1. Programación en Python

Smart City

LAB CTIC UNI

Dr. Manuel Castillo-Cara
Intelligent Ubiquitous Technologies – Smart Cities (IUT-SCi)
Web: www.smartcityperu.org

Índice

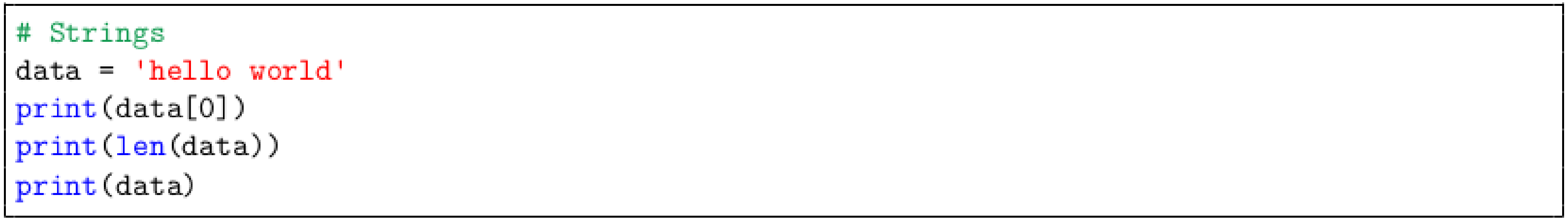
- Asignaciones
- Control de flujo
- Estructuras de datos
- NumPy
- Matplotlib
- Pandas

Smart City

LAB CTIC UNI

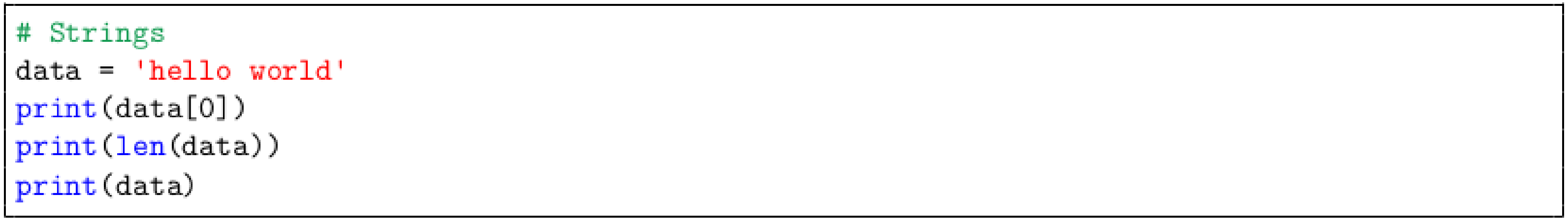
Asignaciones

1. Strings

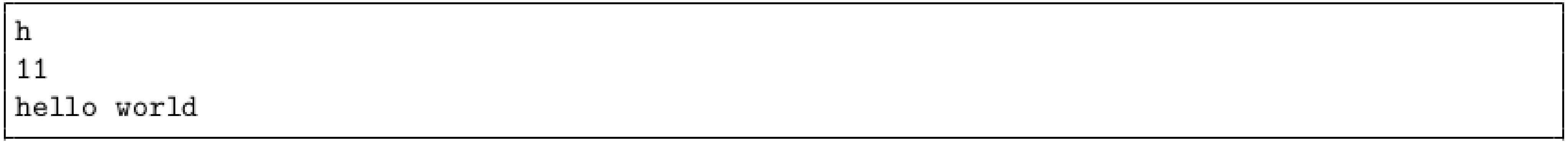


```python
# Strings
data = 'hello world'
print(data[0])
print(len(data))
print(data)
```

1. Strings

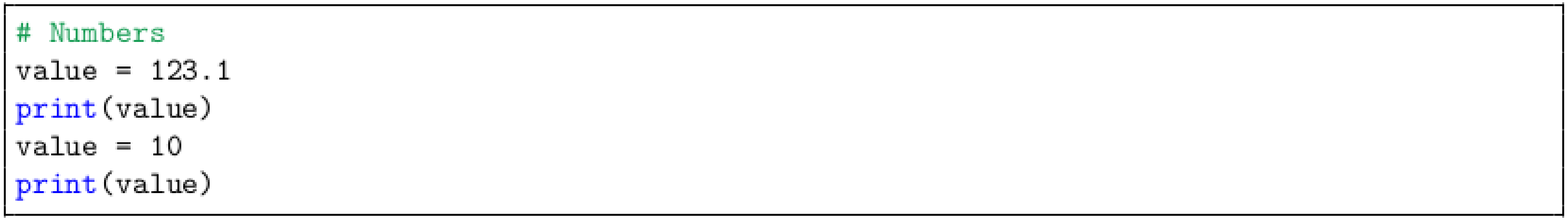


```python
# Strings
data = 'hello world'
print(data[0])
print(len(data))
print(data)
```



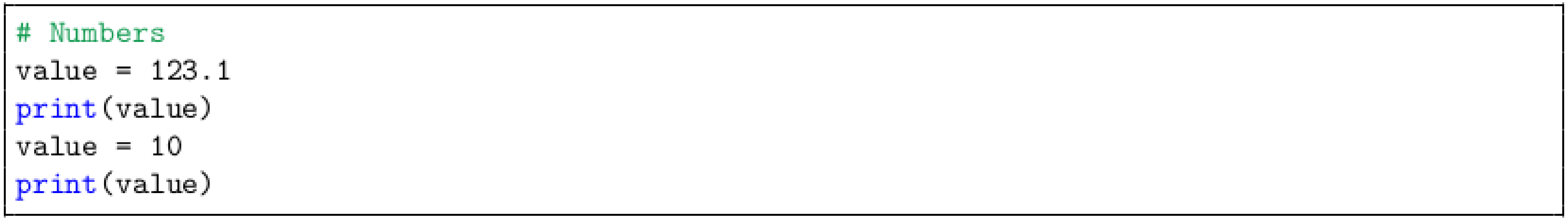
```
h
11
hello world
```

2. Numbers

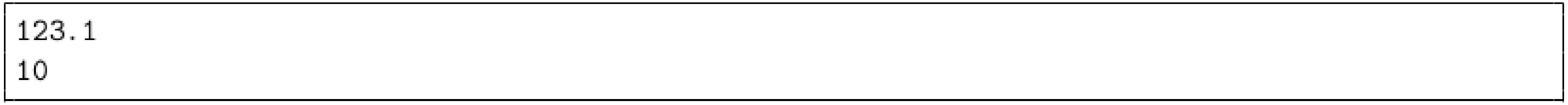


```
# Numbers
value = 123.1
print(value)
value = 10
print(value)
```

2. Numbers

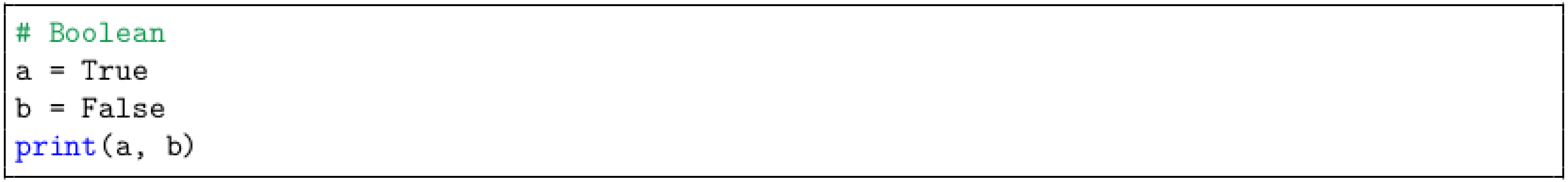


```
# Numbers
value = 123.1
print(value)
value = 10
print(value)
```



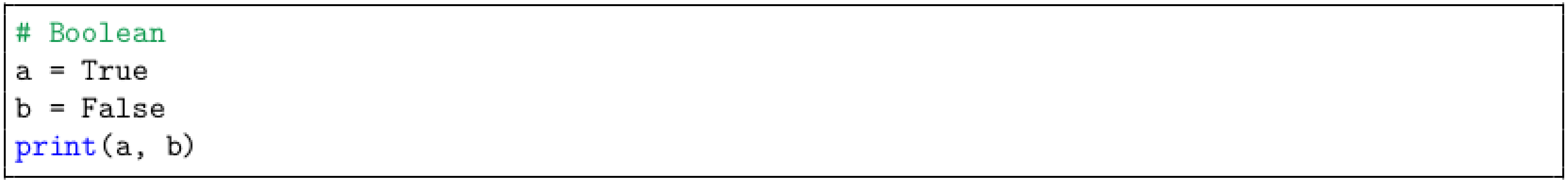
```
123.1
10
```

3. Boolean

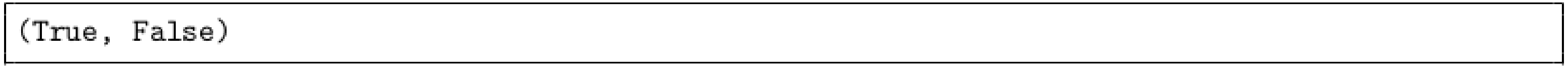


```python
# Boolean
a = True
b = False
print(a, b)
```

3. Boolean

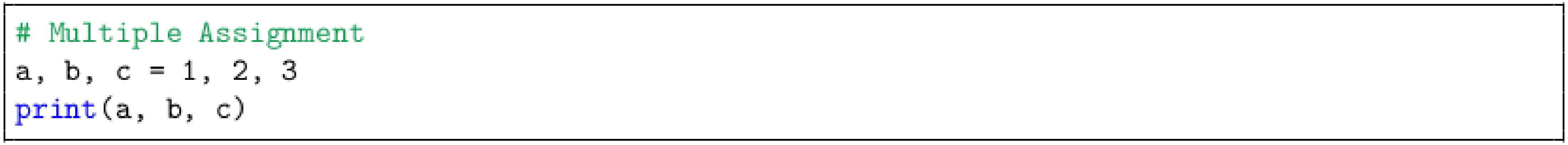


```
# Boolean
a = True
b = False
print(a, b)
```



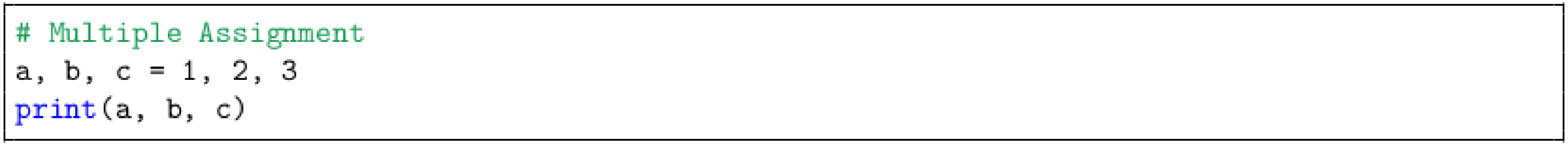
```
(True, False)
```

4. Asignación múltiple

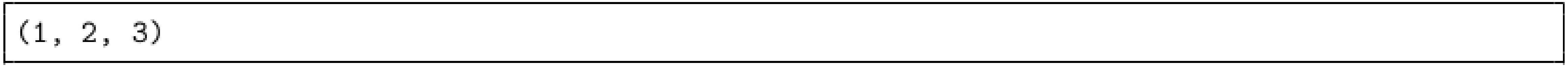


```
# Multiple Assignment
a, b, c = 1, 2, 3
print(a, b, c)
```

4. Asignación múltiple

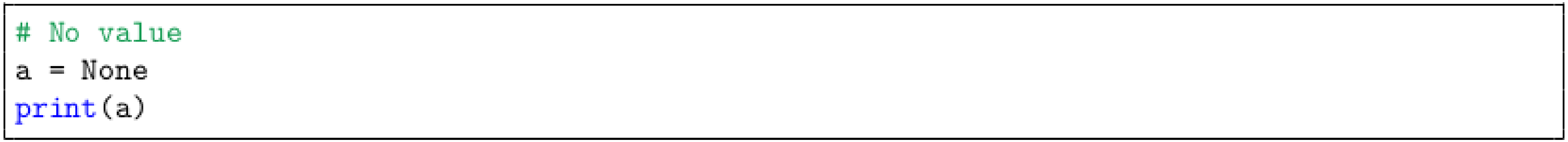


```python
# Multiple Assignment
a, b, c = 1, 2, 3
print(a, b, c)
```



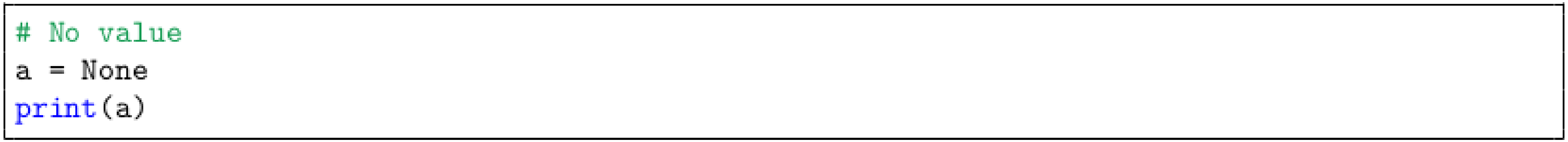
```
(1, 2, 3)
```

5. No value

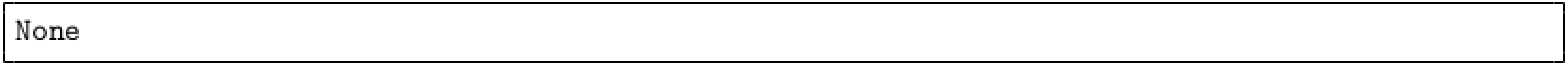


```python
# No value
a = None
print(a)
```

5. No value



```
# No value
a = None
print(a)
```



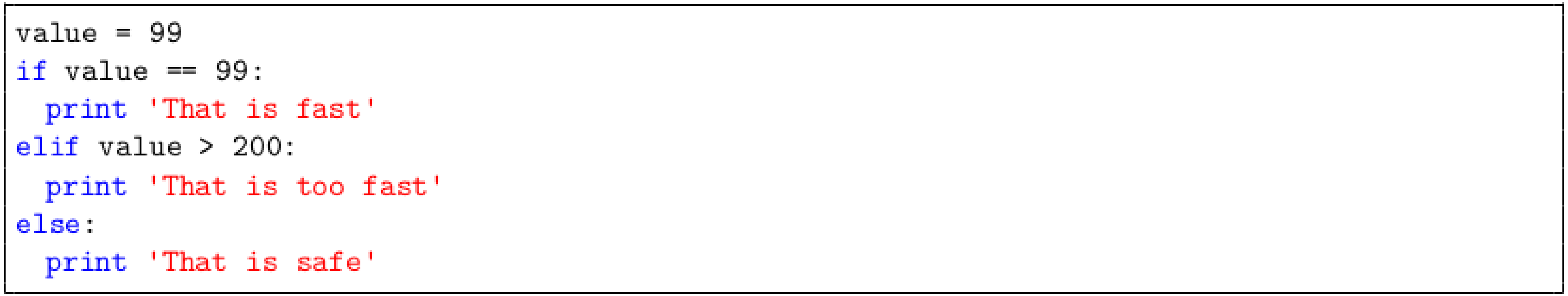
```
None
```

Smart City

LAB CTIC UNI

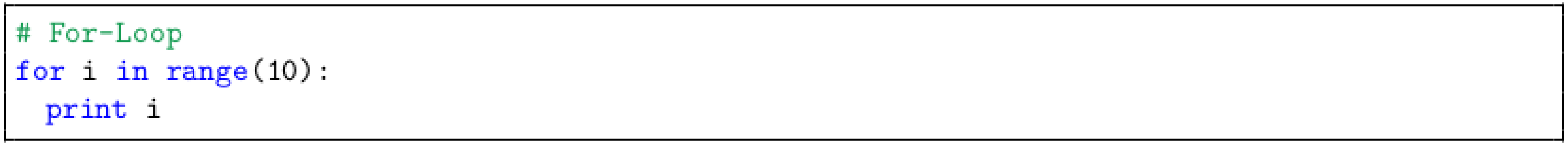
Control de flujo

1. If-Then-Else Conditional



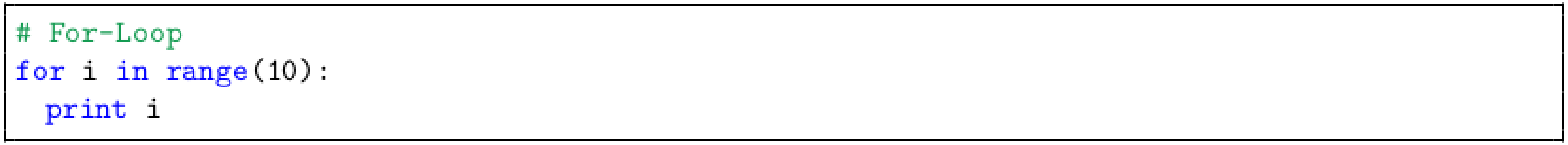
```
value = 99
if value == 99:
  print 'That is fast'
elif value > 200:
  print 'That is too fast'
else:
  print 'That is safe'
```

2. For-Loop

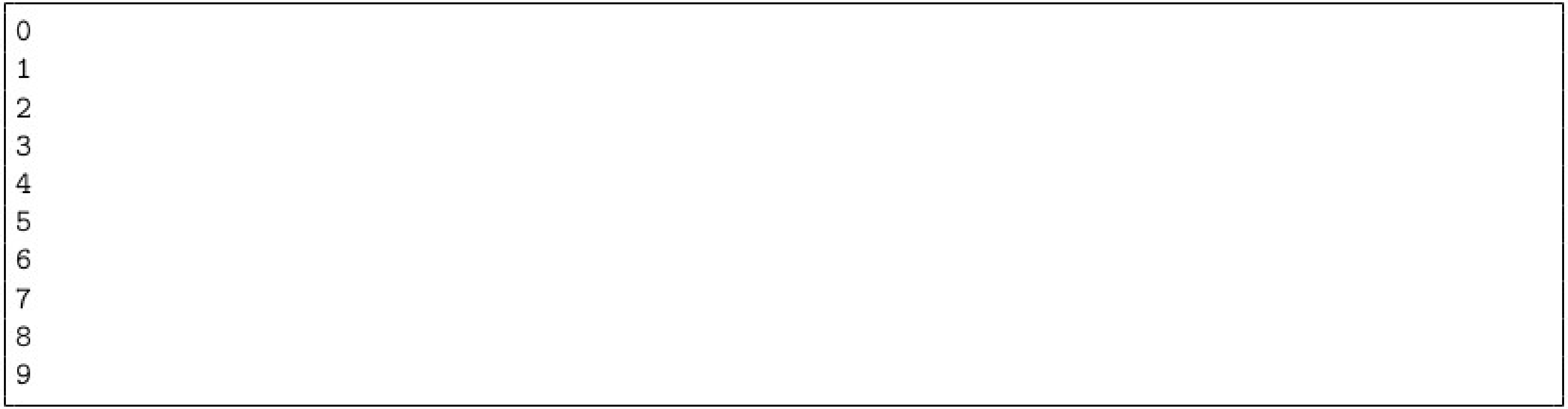


```
# For-Loop
for i in range(10):
  print i
```

2. For-Loop

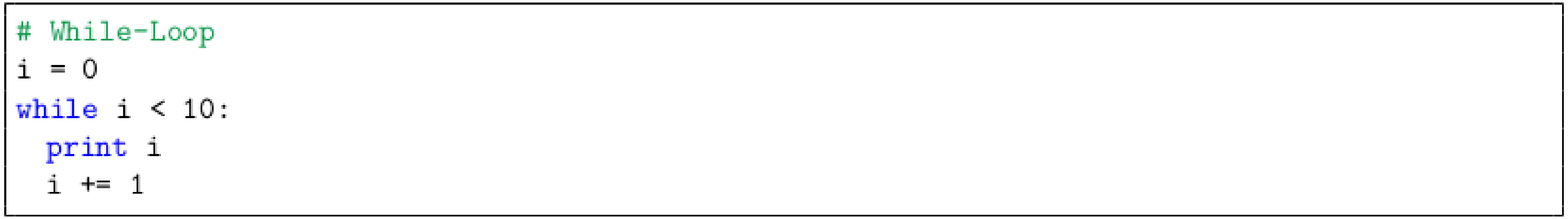


```
# For-Loop
for i in range(10):
  print i
```



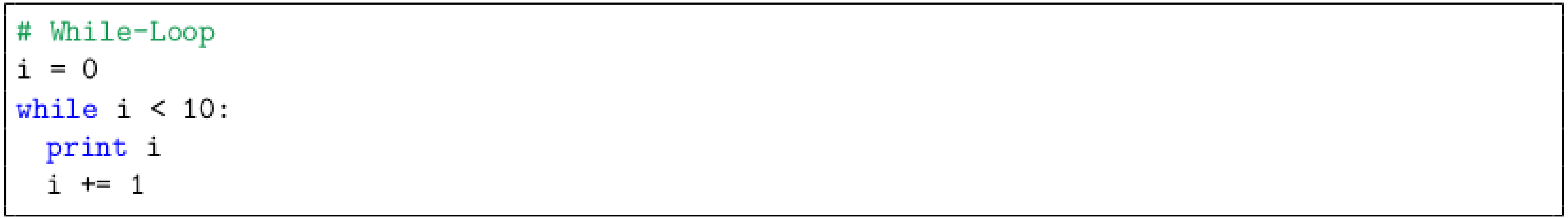
```
0
1
2
3
4
5
6
7
8
9
```

3. While-Loop

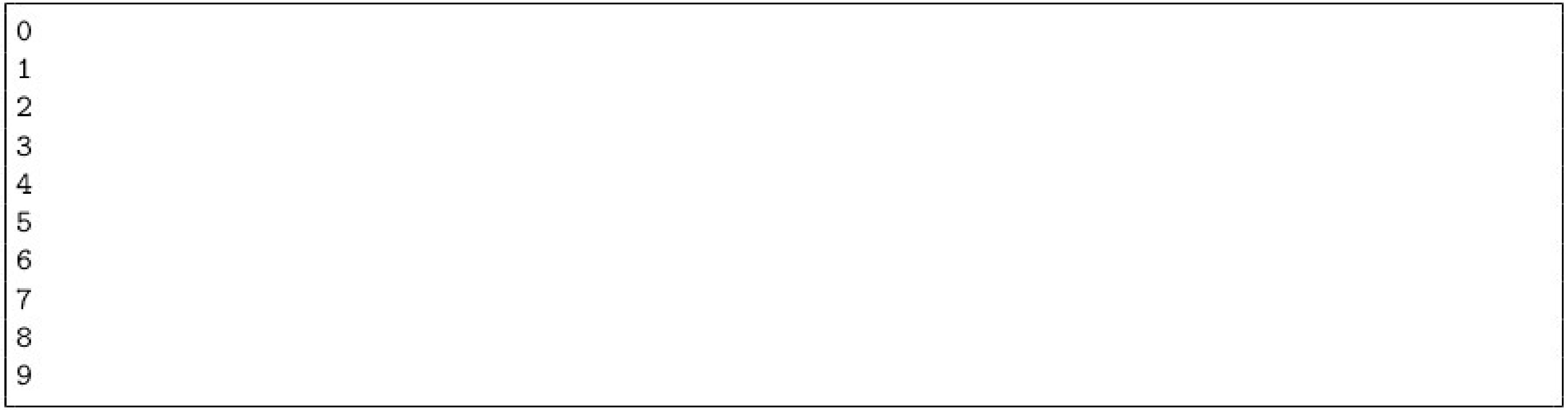


```
# While-Loop
i = 0
while i < 10:
  print i
  i += 1
```

3. While-Loop



```
# While-Loop
i = 0
while i < 10:
  print i
  i += 1
```



```
0
1
2
3
4
5
6
7
8
9
```

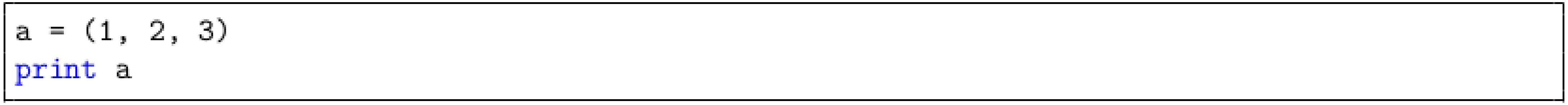
Smart City

LAB CTIC UNI

UNIVERSIDAD NACIONAL DE INGENIERIA
SCIENTIA ET LABOR
LIMA 1876 PERU

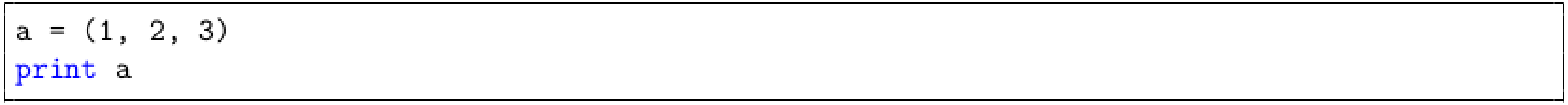
Estructuras de datos

1. Tuple

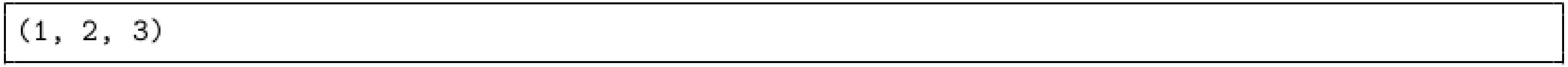


```
a = (1, 2, 3)
print a
```

1. Tuple

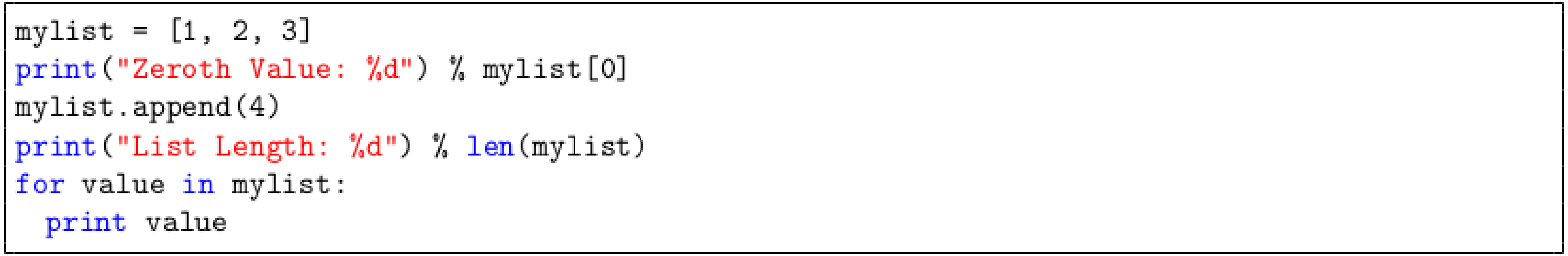


```
a = (1, 2, 3)
print a
```



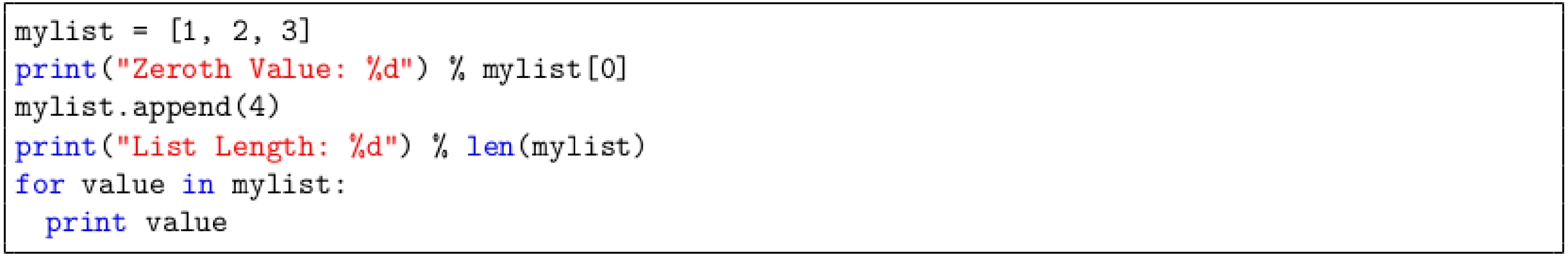
```
(1, 2, 3)
```

2. List

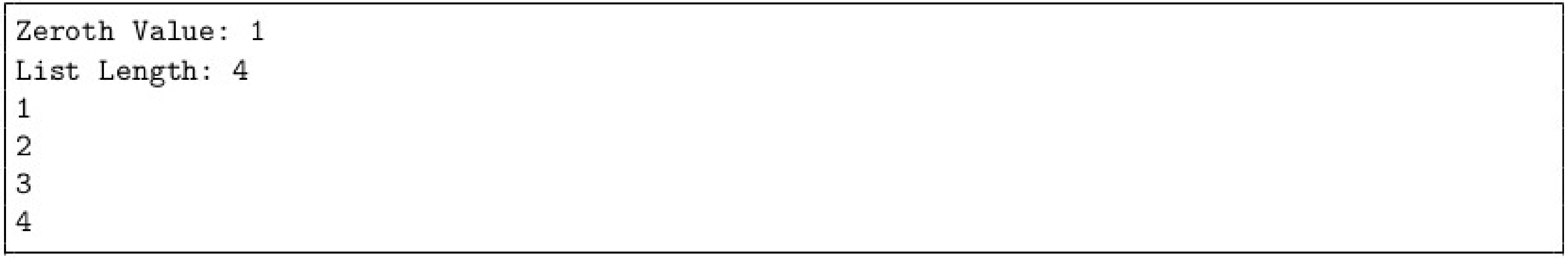


```
mylist = [1, 2, 3]
print("Zeroth Value: %d") % mylist[0]
mylist.append(4)
print("List Length: %d") % len(mylist)
for value in mylist:
  print value
```

2. List

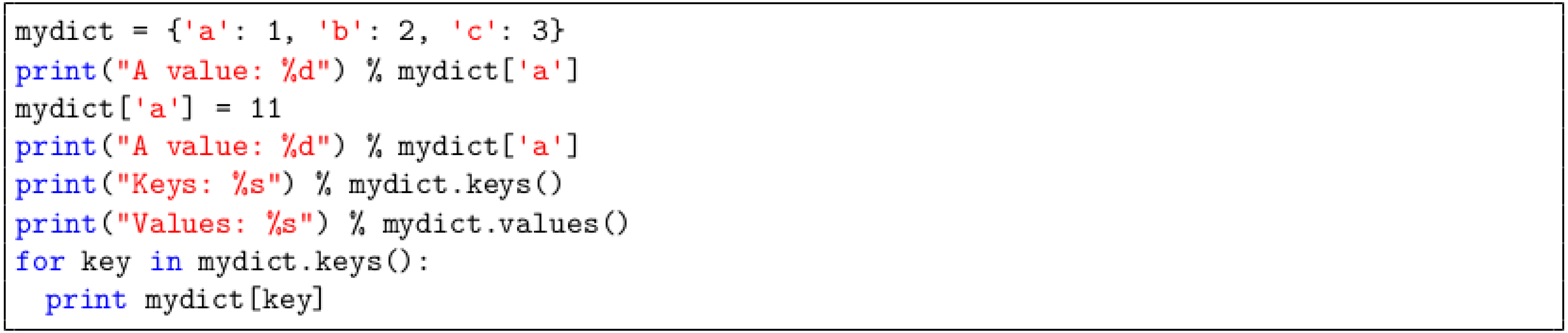


```
mylist = [1, 2, 3]
print("Zeroth Value: %d") % mylist[0]
mylist.append(4)
print("List Length: %d") % len(mylist)
for value in mylist:
  print value
```



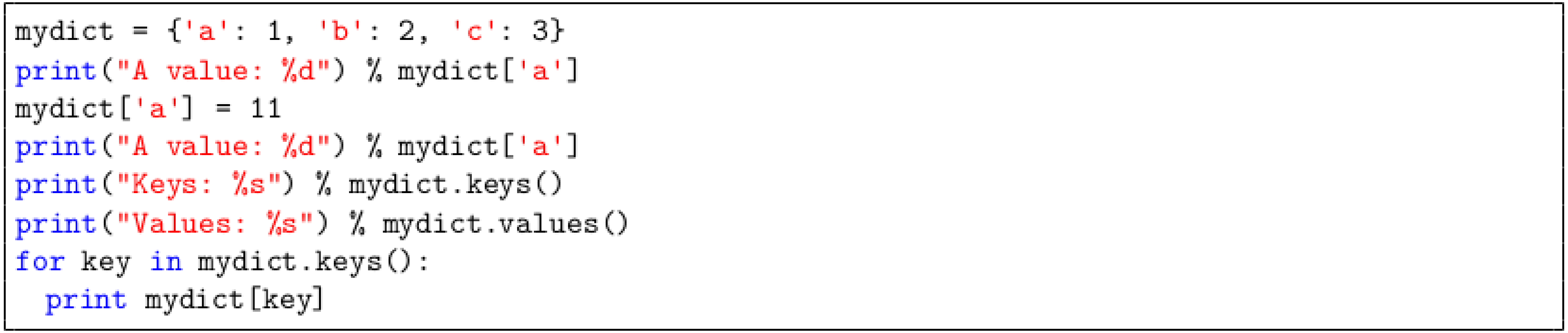
```
Zeroth Value: 1
List Length: 4
1
2
3
4
```

3. Diccionario

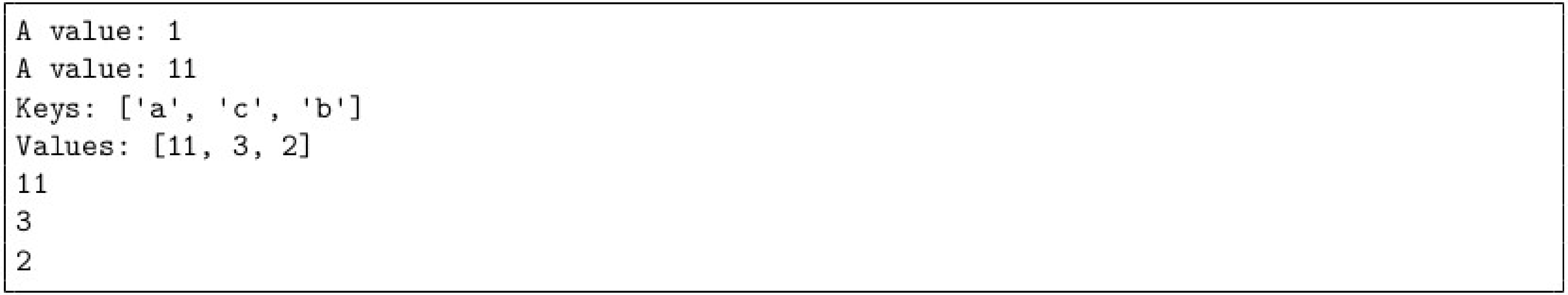


```
mydict = {'a': 1, 'b': 2, 'c': 3}
print("A value: %d") % mydict['a']
mydict['a'] = 11
print("A value: %d") % mydict['a']
print("Keys: %s") % mydict.keys()
print("Values: %s") % mydict.values()
for key in mydict.keys():
  print mydict[key]
```

3. Diccionario

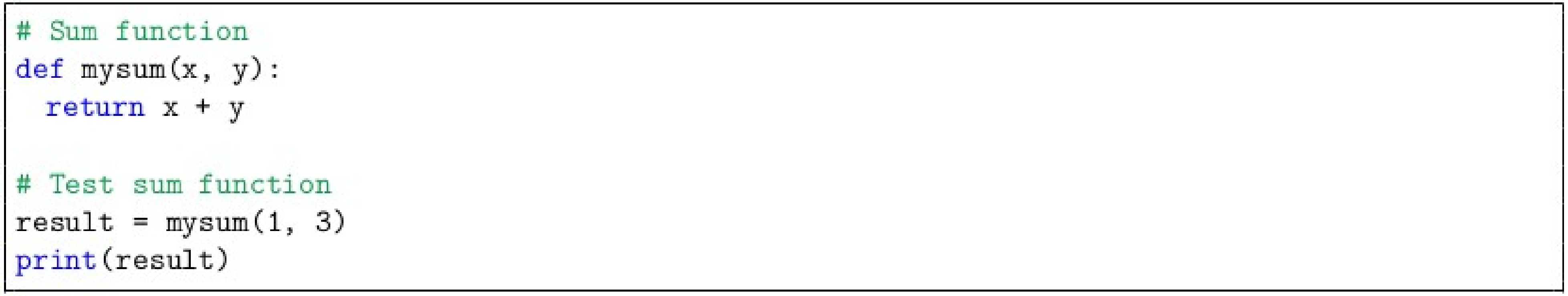


```
mydict = {'a': 1, 'b': 2, 'c': 3}
print("A value: %d") % mydict['a']
mydict['a'] = 11
print("A value: %d") % mydict['a']
print("Keys: %s") % mydict.keys()
print("Values: %s") % mydict.values()
for key in mydict.keys():
  print mydict[key]
```



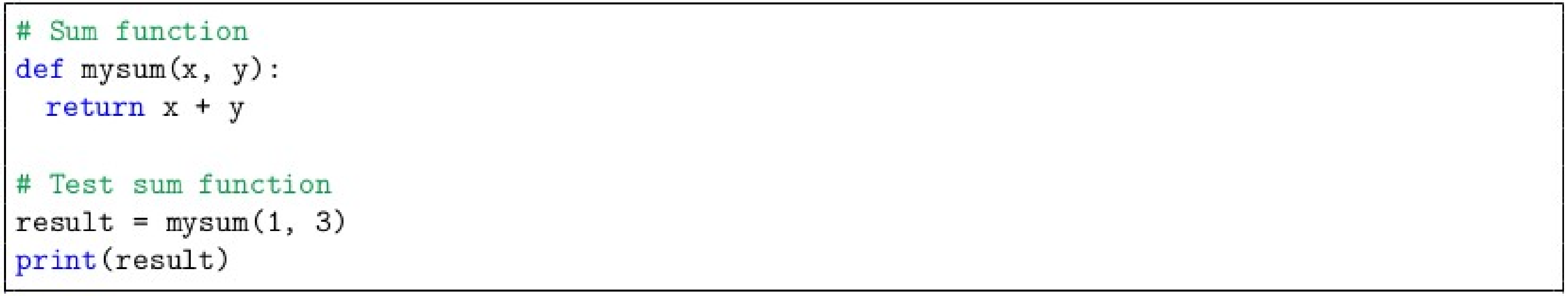
```
A value: 1
A value: 11
Keys: ['a', 'c', 'b']
Values: [11, 3, 2]
11
3
2
```

4. Functions

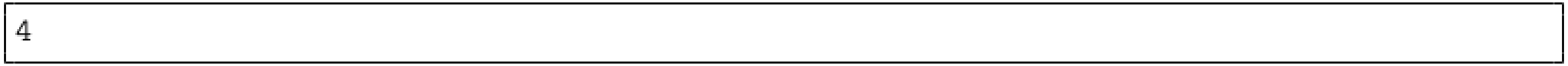


```
# Sum function
def mysum(x, y):
  return x + y

# Test sum function
result = mysum(1, 3)
print(result)
```

4. Functions



```
# Sum function
def mysum(x, y):
  return x + y

# Test sum function
result = mysum(1, 3)
print(result)
```



```
4
```

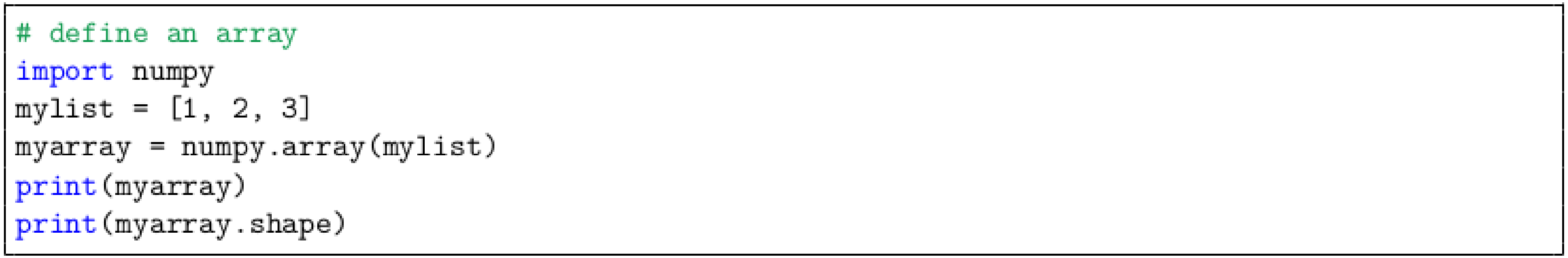
Smart City

LAB CTIC UNI

UNIVERSIDAD NACIONAL DE INGENIERIA
SCIENTIA ET LABOR
LIMA 1876 PERU

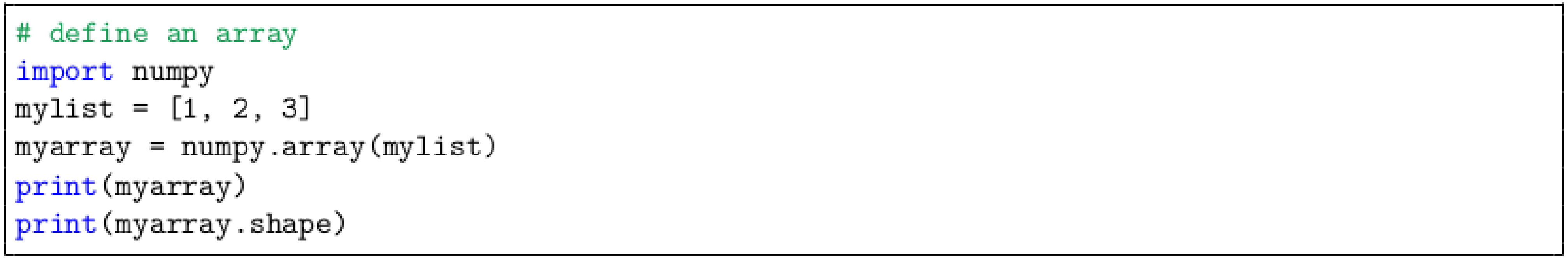
NumPy

1. Create Array

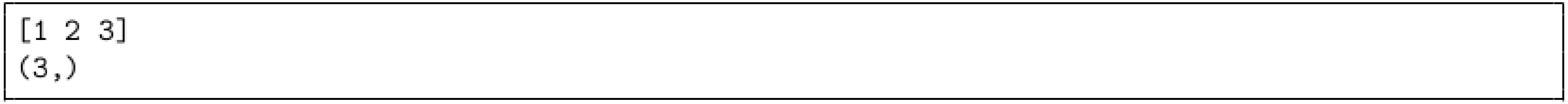


```python
# define an array
import numpy
mylist = [1, 2, 3]
myarray = numpy.array(mylist)
print(myarray)
print(myarray.shape)
```

1. Create Array

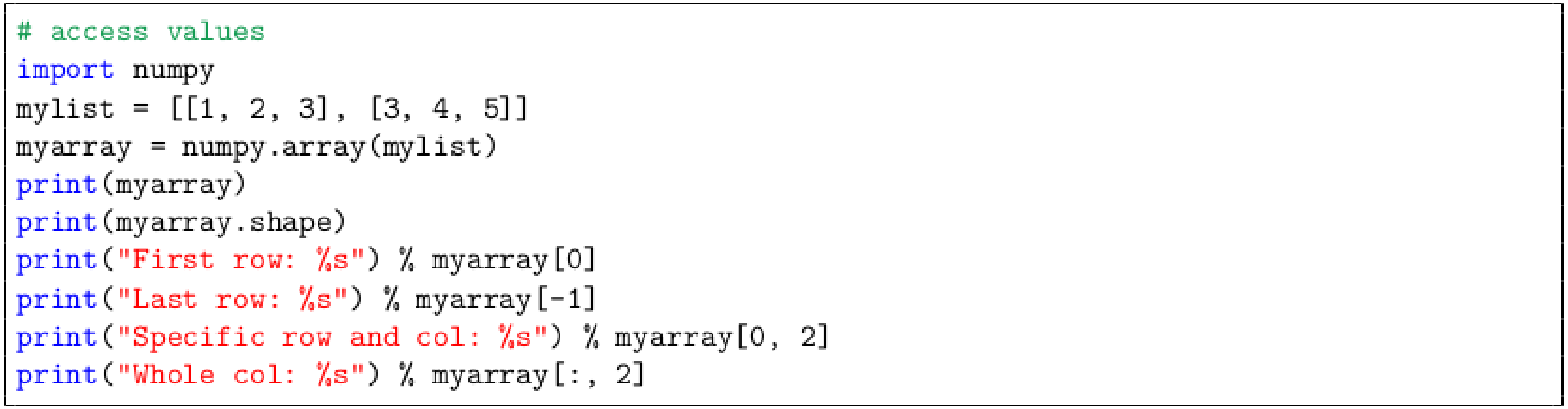


```
# define an array
import numpy
mylist = [1, 2, 3]
myarray = numpy.array(mylist)
print(myarray)
print(myarray.shape)
```



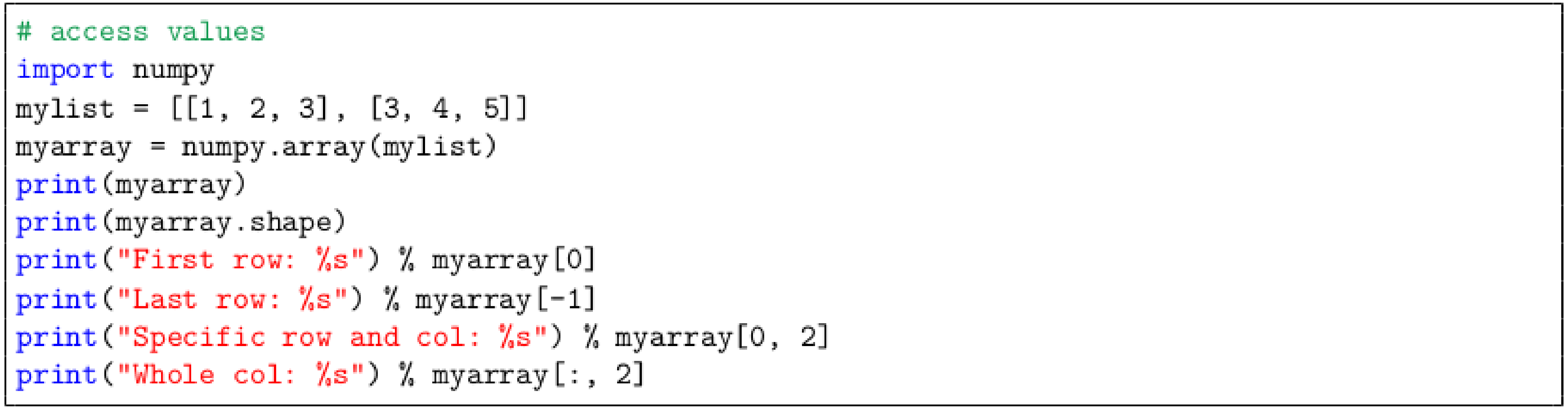
```
[1 2 3]
(3,)
```

2. Access Data

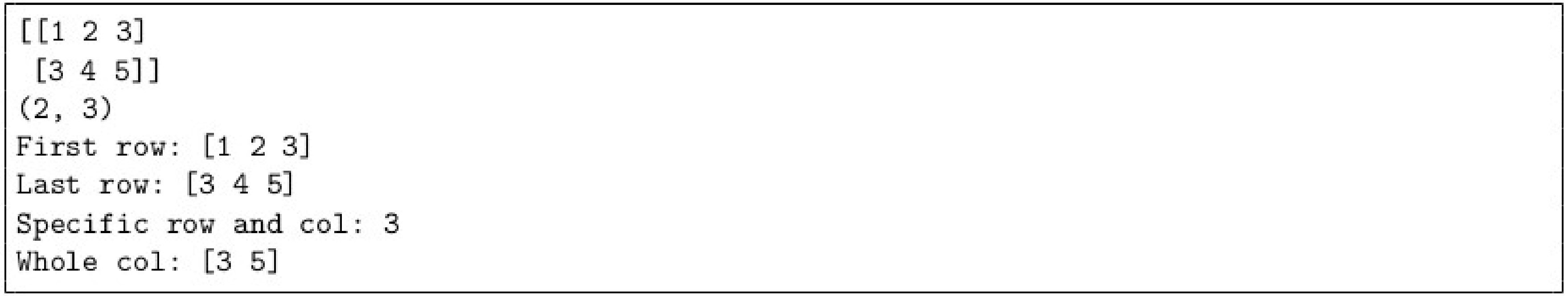


```
# access values
import numpy
mylist = [[1, 2, 3], [3, 4, 5]]
myarray = numpy.array(mylist)
print(myarray)
print(myarray.shape)
print("First row: %s") % myarray[0]
print("Last row: %s") % myarray[-1]
print("Specific row and col: %s") % myarray[0, 2]
print("Whole col: %s") % myarray[:, 2]
```

2. Access Data

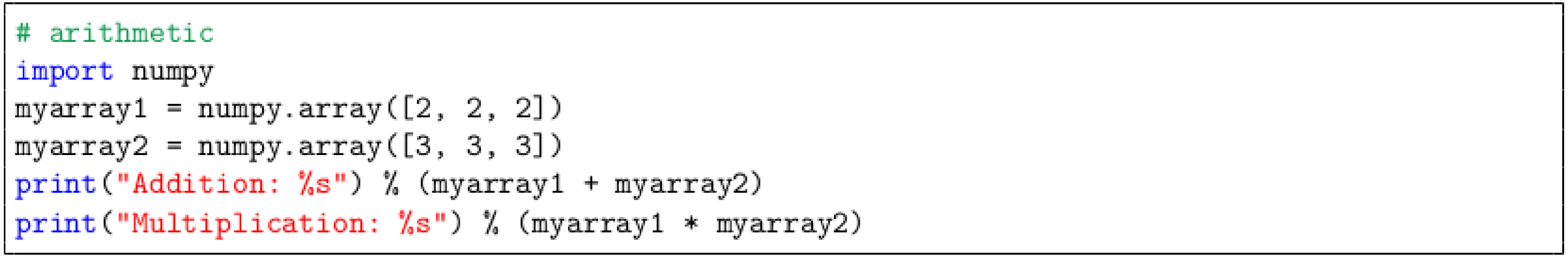


```
# access values
import numpy
mylist = [[1, 2, 3], [3, 4, 5]]
myarray = numpy.array(mylist)
print(myarray)
print(myarray.shape)
print("First row: %s") % myarray[0]
print("Last row: %s") % myarray[-1]
print("Specific row and col: %s") % myarray[0, 2]
print("Whole col: %s") % myarray[:, 2]
```



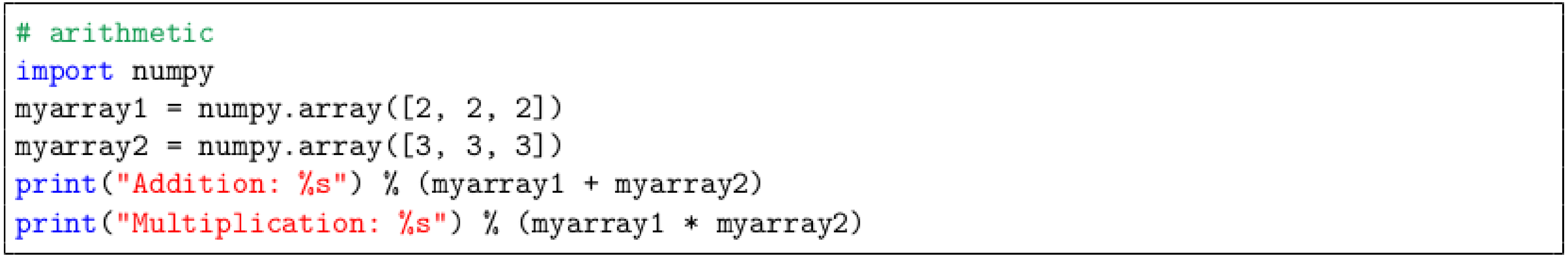
```
[[1 2 3]
 [3 4 5]]
(2, 3)
First row: [1 2 3]
Last row: [3 4 5]
Specific row and col: 3
Whole col: [3 5]
```

3. Arithmetic

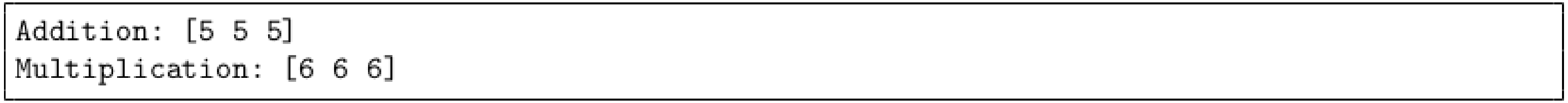


```
# arithmetic
import numpy
myarray1 = numpy.array([2, 2, 2])
myarray2 = numpy.array([3, 3, 3])
print("Addition: %s") % (myarray1 + myarray2)
print("Multiplication: %s") % (myarray1 * myarray2)
```

3. Arithmetic



```
# arithmetic
import numpy
myarray1 = numpy.array([2, 2, 2])
myarray2 = numpy.array([3, 3, 3])
print("Addition: %s") % (myarray1 + myarray2)
print("Multiplication: %s") % (myarray1 * myarray2)
```



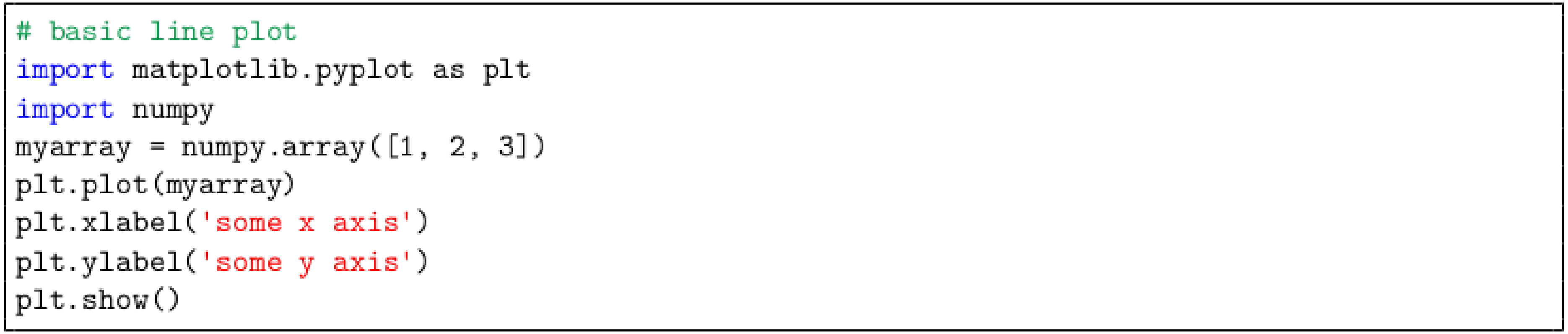
```
Addition: [5 5 5]
Multiplication: [6 6 6]
```

Smart City

LAB CTIC UNI

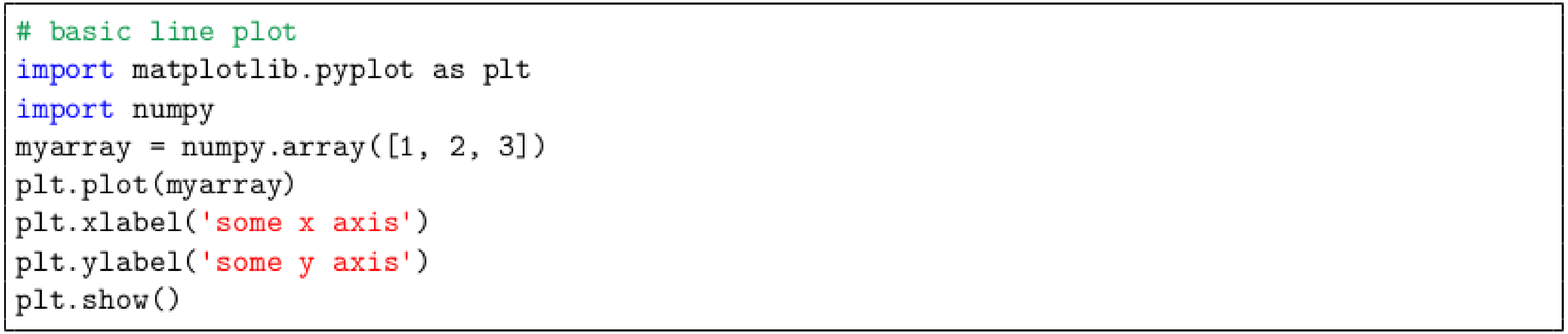
Matplotlib

1. Line Plot

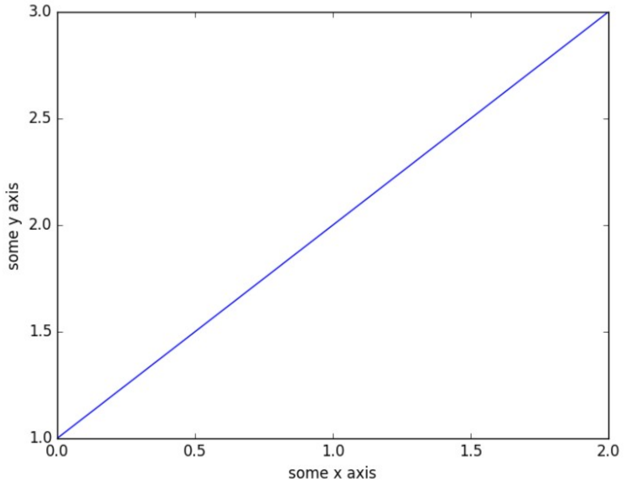


```
# basic line plot
import matplotlib.pyplot as plt
import numpy
myarray = numpy.array([1, 2, 3])
plt.plot(myarray)
plt.xlabel('some x axis')
plt.ylabel('some y axis')
plt.show()
```

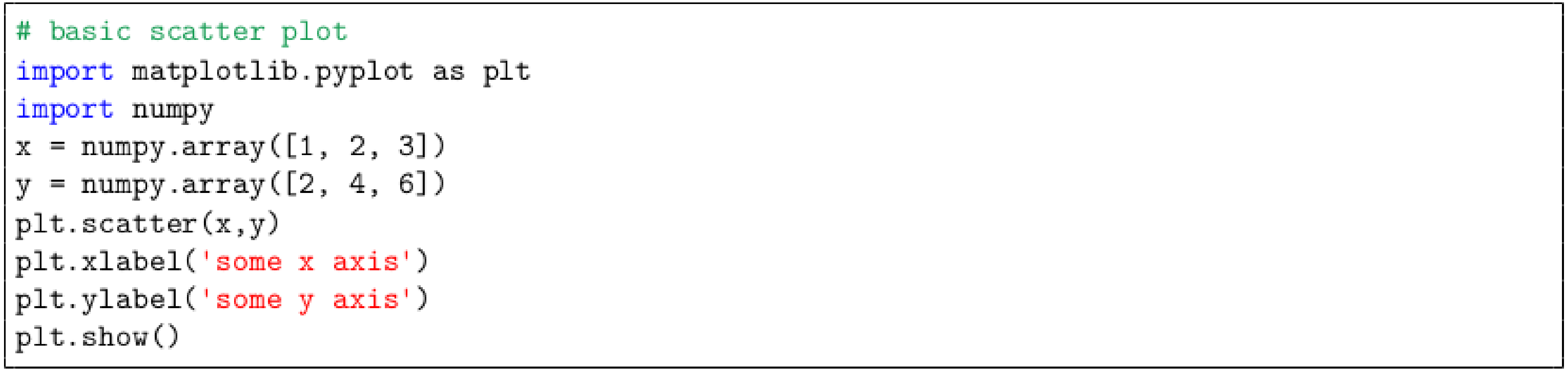
1. Line Plot



```
# basic line plot
import matplotlib.pyplot as plt
import numpy
myarray = numpy.array([1, 2, 3])
plt.plot(myarray)
plt.xlabel('some x axis')
plt.ylabel('some y axis')
plt.show()
```

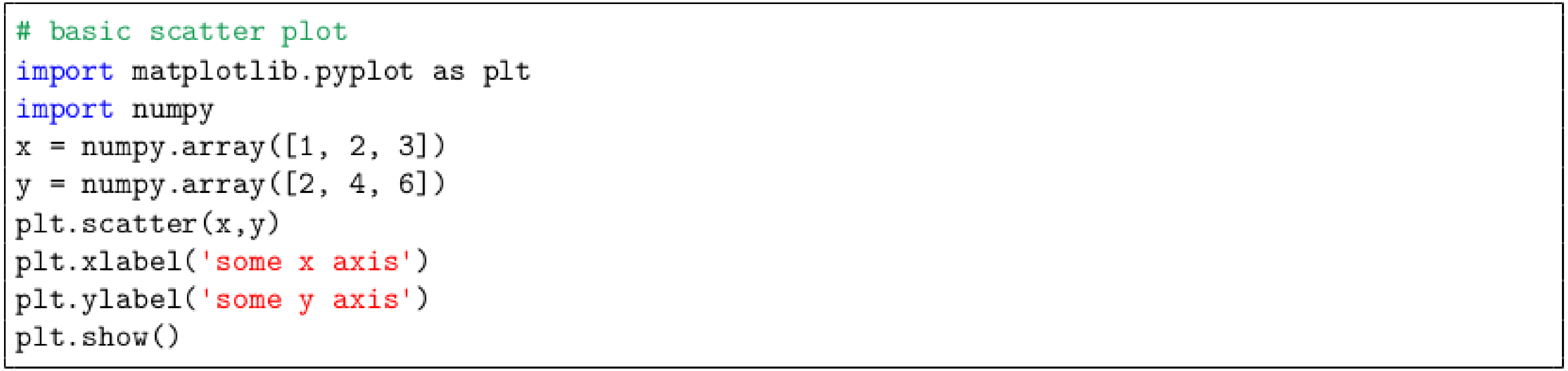


2. Scatter Plot

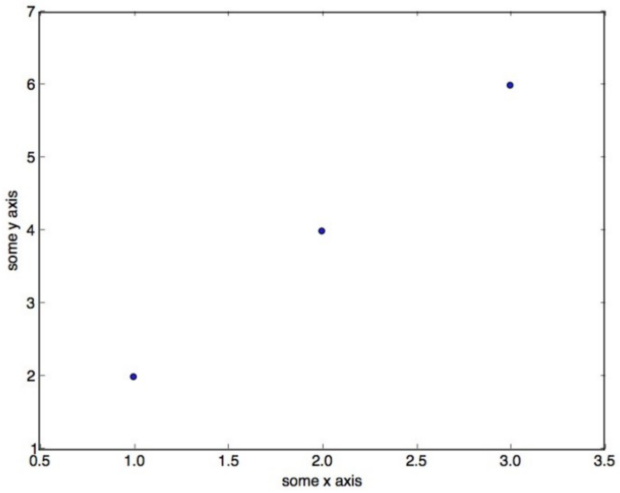


```
# basic scatter plot
import matplotlib.pyplot as plt
import numpy
x = numpy.array([1, 2, 3])
y = numpy.array([2, 4, 6])
plt.scatter(x,y)
plt.xlabel('some x axis')
plt.ylabel('some y axis')
plt.show()
```

2. Scatter Plot



```
# basic scatter plot
import matplotlib.pyplot as plt
import numpy
x = numpy.array([1, 2, 3])
y = numpy.array([2, 4, 6])
plt.scatter(x,y)
plt.xlabel('some x axis')
plt.ylabel('some y axis')
plt.show()
```



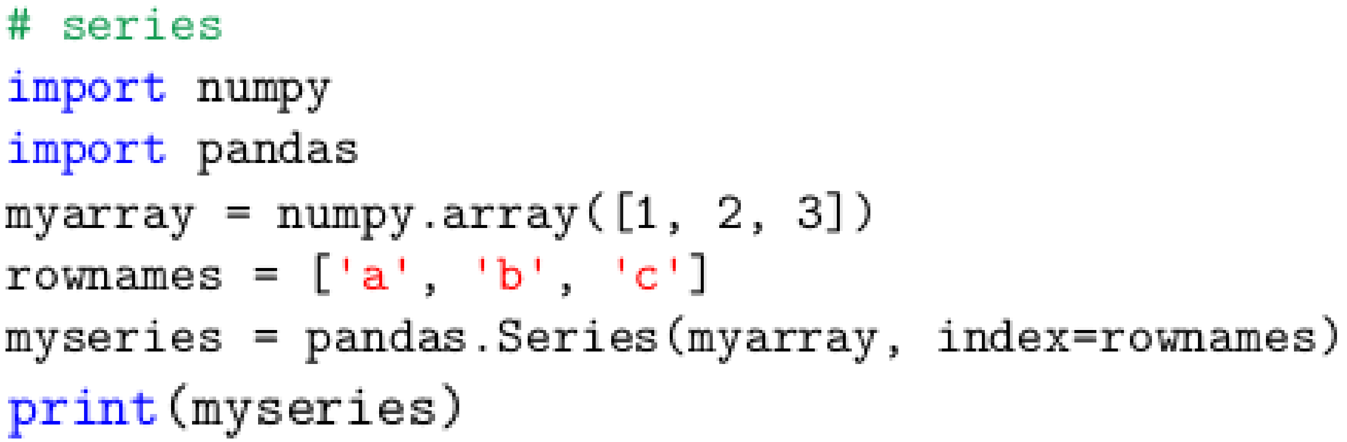
Smart City

LAB CTIC UNI

UNIVERSIDAD NACIONAL DE INGENIERIA
SCIENTIA ET LABOR
LIMA 1876 PERU

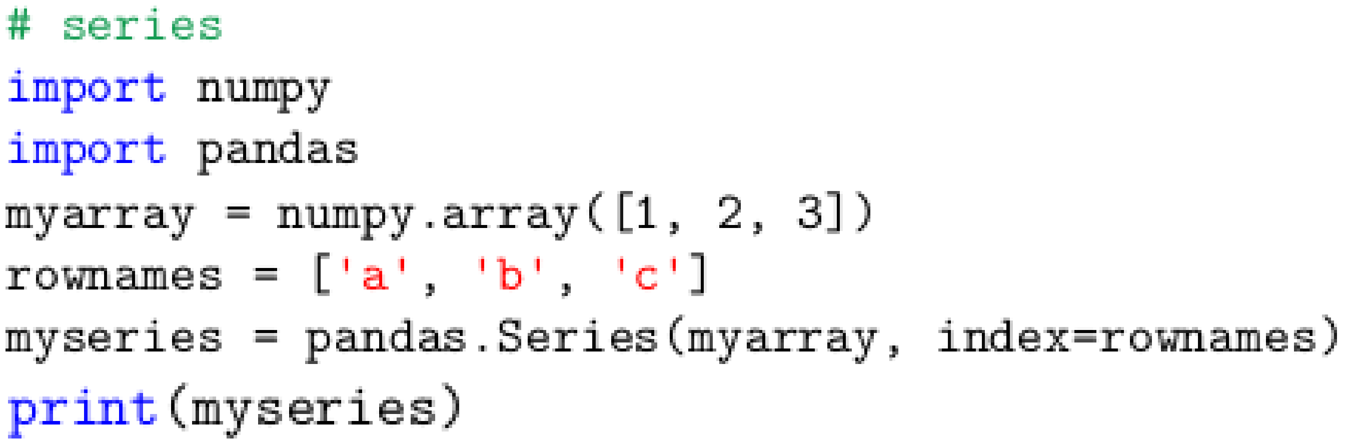
Pandas

1. Series

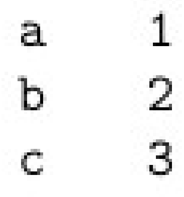


```
# series
import numpy
import pandas
myarray = numpy.array([1, 2, 3])
rownames = ['a', 'b', 'c']
myseries = pandas.Series(myarray, index=rownames)
print(myseries)
```

1. Series

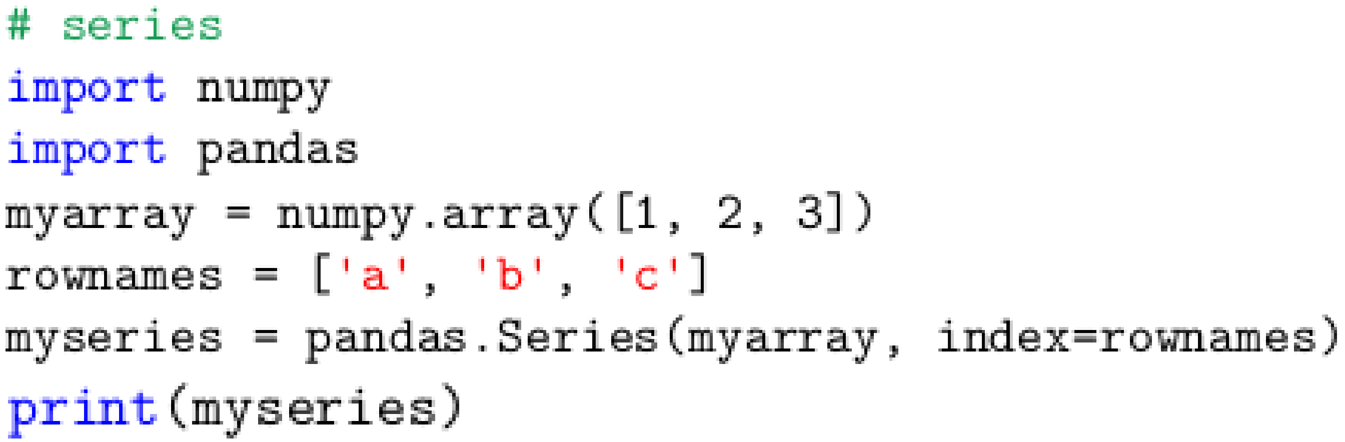


```python
# series
import numpy
import pandas
myarray = numpy.array([1, 2, 3])
rownames = ['a', 'b', 'c']
myseries = pandas.Series(myarray, index=rownames)
print(myseries)
```

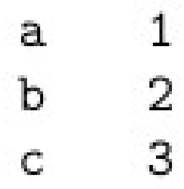


```
a    1
b    2
c    3
```

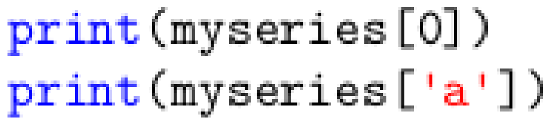
1. Series



```python
# series
import numpy
import pandas
myarray = numpy.array([1, 2, 3])
rownames = ['a', 'b', 'c']
myseries = pandas.Series(myarray, index=rownames)
print(myseries)
```

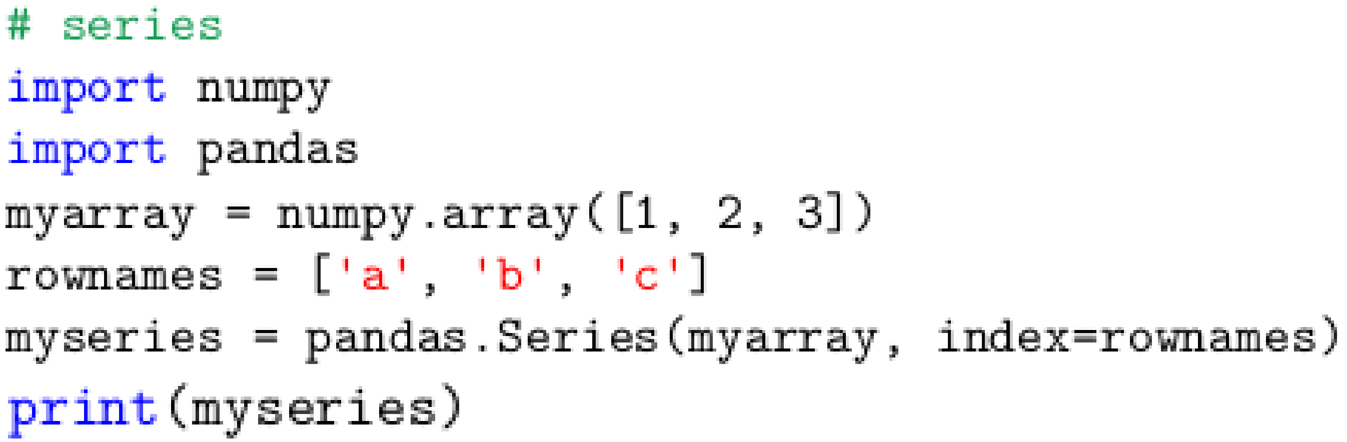


```
a    1
b    2
c    3
```

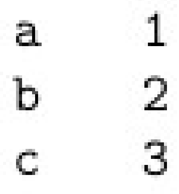


```python
print(myseries[0])
print(myseries['a'])
```

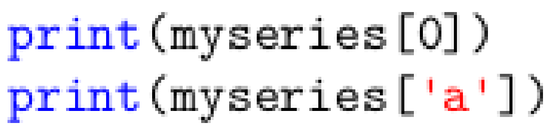
1. Series



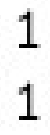
```python
# series
import numpy
import pandas
myarray = numpy.array([1, 2, 3])
rownames = ['a', 'b', 'c']
myseries = pandas.Series(myarray, index=rownames)
print(myseries)
```



```
a    1
b    2
c    3
```

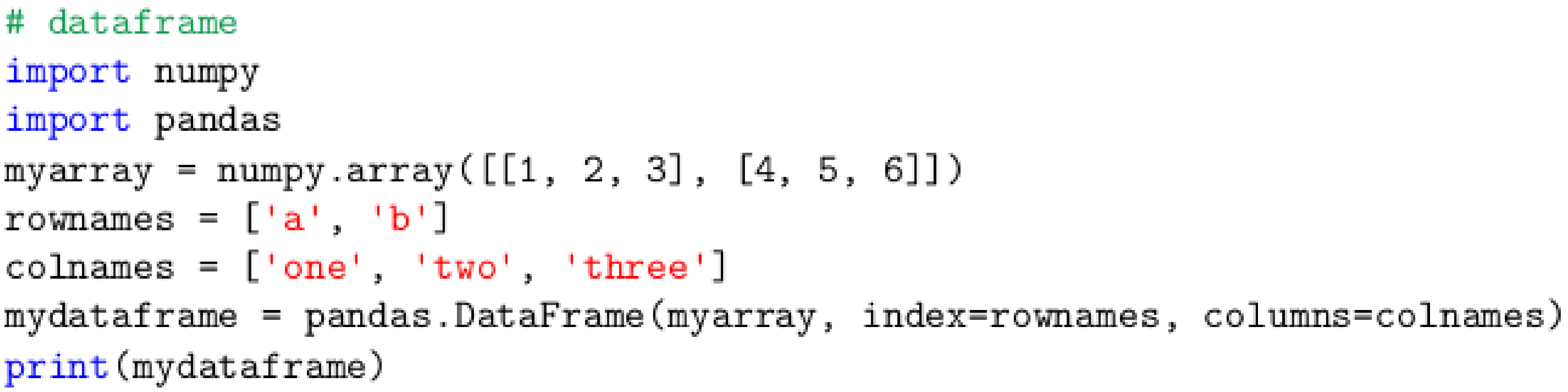


```python
print(myseries[0])
print(myseries['a'])
```



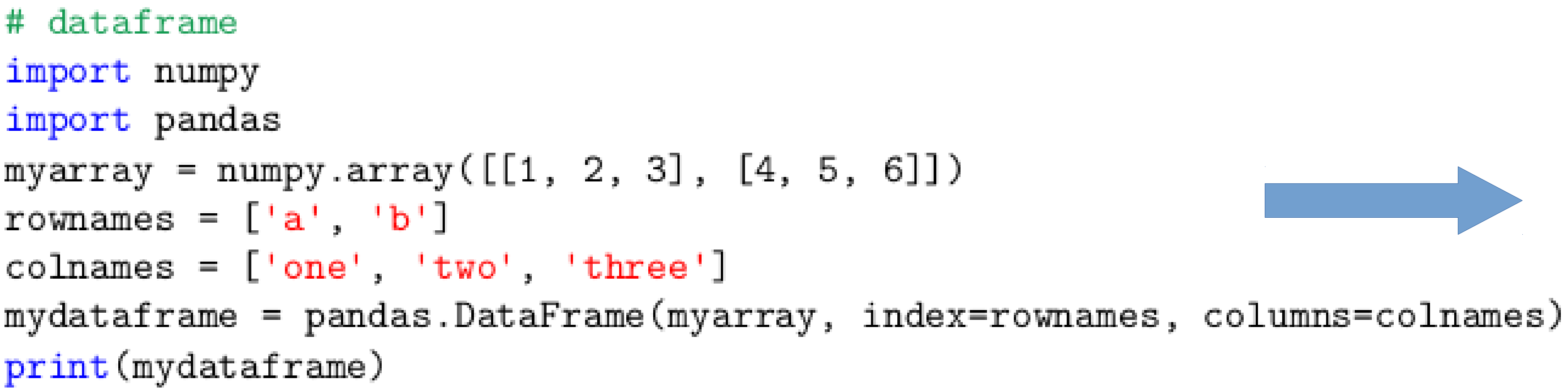
```
1
1
```

2. DataFrame

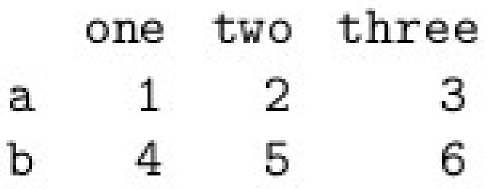


```
# dataframe
import numpy
import pandas
myarray = numpy.array([[1, 2, 3], [4, 5, 6]])
rownames = ['a', 'b']
colnames = ['one', 'two', 'three']
mydataframe = pandas.DataFrame(myarray, index=rownames, columns=colnames)
print(mydataframe)
```

2. DataFrame

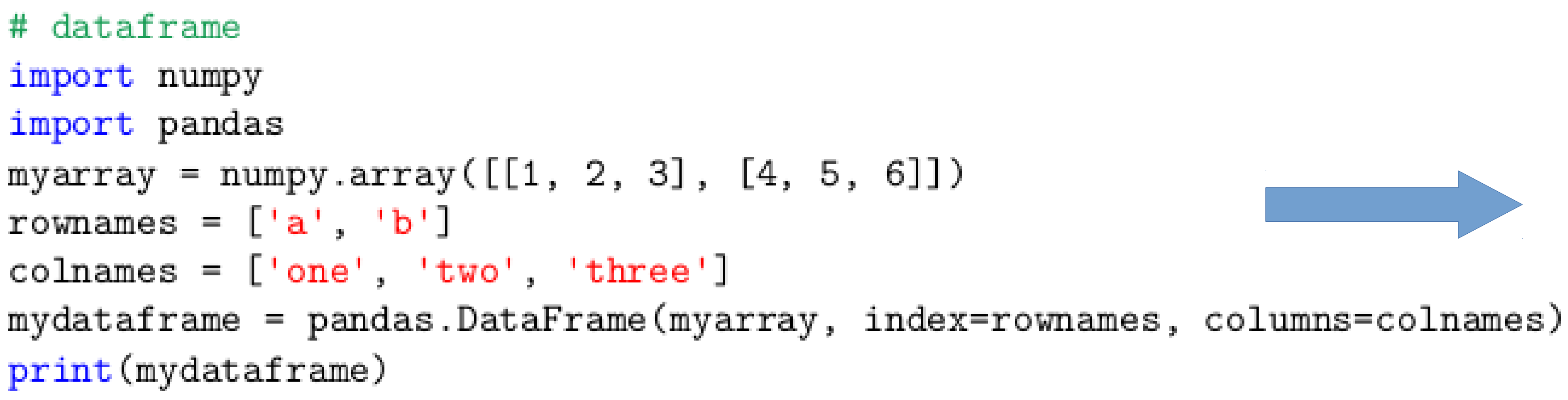


```
# dataframe
import numpy
import pandas
myarray = numpy.array([[1, 2, 3], [4, 5, 6]])
rownames = ['a', 'b']
colnames = ['one', 'two', 'three']
mydataframe = pandas.DataFrame(myarray, index=rownames, columns=colnames)
print(mydataframe)
```

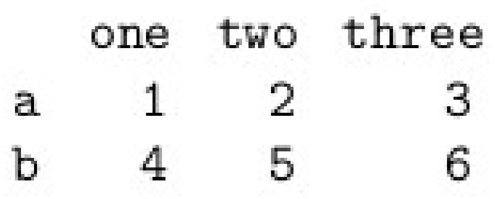


```
   one  two  three
a    1    2      3
b    4    5      6
```

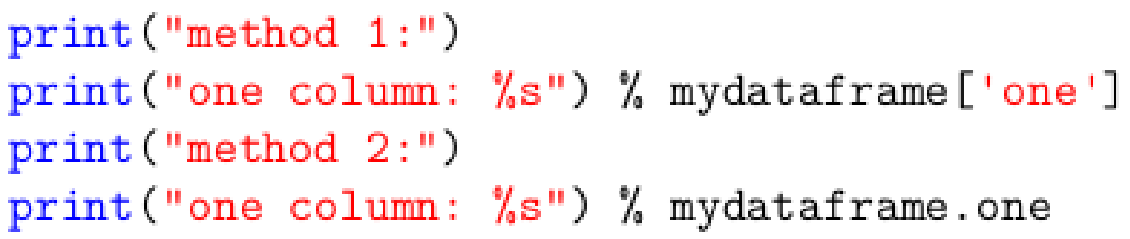
2. DataFrame



```
# dataframe
import numpy
import pandas
myarray = numpy.array([[1, 2, 3], [4, 5, 6]])
rownames = ['a', 'b']
colnames = ['one', 'two', 'three']
mydataframe = pandas.DataFrame(myarray, index=rownames, columns=colnames)
print(mydataframe)
```

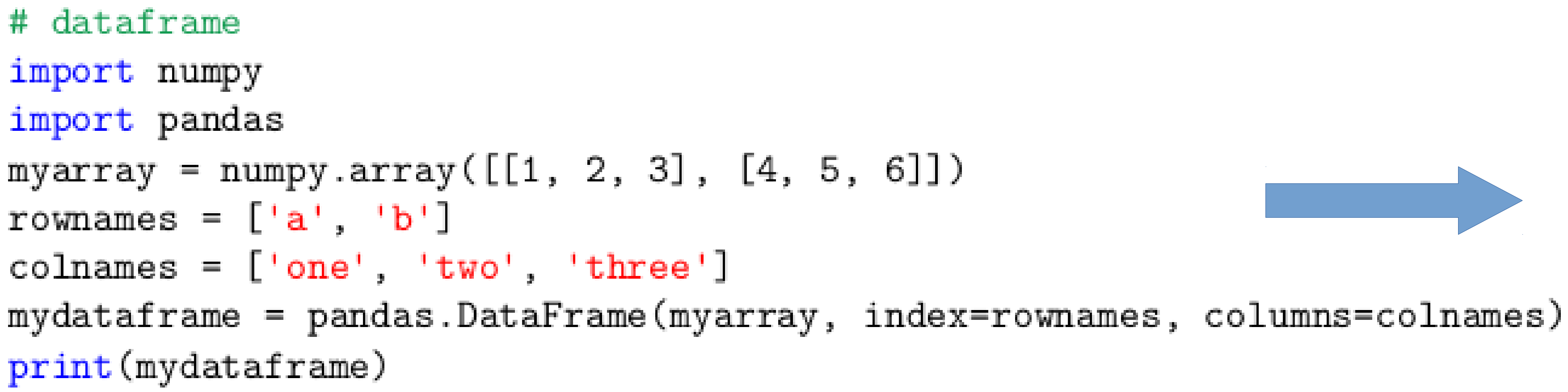


```
   one  two  three
a    1    2      3
b    4    5      6
```

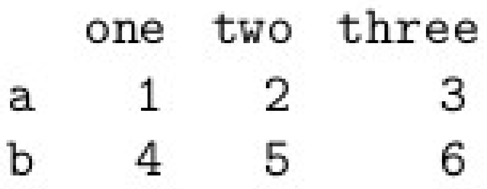


```
print("method 1:")
print("one column: %s") % mydataframe['one']
print("method 2:")
print("one column: %s") % mydataframe.one
```

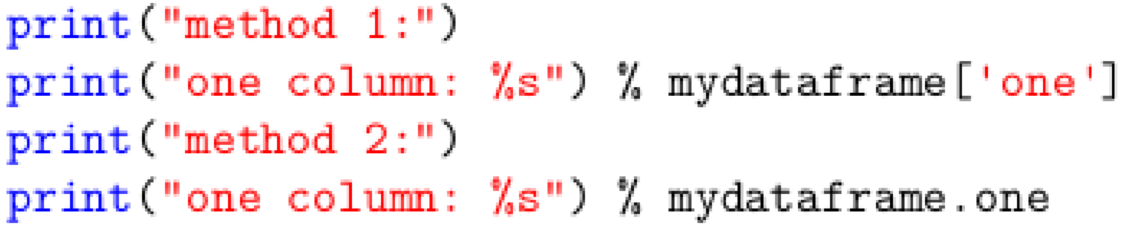
2. DataFrame



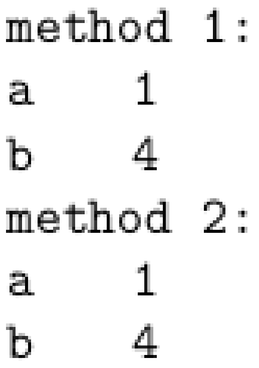
```
# dataframe
import numpy
import pandas
myarray = numpy.array([[1, 2, 3], [4, 5, 6]])
rownames = ['a', 'b']
colnames = ['one', 'two', 'three']
mydataframe = pandas.DataFrame(myarray, index=rownames, columns=colnames)
print(mydataframe)
```



```
   one  two  three
a    1    2      3
b    4    5      6
```



```
print("method 1:")
print("one column: %s") % mydataframe['one']
print("method 2:")
print("one column: %s") % mydataframe.one
```



```
method 1:
a    1
b    4
method 2:
a    1
b    4
```

¡GRACIAS!

Smart City

LAB CTIC UNI

Dr. Manuel Castillo-Cara
Intelligent Ubiquitous Technologies – Smart Cities (IUT-SCi)
Web: www.smartcityperu.org