

**Volume**

**2**

TREINAMENTO PLAYCAREER BUSPLAYS

---

Ferramentas básicas para utilizar o Python para ArcGIS

ArcMap

TREINAMENTO PLAYCAREER

# CURSO DE ARCGIS

---

**Organização:**

Lucas França

Nicolly Gleisy

Tiago Xavier

Bus Plays – [busplays@gmail.com](mailto:busplays@gmail.com)

---

# Sumário

Primeiros passos para utilizar o Python no ArcGIS .....	1
Python 2.7 (32 bits) .....	1
Configurações básicas do ArcMap para o tutorial .....	3
Barra de ferramentas padrão (Standard) .....	5
Janela Catalágo .....	6
Janela Pesquisar.....	8
Janela ArcToolbox .....	10
Descrição do item .....	12
Preparação antes de usar o Python no ArcMap .....	15
Janelas e barras de ferramentas.....	15
Requisitos mínimos para entender as orientações no Python .....	16
Python dentro e fora do ArcMap .....	19
Utilizar dentro do ArcMap.....	19
Biblioteca Arcpy .....	22
Interação com itens e a janela Python .....	24
Python fora.....	26
Escrevendo o código em Python.....	26
Ferramenta de análise Buffer.....	28
Aprendendo os comandos em Python .....	31
Resultados .....	33
Copiar como Python Snippet.....	34
Ferramentas de seleção de feição .....	36
Exemplo .....	39
Exemplo 1 – Utilizar a condição while.....	39
Exemplo 2 – Utilizar a condição for e listas.....	43
Anexos .....	45
Código – Exemplo 1 .....	45
Código – Exemplo 2.....	46

---

## Primeiros passos para utilizar o Python no ArcGIS

### Python 2.7 (32 bits)

A utilização do ArcGIS exige a instalação e uso do Python 2.7 (32 bits). Isso, pois, as ferramentas e scripts do próprio programa utilizam-no como base. Mas não se preocupe, basta que no momento da instalação do ArcGIS, você selecione o modo de instalação completo, o qual já instala o Python 2.7.

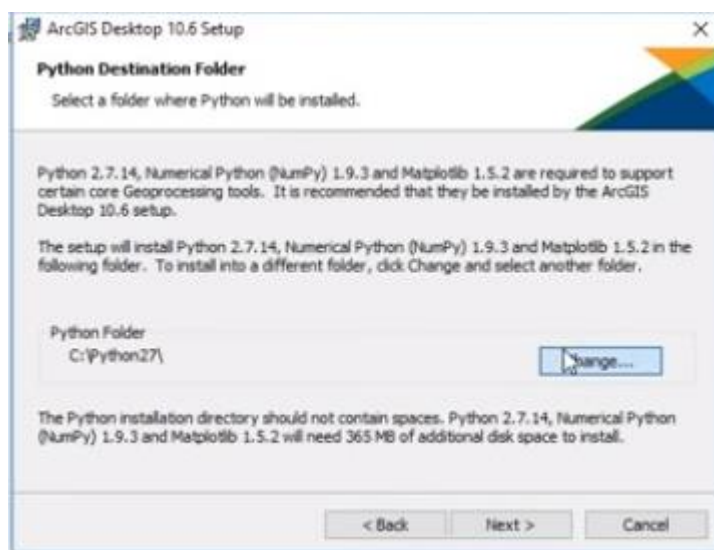


Figura 1- Janela de instalação do ArcGIS orientando a instalação do Python 2.7.14

Caso tenha dúvidas sobre os requisitos do sistema para instalação do ArcGIS e uso do Python, sugiro a leitura do site:

<https://desktop.arcgis.com/en/system-requirements/10.5/arcgis-desktop-system-requirements.htm>

A imagem a seguir retirada do site acima explica em detalhes a importância do Python para o uso do ArcGIS.

Quadro 1- Descrição a necessidade do Python para o ArcGIS. Imagem retirada do site <https://desktop.arcgis.com>

	Description
Python requirement for geoprocessing	ArcMap geoprocessing tools require Python 2.7.13 and Numerical Python 1.9.3 to be installed. If the ArcMap setup does not find either Python 2.7.13 or Numerical Python (NumPy) 1.9.3 installed on the target computer, Python 2.7.13 and Numerical Python 1.9.3 will be installed during a complete installation of ArcMap. You can choose a Custom installation to unselect the Python feature and avoid installing it. Additionally, if the Python setup is executed during the ArcMap installation, you will be provided with the opportunity to choose its installation location. The Python installation location should not include spaces.

## Configurações básicas do ArcMap para o tutorial

O ArcMap é um programa com uma ampla gama de utilidades e ferramentas. Entre elas, podemos citar ferramentas para edição, criação, análise de arquivos shapefile, raster e Excel. A quantidade extensa de utilidades pode causar confusão no usuário. Em vista disso, o programa organiza suas funcionalidades em barras de ferramentas e janelas que podem ser acrescentadas ou retiradas da tela de visualização do programa.

Sugiro para este tutorial que todas as barras de ferramentas e janelas sejam fechadas. Assim, podemos começar do mesmo ponto de partida. Para fechar as janelas, basta clicar no **X** no canto superior direito da janela em específico. Para fechar as barras de ferramentas, é necessário desativá-las. Isso é possível clicando com o botão direito no mouse na parte superior da janela do ArcMap ou acessando a aba Personalizar (“Customize”) e barra de ferramentas.

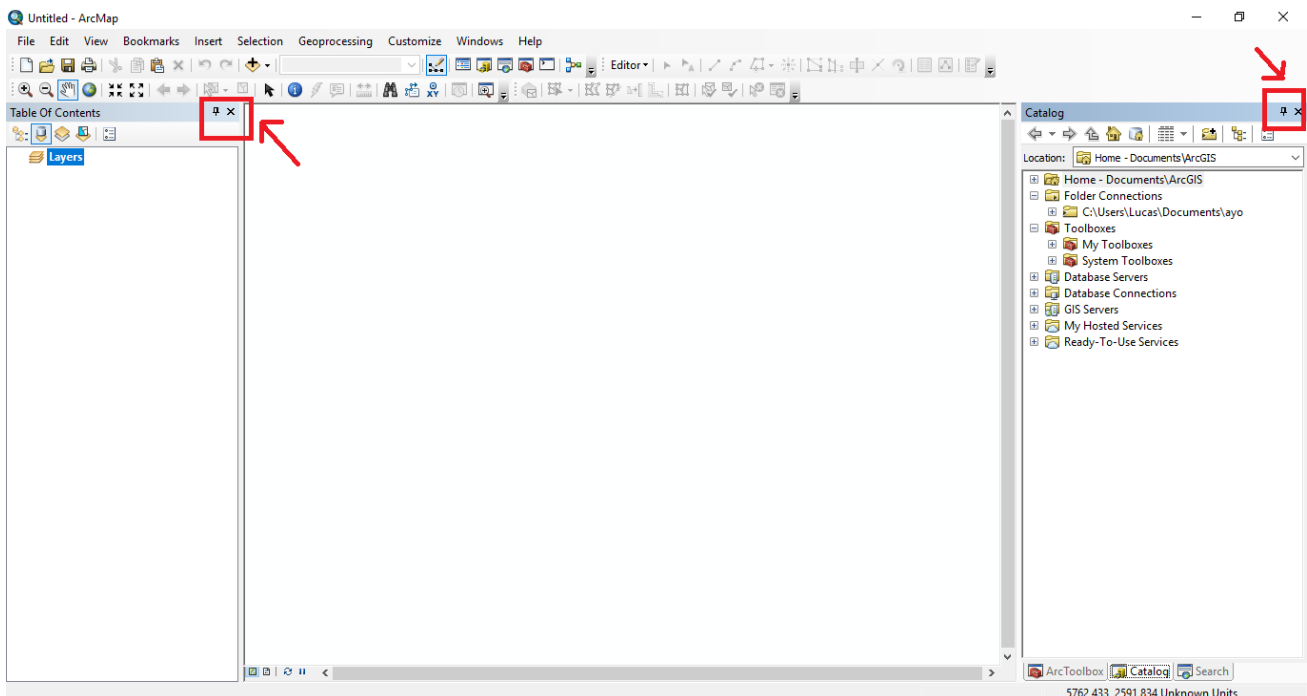


Figura 2- Minha configuração de barras e janelas. O vermelho está indicando onde está o botão para fechar a janela

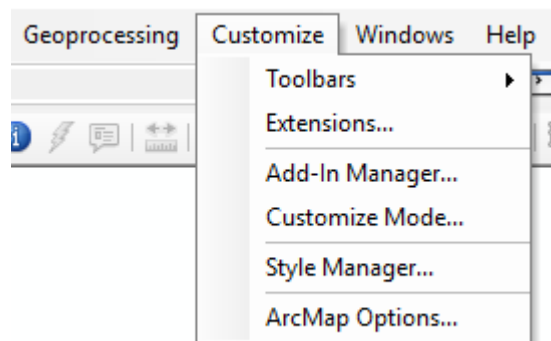


Figura 3- Aba Personalizar > Barra de ferramentas

Quando apagar todas, o seu ArcMap deve ficar com essa aparência:

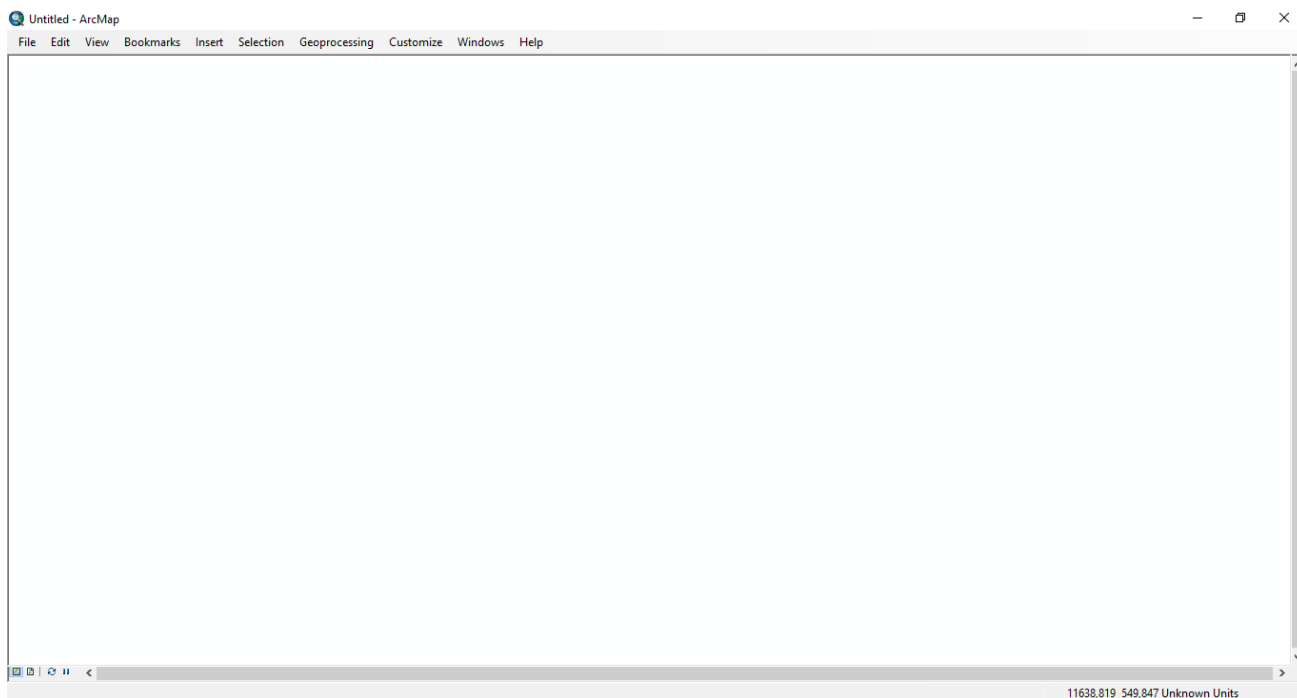


Figura 4- ArcMap sem barras de ferramentas e janelas abertas

A partir desse ponto, irei mostrar barras de ferramentas e janelas fundamentais para o uso do Python no ArcMap.

### Barra de ferramentas padrão (Standard)

Acesse a aba Personalizar (“Customize”) e ative barra de ferramentas “Padrão”. Ela irá aparecer na parte superior da janela do ArcMap.

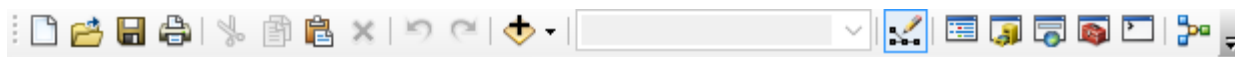












Figura 5- Barra de ferramentas Padrão

A barra de ferramentas Padrão (“Standard”) contém os itens básicos para uso do ArcMap. Ao clicar em seus ícones é possível criar , abrir  e salvar  projetos (mapas), adicionar  shapefiles, tabelas, raster, entre outros formatos. Além disso, ela também dá acesso a janela de camadas ou layers  (“Table of Contents”), a janela Catálogo , a janela de Pesquisa , a janela de ferramentas de geoprocessamento ArcToolbox  e a janela Python .



## Janela Catálogo

Ao clicar no ícone  da barra de ferramentas Padrão (“Standard”) a janela Catálogo será aberta na parte direita da tela.

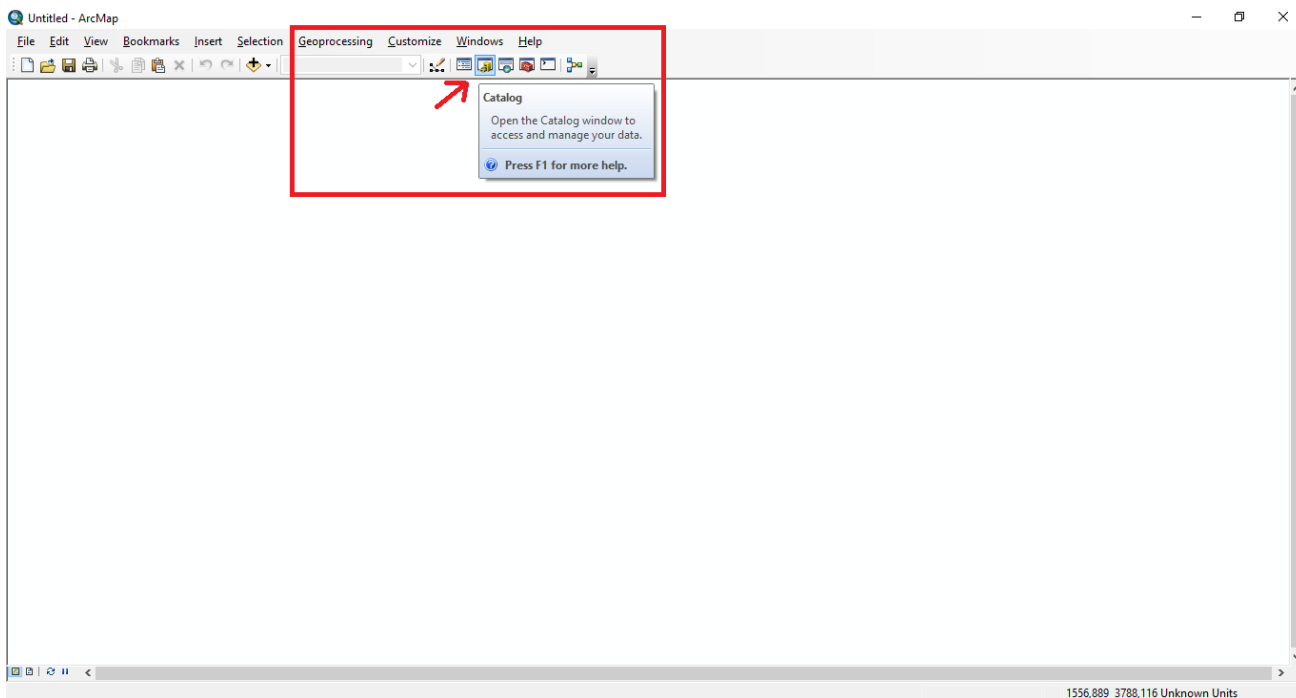


Figura 6- tela e localização do ícone da janela Catálogo

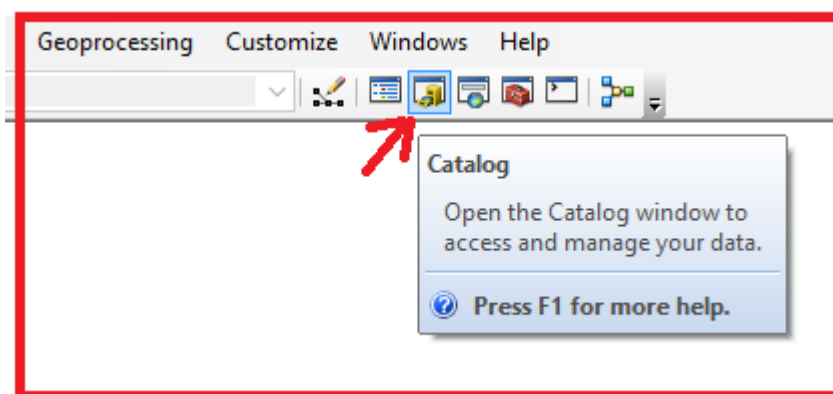


Figura 7- Zoom no local do ícone da janela Catálogo e sua breve descrição

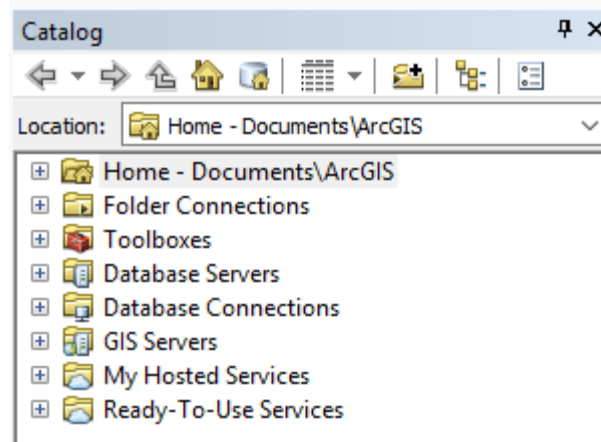



Figura 8- Janela Catálogo

Essa janela permite o acesso e controle de dados das pastas, das ferramentas de geoprocessamento, dos servidores externos.

Ela é parte importante no gerenciamento dos dados que forem criados através do Python, pois, permite visualizar, organizar e apagar os arquivos.

## Janela Pesquisas

Ao clicar no ícone  da barra de ferramentas Padrão (“Standard”) a janela Pesquisas será aberta na parte direita da tela.

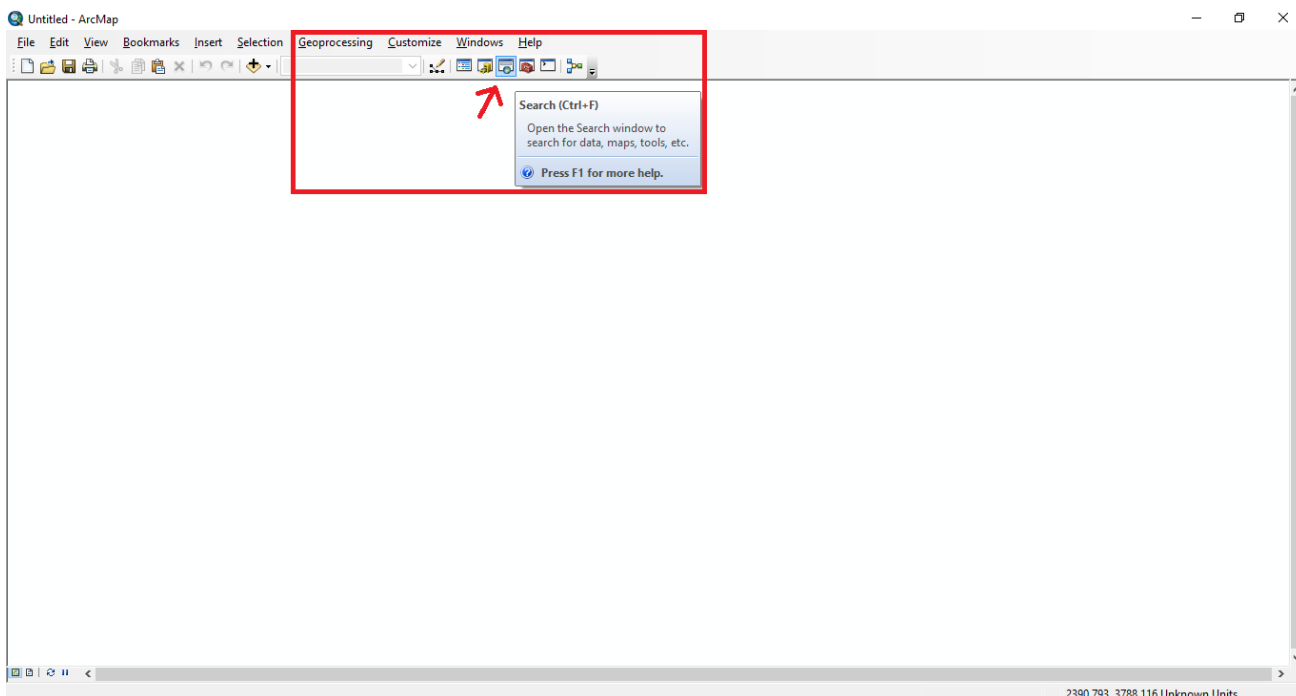


Figura 9- tela e localização do ícone da janela Pesquisas

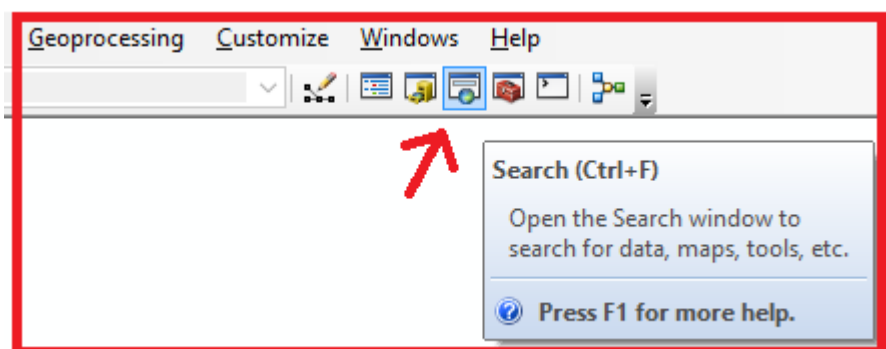


Figura 10- Zoom no local do ícone da janela Pesquisas e sua breve descrição

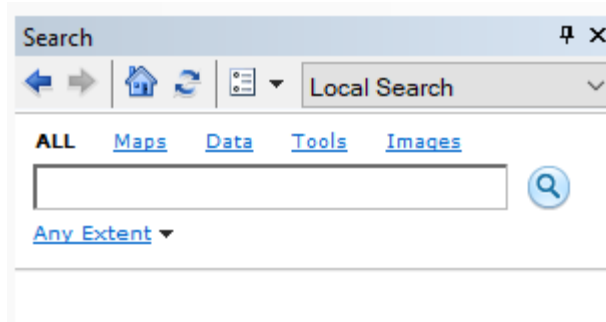



Figura 11- Janela Pesquisar

A janela Pesquisar possibilita a pesquisa de ferramentas para uso no ArcMap. Ela é útil, pois, facilita a busca de ferramentas do Arctoolbox por meio de palavras.

## Janela ArcToolbox

Ao clicar no ícone  da barra de ferramentas Padrão (“Standard”) a janela ArcToolbox será aberta na parte direita da tela.

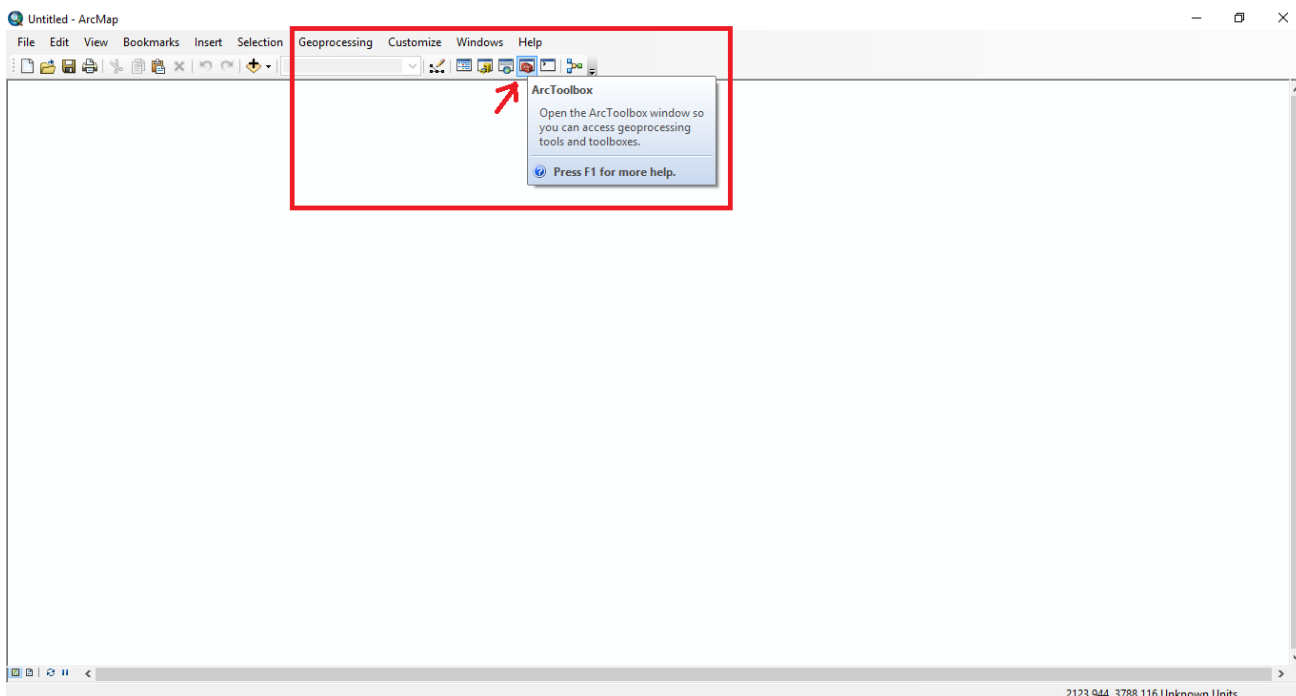


Figura 12- tela e localização do ícone da janela ArcToolbox

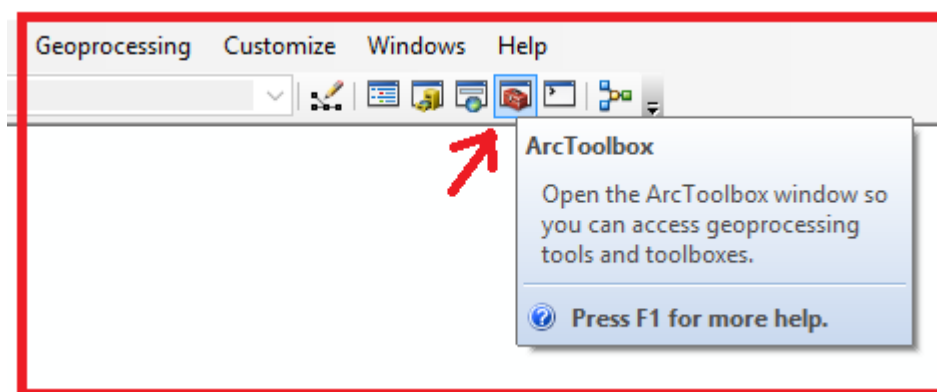


Figura 13- Zoom no local do ícone da janela ArcToolbox e sua breve descrição

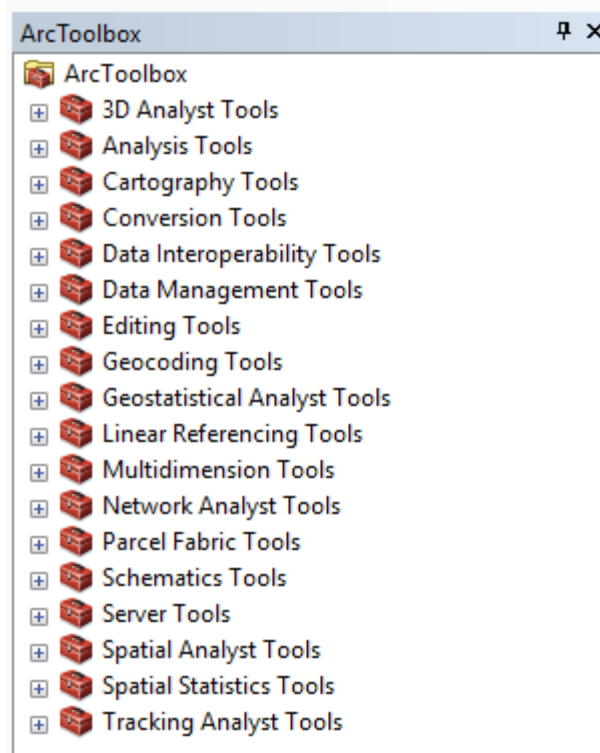


Figura 14- Janela ArcToolbox

A janela ArcToolbox contém ferramentas para utilização no ArcMap. Desde edição, transferência de arquivos e shapefiles à processos de tratamento de imagens, raster e conversão para diferentes formatos. A janela facilita a utilização da linguagem em Python para execução de nossas tarefas. Assim, as ferramentas são potencializadas com o intermédio do Python.

A partir dessa janela, podemos obter as informações de scripts, sintaxe e funcionalidade de forma organizada. Essas informações estão presentes na descrição do item de cada ferramenta. Irei abordar esse assunto a seguir.

## Descrição do item

Ao clicar com o botão direito em alguns itens presentes nas janelas ArcToolbox e Catálogo, há a opção de descrição de item. Normalmente, ele fica entre os últimos das opções abertas.

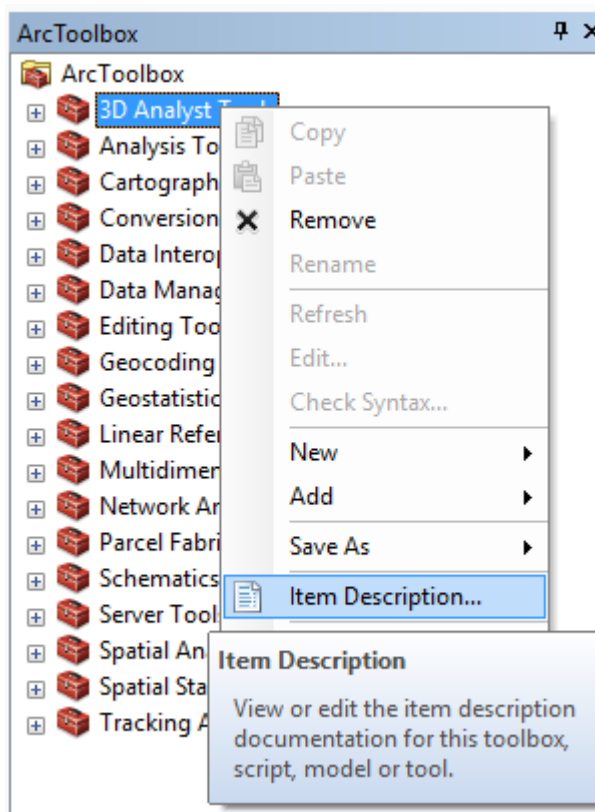


Figura 15- Opção de descrição de item na janela ArcToolbox

Essa opção é importante, pois, contém orientações, dicas e exemplos sobre a utilização do determinado item. Ao utilizá-lo em ferramentas, podemos entender a lógica de funcionamento e evitar erros precoces.

Ao clicar na opção Descrição do item uma nova janela será aberta. Essa janela contém os seguintes itens:

- Título: nome da ferramenta
- Resumo: descrição breve sobre a ferramenta
- Ilustração: imagens mostrando o uso da ferramenta

- Uso: orientações e cuidados na utilização
- Sintaxe: quais são os parâmetros e como escrever essa ferramenta em Python
- Amostras de Código: exemplos de utilização da ferramenta em Python

Por exemplo, caso eu queira utilizar a ferramenta Buffer e queira verificar a descrição do item dessa ferramenta, a seguinte janela pode ser aberta:

## Buffer (Analysis)

**Title** Buffer (Analysis)

### Summary

Creates buffer polygons around input features to a specified distance.

### Illustration

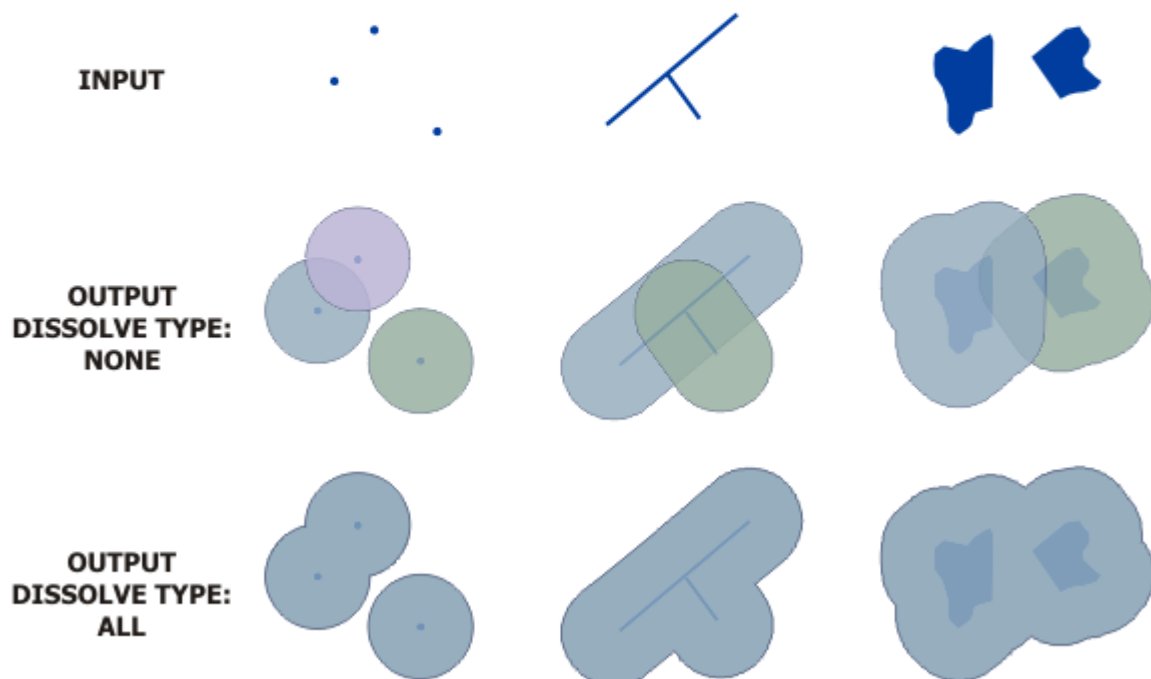


Figura 16- Título, resumo e ilustração da ferramenta Buffer



Usage

- As described in How Buffer works, an important feature of the Buffer tool is the Method parameter which determines how buffers are constructed. There are two basic methods for constructing buffers: Euclidean and geodesic.
    - Euclidean buffers measure distance in a two-dimensional Cartesian plane, where straight-line or Euclidean distances are calculated between two points on a flat surface (the Cartesian plane). Euclidean buffers are the more common type of buffer and work well when analyzing distances around features in a projected coordinate system, which are concentrated in a relatively small area (such as one UTM zone).
    - Geodesic buffers are those that account for the actual shape of the earth (an ellipsoid, or more properly, a geoid). Distances are calculated between two points on a curved surface (the geoid) as opposed to two points on a flat surface (the Cartesian plane). You should always consider creating geodesic buffers when
      - Your input features are dispersed (cover multiple UTM zones, large regions, or even the whole globe).
      - The spatial reference (map projection) of your input features distorts distances in order to preserve other properties such as area.Geodesic buffers may appear unusual on a flat map, but when displayed on a globe, these buffers will look correct (you can also use the ArcGlobe or ArcGIS Explorer applications to view geographic data on a three-dimensional globe). For more information, see How Buffer works.
- The Method parameter determines how buffers are created.

Figura 17- Orientações sobre o uso da ferramenta Buffer

Syntax

Buffer\_analysis (in\_features, out\_feature\_class, buffer\_distance\_or\_field, {line\_side}, {line\_end\_type}, {dissolve\_option}, {dissolve\_field}, {method})

Parameter	Explanation	Data Type
in_features	<b>Dialog Reference</b> The input point, line, or polygon features to be buffered.  <b>Python Reference</b> The input point, line, or polygon features to be buffered.	Feature Layer
out_feature_class	<b>Dialog Reference</b> The feature class containing the output buffers.	Feature Class

Figura 18- Sintaxe e descrição dos parâmetros da ferramenta Buffer

Code Samples

Buffer example 1 (Python window)

The following Python window script demonstrates how to use the Buffer tool.

```
import arcpy
arcpy.env.workspace = "C:/data"
arcpy.Buffer_analysis("roads", "C:/output/majorroadsBuffered", "100 Feet", "FULL", "ROUND", "LIST", "Distance")
```

Buffer example 2 (stand-alone script)

Find areas of suitable vegetation that exclude areas heavily impacted by major roads.

```
# Name: Buffer.py
# Description: Find areas of suitable vegetation which exclude areas heavily impacted by major roads

# import system modules
import arcpy
from arcpy import env

# Set environment settings
env.workspace = "C:/data/Habitat_Analysis.gdb"

# Select suitable vegetation patches from all vegetation
veg = "vegtype"
suitableVeg = "C:/output/Output.gdb/suitable_vegetation"
whereClause = "HABITAT = 1"
```

Figura 19- Exemplos de código da ferramenta Buffer

## Preparação antes de usar o Python no ArcMap

Antes de iniciar a utilização do Python para realizar tarefas no ArcGIS, sugiro configurar as barras de ferramentas e janelas do ArcMap e ter um conhecimento básico sobre a linguagem em Python. Portanto, irei apresentar qual a configuração eu utilizo no ArcMap e recomendações para linguagem em Python.

### Janelas e barras de ferramentas

Sugiro as seguintes barras de ferramentas:

- Padrão (Standard)
- Ferramentas (Tools)

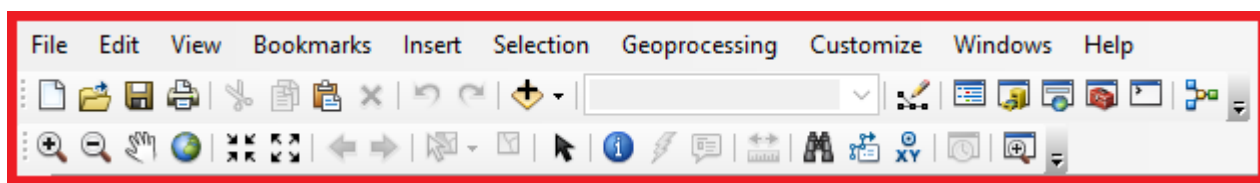


Figura 20- Barra de ferramentas Standard e Tools

Sugiro as seguintes janelas:

- Tabela de camadas (Table of Contents)
- Catálogo
- Pesquisar
- ArcToolbox

Sugiro clicar no ícone de fixação próximo ao botão de fechar dessa janela para fixar a janela do ArcToolbox. O mesmo deve ser feito para a janela Pesquisar e a janela Catálogo. Fixá-las facilita o manuseio das mesmas, o qual é corrente na utilização do ArcMap.

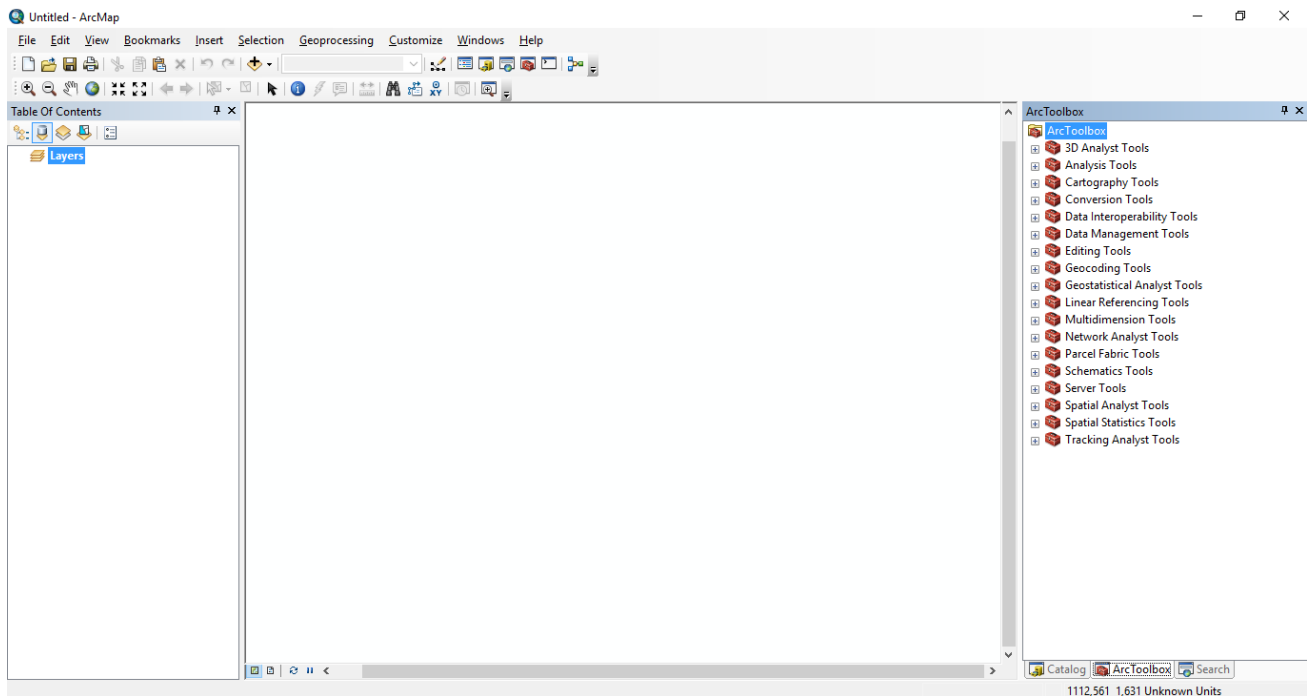


Figura 21- ArcMap com minha configuração

### **Requisitos mínimos para entender as orientações no Python**

O conhecimento básico sobre a linguagem Python permite um uso extenso e potencializado das ferramentas do ArcMap. Isso, pois, escrever em Python te dá uma liberdade para criar códigos e funcionalidades personalizadas.

Sugiro ler o início do tutorial do Python. Ele está disponível no IDLE do Python na aba Help > Python Docs.

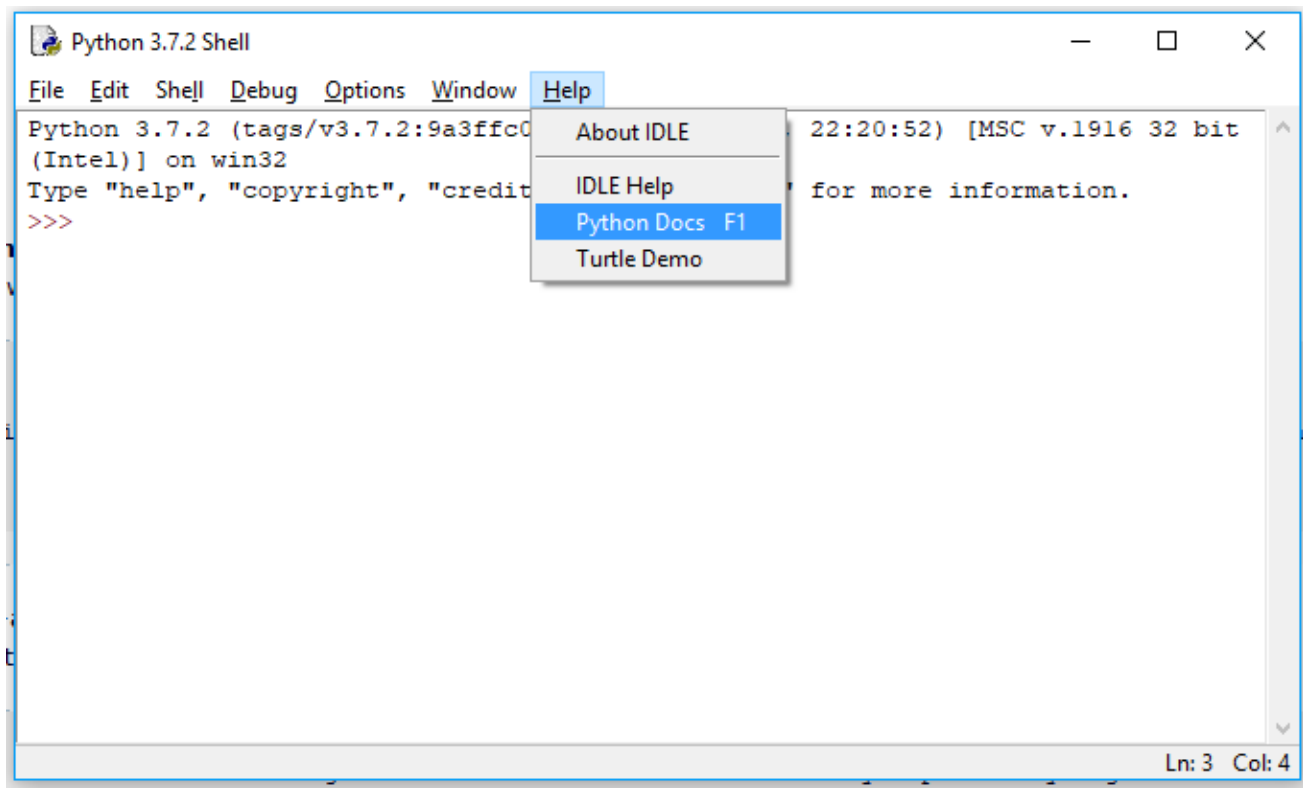


Figura 22- Python IDLE com aba Help aberta e seleciona em Python Docs

Ao abrir a janela de documentação do Python, selecione o item tutoriais do Python. Nele, há itens como Uma introdução informal no Python (“An Informal Introduction to Python”) e Mais Ferramentas de Controle (“More Control Flow Tools”), sugiro que leia esses.

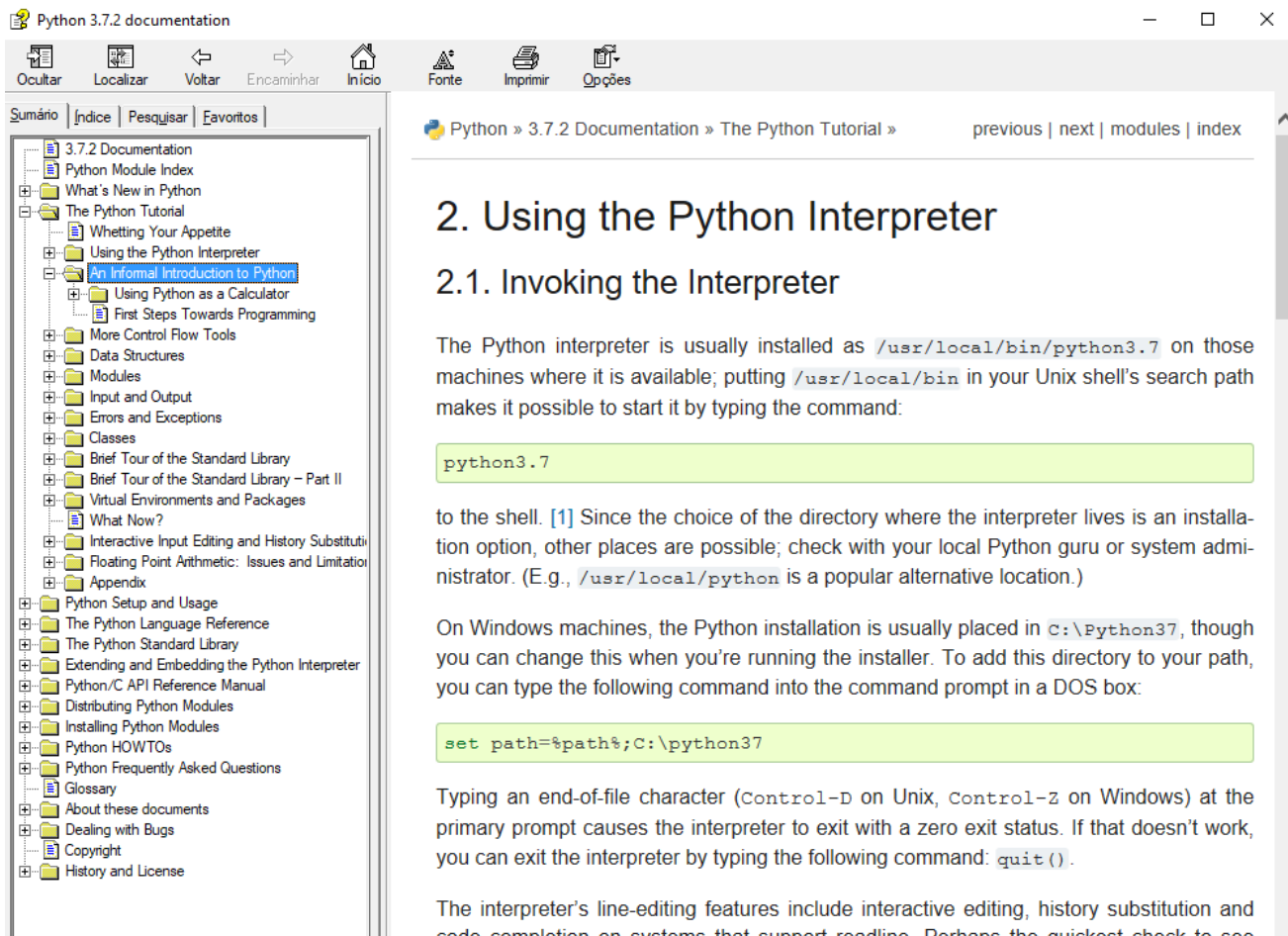



Figura 23- Janela de documentação do Python

Esses itens abordam sobre os conceitos de variável, string, lista, rotinas com for, while e alguns métodos que auxiliam no uso desses conceitos. Isso permite a execução de rotinas de repetição dentro do ArcMap. Irei explicar mais a frente, como utilizar esses conceitos dentro da construção do código do ArcMap.

## Python dentro e fora do ArcMap

Você pode escrever e executar o seu código em Python dentro ou fora do programa ArcMap. Para utilizar fora: você deve importar a biblioteca “arcpy” do ArcGIS. Para utilizar dentro: você deve abrir a janela de interação do Python no ArcMap.

### Utilizar dentro do ArcMap

Com a barra de ferramentas “Padrão” ativada, clique no ícone Python  ou clique na aba Geoprocessamento e clique no ícone Python.

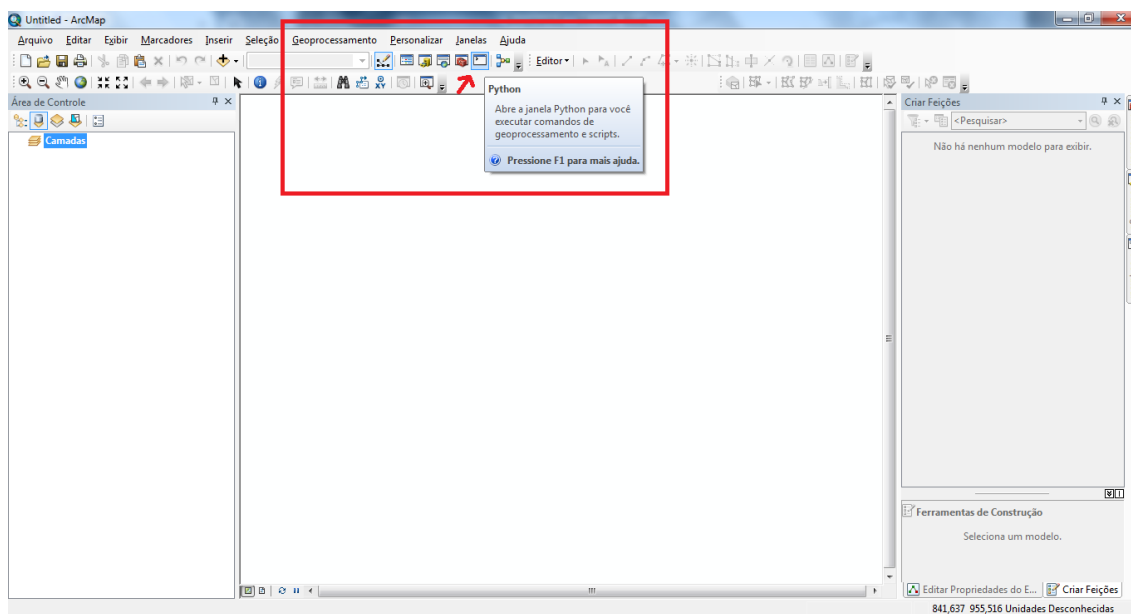


Figura 24– tela e localização do ícone da janela Python

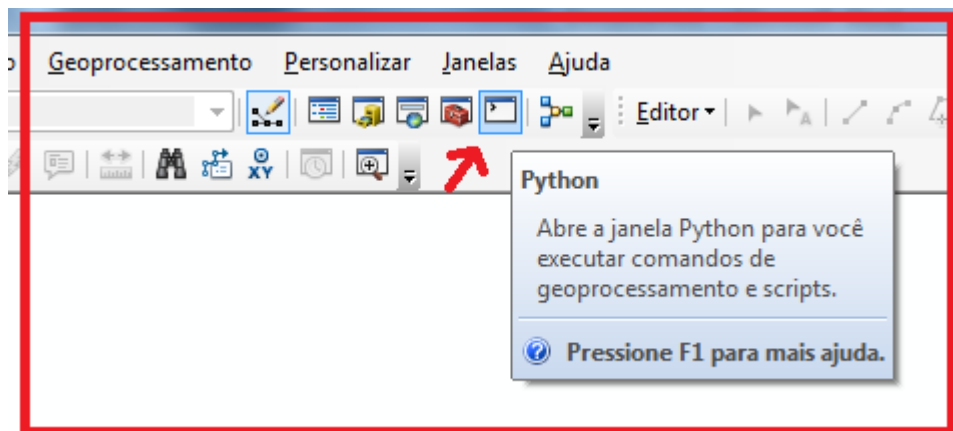


Figura 25- Zoom no local do ícone da janela Python e sua breve descrição

Ao clicar no ícone, o programa irá exibir uma janela na parte inferior da tela para escrever e executar os comandos em Python. Essa janela é dividida em duas partes

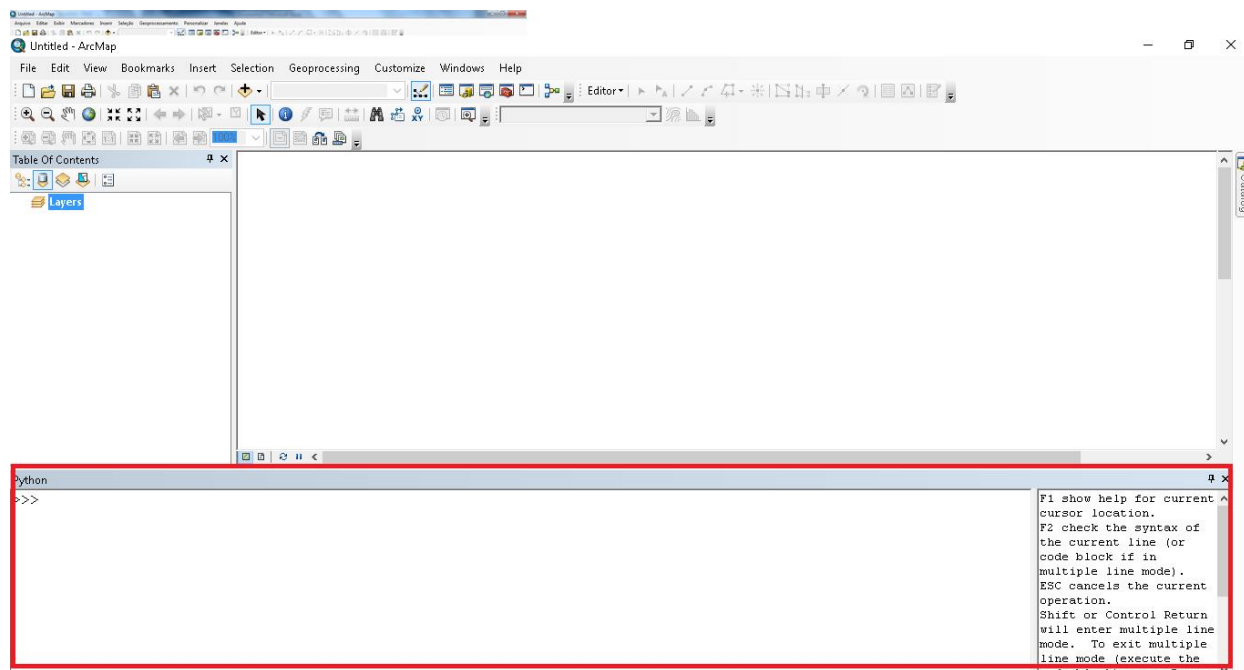


Figura 26- Tela e janela do Python

Essa janela é composta de duas partes: a parte que permite escrever o código e visualizar os erros e avisos obtidos e a parte que mostra informações de ajuda para escrever e executar o código:

```
Python
>>> print 'Hello World'
Hello World
>>> print 100 + 100
200
>>>
```

Figura 27- Parte da janela Python para escrever o código

```
print(value, ..., sep='
', end='\n',
file=sys.stdout)
Prints the values to a
stream, or to sys.stdout
by default.
Optional keyword
arguments: file: a file-
like object (stream);
defaults to the current
sys.stdout. sep: string
inserted between values,
default a space. end:
```

Figura 28- Parte da janela Python que fornece informações de ajuda



## Biblioteca Arcpy

Para utilizar alguma ferramenta do ArcMap é necessário iniciar a escrita com “arcpy” seguido de um ponto. Isso significa que você está acessando a biblioteca das ferramentas do ArcGIS. Ao escrever dessa forma, o programa irá te auxiliar abrindo uma janela com as opções de ferramentas disponíveis.

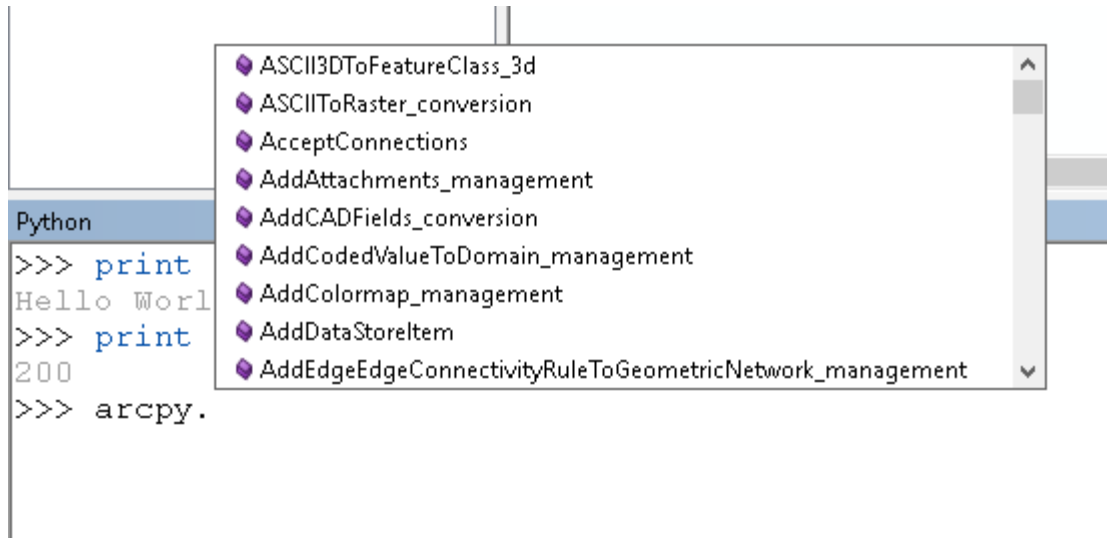


Figura 29- Uso da biblioteca arcpy na janela Python do ArcMap

Por exemplo, ao utilizar a ferramenta Buffer, a escrita fica igual a: “arcpy.Buffer\_analysis()”. Vale lembrar que a ferramenta buffer é utilizada para criar um polígono envolta de sua feição de entrada (arquivo de entrada).



Figura 30- Código para chamar a ferramenta Buffer na janela Python

Sendo que dentro dos parênteses são colocados os parâmetros da ferramenta. É possível consultar esses parâmetros a partir da janela de ajuda (canto direito).

```
Buffer analysis
(in_features,
out_feature_class,
buffer_distance_or_field
, {line_side},
{line_end_type},
{dissolve_option},
{dissolve_field;dissolve
_field...}, {method})
Creates buffer polygons
around input features to
a specified distance.
```

Figura 31- Parâmetros da ferramenta buffer mostrados na janela de ajuda Python

O item grifado de amarelo queimado é o parâmetro que deve ser inserido. Para inserir outro parâmetro, coloca-se a vírgula e o parâmetro. Aqueles que estão entre chaves “{}” não são obrigatórios.

## Interação com itens e a janela Python

Uma das vantagens de se utilizar o Python dentro do ArcMap é a facilidade de interação com os outros itens do programa ArcMap. Por exemplo, caso eu tenha uma camada em shapefile com o nome “teste” e queira usá-la na escrita do código em Python, eu posso inseri-la diretamente no código. Basta clicar com o mouse na camada e arrastar o item, ele aparece direto no código.

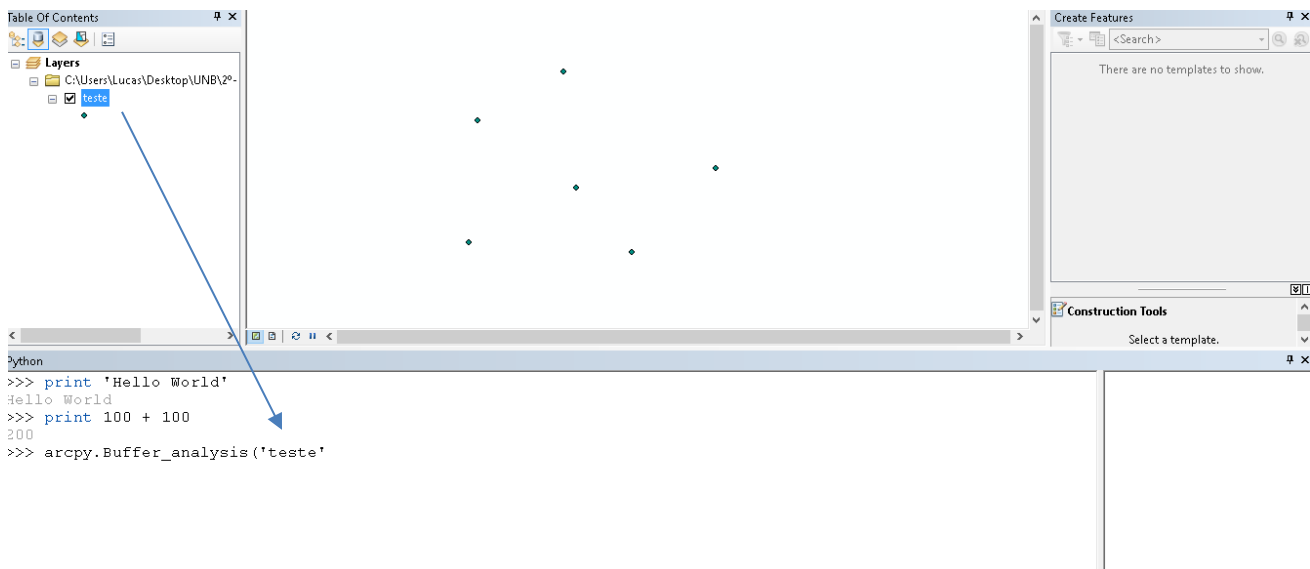


Figura 32- Interação entre a janela Tabela de camadas e a janela Python

Também, posso clicar em uma pasta fora da janela do ArcMap e indicar a pasta como o endereço para o arquivo de saída da ferramenta em uso, basta arrastar a pasta para a janela do Python e o caminho dela será inserido.

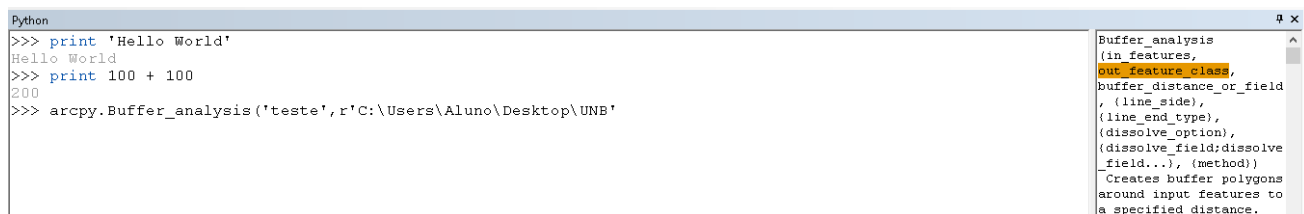


Figura 33- Interação entre uma janela externa e a janela Python

Ao complementar o código e executá-lo, a janela Python marca o final do processo e o resultado pode ser visto no ArcMap.

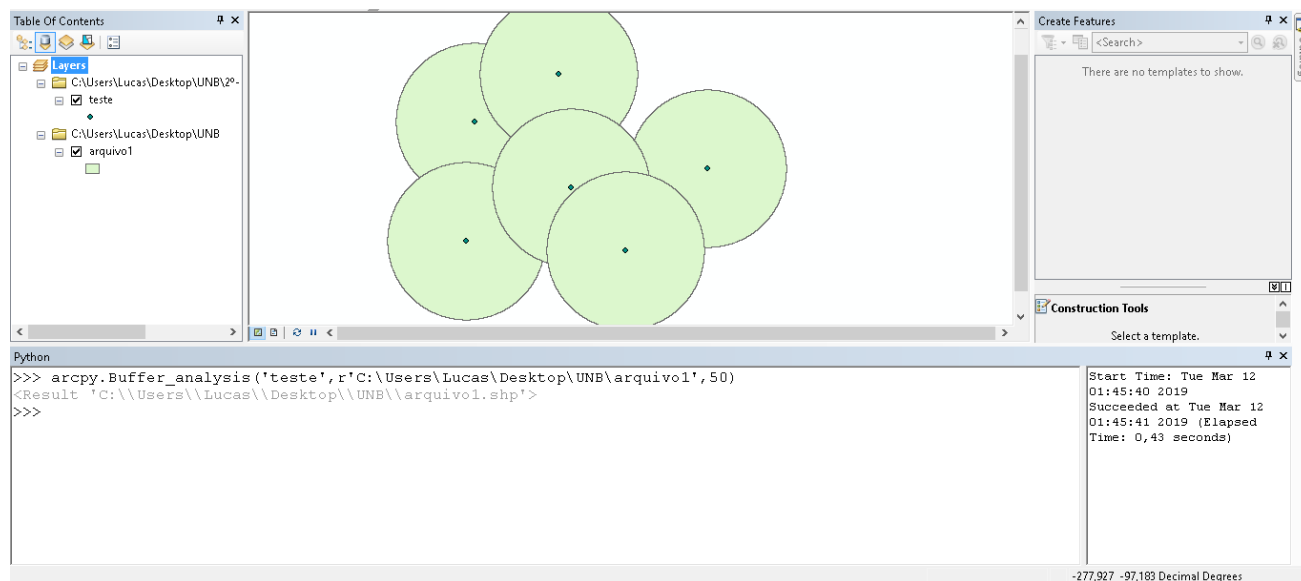


Figura 34- Resultado do uso da ferramenta Buffer\_analysis

## Python fora

A utilização do Python fora da janela Python do ArcMap pode ser realizada por meio de um programa IDE (um ambiente para desenvolvimento). Eu utilizo o Microsoft Visual Studio 2017 para escrever e executar o código em Python. Entretanto, ele pode ser feito em outros programas, como Pycharm, o próprio IDLE do Python, Pythonwin 2.7, etc.

Ao utilizar o Python fora do ArcMap, é importante lembrar que a biblioteca “arcpy” deve ser executada em um ambiente virtual Python 2.7 (32 bits). No caso do Visual Studio essa informação está presente na seguinte janela:

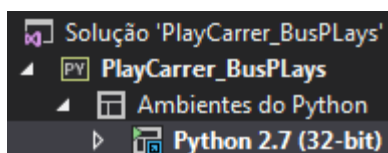


Figura 35- Ambiente do Python 2.7 (32 bits) selecionado

## Escrevendo o código em Python

A escrita do Python utiliza-se de variáveis, funções e módulos para construção de um código. Inicialmente, devemos importar todos os módulos que iremos utilizar. Para utilizar as ferramentas que o programa ArcMap possui, é necessário importar a biblioteca “arcpy”

```
1 #Importar biblioteca
2 import arcpy
3
```

Figura 36- Comando para importar o módulo arcpy

Após a importação, podemos definir o local da pasta de trabalho. No momento de instalação do ArcMap, o programa define uma pasta padrão para que arquivos shapefiles, raster sejam salvos automaticamente. O programa procura os arquivos nessa pasta no momento de executar o código Python. Caso deseje definir o endereço de pasta no qual o ArcMap irá procurar os arquivos, basta executar o seguinte comando “arcpy.env.workspace”:

```
4 #Variáveis recebem os seguintes valores:
5 nomedapasta = r'C:\Users\Lucas\Desktop\UNB'
6
7 # Estabelecer o ambiente de trabalho
8 arcpy.env.workspace = nomedapasta
9
```

Figura 37- Comando para selecionar a pasta que será usada como ambiente de trabalho

## Ferramenta de análise Buffer

Utilizarei a ferramenta Buffer como exemplo para executar o código em Python. Ela é uma ferramenta simples que cria um polígono em torno de um elemento com um raio de distância que pode ser determinado. Ela pode ser acessada no ArcMap ao clicar na aba “Geoprocessamento” e escolher a opção Buffer. Essa ferramenta pode ser executada pelo comando em Python através do comando “Buffer\_analysis”, o qual possui três parâmetros que devem ser escritos obrigatoriamente: o shapefile de base para o buffer, o nome de saída do arquivo e a distância do buffer.

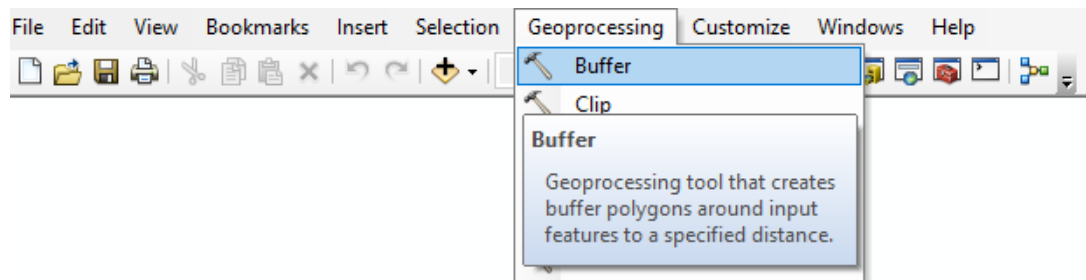


Figura 38- Selecionar a ferramenta Buffer

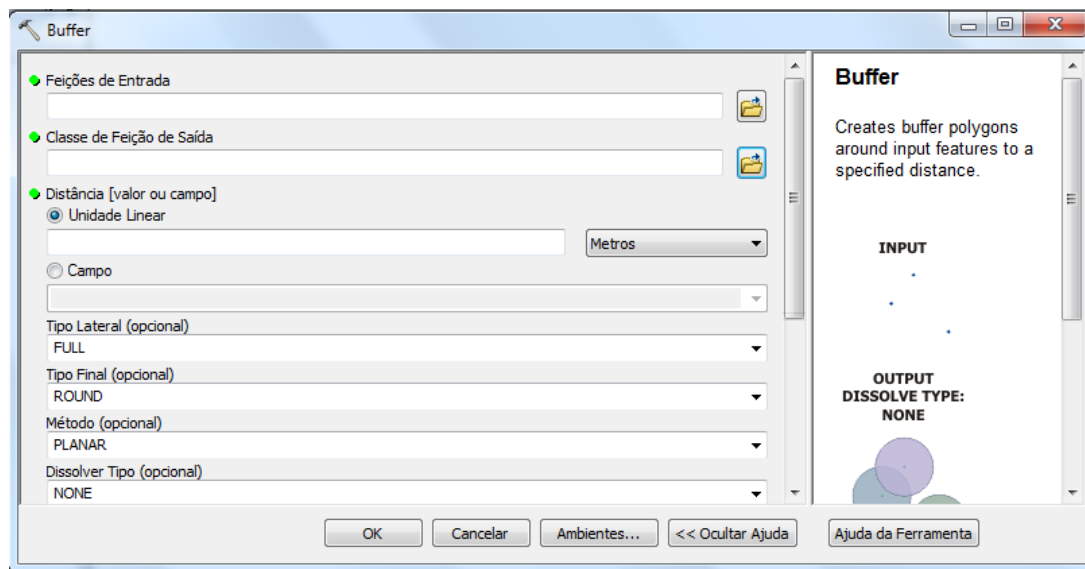


Figura 39- Janela da ferramenta Buffer

Esse comando também se encontra na janela Pesquisar, basta procurar pela palavra “Buffer”.

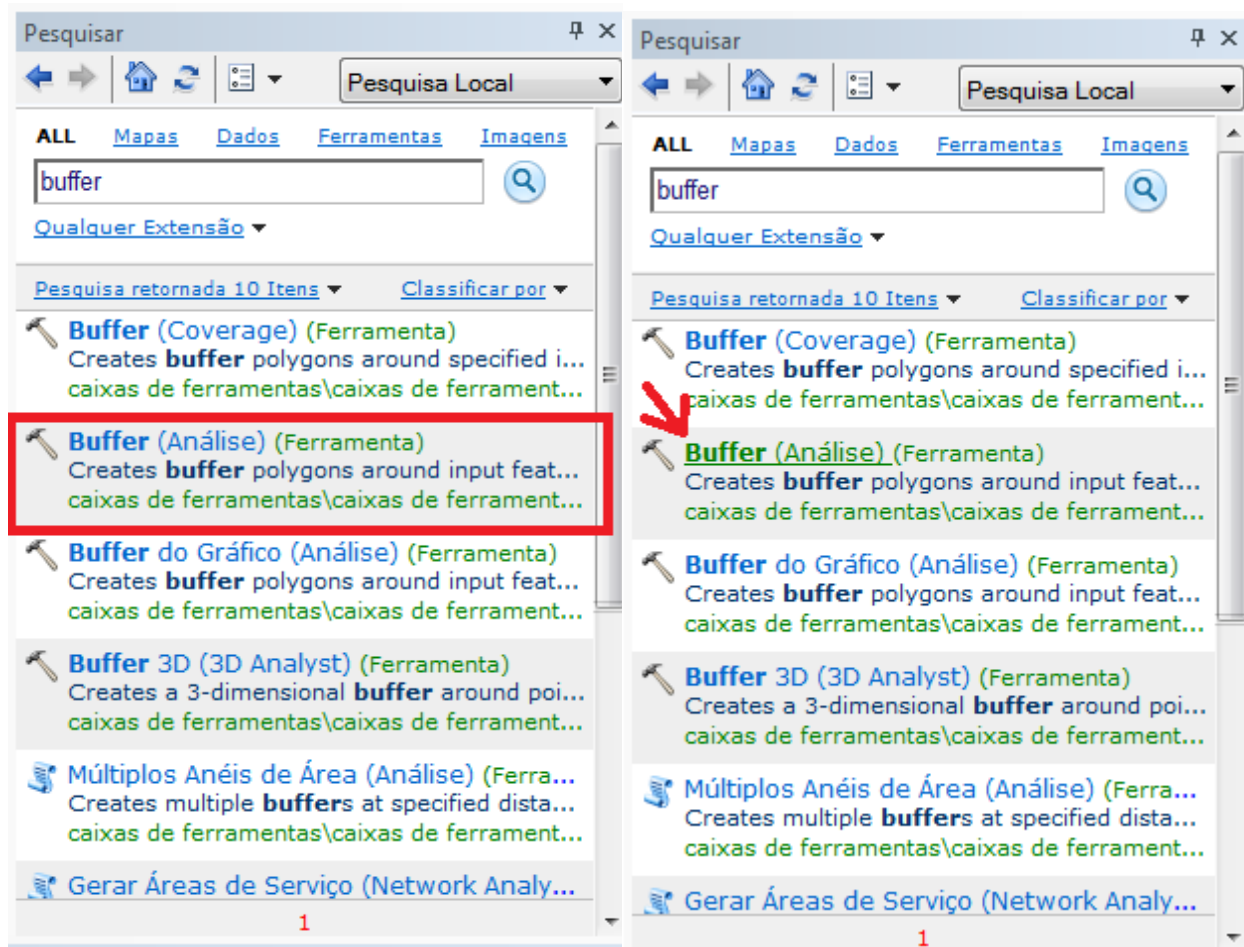


Figura 40- Procure a ferramenta Buffer na janela Pesquisar

Para selecionar a ferramenta, basta clicar em cima de seu nome. Entretanto, utilizaremos a barra de busca para outros fins. Para isso, clique sobre o ‘link’ abaixo de “Buffer (Análise)” e da descrição em azul. Ao clicar no link, você será redirecionado para a janela Catálogo diretamente na ferramenta que você estava procurando. No caso, a ferramenta Buffer.



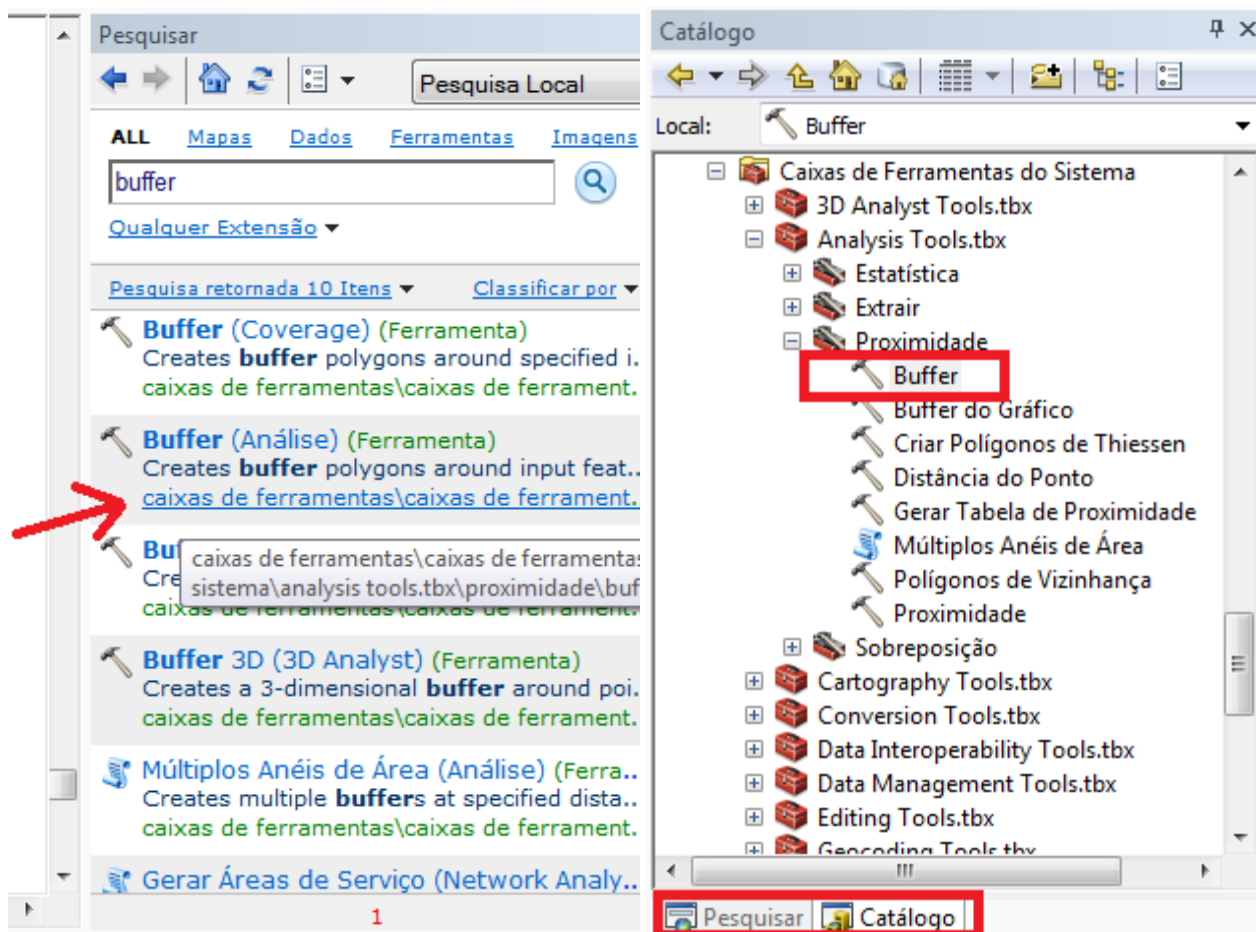


Figura 41- Ferramenta Buffer, Janela Pesquisar e Janela Catálogo

## Aprendendo os comandos em Python

Os comandos em Python não são tão intuitivos como a interface do ArcMap pode ser. Em vista disso, o programa oferece uma série de ferramentas para facilitar a escrita em Python. Voltando a janela Catálogo, clique com o botão direito do mouse na ferramenta que deseja selecionar. No nosso caso, essa ferramenta é o Buffer. Dessa forma, abrirá algumas opções sobre a ferramenta. Clique na descrição do item para ter mais informações sobre a mesma.

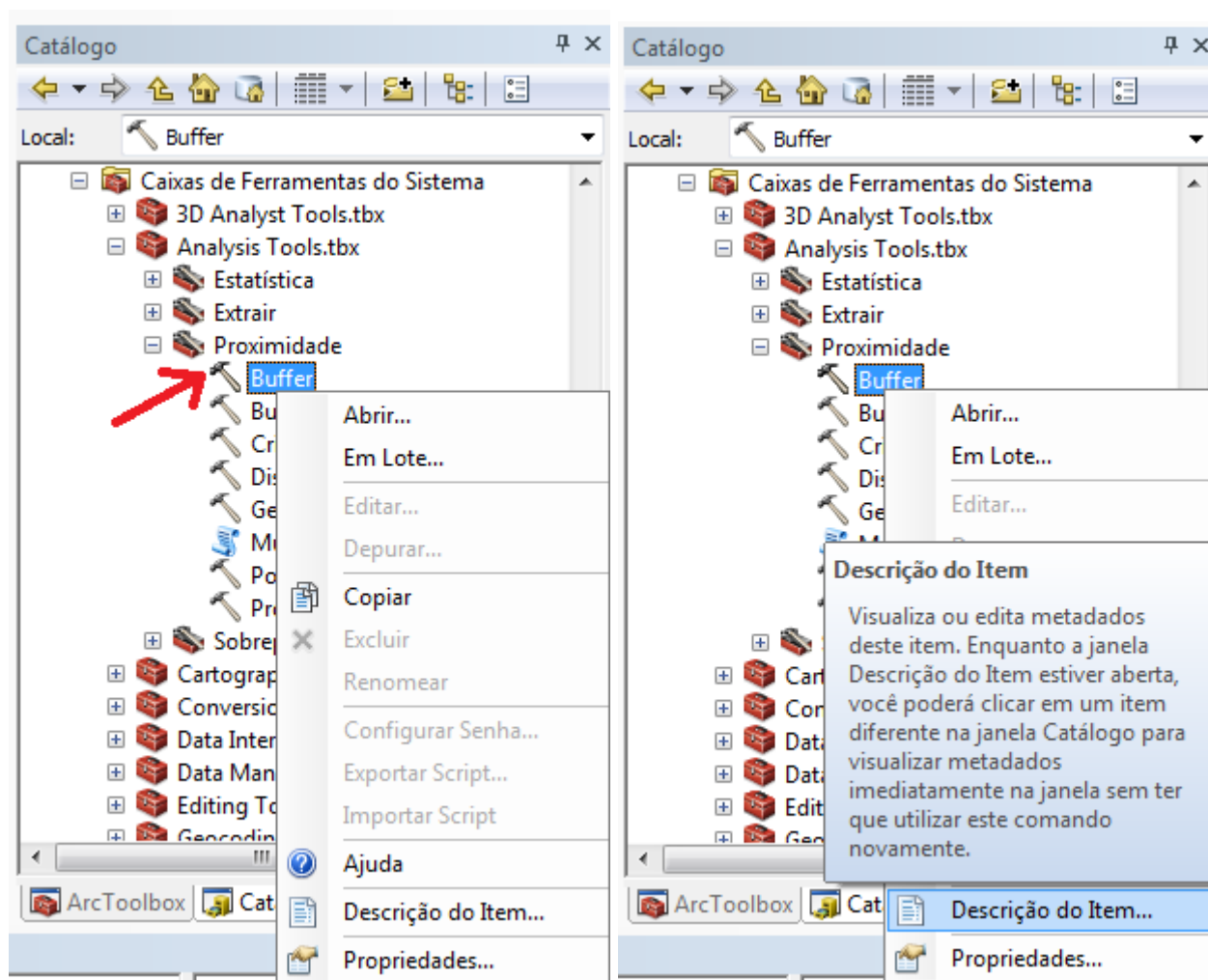


Figura 42- Opções ao clicar com o botão direito nas ferramentas

Como já visto anteriormente nas configurações básicas do ArcMap para o tutorial > Descrição do item, a janela a seguir irá aparecer:

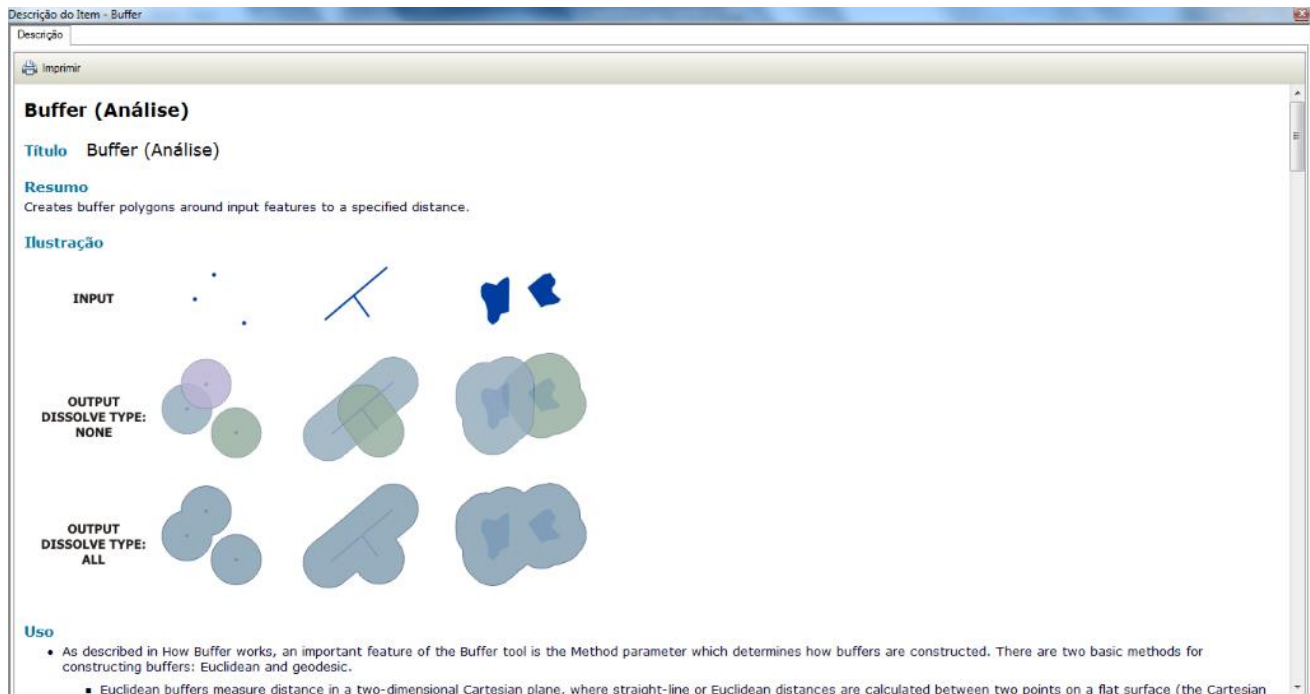


Figura 43- Descrição do item Buffer

Essa janela contém três partes fundamentais para escrever algo em Python. Ele indica dicas e cuidados no momento de escolher os parâmetros da ferramenta, a sintaxe que deve ser utilizada e exemplos em Python.

## Resultados

Toda vez que utilizamos uma ferramenta a partir da janela de Catálogos ou do ArcToolbox o ArcMap registra suas ações no seu sistema. Isso permite o monitoramento dos seus processos e a obtenção do seu código a partir desse resultado. Para acessar essa função, clique na aba “Geoprocessamento” (Geoprocessing) e depois em “Resultados” (Results).

A janela Resultados marca as ações que foram feitas no seu ArcMap. Ela dá acesso aos resultados das ações que foram executadas no ArcMap. Ao executar, por exemplo, a ferramenta Buffer, o resultado ficará marcado. Clique no item Buffer para visualizar suas opções.

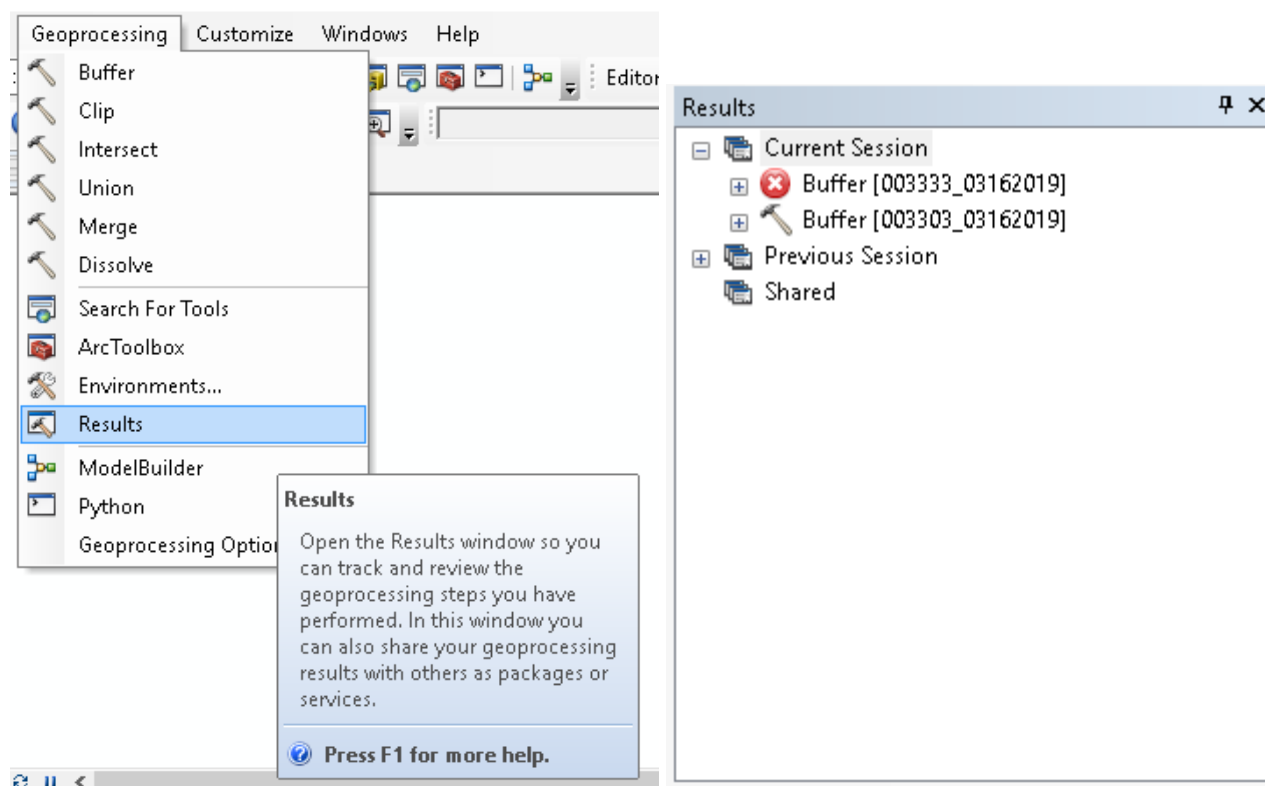


Figura 44- Aba geoprocessamento e janela Resultados

## Copiar como Python Snippet

A janela Resultados possui a opção “Copiar como Python Snippet” (Copy As Python Snippet). Ela permite que as ações que foram realizadas no ArcMap sejam escritas em Python. No caso do Buffer, clique em Resultados e depois clique com o botão direito sobre o item que deseja utilizar. O código será copiado para a área de transferência. Basta, colar para visualizar o comando.

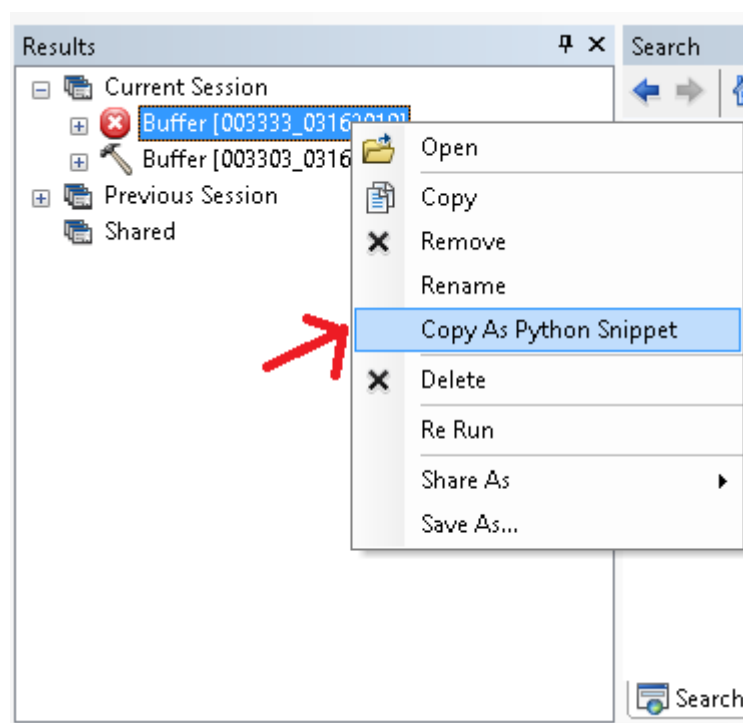


Figura 45- Janela Resultados e opção Copy As Python Snippet

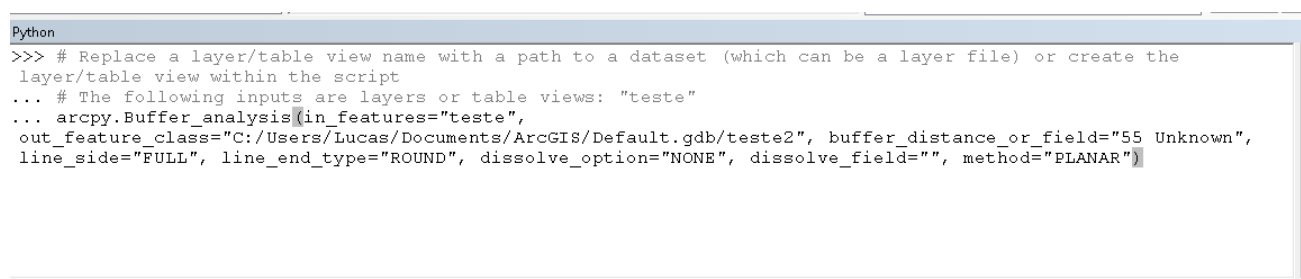


Figura 46- Texto copiado do Copy As Python Snippet e colado na janela Python

O Snippet Python permite que você execute comandos dentro do ArcMap, aproveitando-se de sua interface mais intuitiva e posteriormente utilize a sintaxe da ferramenta e os parâmetros que foram utilizados. Assim, podemos ver seus resultados, importar e reproduzir os comandos no Python.

O Snippet Python deve ser utilizado em conjunto com a Janela Pesquisar, a Janela ArcToolbox e Janela Catálogo, pois assim, procuramos a ferramenta desejada, verificamos seu uso na descrição do item, executamos no ArcMap e copiamos o comando para o Python. Cabe salientar que algumas ferramentas não são marcadas na Janela Resultados, isso acontece, por exemplo, ao clicar na aba Seleção e escolher as ferramentas de seleção por atributos ou por localização. Quando isso acontecer, basta procurar uma ferramenta equivalente na Janela Pesquisar e executá-la.

## Ferramentas de seleção de feição

O uso da ferramenta de selecionar feições, seja por atributos ou localização não é registrada nos resultados quando ela é utilizada por meio da aba “Seleção” (Selection). Para que o uso da mesma seja registrado, basta utilizar a barra de pesquisa e procurar pela ferramenta. Sempre que algo não for registrado tente procurar essa ferramenta na barra de pesquisa ou ArcToolbox.

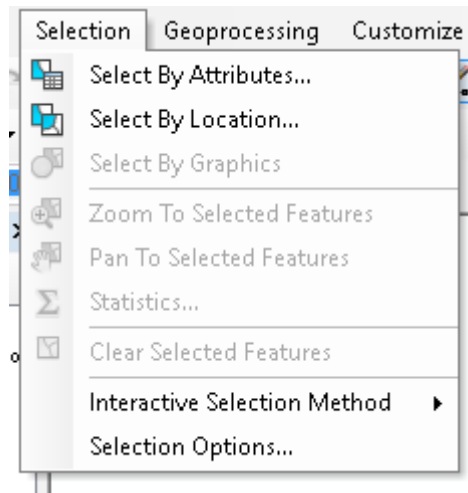


Figura 47- Utilizando comandos pela aba Seleção

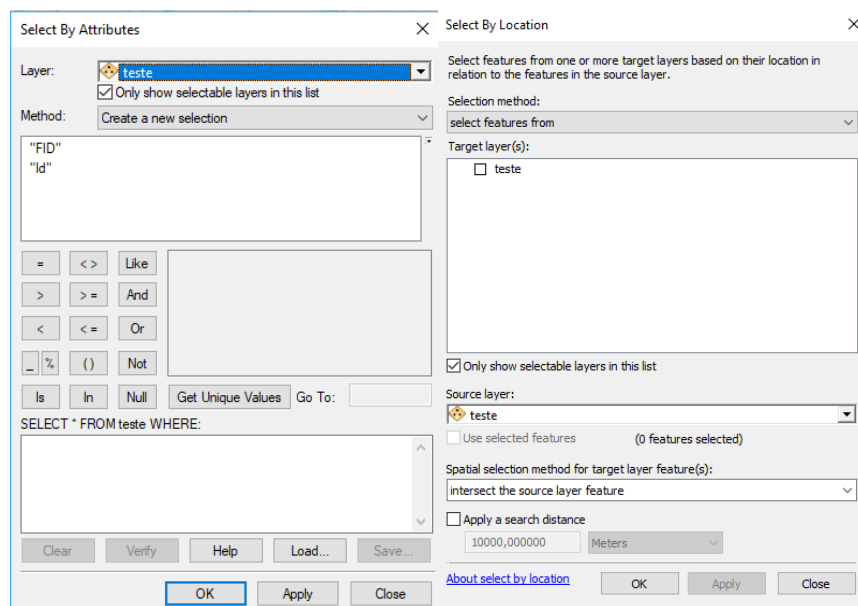


Figura 48- Janela Seleção por atributos e Janela Seleção por localização

Posso procurar na janela Pesquisar as ferramentas Seleccionar por atributos ou localização. Para isso, basta, procurar por uma palavra semelhante as ferramentas, no meu caso em que utilizo o ArcMap em inglês devo pesquisar nesse idioma “Selection”. Ao clicar sobre os itens, uma janela equivalente ao visto anteriormente será executada.

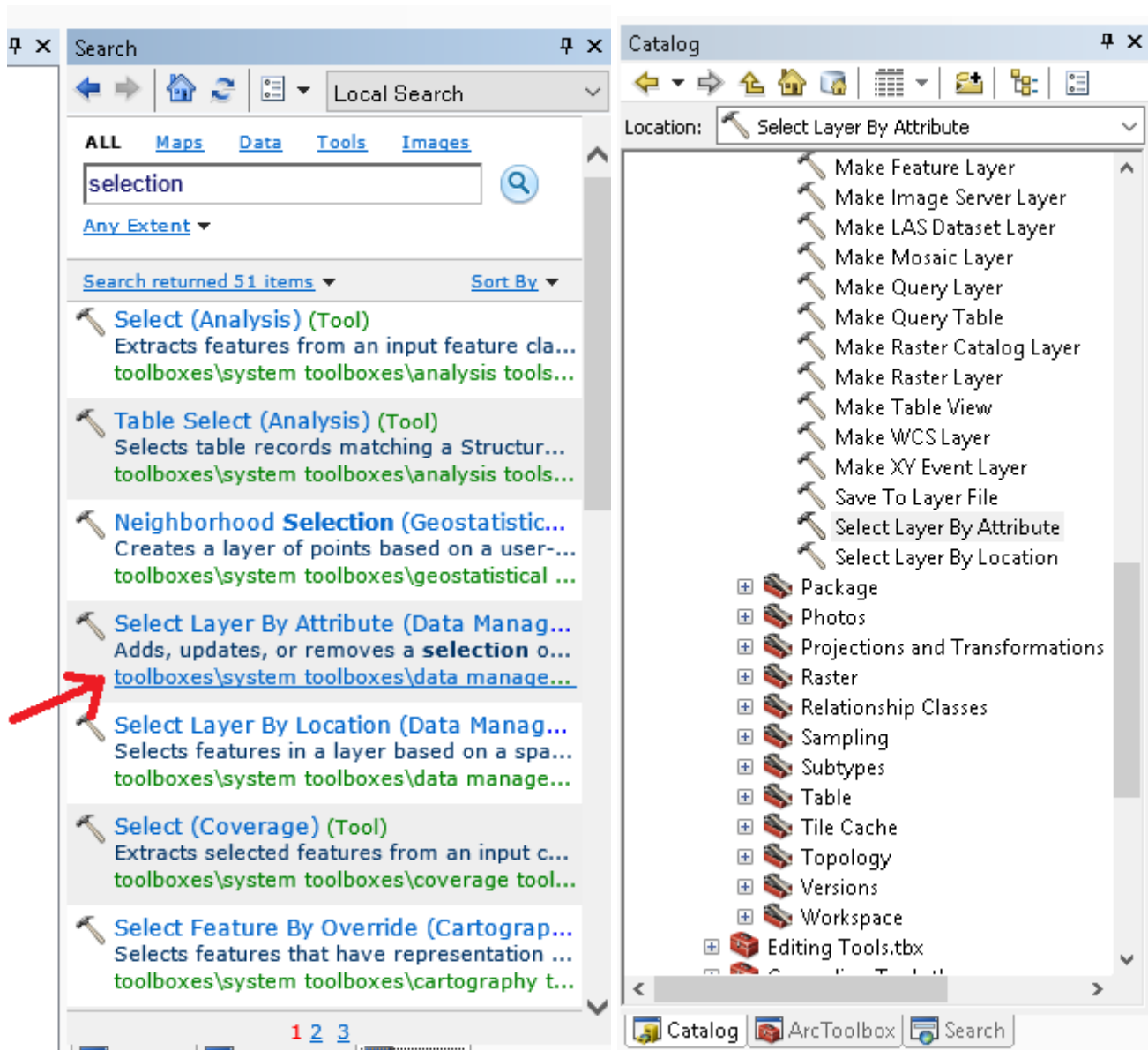


Figura 49- Janela Pesquisar e resultado na Janela Catálogo



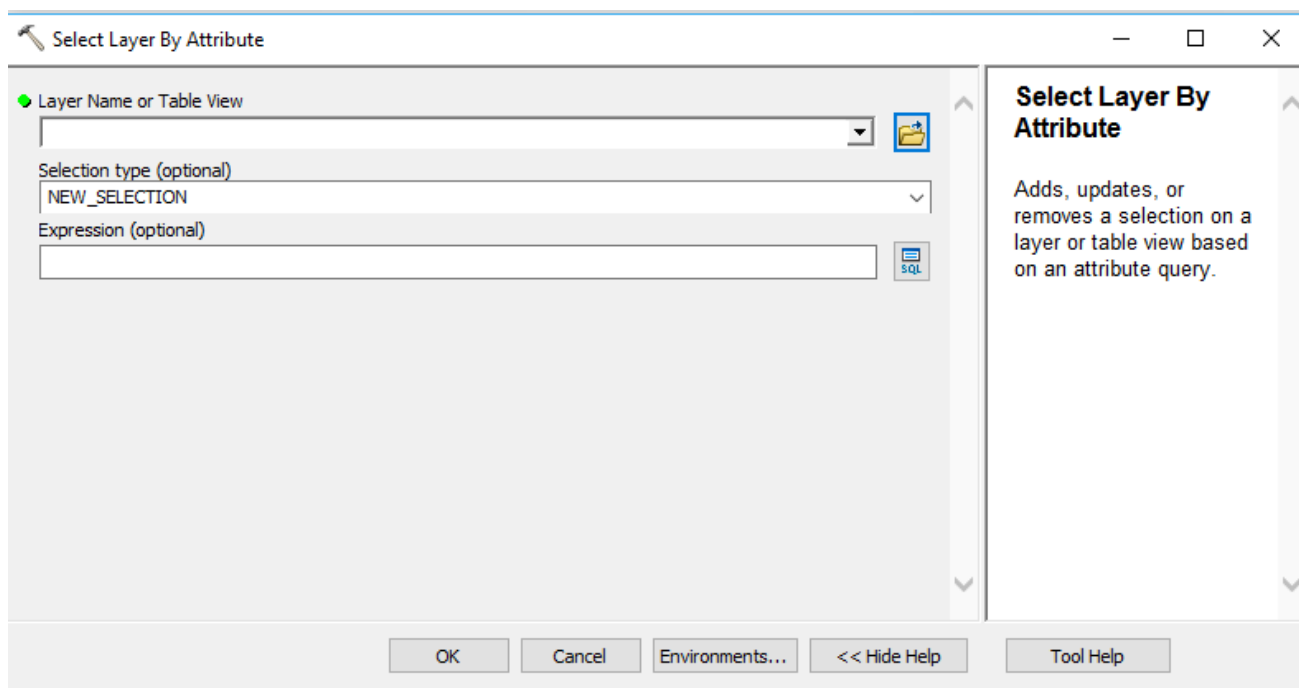


Figura 50- Janela equivalente para a ferramenta Selecionar por atributos

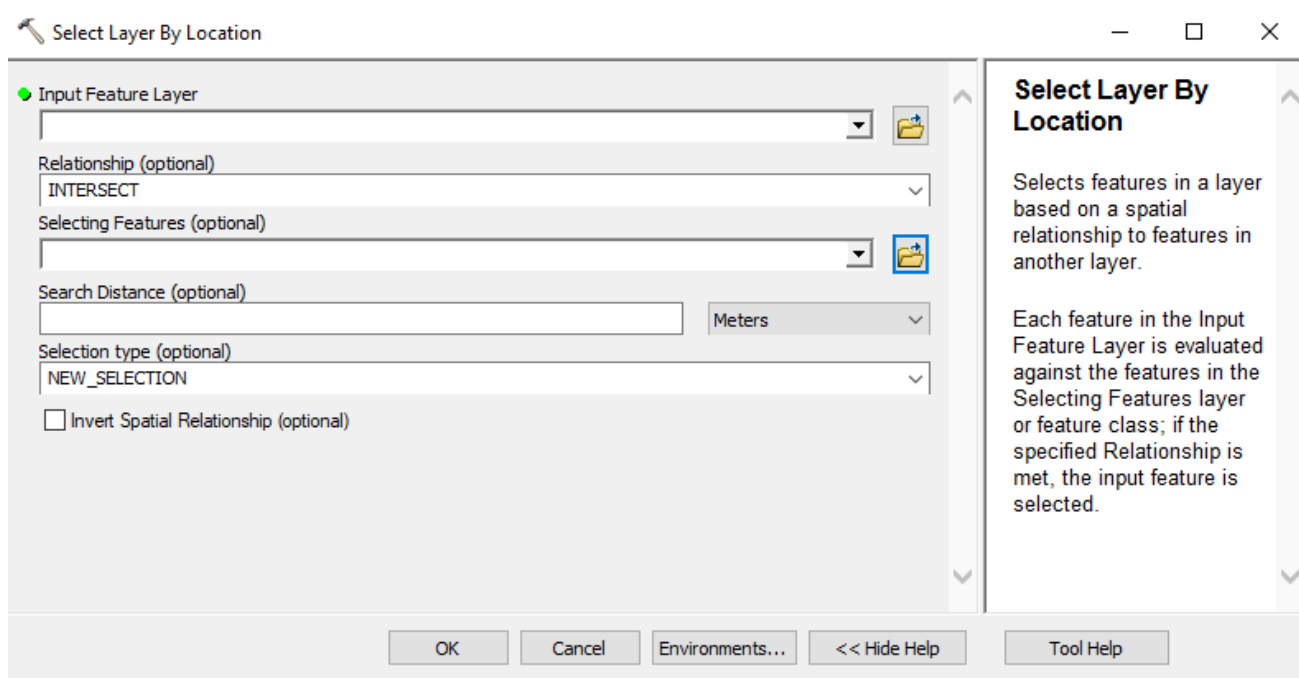


Figura 51- Janela equivalente para a ferramenta Selecionar por localização

## Exemplo

### Exemplo 1 – Utilizar a condição while

Neste exemplo, irei utilizar a ferramenta seleção por atributos e a ferramenta Buffer com o auxílio da condição while. Caso ocorra algum problema em executar o código fora do ArcMap, sugiro escrever o código fora e copiar e colar dentro da Janela do Python do ArcMap.

Irei utilizar a condição while, pois, ela me permite que eu repita o uso da ferramenta e que crie arquivos com diferentes nomes. No caso, minha ideia é selecionar as feições pelo número de identificação delas (FID) e criar um polígono com a ferramenta buffer para cada uma delas, sendo que cada uma terá uma distância diferente proporcional ao seu FID.

Neste caso, a camada 'teste' que representa um shapefile de pontos de várias cidades próximas ao Distrito Federal possui as seguintes informações em seus atributos

Table

teste

	FID	Shape *	Id	name
▶	0	Point	0	Brasilia
	1	Point	0	Goiânia
	2	Point	0	Luziânia
	3	Point	0	Anápolis
	4	Point	0	Formosa

Figura 52- Janela da Tabela de atributos da camada 'teste'

```

1  # coding=utf-8
2  # Importando a biblioteca
3
4  import arcpy
5
6  # A variável recebe os seguintes valores
7
8  path = r'C:\Users\Lucas\Desktop\UNB'
9  arcpy.env.workspace = path
10
11 # Definir variáveis para usar como números
12 i = 0
13
14 # Executando a ferramenta selecionar
15
16 while i < 5:
17     arcpy.SelectLayerByAttribute_management(in_layer_or_view="teste",
18                                             selection_type="NEW_SELECTION",
19                                             where_clause='"FID" = ' + str(i))
20     arcpy.Buffer_analysis('teste', 'teste_' + str(i) + '.shp', 10000 + 5000*i)
21     i = i + 1
22

```

Figura 53- Código para selecionar e executar o Buffer

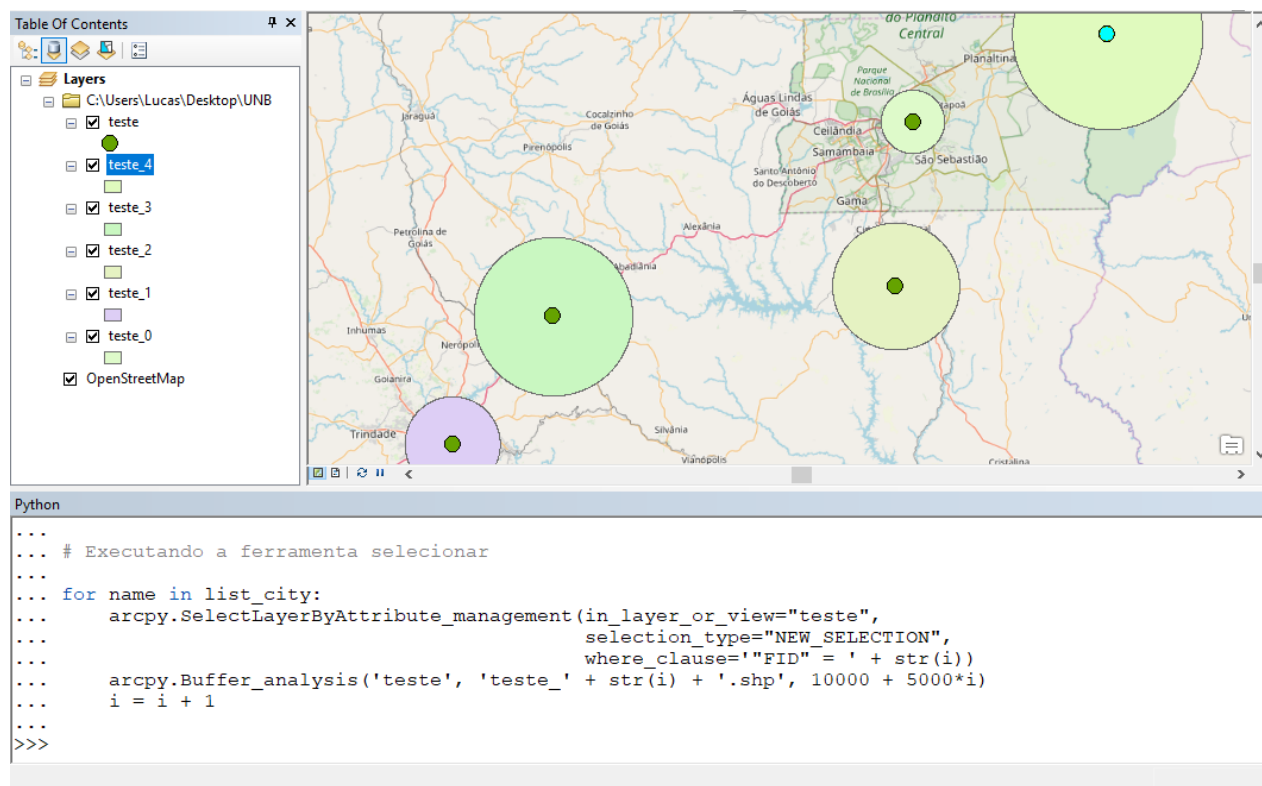


Figura 54- Resultado da execução do comando

Alguns pontos devem ser observados:

- Ao alterar o parâmetro `where_clause` devemos acrescentar o valor `i` como string. Esse parâmetro está executando uma expressão em SQL para selecionar os arquivos desejados, como se fosse uma linguagem dentro de outra. Portanto, é importante verificar o uso das aspas e variáveis, pois, os mesmos podem gerar diversos erros.
- Verifique se o arquivo que você quer criar já existe, pois, o ArcMap não substitui o arquivo diretamente. Caso execute com um nome já existente na mesma pasta, o código apresentará erro
- Utilize aspas simples no começo da escrita da expressão em SQL, pois, a mesma necessita de aspas duplas para identificar a string "FID"

```
while i < 5:  
    arcpy.SelectLayerByAttribute_management(in_layer_or_view="teste",  
                                           selection_type="NEW_SELECTION",  
                                           where_clause="'FID' = " + str(i))
```

Figura 55- Escrita do código está errada. Iniciar a expressão SQL com aspas " está incorreto

```
Runtime error Traceback (most recent call last):  File "<string>", line 19, in <module>  File "c:\program  
files (x86)\arcgis\desktop10.5\arcpy\arcpy\management.py", line 7713, in SelectLayerByAttribute      raise e  
ExecuteError: ERROR 000358: Invalid expression Failed to execute (SelectLayerByAttribute).  
>>>
```

Figura 56- Erro na expressão SQL

- Caso utilize alguma expressão SQL em que utilizará nomes (strings) é preciso ter cuidado com as aspas ao escrever a função. Utilizarei o exemplo anterior com o uso da condição "for" para exemplificar.

```

1  # coding=utf-8
2  # Importando a biblioteca
3
4  import arcpy
5
6  # A variável recebe os seguintes valores
7
8  path = r'C:\Users\Lucas\Desktop\UNB'
9  arcpy.env.workspace = path
10
11 # Definir lista com os nomes das cidades
12
13 list_city = ['Brasília', 'Goiânia', 'Luziânia', 'Anápolis', 'Formosa']
14
15 # Executando a ferramenta selecionar
16
17 for name in list_city:
18     print('"name" = ' + name)
19

```

Figura 57- Verificação de como o programa entende a string na expressão SQL

```

... # Executando a ferramenta selecionar
...
... for name in list_city:
...     print('"name" = ' + name)
...
"name" = Brasília
"name" = Goiânia
"name" = Luziânia
"name" = Anápolis
"name" = Formosa

```

Figura 58- Resultado incorreto para a expressão SQL

O resultado não coloca o nome das cidades em aspas, portanto, a expressão SQL estará incorreta. Uma forma de resolver isso é rescrever dessa forma:

```

... # Executando a ferramenta selecionar
...
... for name in list_city:
...     print('"name" = ' + '\'' + name + '\')
...
"name" = 'Brasília'
"name" = 'Goiânia'
"name" = 'Luziânia'
"name" = 'Anápolis'
"name" = 'Formosa'

```

Figura 59- Forma correta de se escrever a string para a expressão SQL.

### Exemplo 2 – Utilizar a condição for e listas

Levando em consideração as observações feitas no exemplo 1, também posso reescrever o meu código e utilizar listas e a condição “for” para executar a ferramenta Buffer. Irei definir uma lista com os nomes das cidades, os quais já estão adicionados como um atributo da camada ‘teste’. A partir desses nomes utilizarei a condição for para selecionar cada um deles e criar um Buffer específico.

```

1  # coding=utf-8
2  # Importando a biblioteca
3
4  import arcpy
5
6  # A variável recebe os seguintes valores
7
8  path = r'C:\Users\Lucas\Desktop\UNB'
9  arcpy.env.workspace = path
10
11 # Definir lista com os nomes das cidades
12
13 list_city = ['Brasília', 'Goiânia', 'Luziânia', 'Anápolis', 'Formosa']
14
15 # Executando a ferramenta selecionar
16
17 for name in list_city:
18     arcpy.SelectLayerByAttribute_management(in_layer_or_view="teste",
19                                             selection_type="NEW_SELECTION",
20                                             where_clause='"name" = ' + '\'' + name + '\')
21     arcpy.Buffer_analysis('teste', 'teste' + name + '.shp', 10000)
22

```

Figura 60- Exemplo anterior com o uso de listas e a condição "for"

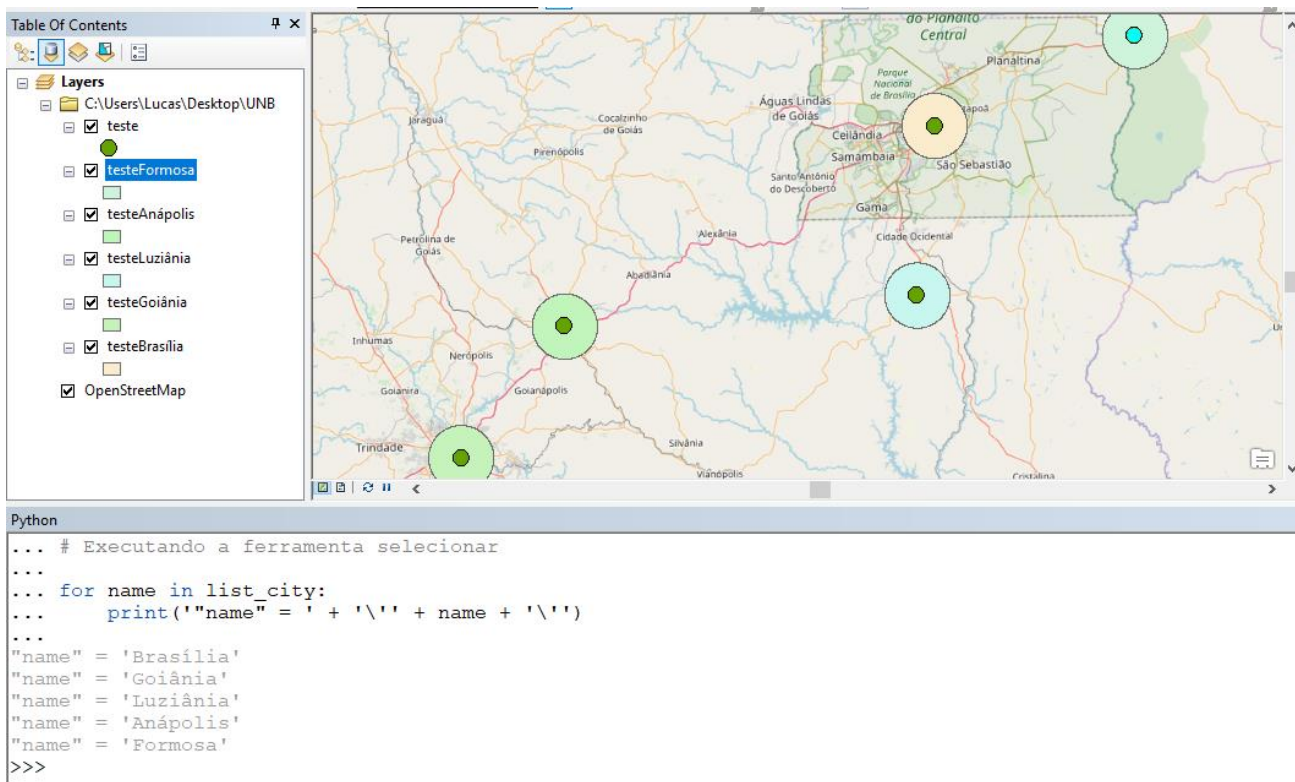


Figura 61- Resultado da execução do comando

## Anexos

### Código – Exemplo 1

```
# coding=utf-8

# Importando a biblioteca

import arcpy

# A variável recebe os seguintes valores

path = r'C:\Users\Lucas\Desktop\UNB'
arcpy.env.workspace = path

# Definir variáveis para usar como números
i = 0

# Executando a ferramenta selecionar

for name in list_city:
    arcpy.SelectLayerByAttribute_management(in_layer_or_view="teste",
                                             selection_type="NEW_SELECTION",
                                             where_clause="FID" = ' + str(i))
    arcpy.Buffer_analysis('teste', 'teste_' + str(i) + '.shp', 10000 + 5000*i)
    i = i + 1
```



## **Código – Exemplo 2**

```
# coding=utf-8

# Importando a biblioteca

import arcpy

# A variável recebe os seguintes valores

path = r'C:\Users\Lucas\Desktop\UNB'
arcpy.env.workspace = path

# Definir lista com os nomes das cidades

list_city = ['Brasília', 'Goiânia', 'Luziânia', 'Anápolis', 'Formosa']

# Executando a ferramenta selecionar

for name in list_city:
    arcpy.SelectLayerByAttribute_management(in_layer_or_view="teste",
                                             selection_type="NEW_SELECTION",
                                             where_clause="name" + ' + \' + name + \'')
    arcpy.Buffer_analysis('teste', 'teste' + name + '.shp', 10000)
```