UNIVERSITY OF CALGARY

CPSC203 – Introduction to Problem Solving and Using Application Software

Fall 2009

Tutorial 25, Mehrdad Nurolahzade

Introduction

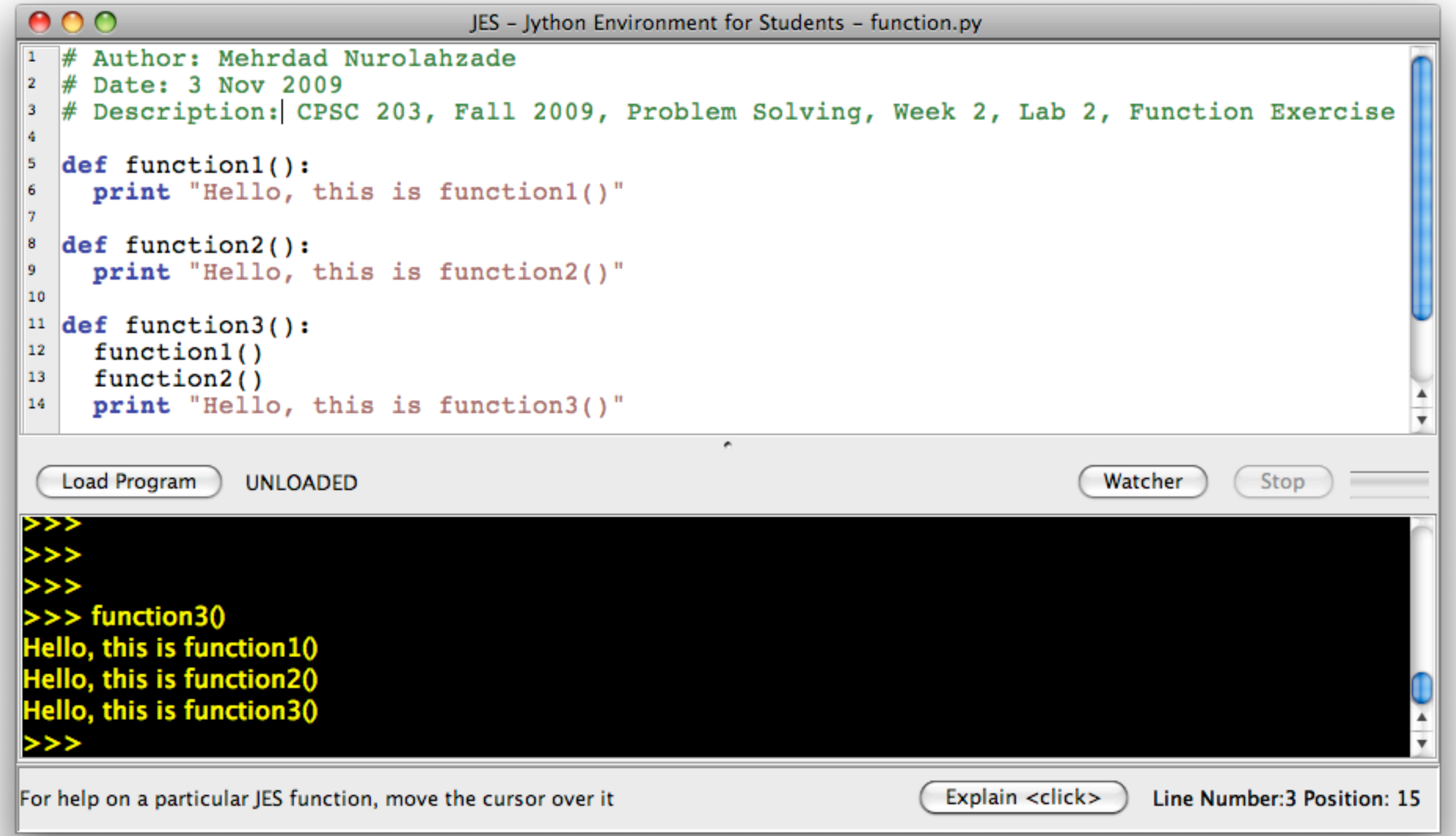
- Functions
- Graphics

Multiple Functions (1)

- Jython programs can have more than a function.
- A function can call (=invoke) other functions.

- When function A calls function B, all instructions in function B get executed first. Then, once function B returns (=exits) function A continues execution.

Multiple Functions (2)



```
# Author: Mehrdad Nurolahzade
# Date: 3 Nov 2009
# Description: CPSC 203, Fall 2009, Problem Solving, Week 2, Lab 2, Function Exercise

def function1():
  print "Hello, this is function1()"

def function2():
  print "Hello, this is function2()"

def function3():
  function1()
  function2()
  print "Hello, this is function3()"
```

```
>>>
>>>
>>>
>>> function3()
Hello, this is function1()
Hello, this is function2()
Hello, this is function3()
>>>
```

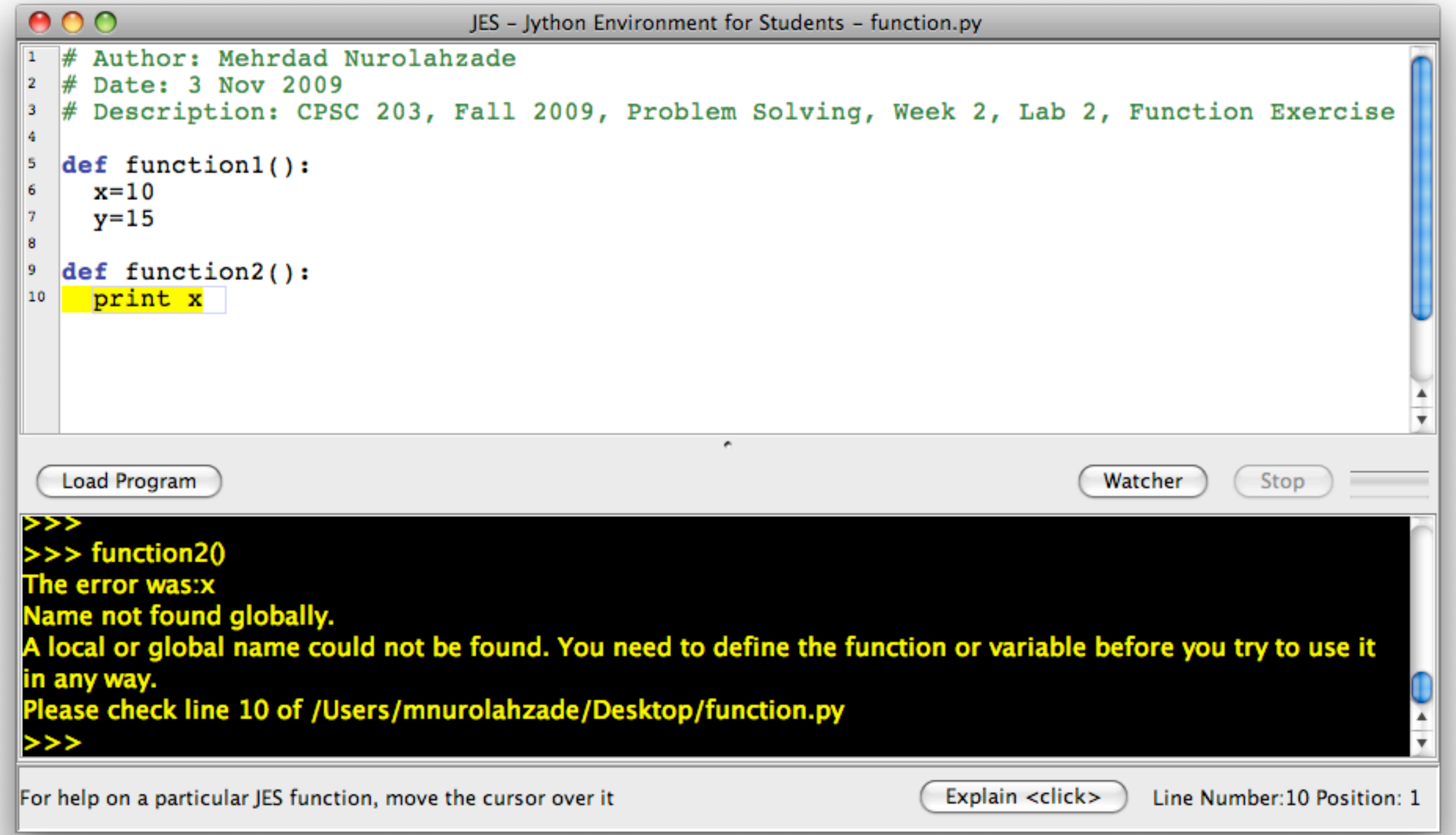
For help on a particular JES function, move the cursor over it

Line Number:3 Position: 15

Variable Scope (1)

- All variables defined inside the body of a function are local to that function, i.e. cannot be accessed in other functions.

Variable Scope (2)


JES – Jython Environment for Students – function.py
Load Program
Watcher
Stop
Explain <click>
For help on a particular JES function, move the cursor over it
Line Number:10 Position: 1

```
# Author: Mehrdad Nurolahzade
# Date: 3 Nov 2009
# Description: CPSC 203, Fall 2009, Problem Solving, Week 2, Lab 2, Function Exercise

def function1():
  x=10
  y=15

def function2():
  print x
```

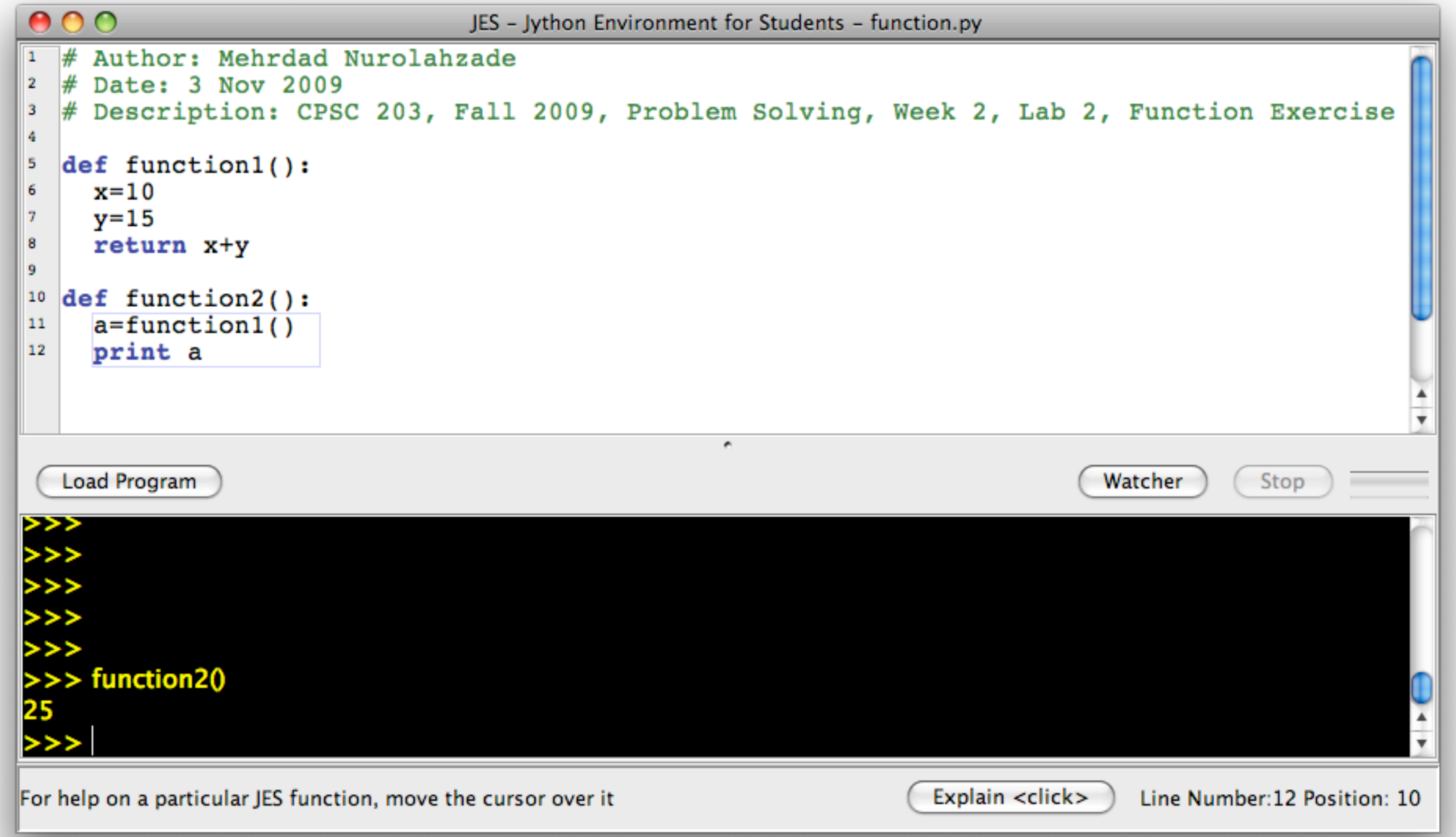
```
>>>
>>> function2()
The error was:x
Name not found globally.
A local or global name could not be found. You need to define the function or variable before you try to use it
in any way.
Please check line 10 of /Users/mnurolahzade/Desktop/function.py
>>>
```

Return Value (1)

- A function can return a value upon exit.
- Function A calling function B receives its return value.
- General syntax:

```
def function-name(zero or more arguments):
    statement1
    statement2
    …
    statementN
    return return-value
```

Return Value (2)



```
# Author: Mehrdad Nurolahzade
# Date: 3 Nov 2009
# Description: CPSC 203, Fall 2009, Problem Solving, Week 2, Lab 2, Function Exercise

def function1():
  x=10
  y=15
  return x+y

def function2():
  a=function1()
  print a
```

```
>>>
>>>
>>>
>>>
>>>
>>> function2()
25
>>>
```

For help on a particular JES function, move the cursor over it

Line Number:12 Position: 10

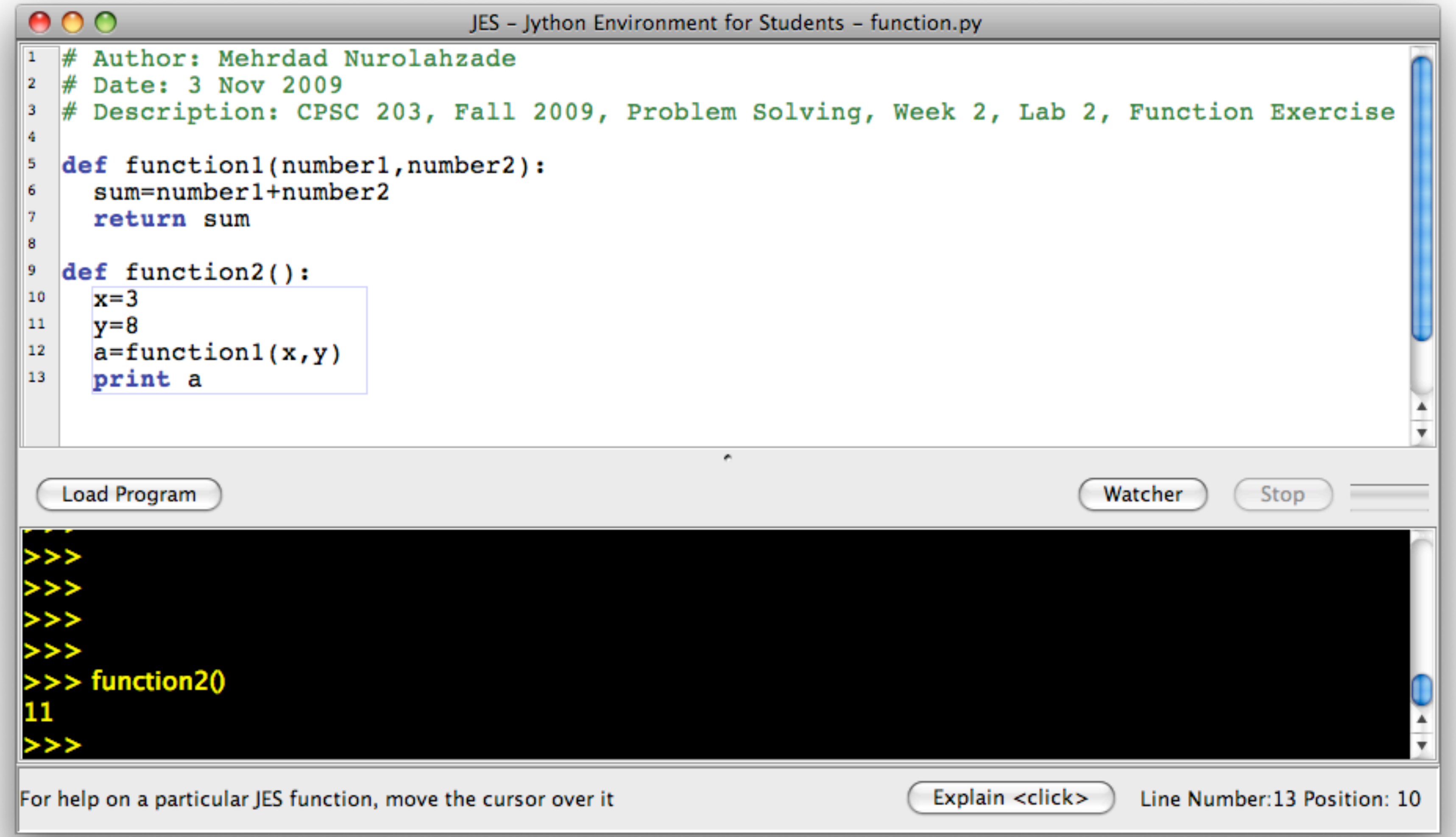
Function Arguments (1)

- Function arguments are a list of parameters that are passed to the function.
- A function can have zero or more arguments.

- General syntax:

```
def function-name(arg1, arg2, …, argN):
  statements
  return return-value
```

Function Arguments (2)

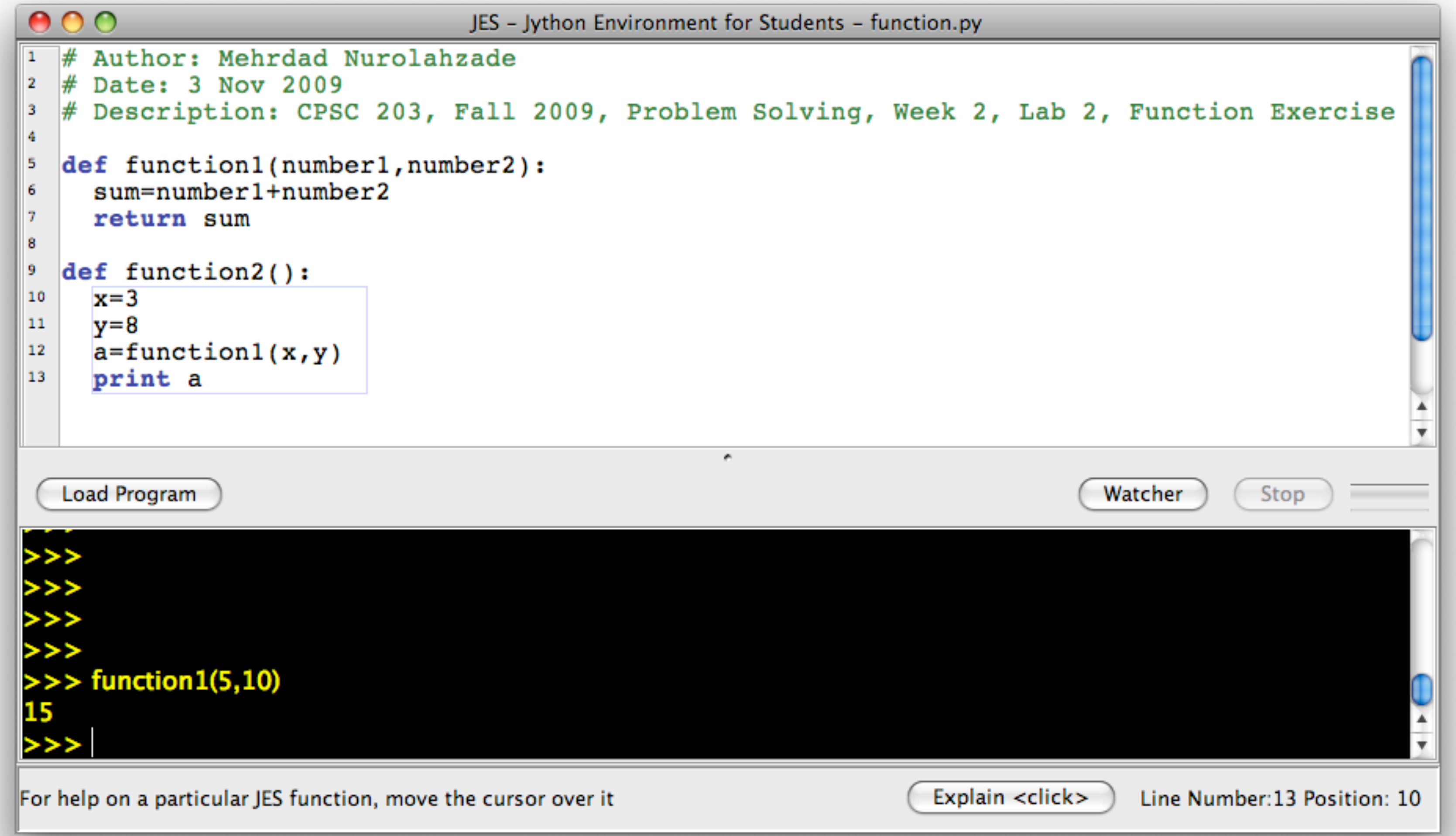

JES – Jython Environment for Students – function.py
Load Program
Watcher
Stop
Explain <click>
Line Number:13 Position: 10

```
# Author: Mehrdad Nurolahzade
# Date: 3 Nov 2009
# Description: CPSC 203, Fall 2009, Problem Solving, Week 2, Lab 2, Function Exercise

def function1(number1,number2):
  sum=number1+number2
  return sum

def function2():
  x=3
  y=8
  a=function1(x,y)
  print a
```

```
>>>
>>>
>>>
>>>
>>> function2()
11
>>>
```

For help on a particular JES function, move the cursor over it

Function Arguments (3)


JES – Jython Environment for Students – function.py

```
# Author: Mehrdad Nurolahzade
# Date: 3 Nov 2009
# Description: CPSC 203, Fall 2009, Problem Solving, Week 2, Lab 2, Function Exercise

def function1(number1,number2):
  sum=number1+number2
  return sum

def function2():
  x=3
  y=8
  a=function1(x,y)
  print a
```

```
>>>
>>>
>>>
>>>
>>> function1(5,10)
15
>>>
```

For help on a particular JES function, move the cursor over it

Line Number:13 Position: 10

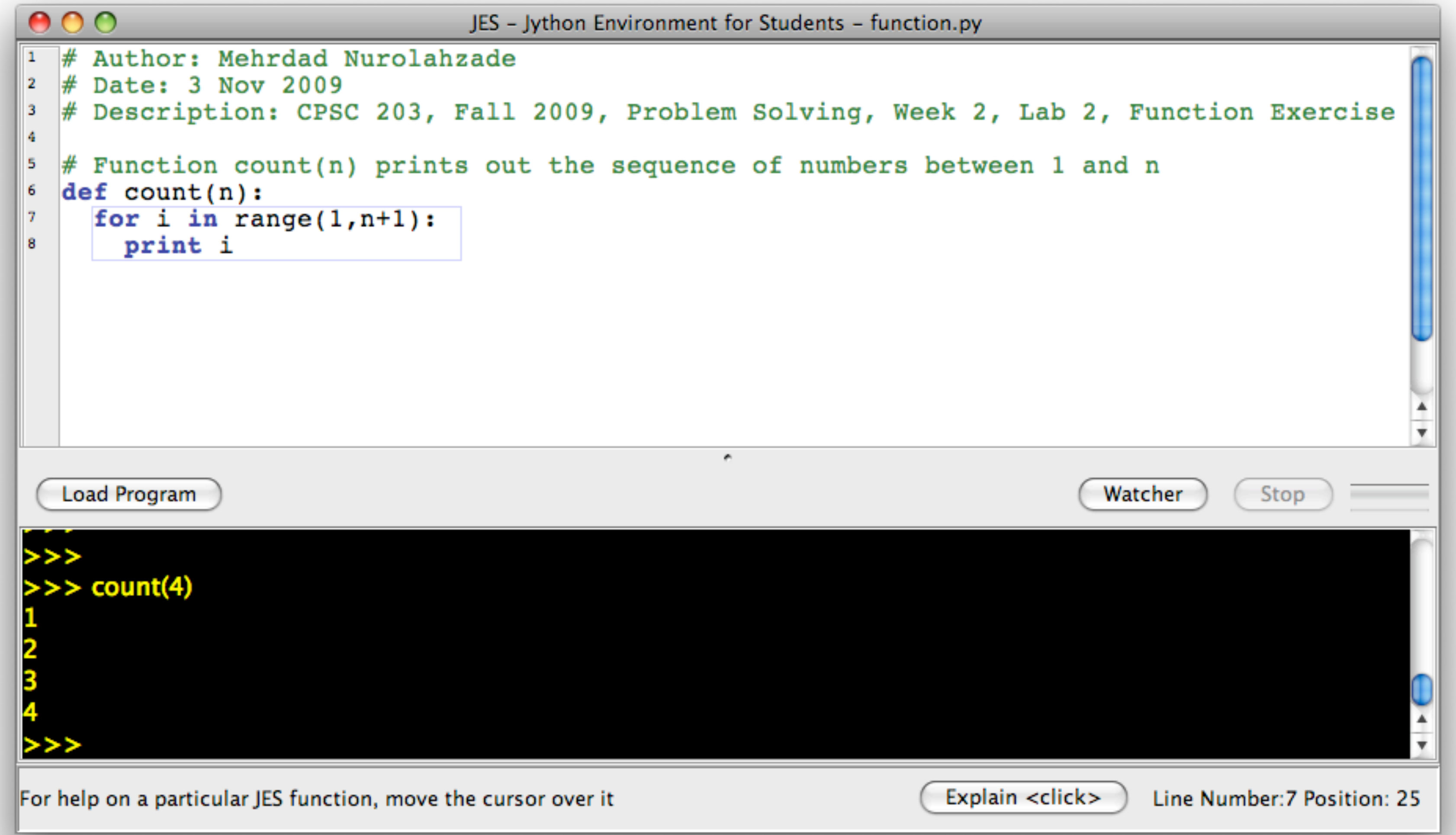
Function Exercise (1)

- Write the Jython function **count(n)** that given the argument **n** of type Integer prints out numbers from 1 to **n**.

- Example run:

```
count(4)
1
2
3
4
```

Function Exercise (2)



```
# Author: Mehrdad Nurolahzade
# Date: 3 Nov 2009
# Description: CPSC 203, Fall 2009, Problem Solving, Week 2, Lab 2, Function Exercise

# Function count(n) prints out the sequence of numbers between 1 and n
def count(n):
  for i in range(1,n+1):
    print i
```

```
>>>
>>> count(4)
1
2
3
4
>>>
```

For help on a particular JES function, move the cursor over it

Line Number:7 Position: 25

Function Exercise (3)

- Write the Jython function **minimum(S)** that finds the minimum element in the given list **S** of Integers.
- Note: you are not allowed to use the **min()** function.

- Example run:

```
minimum([7, 3, 5, 6, 4, 2, 3])
2
```

Function Exercise (4)

```
def minimum(S):
   if len(S)==0:
        return 'undefined'
   else:
        min_so_far = S[0]
        for i in range(1,len(S)):
                if S[i] < min_so_far:
                        min_so_far = S[i]
   return min_so_far
```

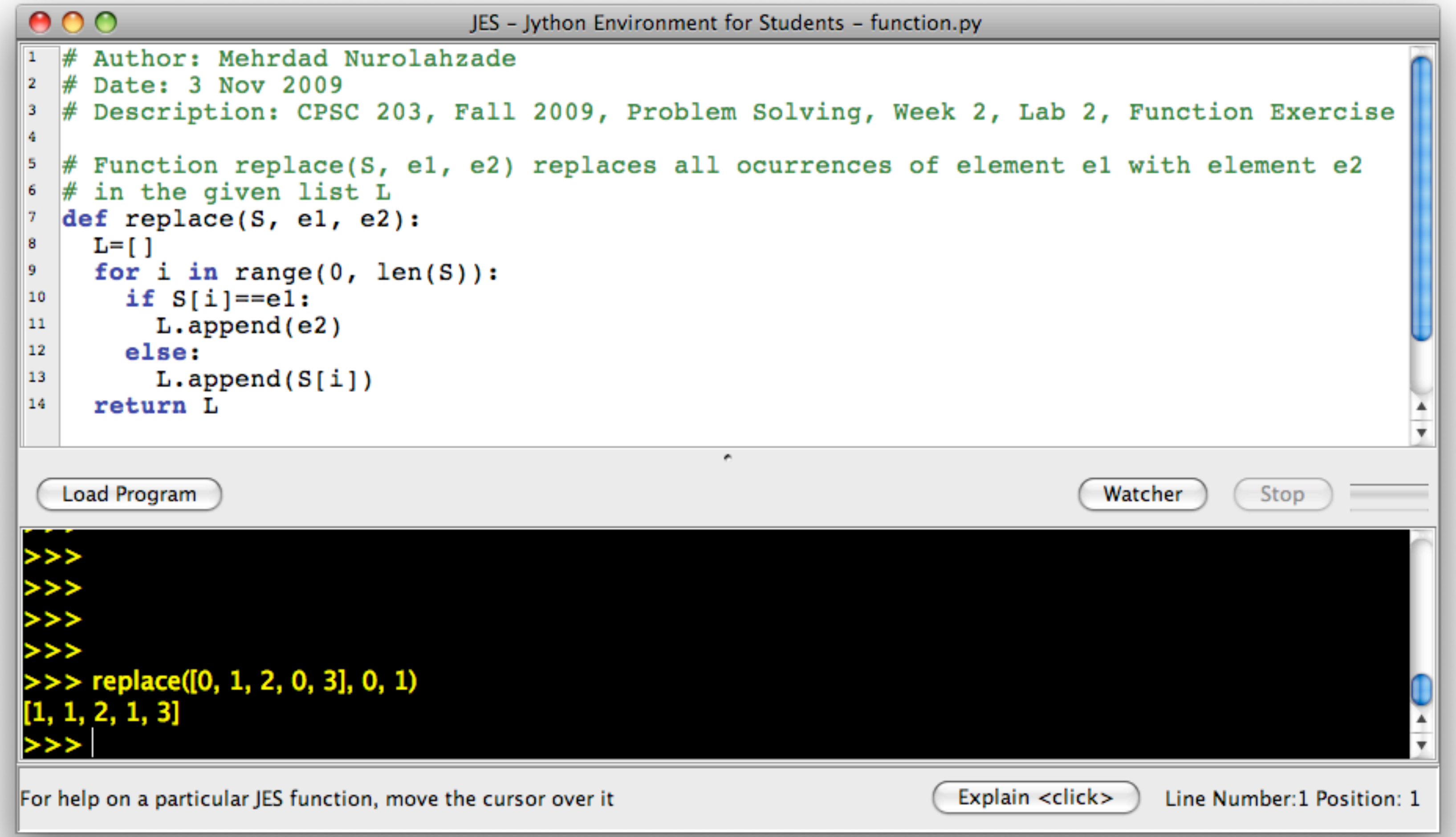
Function Exercise (5)

- Write the Jython function **replace(L, e1, e2)** that replaces all occurrences of element **e1** with element **e2** in the given list **L**.

- Example run:

```
replace([‘a’, ‘d’, ‘a’, ‘c’, ‘b’], ‘a’, ‘b’)
[‘b’, ‘d’, ‘b’, ‘c’, ‘b’]
```

Function Exercise (6)


JES – Jython Environment for Students – function.py
Load Program
Watcher
Stop
Explain <click>
Line Number:1 Position: 1

```python
# Author: Mehrdad Nurolahzade
# Date: 3 Nov 2009
# Description: CPSC 203, Fall 2009, Problem Solving, Week 2, Lab 2, Function Exercise

# Function replace(S, e1, e2) replaces all ocurrences of element e1 with element e2
# in the given list L
def replace(S, e1, e2):
  L=[]
  for i in range(0, len(S)):
    if S[i]==e1:
      L.append(e2)
    else:
      L.append(S[i])
  return L
```

```
>>>
>>>
>>>
>>>
>>> replace([0, 1, 2, 0, 3], 0, 1)
[1, 1, 2, 1, 3]
>>>
```

For help on a particular JES function, move the cursor over it

Graphics in Jython

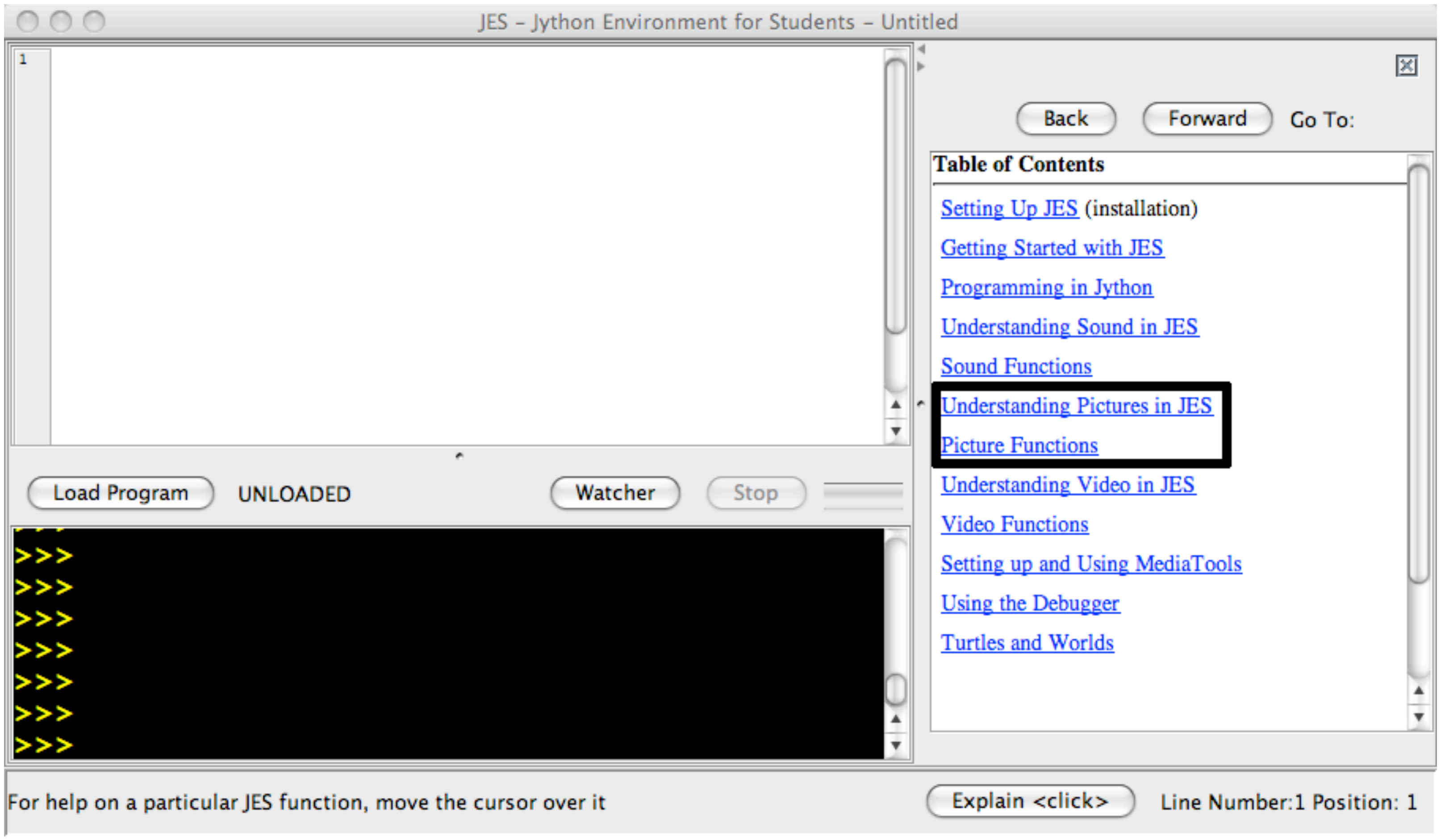

JES – Jython Environment for Students – Untitled
Back
Forward
Go To:
Load Program
UNLOADED
Watcher
Stop
Explain <click>
Line Number:1 Position: 1

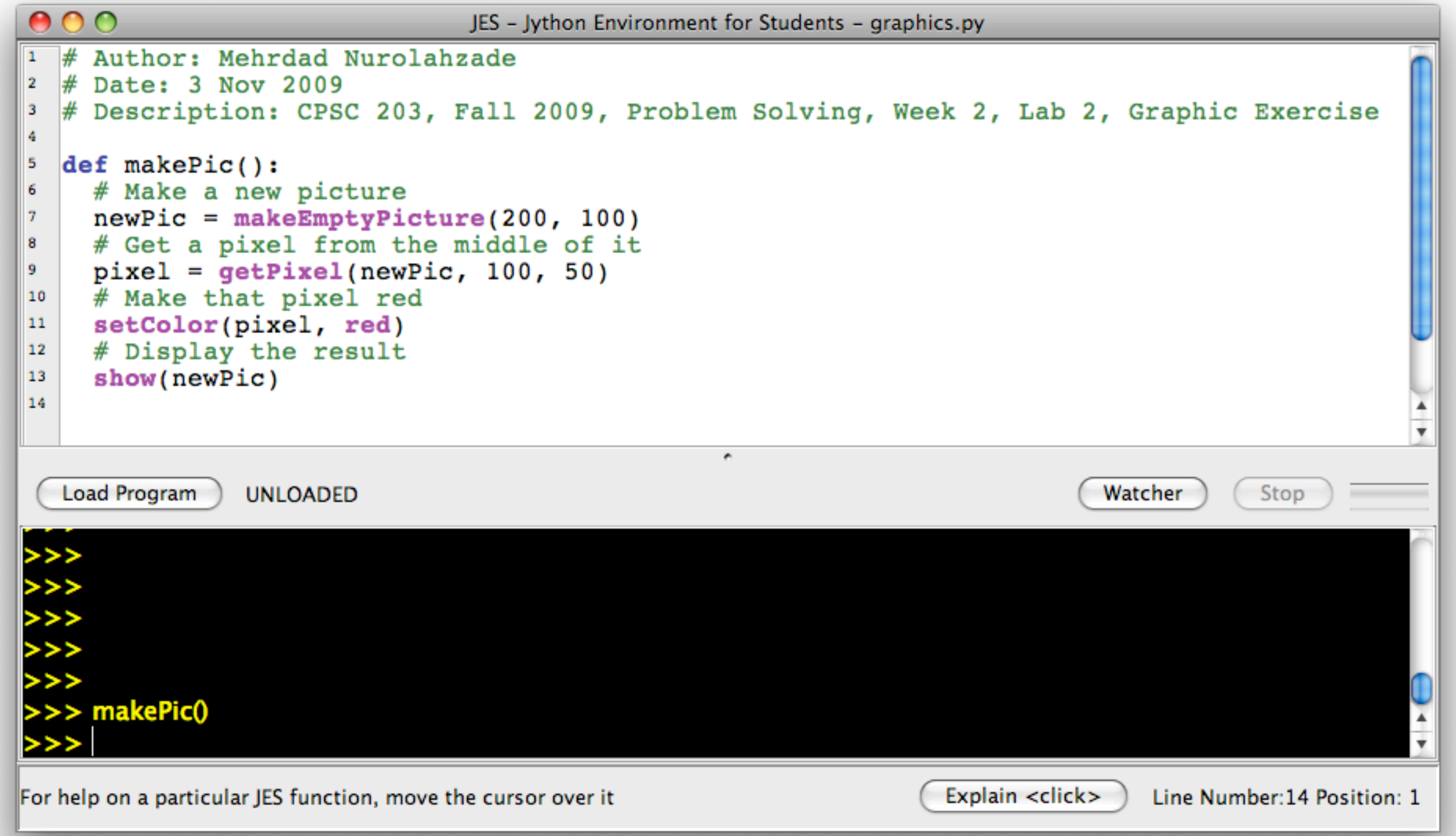
Table of Contents

Setting Up JES (installation)
Getting Started with JES
Programming in Jython
Understanding Sound in JES
Sound Functions
Understanding Pictures in JES
Picture Functions
Understanding Video in JES
Video Functions
Setting up and Using MediaTools
Using the Debugger
Turtles and Worlds

```
>>>
>>>
>>>
>>>
>>>
>>>
>>>
```

For help on a particular JES function, move the cursor over it

Making A Red Dot


JES – Jython Environment for Students – graphics.py
Load Program
UNLOADED
Watcher
Stop
For help on a particular JES function, move the cursor over it
Explain <click>
Line Number:14 Position: 1

```
# Author: Mehrdad Nurolahzade
# Date: 3 Nov 2009
# Description: CPSC 203, Fall 2009, Problem Solving, Week 2, Lab 2, Graphic Exercise

def makePic():
  # Make a new picture
  newPic = makeEmptyPicture(200, 100)
  # Get a pixel from the middle of it
  pixel = getPixel(newPic, 100, 50)
  # Make that pixel red
  setColor(pixel, red)
  # Display the result
  show(newPic)
```

```
>>>
>>>
>>>
>>>
>>>
>>> makePic()
>>>
```

makeEmptyPicture()

makeEmptyPicture(width, height[, color]):

width: the width of the empty picture

height: height of the empty picture

color: background color of the empty picture (optional)

returns: a new picture object with all the pixels set to the specified color

Makes a new "empty" picture and returns it to you. The width and height must be between 0 and 10000. Default color is white.

getPixel()

getPixel(picture, xpos, ypos):

picture: the picture you want to get the pixel from

xpos: the x-coordinate of the pixel you want

ypos: the y-coordinate of the pixel you want

Takes a picture, an x position and a y position (two numbers), and returns the Pixel object at that point in the picture.

setColor()

setColor(pixel, color):

pixel: the pixel you want to set the color of

color: the color you want to set the pixel to

Takes in a pixel and a color, and sets the pixel to the given color.

show()

show(picture):

picture: the picture you want to see

Shows the picture provided as input.

addLine()

addLine(picture, startX, startY, endX, endY[, color]):

picture: the picture you want to draw the line on

startX: the x position you want the line to start

startY: the y position you want the line to start

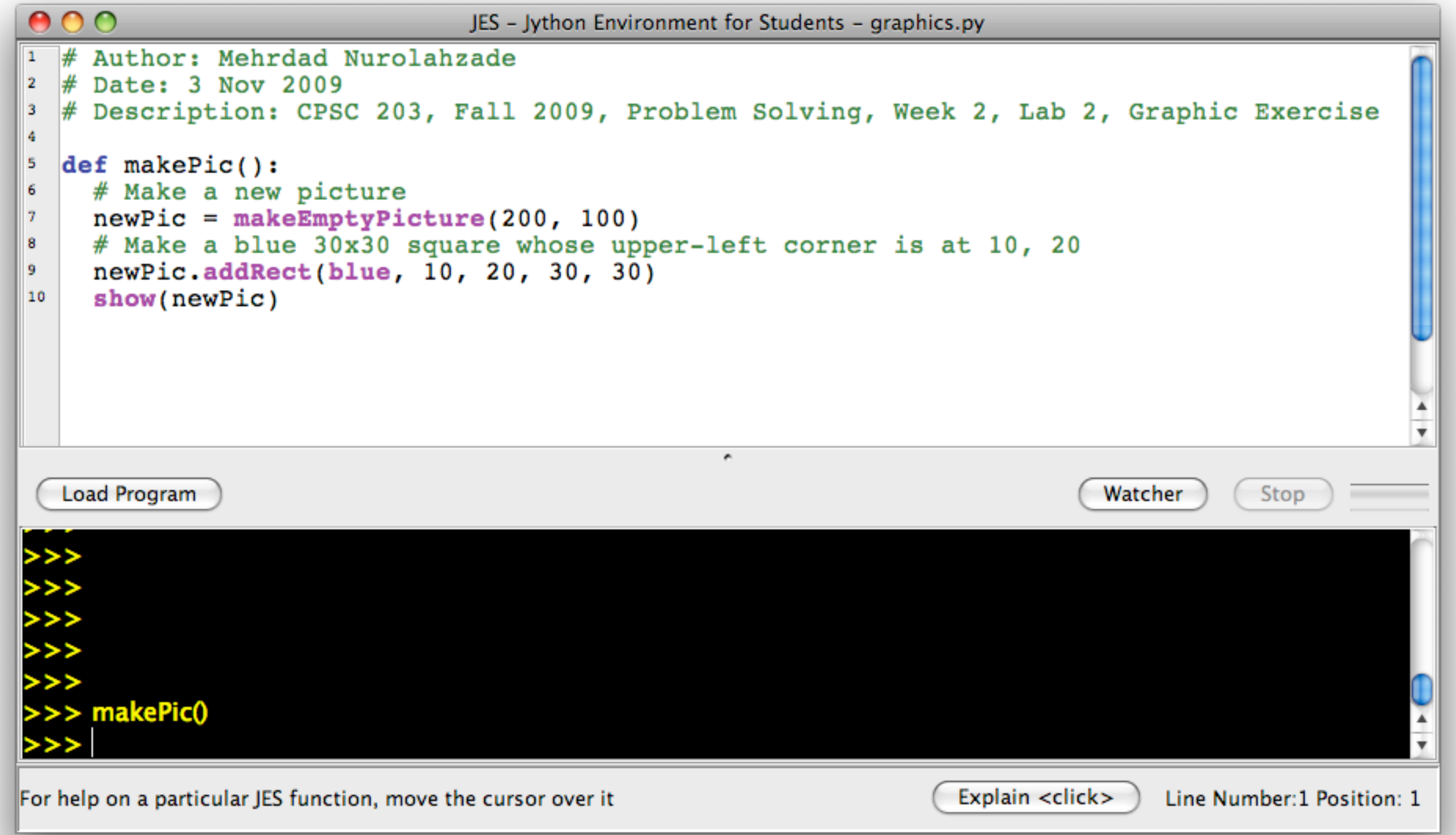
endX: the x position you want the line to end

endY: the y position you want the line to end

color: the color you want to draw in (optional)

Takes a picture, a starting (x, y) position (two numbers), and an ending (x, y) position (two more numbers, four total), and (optionally) a color as input. Adds a line from the starting point to the ending point in the picture. Default color is black.

Draw A Blue Rectangle


JES - Jython Environment for Students - graphics.py

```python
# Author: Mehrdad Nurolahzade
# Date: 3 Nov 2009
# Description: CPSC 203, Fall 2009, Problem Solving, Week 2, Lab 2, Graphic Exercise

def makePic():
  # Make a new picture
  newPic = makeEmptyPicture(200, 100)
  # Make a blue 30x30 square whose upper-left corner is at 10, 20
  newPic.addRect(blue, 10, 20, 30, 30)
  show(newPic)
```

Load Program
Watcher
Stop

```
>>>
>>>
>>>
>>>
>>>
>>> makePic()
>>>
```

For help on a particular JES function, move the cursor over it
Explain <click>
Line Number:1 Position: 1

addRect()

addRect(picture, startX, startY, width, height[, color]):

picture: the picture you want to draw the rectangle on

startX: the x-coordinate of the upper left-hand corner of the rectangle

startY: the y-coordinate of the upper left-hand corner of the rectangle

width: the width of the rectangle

height: the height of the rectangle

color: the color you want to draw in (optional)

Takes a picture, a starting (x, y) position (two numbers), a width and height (two more numbers, four total), and (optionally) a color as input. Adds a rectangular outline of the specified dimensions using the (x,y) as the upper left corner. Default color is black.

addOval()

addOval(picture, startX, startY, width, height[, color]):

picture: the picture you want to draw the rectangle on

startX: the x-coordinate of the upper left-hand corner of the bounding rectangle of the oval

startY: the y-coordinate of the upper left-hand corner of the bounding rectangle of the oval

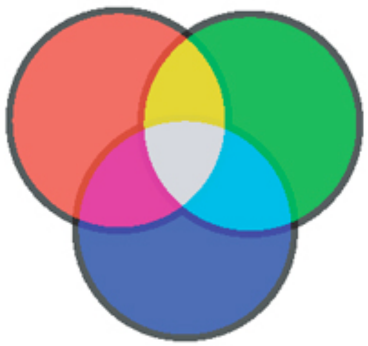
width: the width of the oval

height: the height of the oval

color: the color you want to draw in (optional)

Takes a picture, a starting (x, y) position (two numbers), a width and height (two more numbers, four total), and (optionally) a color as input. Adds an oval outline of the given dimensions using the (x,y) as the upper left corner of the bounding rectangle. Default color is black.

makeColor()



makeColor(red[, green, blue]):

red: the amount of red you want in the color (or a Color object you want to duplicate)

green: the amount of green you want in the color (optional)

blue: the amount of blue you want in the picture (optional)

returns: the color made from the inputs

Takes three integer inputs for the red, green, and blue components (in order) and returns a color object. If green and blue are omitted, the red value is used as the intensity of a gray color. Also it works with only a color as input and returns a new color object with the same RGB values as the original.

addText()

addText(picture, xpos, ypos, text[, color]):

picture: the picture you want to add the text to

xpos: the x-coordinate where you want to start writing the text

ypos: the y-coordinate where you want to start writing the text

text: string containing the text you want written

color: the color you want to draw in (optional)

Takes a picture, an x position and a y position (two numbers), and some text as a string, which will get drawn into the picture, in the specified color. Default is black.

writePictureTo()

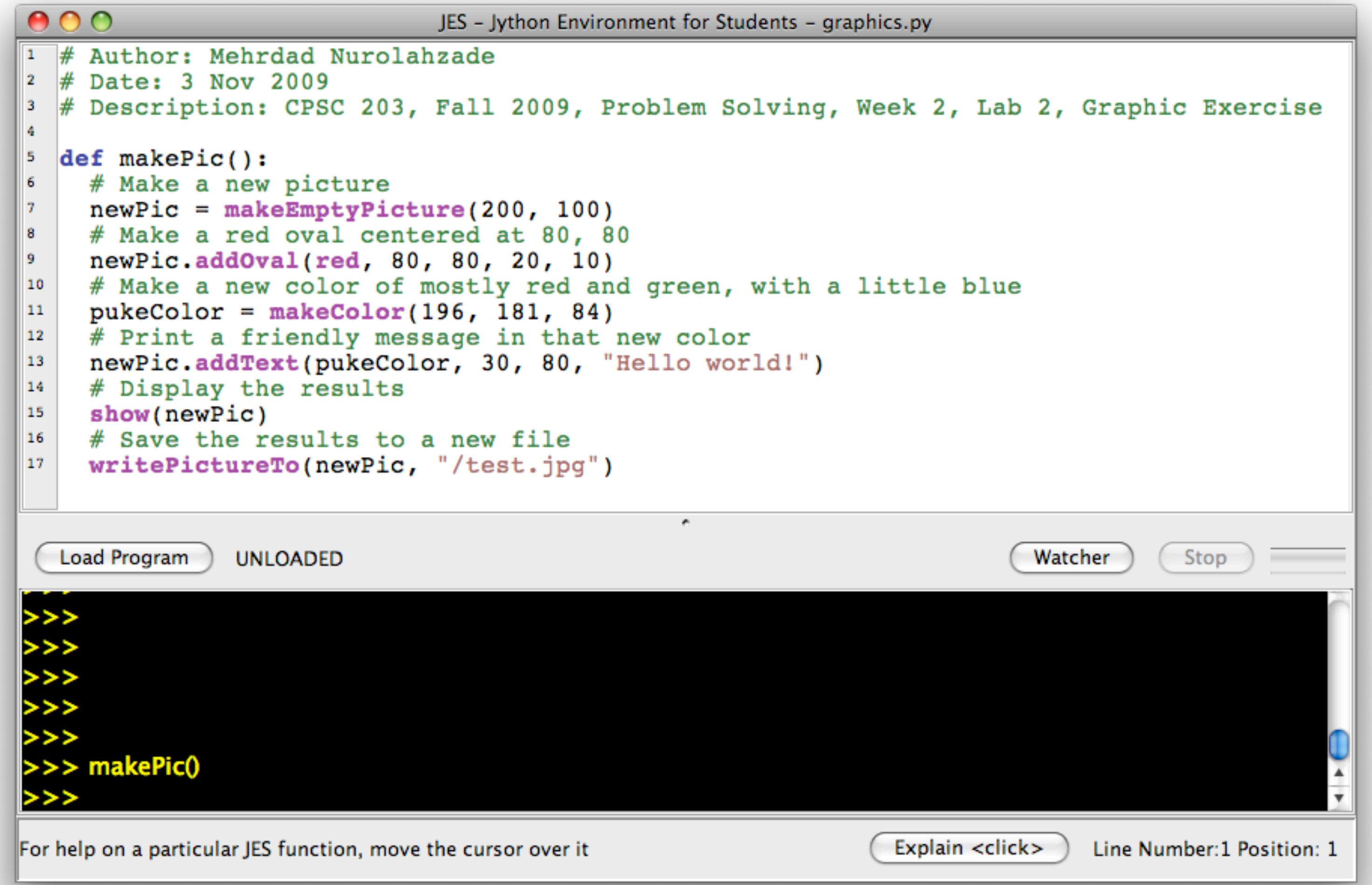
writePictureTo(picture, path):

picture: the picture you want to be written out to a file

path: the path to the file you want the picture written to

Takes a picture and a file name (string) as input, then writes the picture to the file as a JPEG, PNG, or BMP. (Be sure to end the filename in ".jpg" or ".png" or ".bmp" for the operating system to understand it well.)

Graphics Example



```python
# Author: Mehrdad Nurolahzade
# Date: 3 Nov 2009
# Description: CPSC 203, Fall 2009, Problem Solving, Week 2, Lab 2, Graphic Exercise

def makePic():
  # Make a new picture
  newPic = makeEmptyPicture(200, 100)
  # Make a red oval centered at 80, 80
  newPic.addOval(red, 80, 80, 20, 10)
  # Make a new color of mostly red and green, with a little blue
  pukeColor = makeColor(196, 181, 84)
  # Print a friendly message in that new color
  newPic.addText(pukeColor, 30, 80, "Hello world!")
  # Display the results
  show(newPic)
  # Save the results to a new file
  writePictureTo(newPic, "/test.jpg")
```

```
>>>
>>>
>>>
>>>
>>>
>>> makePic()
>>>
```

For help on a particular JES function, move the cursor over it
Explain <click>
Line Number:1 Position: 1