou 32bits, systeminfo me retorna o tipo de sistema.
2. # quando coloco o [1] que dizer que quero somente o segundo valor da lista, que eh o resultado do cmd.
3. sys = getstatusoutput("systeminfo | findstr //C:\"Tipo de sistema\" /C:\"System type\"")[1]
4. if 'x64' in sys.lower():
5. reg.reg\_path = "SOFTWARE\Wow6432Node\Microsoft\Internet Explorer\Main\FeatureControl\FEATURE\_BFCACHE"
6. else:
7. reg.reg\_path = "SOFTWARE\Microsoft\Internet Explorer\Main\FeatureControl\FEATURE\_BFCACHE"
8. # cria a key
9. key\_open = reg.open\_key\_reg("machine")
10. # Configuro as variaveis.
11. reg.key\_name = "iexplore.exe"
12. reg.value = 0
13. # Verifica se a chave jah existe.
14. # caso nao exista ele inseri.
15. verified = reg.query\_path(key\_open)
16. if not verified == 0:
17. reg.set\_reg\_path\_dword(key\_open)
18. print("FEATURE\_BFCACHE adicionado.")
19. else:
20. print("Registro jah adicionado.")



Em todo registro temos objetos comum na nossa classe, nome da chave, tipo da chave, valor, caminho, que passamos por parâmetro para a classe ou para a função.

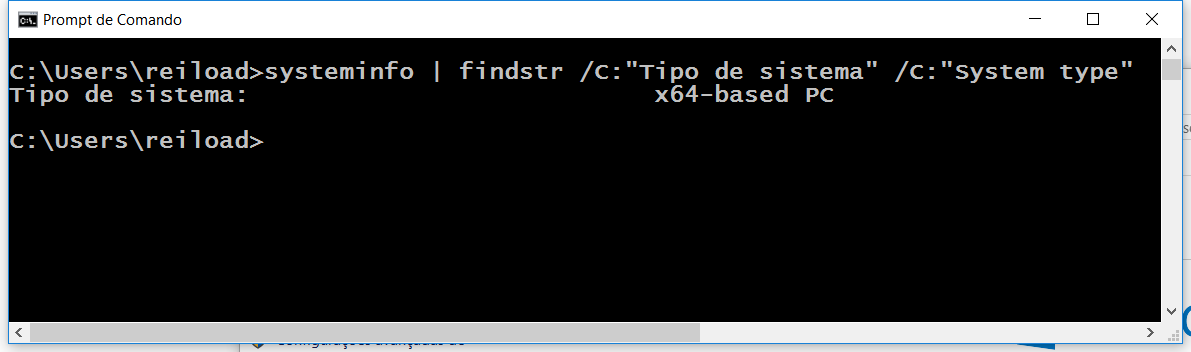
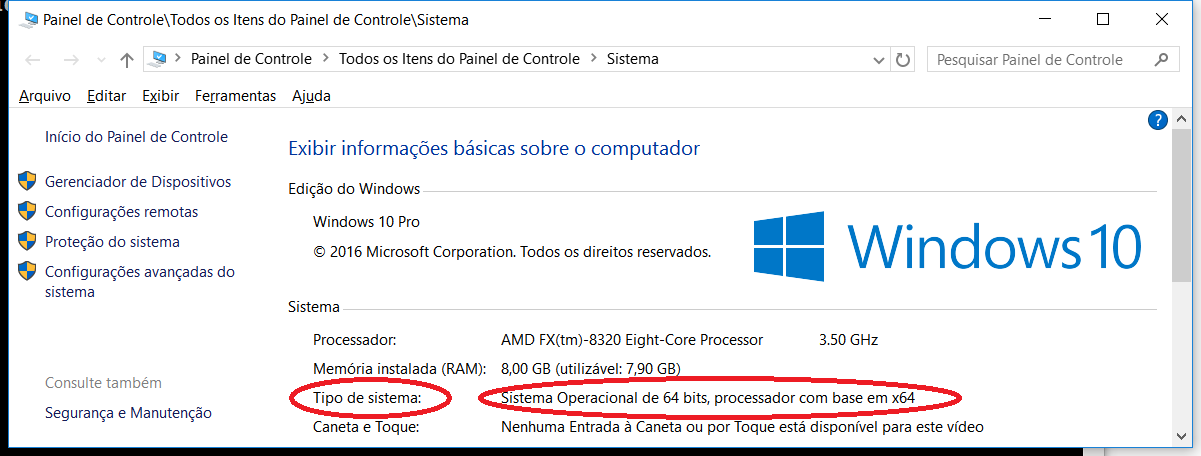
**Registro:**

|  |  |  |
| --- | --- | --- |
| **Nome** | **Tipo do valor** | **Descrição** |
| Valor binário | REG\_BINARY | - Dados binários não-processados. Grande parte das informações de componente de hardware é armazenada como dado binário e exibida no Editor do Registro no formato hexadecimal. |
| Valor DWORD | REG\_DWORD | - Dados representados por um número de 4 bytes (um inteiro de 32 bits). Muitos parâmetros para drivers e serviços de dispositivo são desse tipo e são exibidos no Editor do Registro no formato binário, hexadecimal ou decimal. |
| Valor da seqüência | REG\_SZ | - Seqüência de texto com extensão fixa. |

Fonte: support.microsoft.com

**Outra coisa legal que fizemos pegamos o tipo de sistema operacional do windows:**

Se você digitar systeminfo no cmd, vai trazer várias informações sobre o sistema operacional, no python você pode usar o **import platform**  e usar a função **platform.architecture()**, mas pode dar errado quando o sistema operacional é de 32 bits instalado numa maquina x64, isso pode ocorrer.



**Referências:**

[1] https://github.com/reinaldorossetti/tests\_solution

[2] https://support.microsoft.com/pt-br/kb/256986

[3] https://docs.python.org/3.6/library/platform.html

[4] https://technet.microsoft.com/pt-br/library/cc771190(v=ws.10).aspx