Exposure CS 2020 for CS1=Honors

Chapter 3 Section 1-4 Slides

Introduction to Python Coding:
Using jGRASP, Output with print & Comments

PowerPoint Presentation
created by:
Mr. John L. M. Schram
and Mr. Leon Schram
Authors of Exposure
Computer Science

Section 3.1

Introduction

Introduction

This chapter will start by showing you the necessary tools to write simple programs. Keep in mind that you will be creating your programs in a high-level language, Python. This means you need some type of editing environment in which to write your programs. After the program is written you need to translate and execute your program. For now, learning those fundamental program-writing skills is your primary concern.

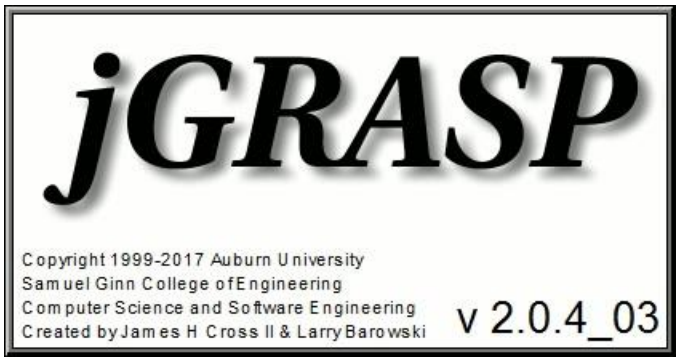
Section 3.2

Using jGRASP for Python

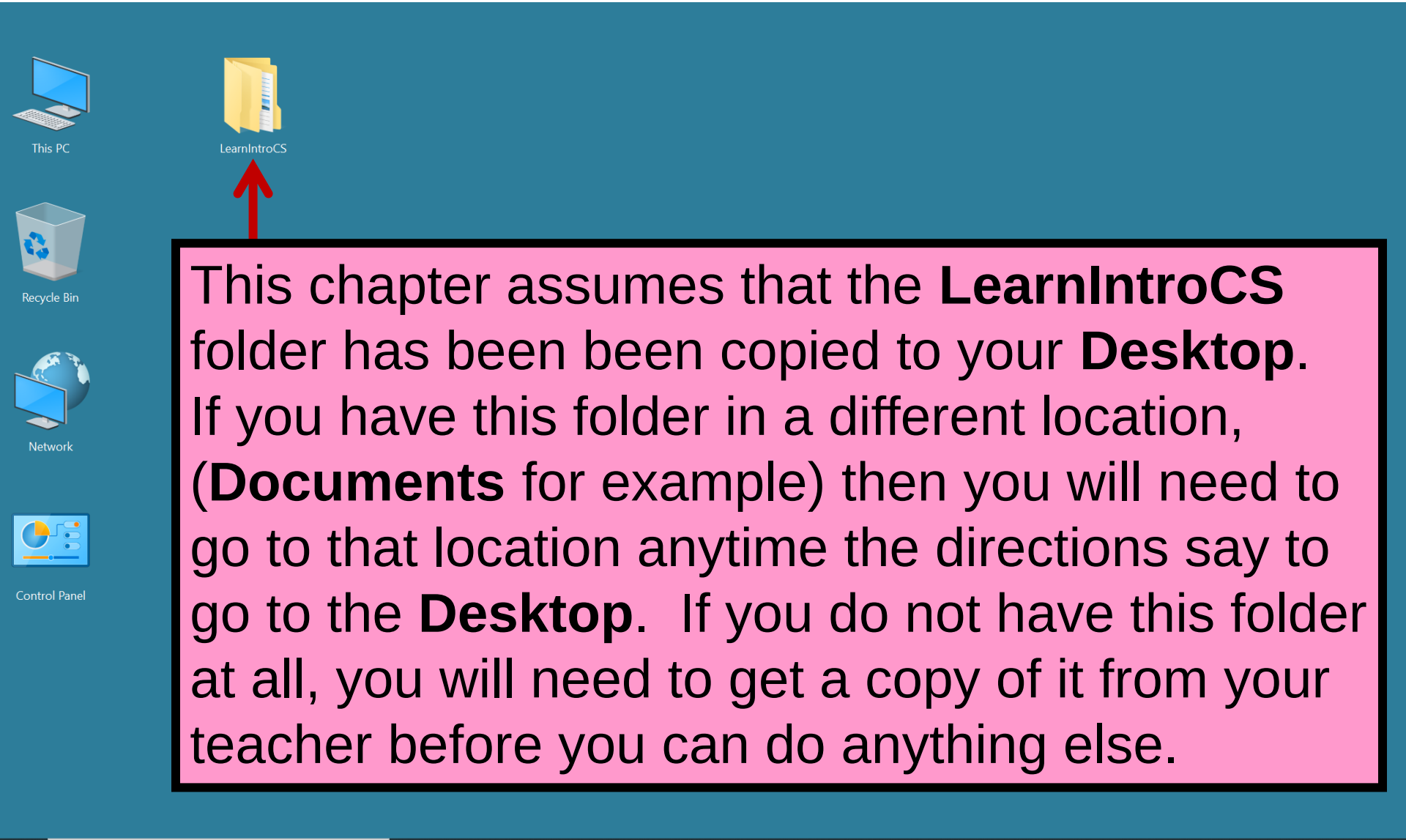
Required Software

In Chapter 2 you installed the **Java**, **jGRASP** and **Python** in order to write and execute Python programs. You may wonder why you had to install Java to run Python programs. The answer is **jGRASP** was written in **Java** and cannot execute without it.

python



The LearnIntroCS Folder



What is an IDE?

IDE stands for **I***ntegrated* **D***evelopment* **E***nvironment*.

An IDE is an “environment” that contains everything you need to “develop” programs all “integrated” together.

Features of an IDE	
Text Editor	A full screen editor for writing your programs
	A file manager to save programs and load them again later
Translator	Used to translate your source code into machine code. Most languages use a compiler. Python uses an interpreter.
Output/Message Window	Show the output of your programs or displays error messages

jGRASP

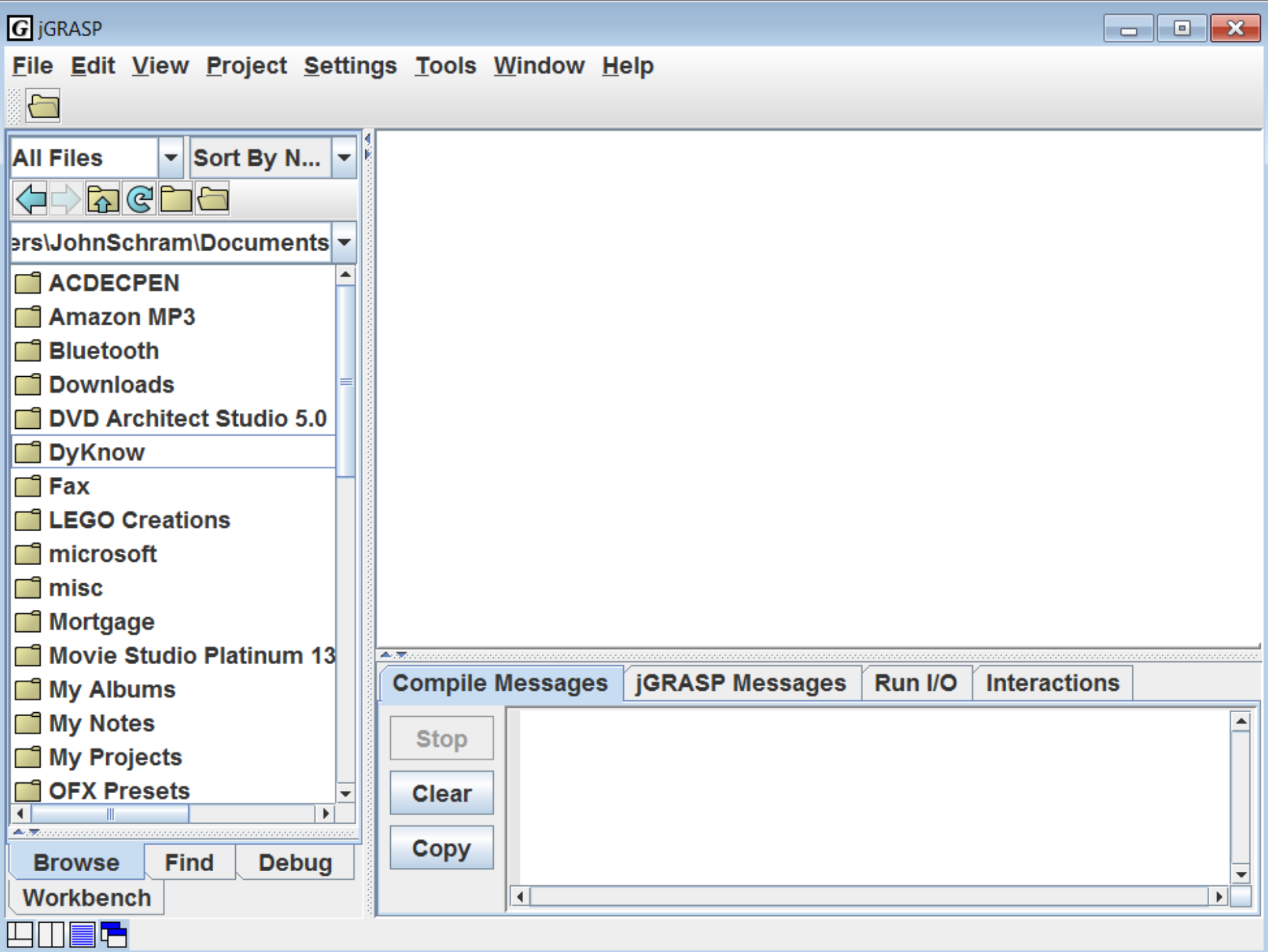
The IDE that we will be using is **jGRASP**.

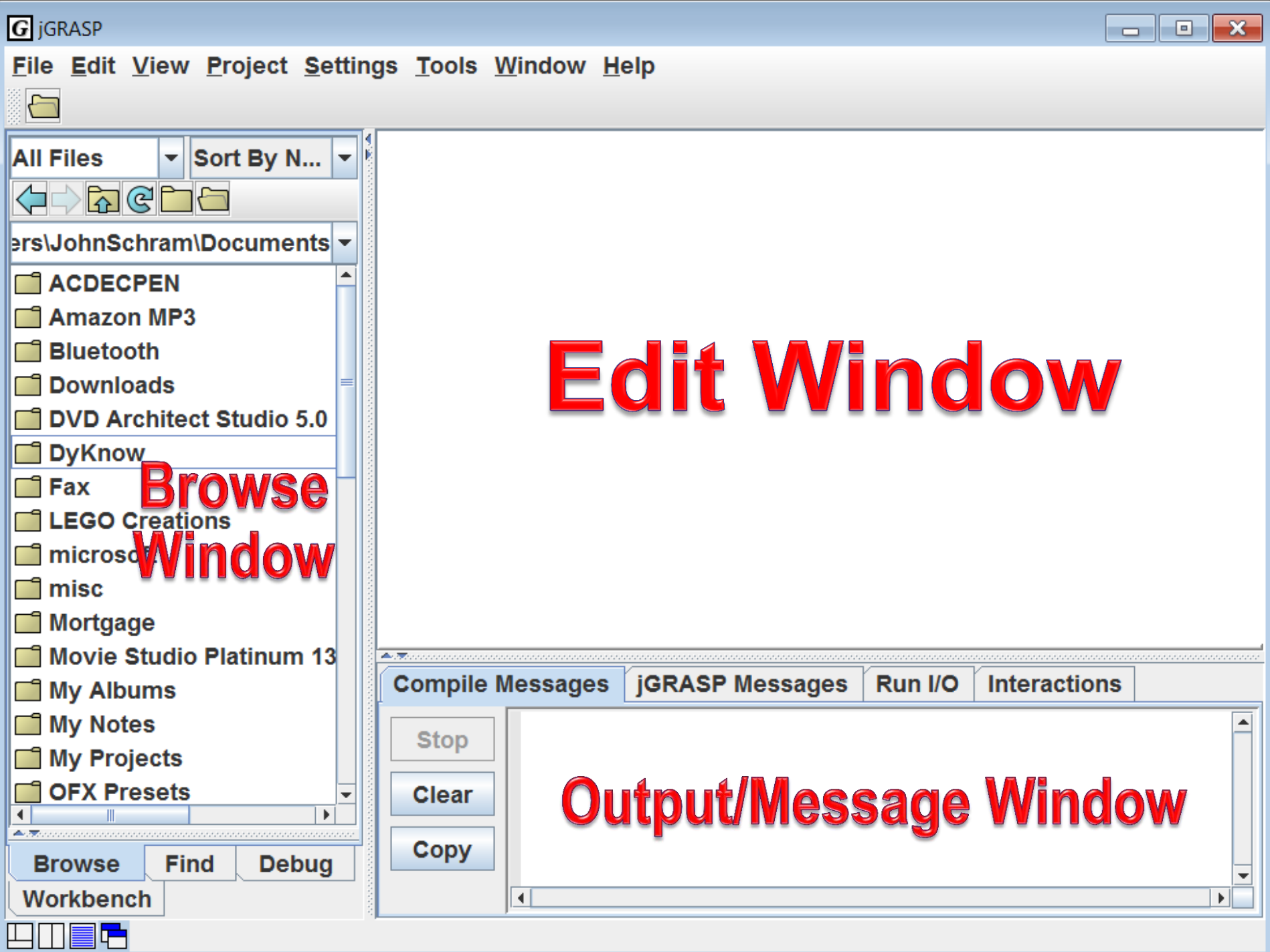
One of the main reasons that we are using **jGRASP** is that it is *freeware*.

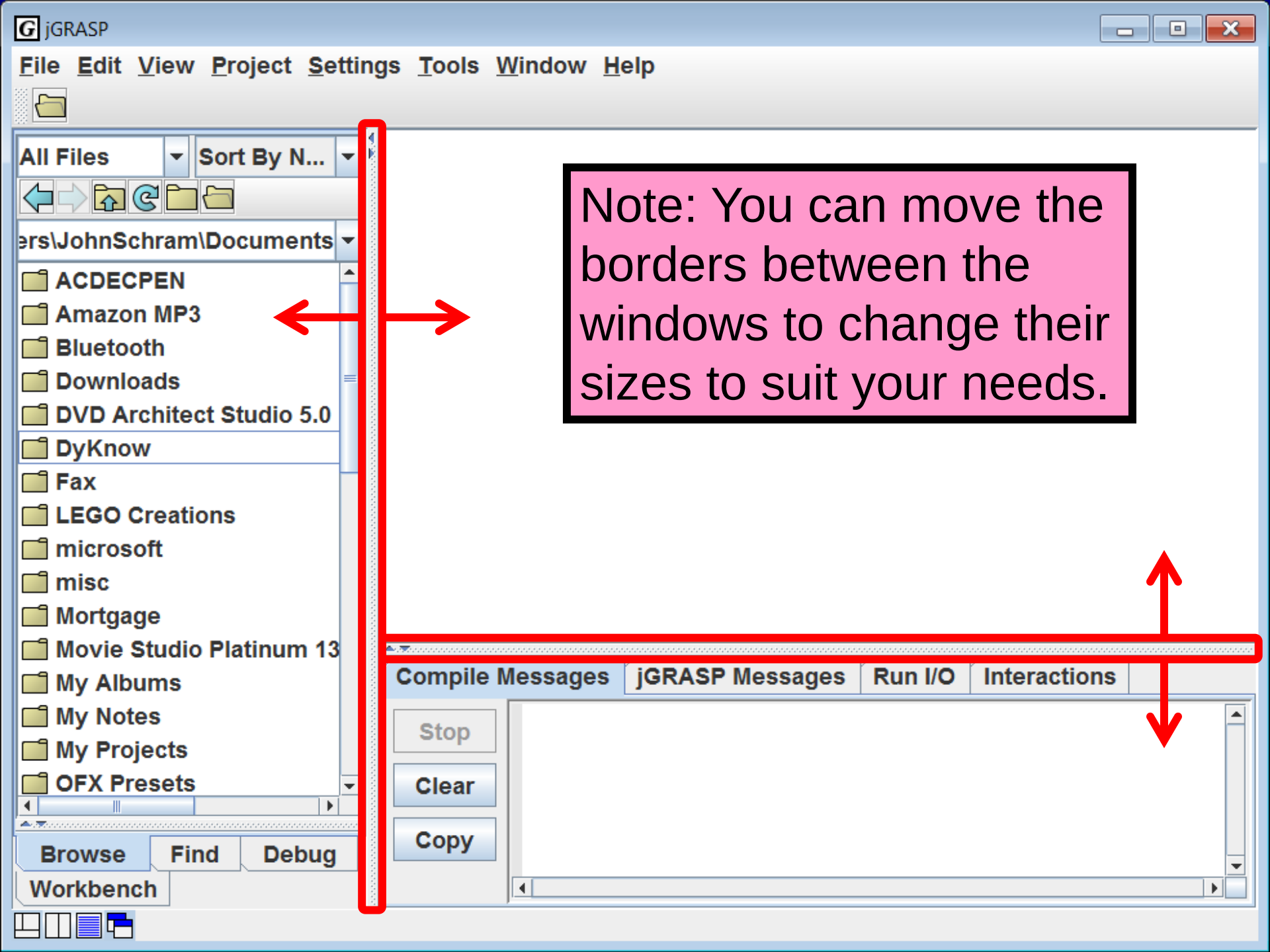
jGRASP also has other features that make it work well for our lab lectures and lab assignments.

You will find the **jGRASP** icon either on your Desktop, Start menu, Taskbar, Dock, or amoung other apps in the Launcher.

Click/Double-click the icon to load **jGRASP**.

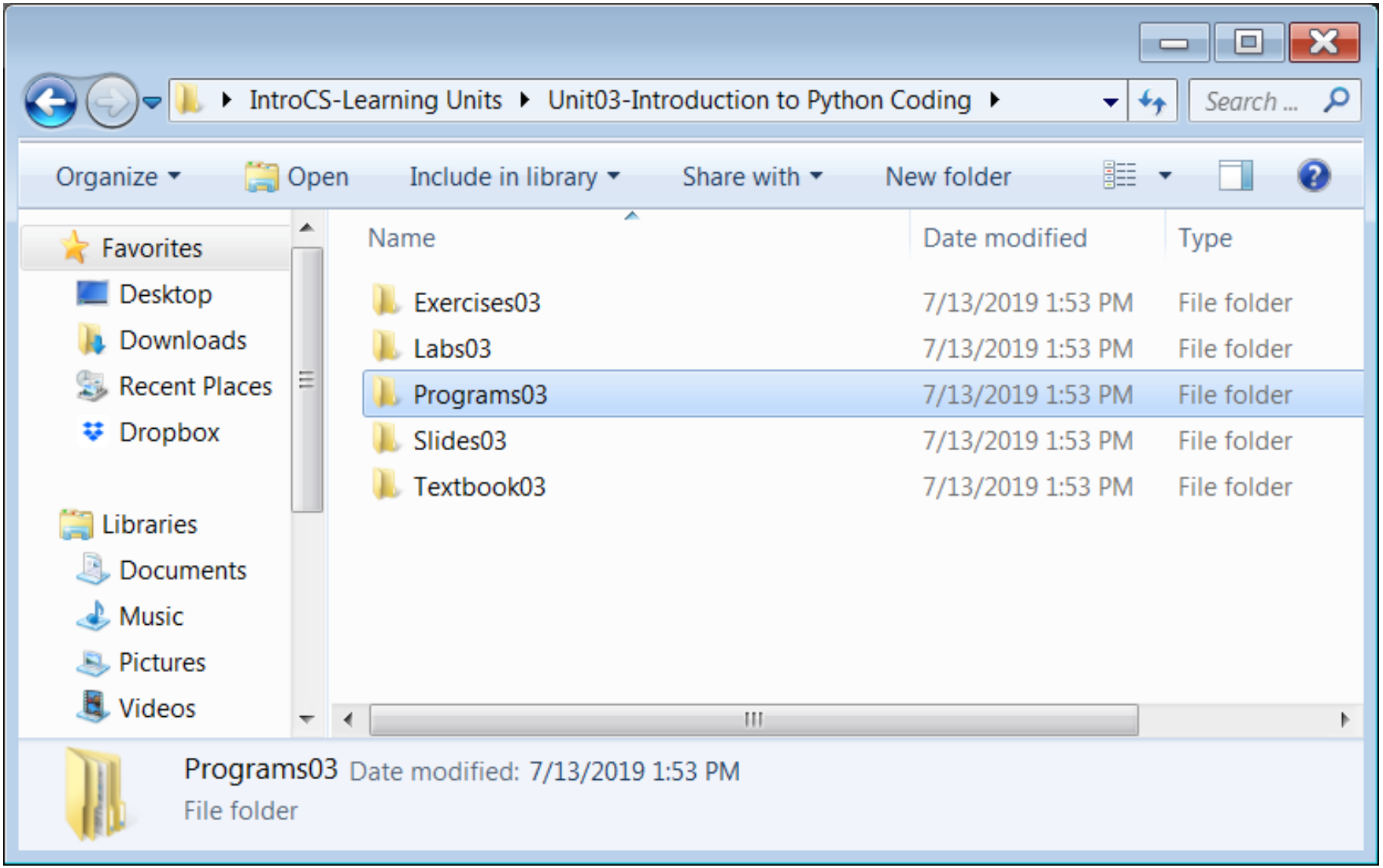

jGRASP
File Edit View Project Settings Tools Window Help
All Files
Sort By N...
ers\JohnSchram\Documents
ACDECPEN
Amazon MP3
Bluetooth
Downloads
DVD Architect Studio 5.0
DyKnow
Fax
LEGO Creations
microsoft
misc
Mortgage
Movie Studio Platinum 13
My Albums
My Notes
My Projects
OFX Presets
Browse
Find
Debug
Workbench
Compile Messages
jGRASP Messages
Run I/O
Interactions
Stop
Clear
Copy


jGRASP
File Edit View Project Settings Tools Window Help
All Files
Sort By N...
ers\JohnSchram\Documents
ACDECPEN
Amazon MP3
Bluetooth
Downloads
DVD Architect Studio 5.0
DyKnow
Fax
LEGO Creations
misc
Mortgage
Movie Studio Platinum 13
My Albums
My Notes
My Projects
OFX Presets
Browse Window
Edit Window
Compile Messages
jGRASP Messages
Run I/O
Interactions
Stop
Clear
Copy
Output/Message Window
Browse
Find
Debug
Workbench

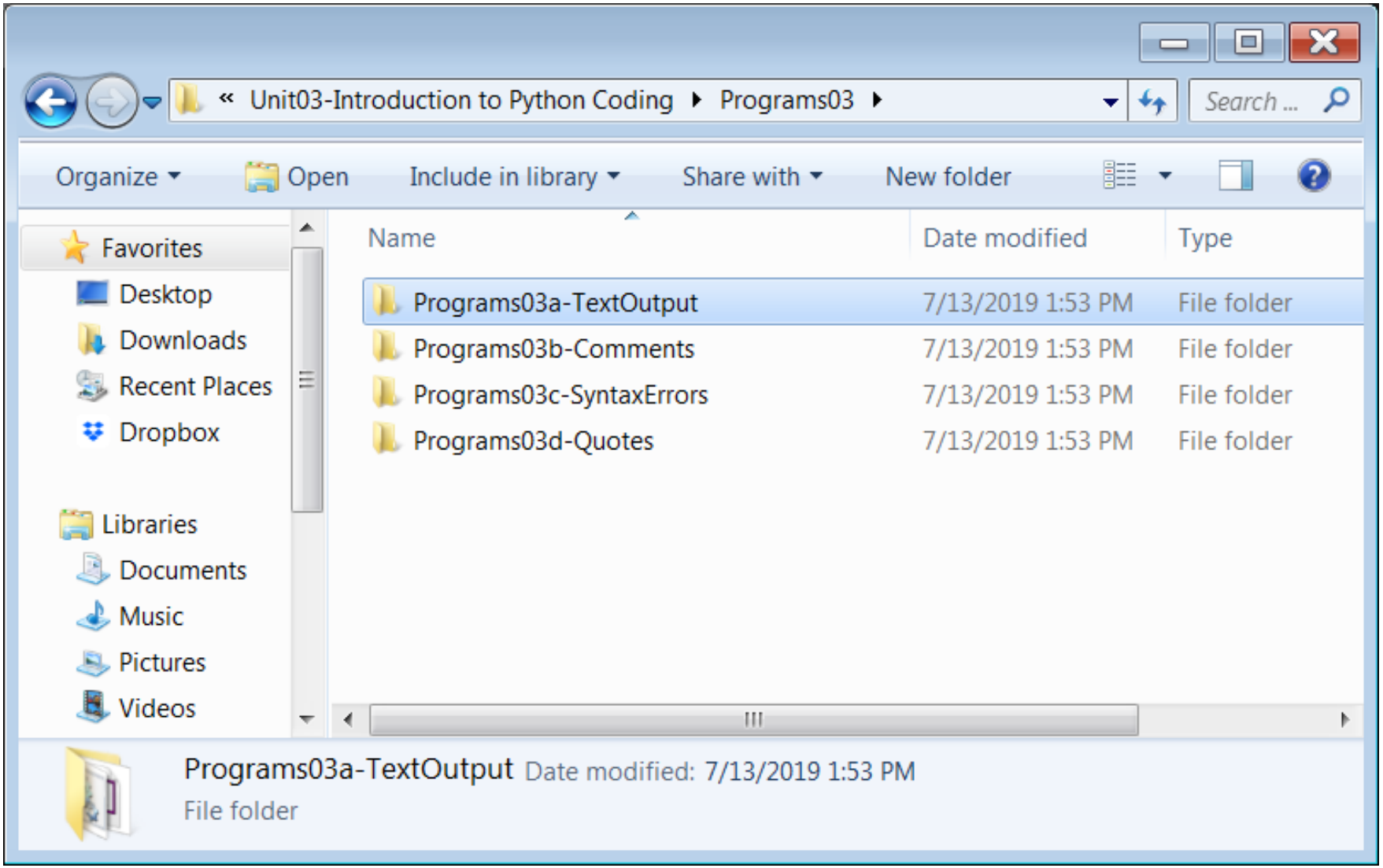

jGRASP
File Edit View Project Settings Tools Window Help
All Files
Sort By N...
ers\JohnSchram\Documents
ACDECPEN
Amazon MP3
Bluetooth
Downloads
DVD Architect Studio 5.0
DyKnow
Fax
LEGO Creations
microsoft
misc
Mortgage
Movie Studio Platinum 13
My Albums
My Notes
My Projects
OFX Presets
Browse
Find
Debug
Workbench
Note: You can move the borders between the windows to change their sizes to suit your needs.
Compile Messages
jGRASP Messages
Run I/O
Interactions
Stop
Clear
Copy

Organization of Program Examples

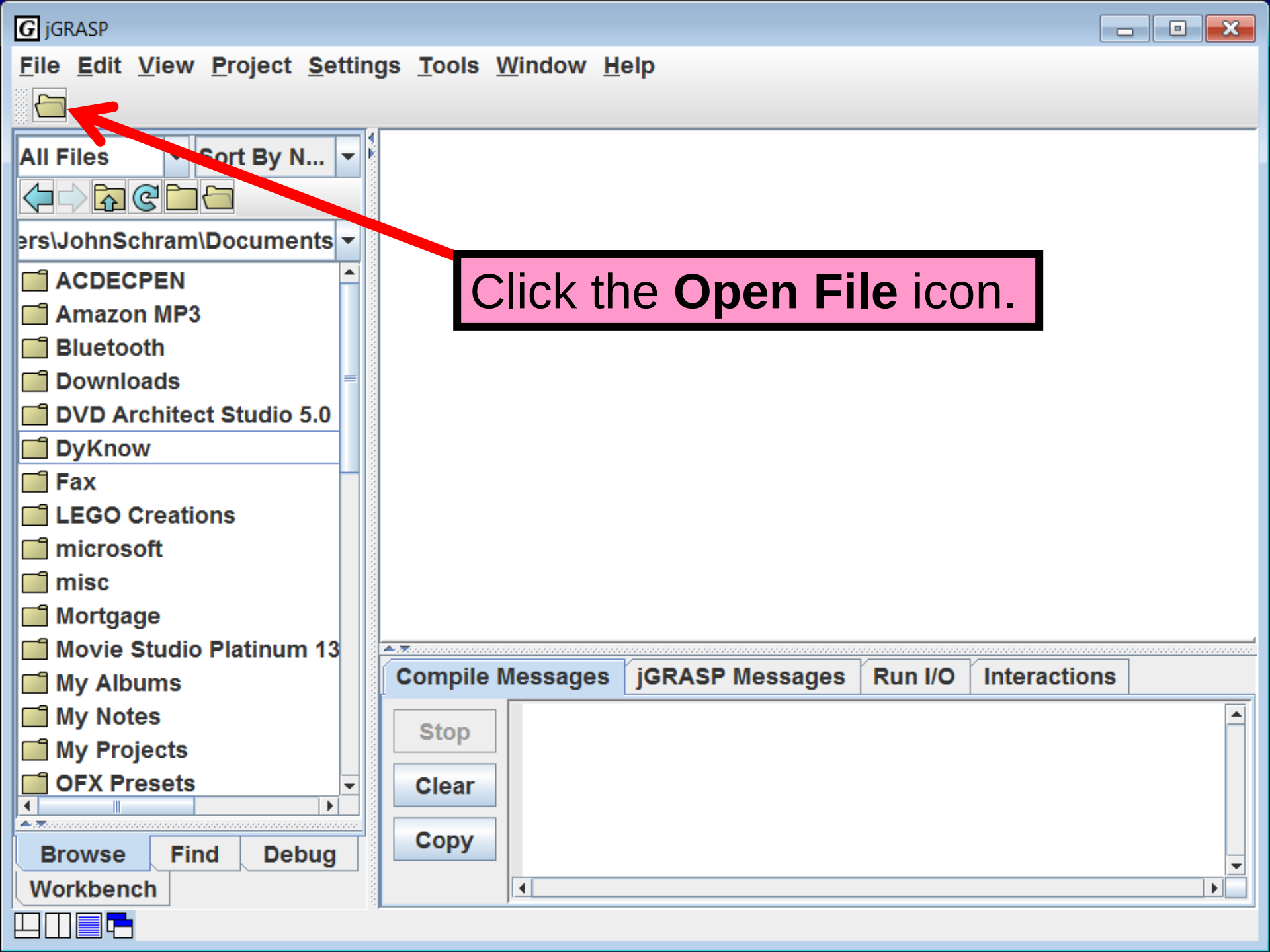
Organization of Program Examples in LearnIntroCS

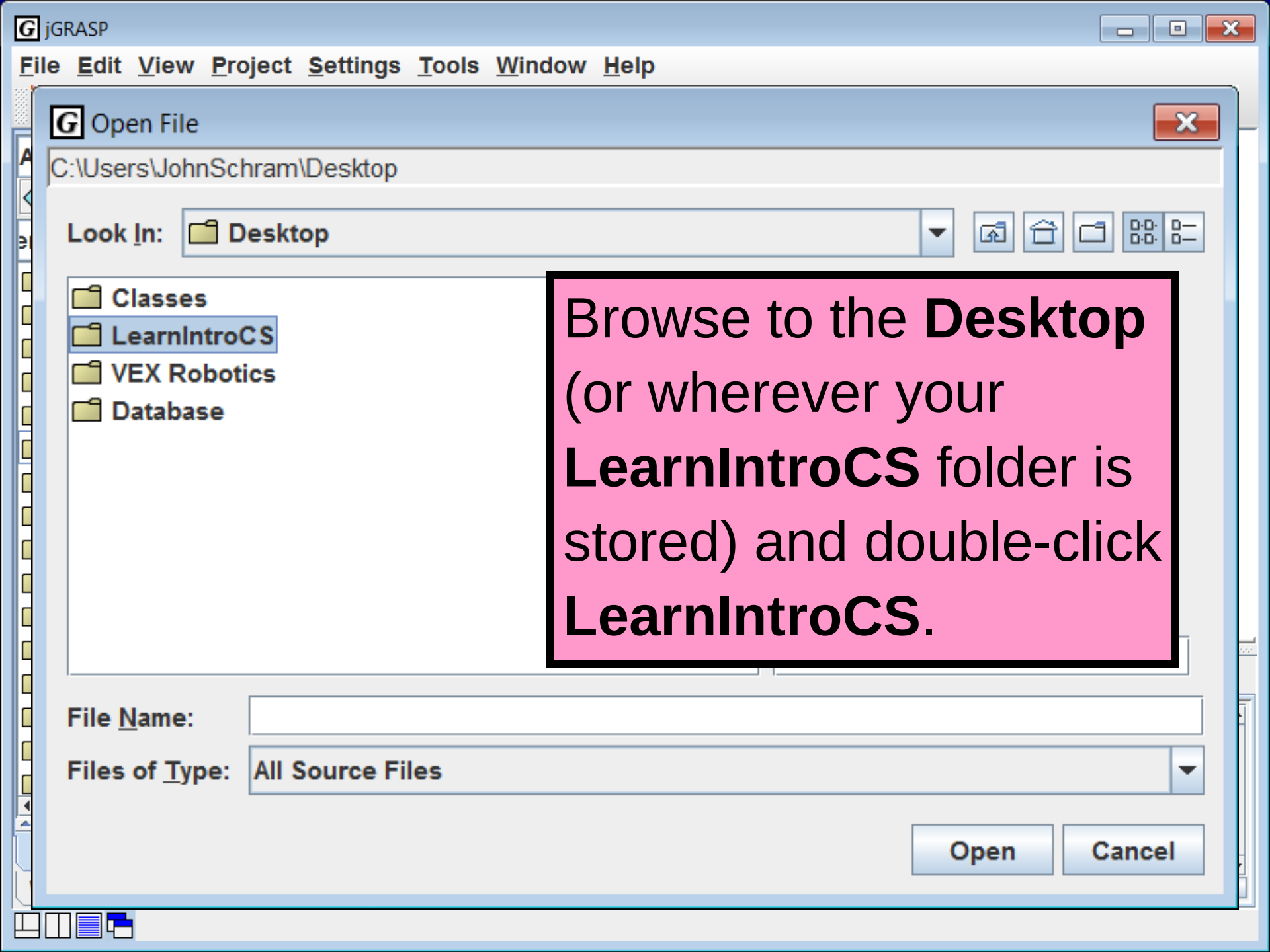


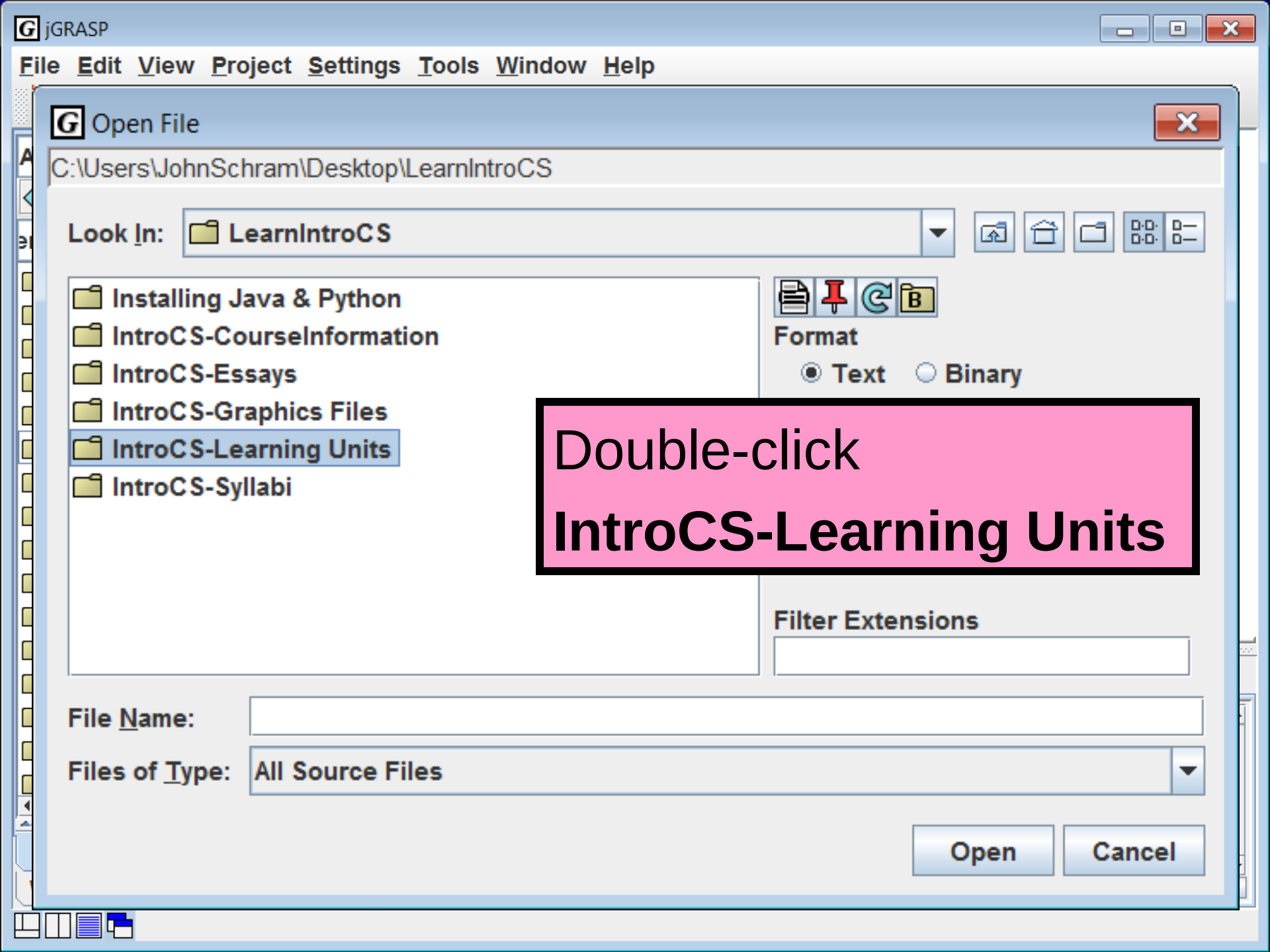
Organization of Program Examples in LearnIntroCS

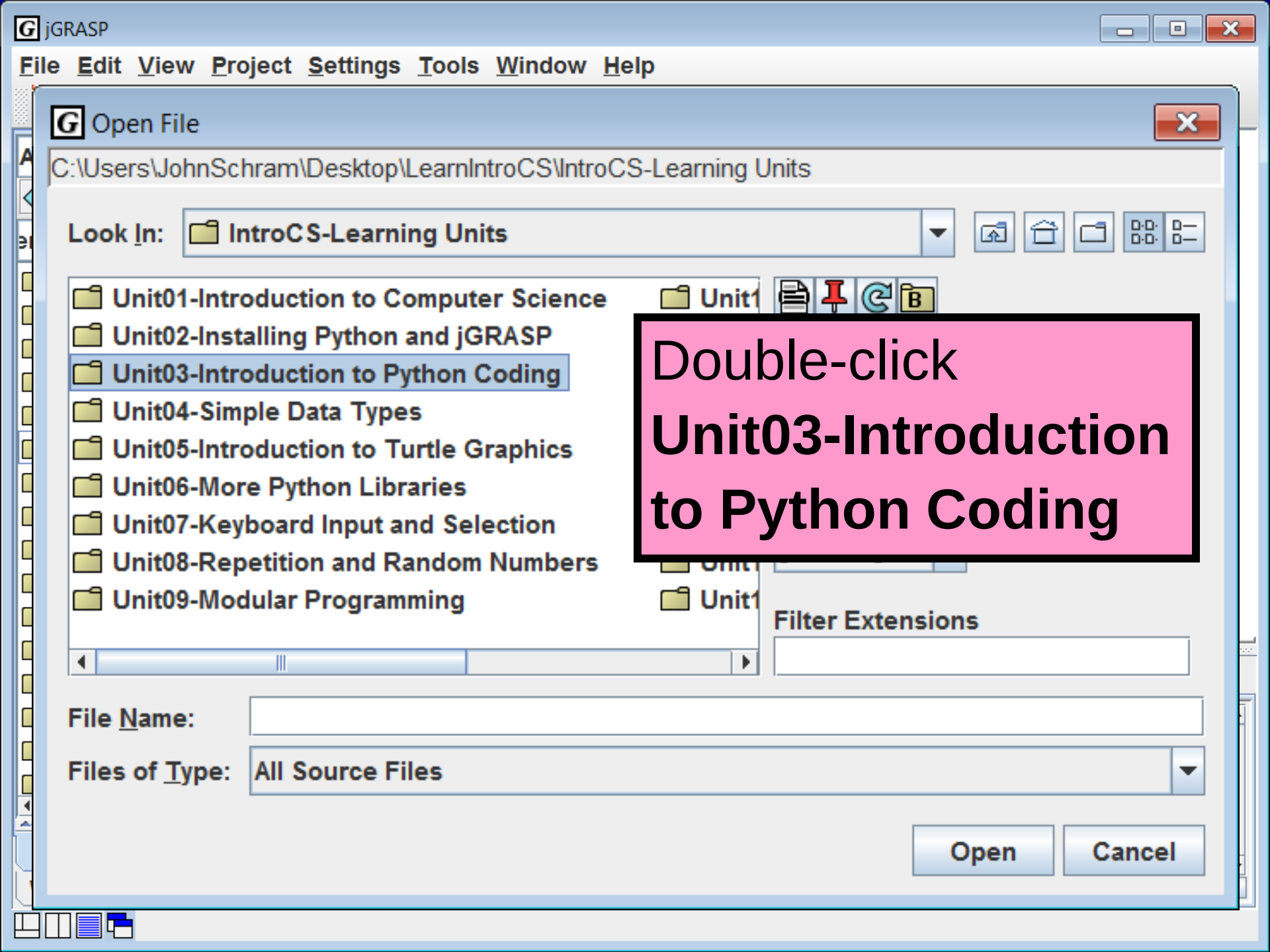


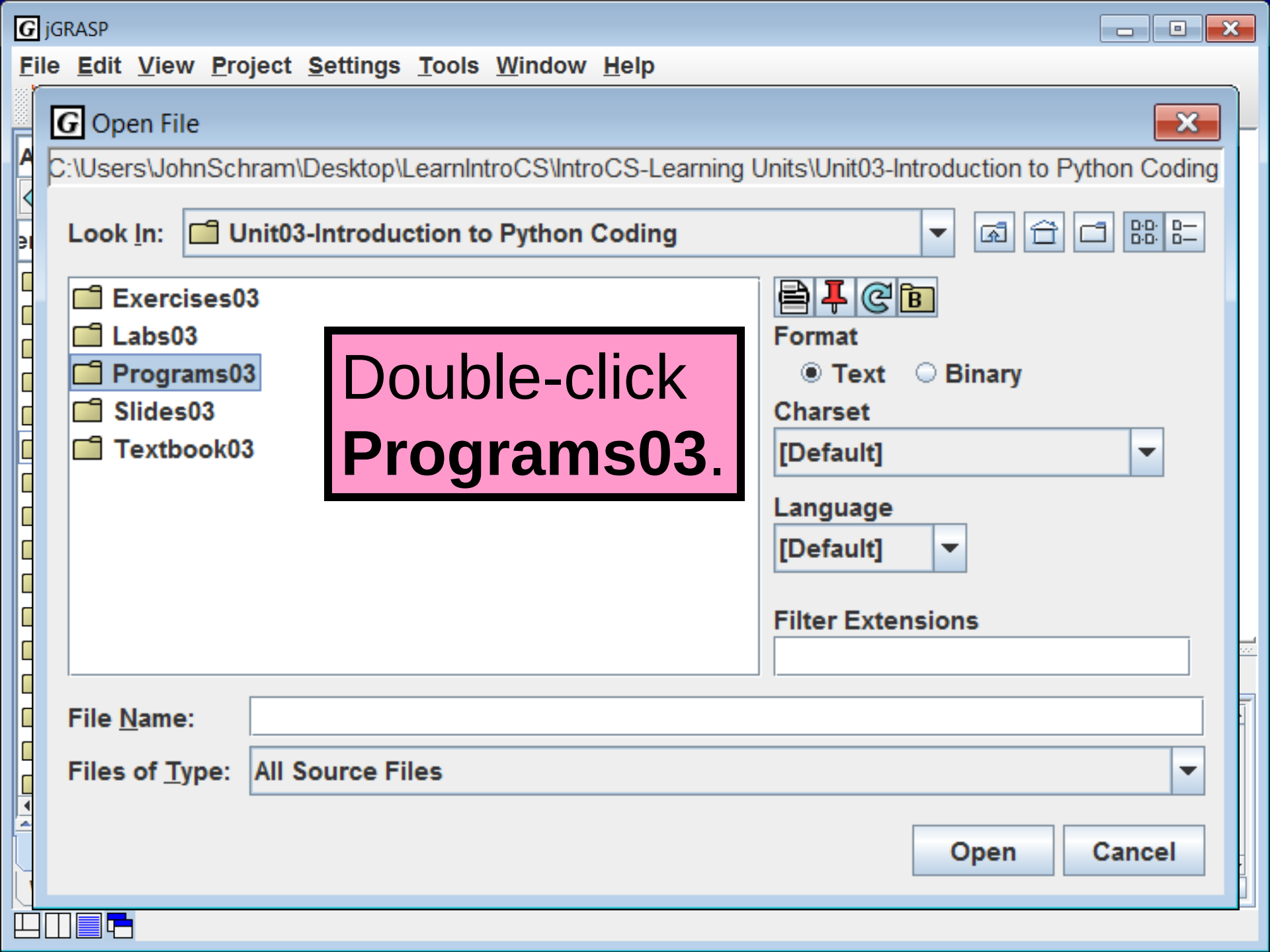
Loading, Executing & Editing Python Files in jGRASP

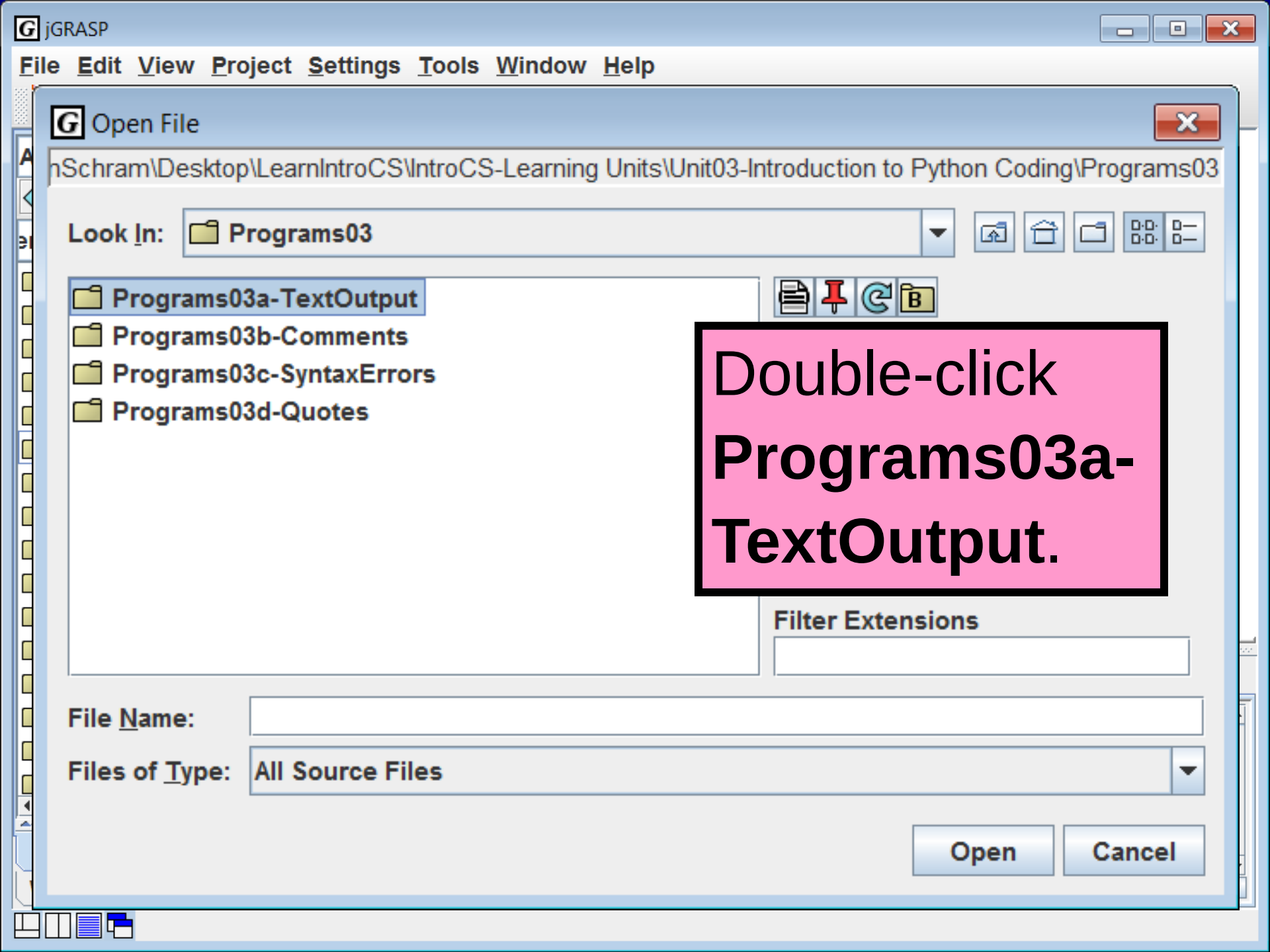

jGRASP
File Edit View Project Settings Tools Window Help
All Files
Sort By N...
ers\JohnSchram\Documents
ACDECPEN
Amazon MP3
Bluetooth
Downloads
DVD Architect Studio 5.0
DyKnow
Fax
LEGO Creations
microsoft
misc
Mortgage
Movie Studio Platinum 13
My Albums
My Notes
My Projects
OFX Presets
Browse
Find
Debug
Workbench
Click the Open File icon.
Compile Messages
jGRASP Messages
Run I/O
Interactions
Stop
Clear
Copy

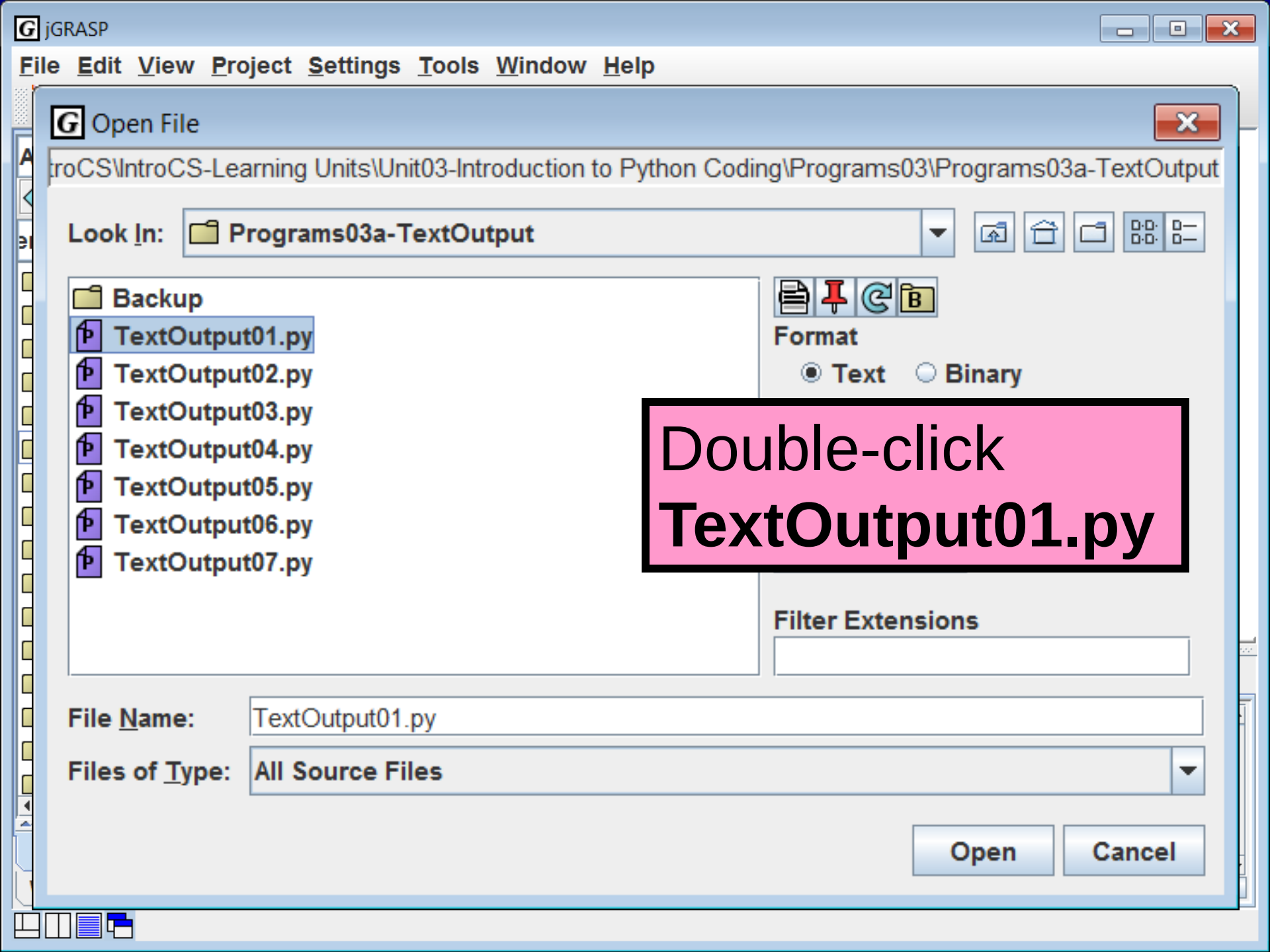

jGRASP
File Edit View Project Settings Tools Window Help
Open File
C:\Users\JohnSchram\Desktop
Look In: Desktop
Classes
LearnIntroCS
VEX Robotics
Database
Browse to the Desktop (or wherever your LearnIntroCS folder is stored) and double-click LearnIntroCS.
File Name:
Files of Type: All Source Files
Open
Cancel

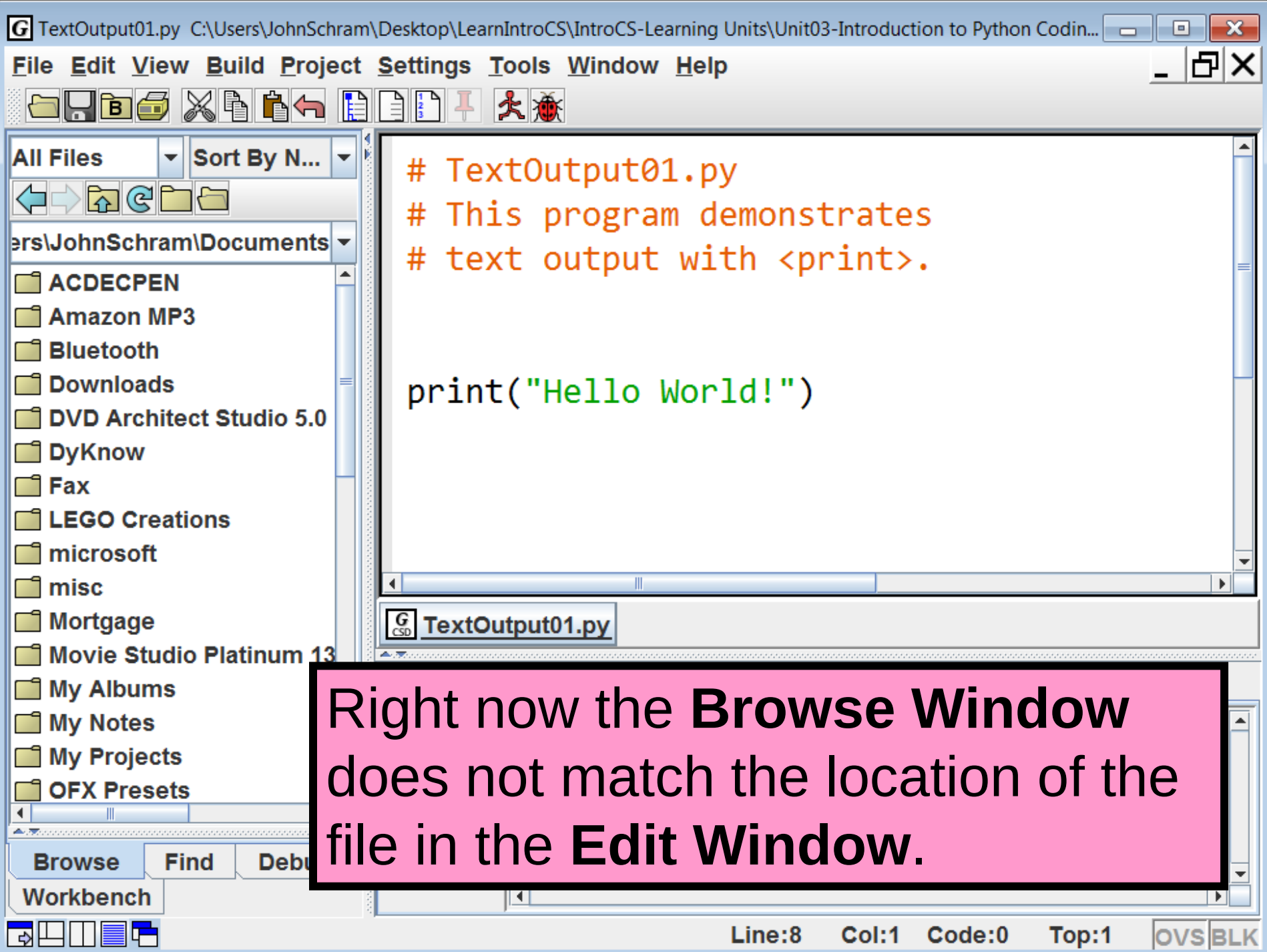

jGRASP
File Edit View Project Settings Tools Window Help
Open File
C:\Users\JohnSchram\Desktop\LearnIntroCS
Look In: LearnIntroCS
Installing Java & Python
IntroCS-CourseInformation
IntroCS-Essays
IntroCS-Graphics Files
IntroCS-Learning Units
IntroCS-Syllabi
Format
Text
Binary
Double-click
IntroCS-Learning Units
Filter Extensions
File Name:
Files of Type: All Source Files
Open
Cancel

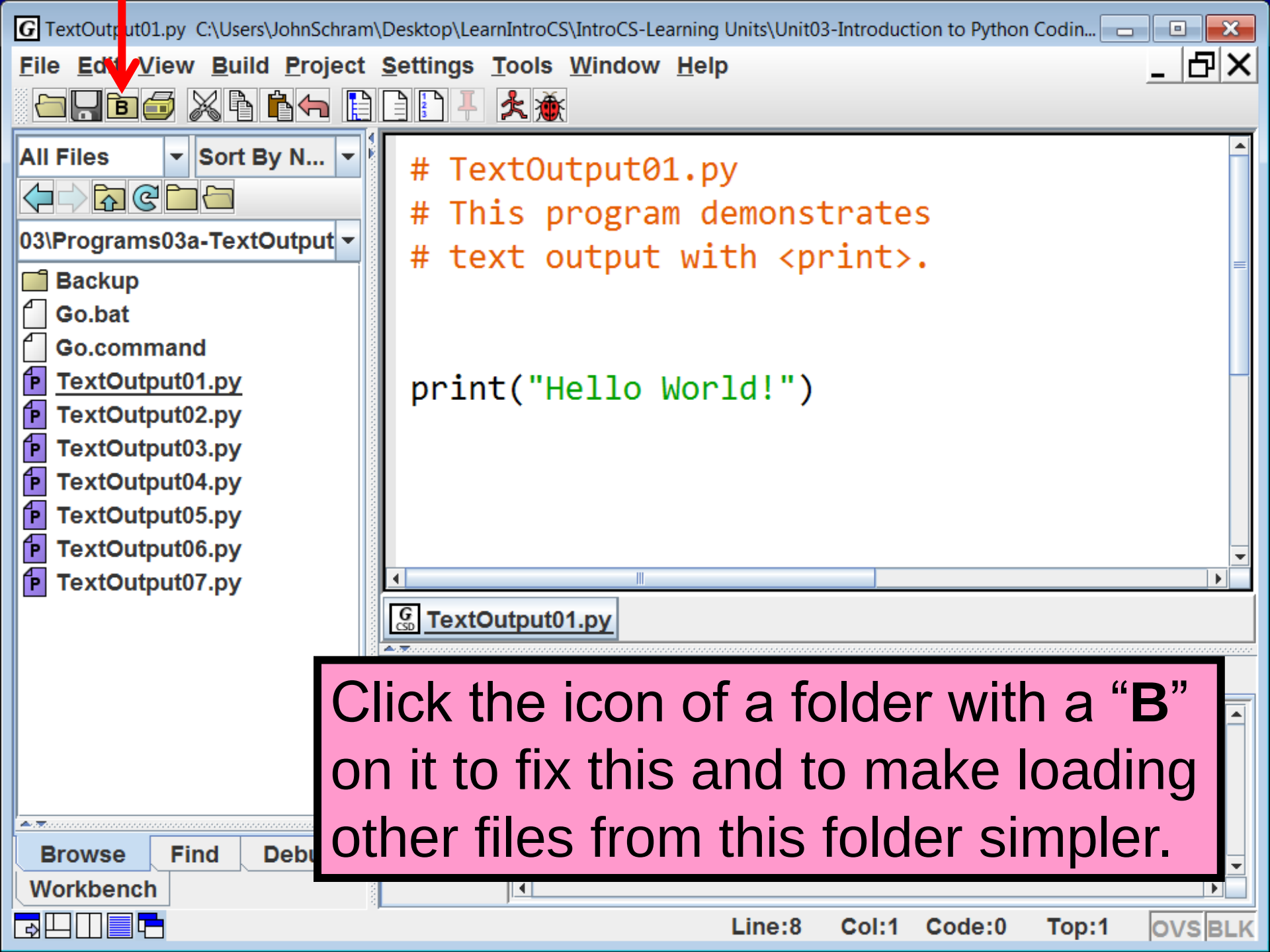

jGRASP
File Edit View Project Settings Tools Window Help
Open File
C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units
Look In:
IntroCS-Learning Units
Unit01-Introduction to Computer Science
Unit02-Installing Python and jGRASP
Unit03-Introduction to Python Coding
Unit04-Simple Data Types
Unit05-Introduction to Turtle Graphics
Unit06-More Python Libraries
Unit07-Keyboard Input and Selection
Unit08-Repetition and Random Numbers
Unit09-Modular Programming
Filter Extensions
File Name:
Files of Type:
All Source Files
Open
Cancel
Double-click
Unit03-Introduction to Python Coding

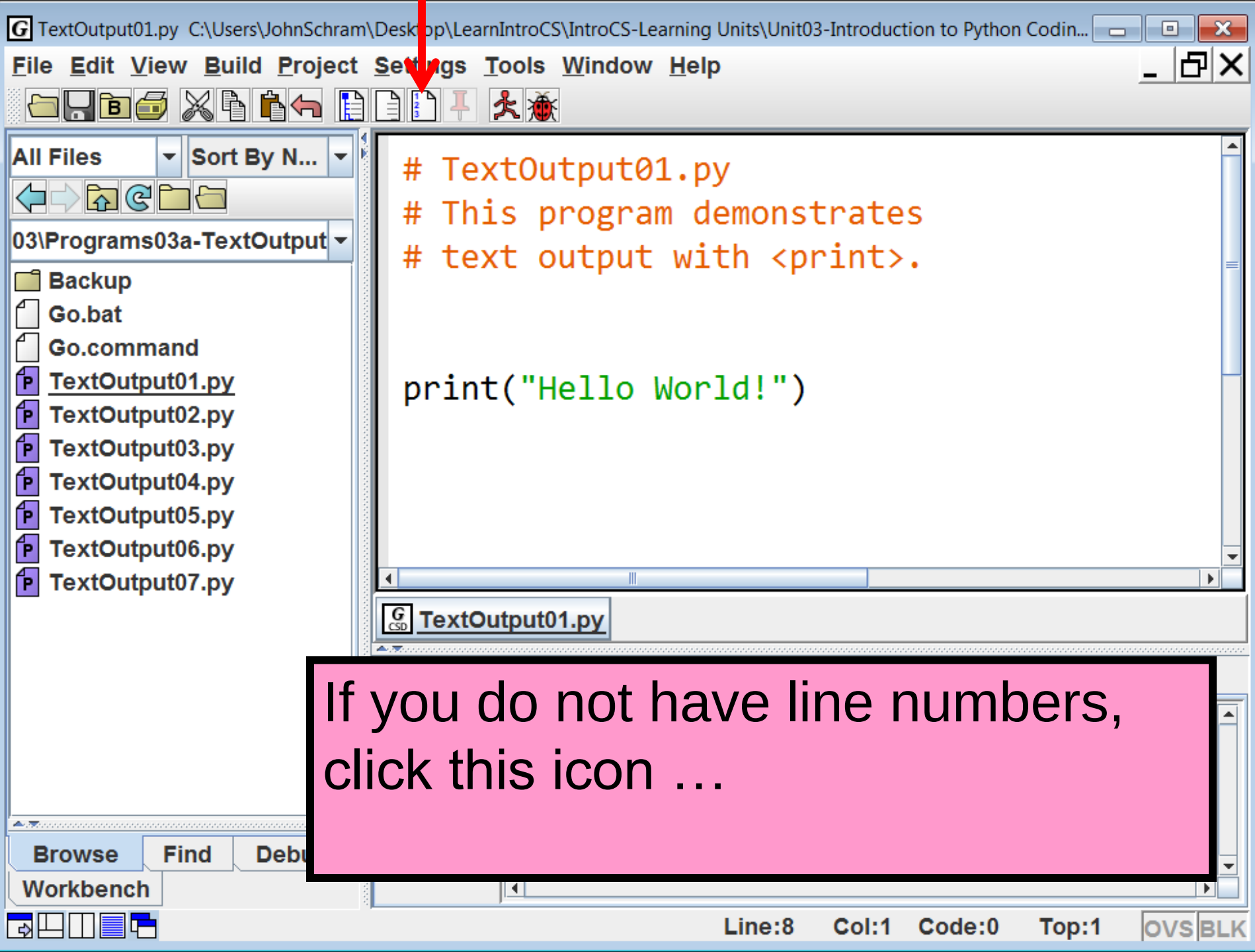

jGRASP
File Edit View Project Settings Tools Window Help
Open File
C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Coding
Look In:
Unit03-Introduction to Python Coding
Exercises03
Labs03
Programs03
Slides03
Textbook03
Double-click
Programs03.
Format
Text
Binary
Charset
[Default]
Language
[Default]
Filter Extensions
File Name:
Files of Type:
All Source Files
Open
Cancel

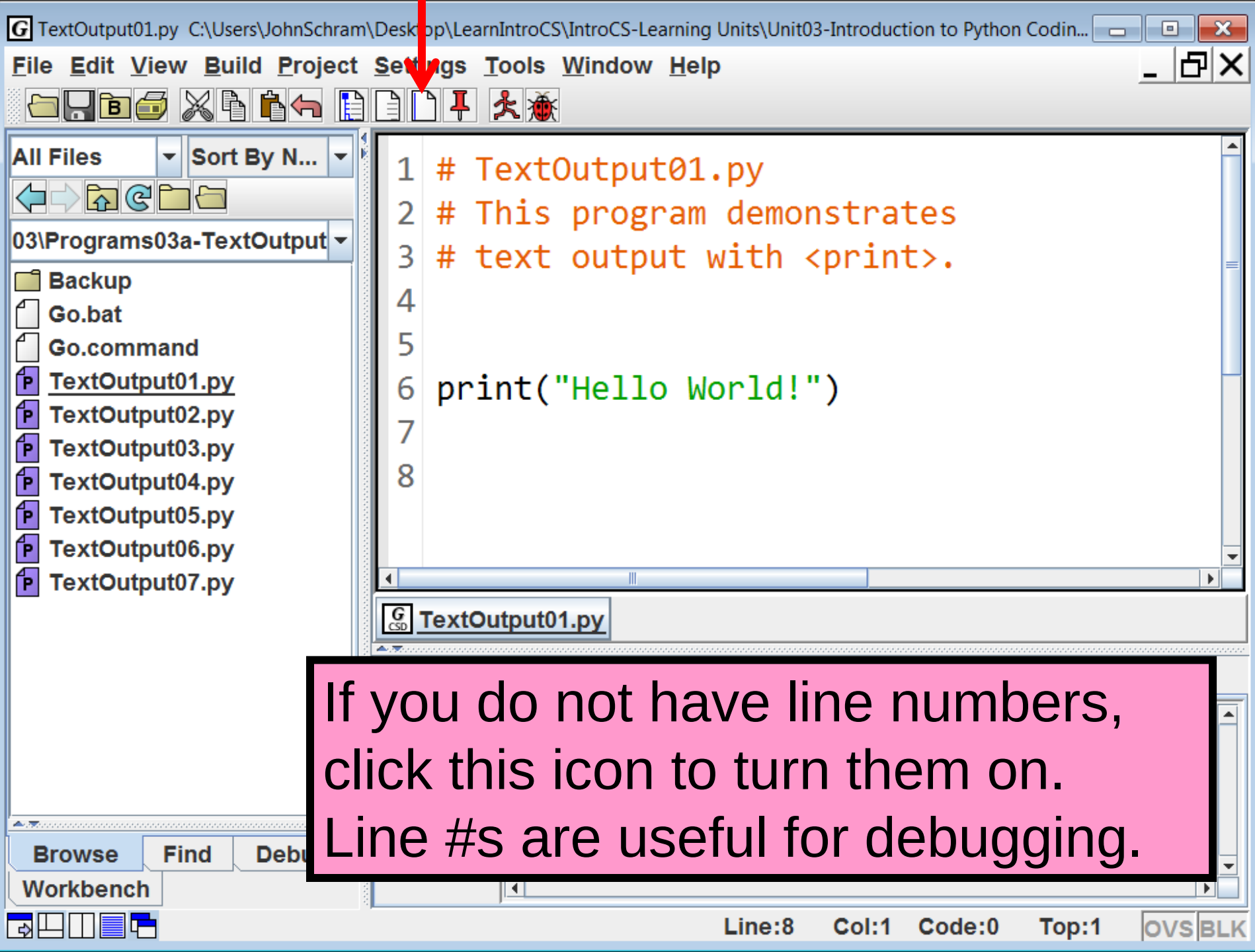

jGRASP
File Edit View Project Settings Tools Window Help
Open File
nSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Coding\Programs03
Look In: Programs03
Programs03a-TextOutput
Programs03b-Comments
Programs03c-SyntaxErrors
Programs03d-Quotes
Double-click
Programs03a-
TextOutput.
Filter Extensions
File Name:
Files of Type: All Source Files
Open
Cancel

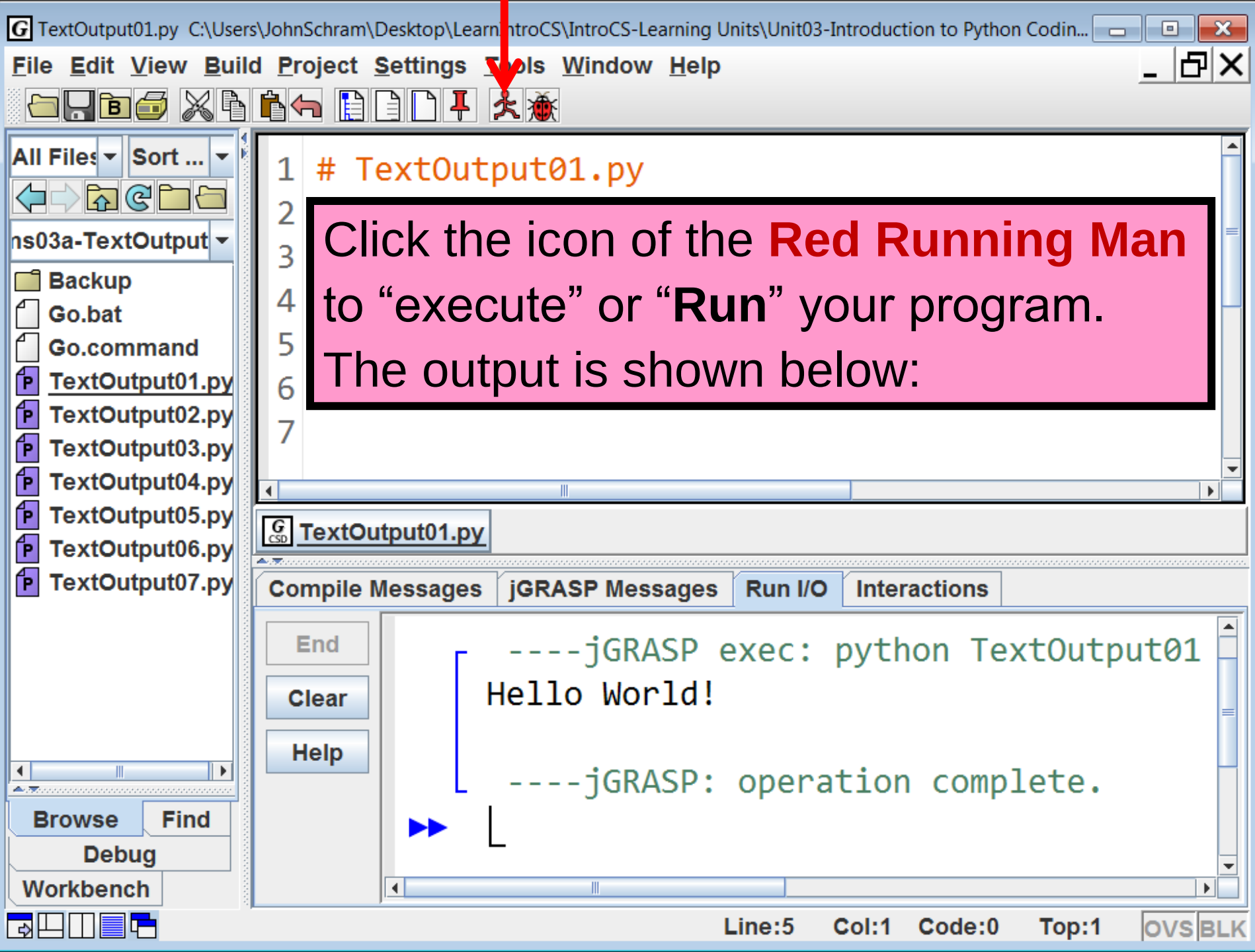

jGRASP
File Edit View Project Settings Tools Window Help
Open File
troCS\IntroCS-Learning Units\Unit03-Introduction to Python Coding\Programs03\Programs03a-TextOutput
Look In: Programs03a-TextOutput
Backup
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
Format
Text
Binary
Double-click
TextOutput01.py
Filter Extensions
File Name: TextOutput01.py
Files of Type: All Source Files
Open
Cancel

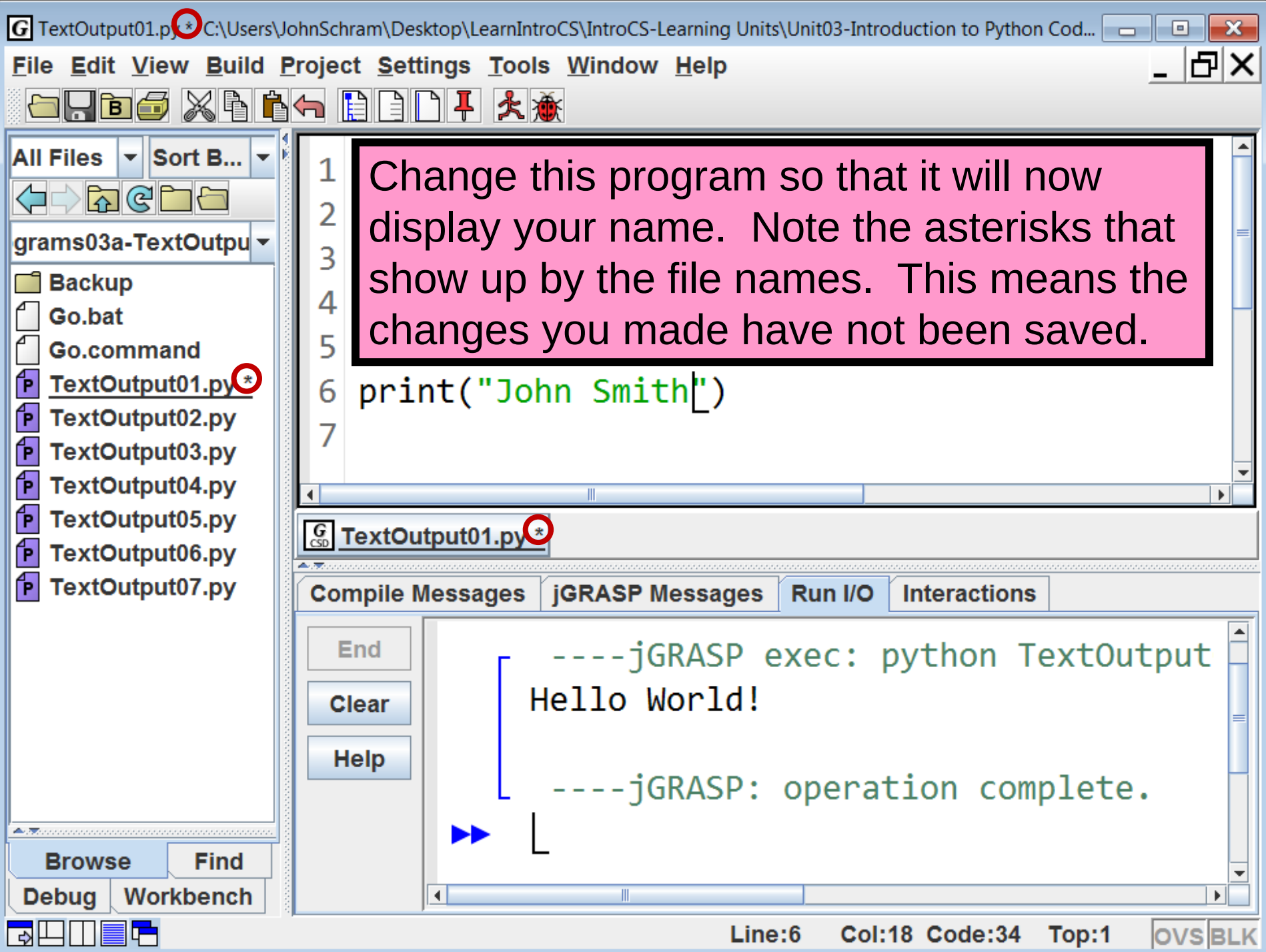

TextOutput01.py C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Codin...
File Edit View Build Project Settings Tools Window Help
All Files
Sort By N...
ers\JohnSchram\Documents
ACDECPEN
Amazon MP3
Bluetooth
Downloads
DVD Architect Studio 5.0
DyKnow
Fax
LEGO Creations
microsoft
misc
Mortgage
Movie Studio Platinum 13
My Albums
My Notes
My Projects
OFX Presets
Browse
Find
Debu
Workbench
# TextOutput01.py
# This program demonstrates
# text output with <print>.
print("Hello World!")
TextOutput01.py
Right now the Browse Window does not match the location of the file in the Edit Window.
Line:8 Col:1 Code:0 Top:1
OVS BLK

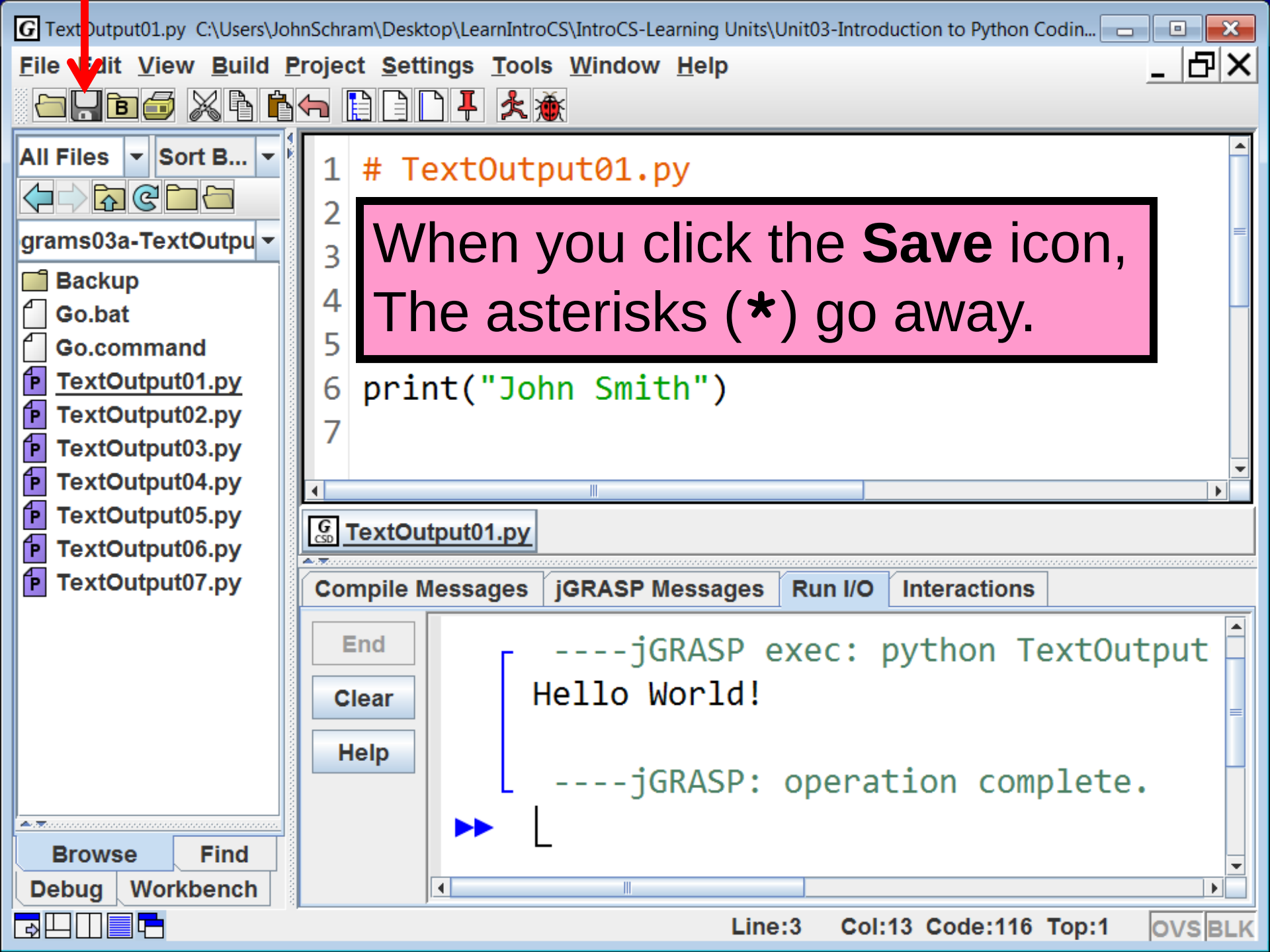

TextOutput01.py C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Codin...
File Edit View Build Project Settings Tools Window Help
All Files
Sort By N...
03\Programs03a-TextOutput
Backup
Go.bat
Go.command
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
# TextOutput01.py
# This program demonstrates
# text output with <print>.
print("Hello World!")
TextOutput01.py
Browse
Find
Workbench
Line:8 Col:1 Code:0 Top:1
OVS BLK
Click the icon of a folder with a "B" on it to fix this and to make loading other files from this folder simpler.

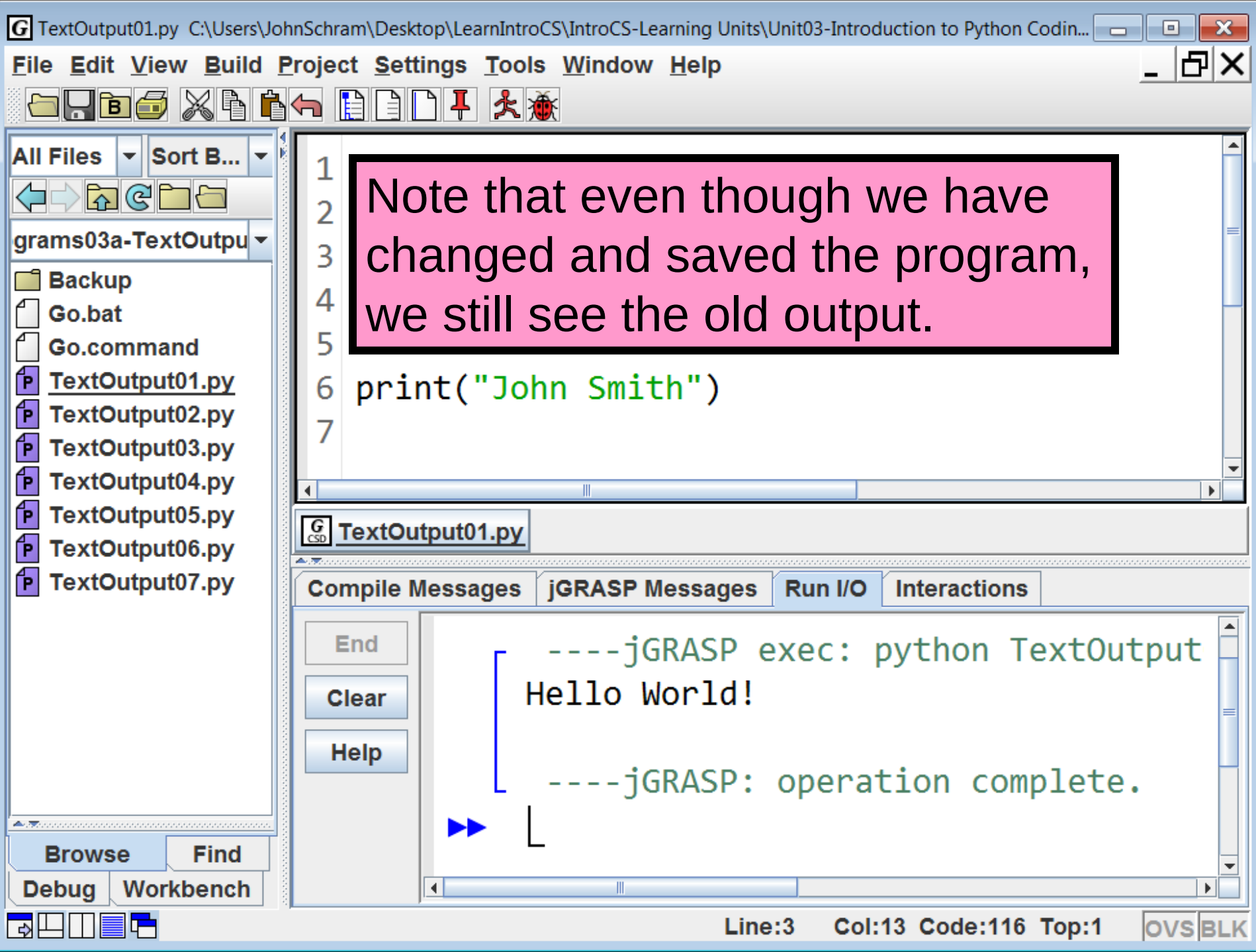

TextOutput01.py C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Codin...
File Edit View Build Project Settings Tools Window Help
All Files
Sort By N...
03\Programs03a-TextOutput
Backup
Go.bat
Go.command
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
# TextOutput01.py
# This program demonstrates
# text output with <print>.
print("Hello World!")
TextOutput01.py
If you do not have line numbers, click this icon …
Browse
Find
Workbench
Line:8
Col:1
Code:0
Top:1
OVS
BLK

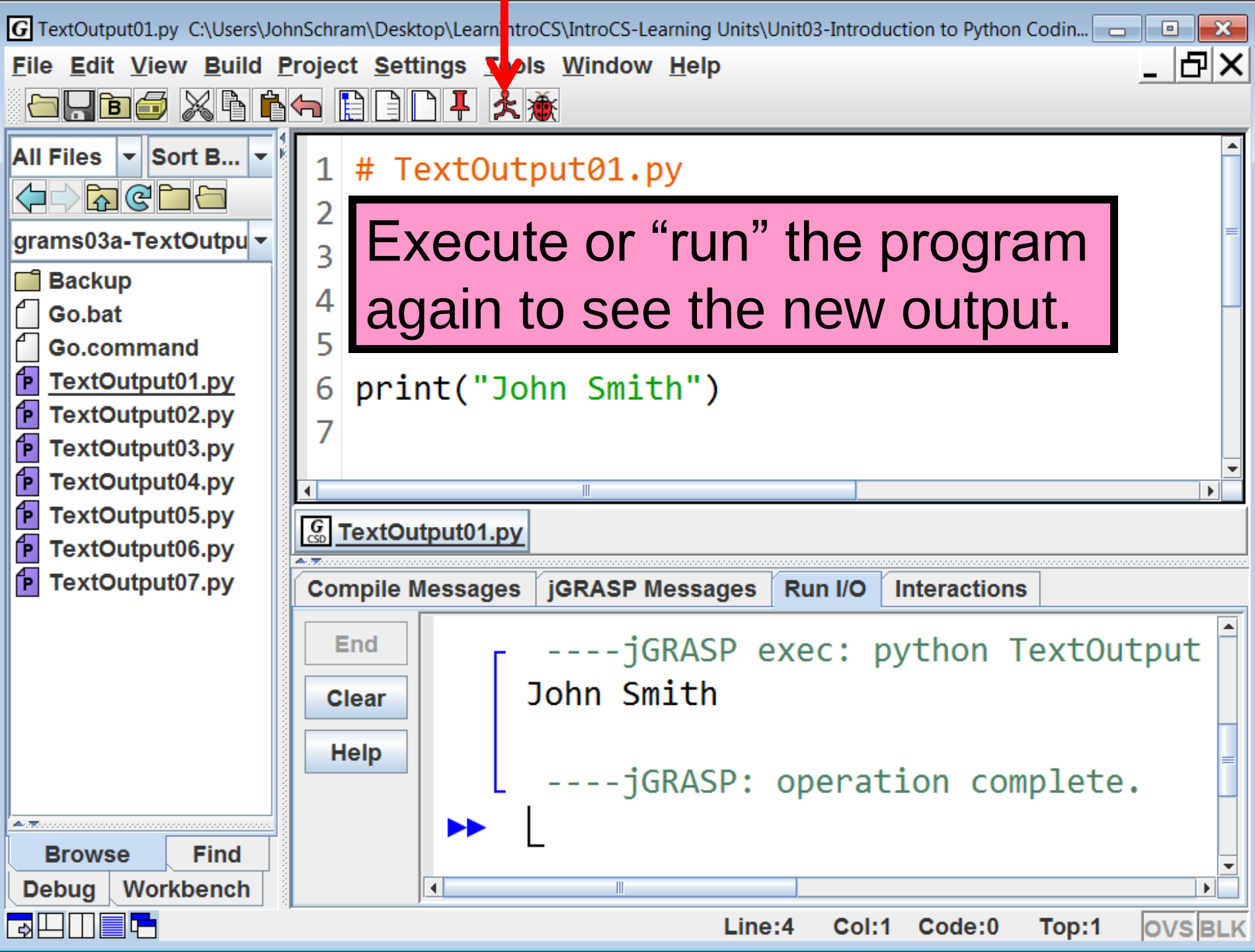

TextOutput01.py C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Codin...
File Edit View Build Project Settings Tools Window Help
All Files
Sort By N...
03\Programs03a-TextOutput
Backup
Go.bat
Go.command
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
1 # TextOutput01.py
2 # This program demonstrates
3 # text output with <print>.
4
5
6 print("Hello World!")
7
8
TextOutput01.py
If you do not have line numbers, click this icon to turn them on. Line #s are useful for debugging.
Browse
Find
Workbench
Line:8 Col:1 Code:0 Top:1
OVS BLK

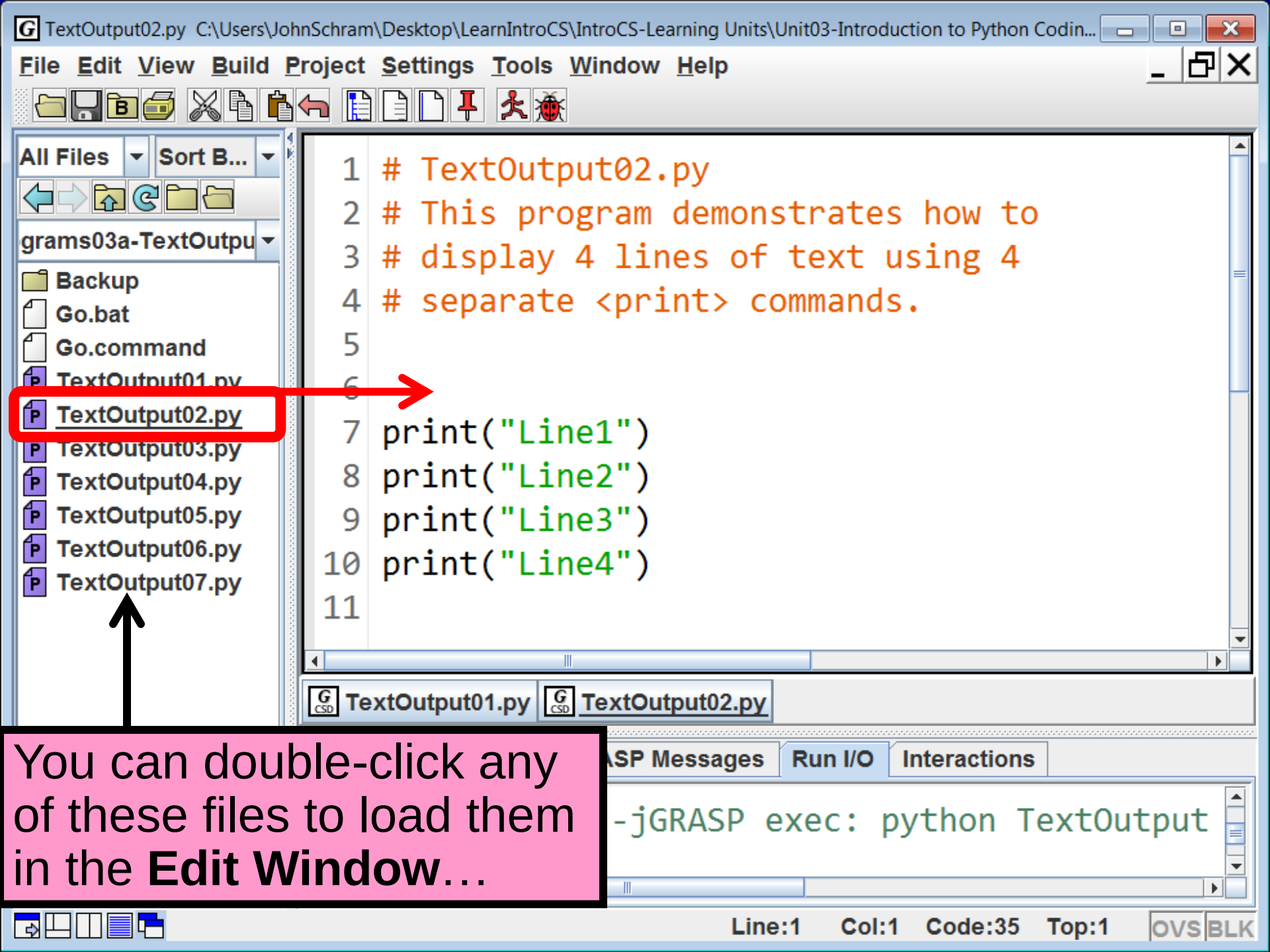

TextOutput01.py C:\Users\JohnSchram\Desktop\Learn ntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Codin...
File Edit View Build Project Settings Tools Window Help
All Files
Sort ...
ns03a-TextOutput
Backup
Go.bat
Go.command
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
Browse
Find
Debug
Workbench
1 # TextOutput01.py
2
3 Click the icon of the Red Running Man
4 to "execute" or "Run" your program.
5 The output is shown below:
6
7
TextOutput01.py
Compile Messages
jGRASP Messages
Run I/O
Interactions
End
Clear
Help
----jGRASP exec: python TextOutput01
Hello World!
----jGRASP: operation complete.
Line:5 Col:1 Code:0 Top:1
OVS BLK

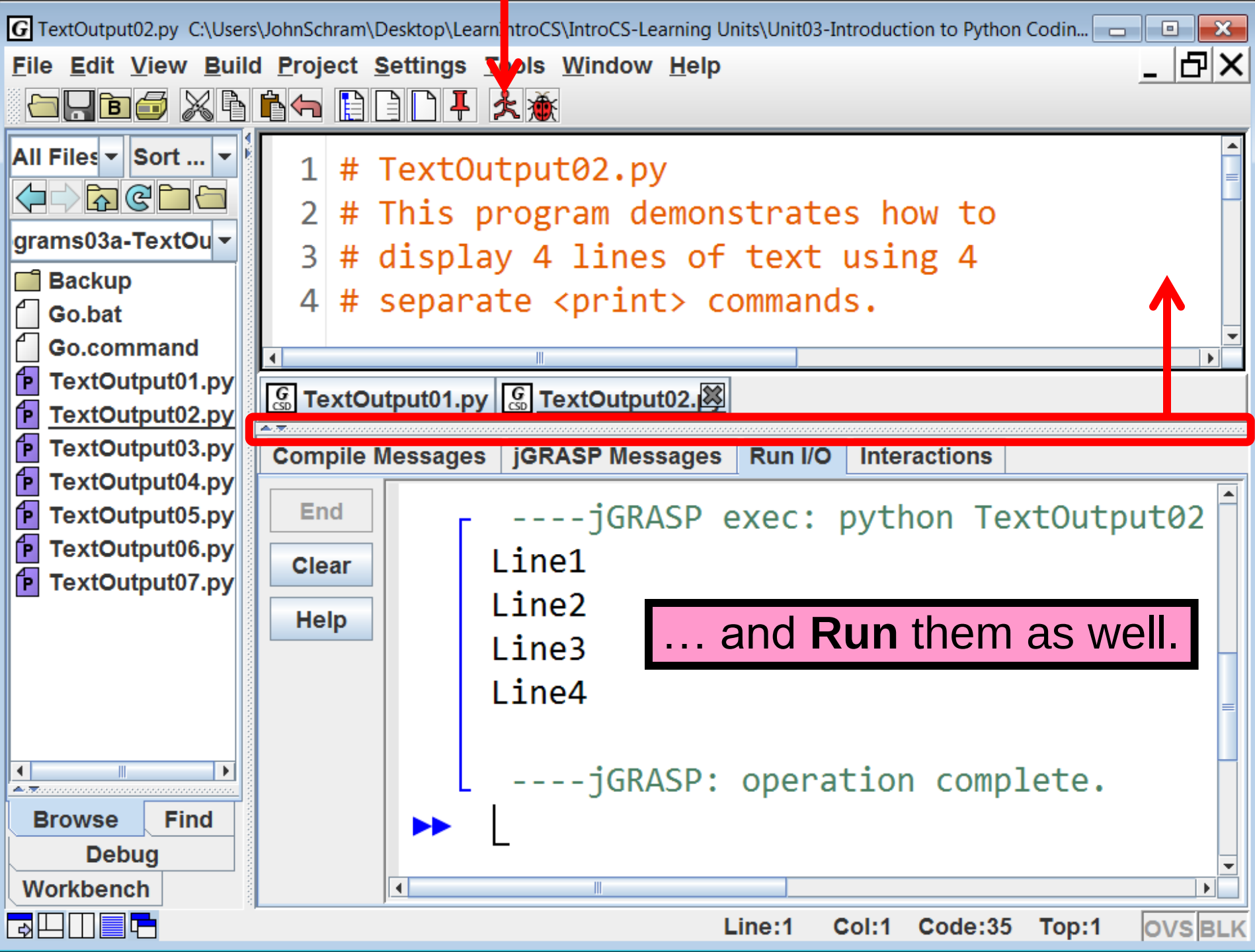

TextOutput01.py *
C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Cod...
File Edit View Build Project Settings Tools Window Help
All Files
Sort B...
grams03a-TextOutpu
Backup
Go.bat
Go.command
TextOutput01.py *
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
Browse
Find
Debug
Workbench
Change this program so that it will now display your name. Note the asterisks that show up by the file names. This means the changes you made have not been saved.
1
2
3
4
5
6
7
print("John Smith")
TextOutput01.py *
Compile Messages
jGRASP Messages
Run I/O
Interactions
End
Clear
Help
----jGRASP exec: python TextOutput
Hello World!
----jGRASP: operation complete.
Line:6
Col:18
Code:34
Top:1
OVS
BLK

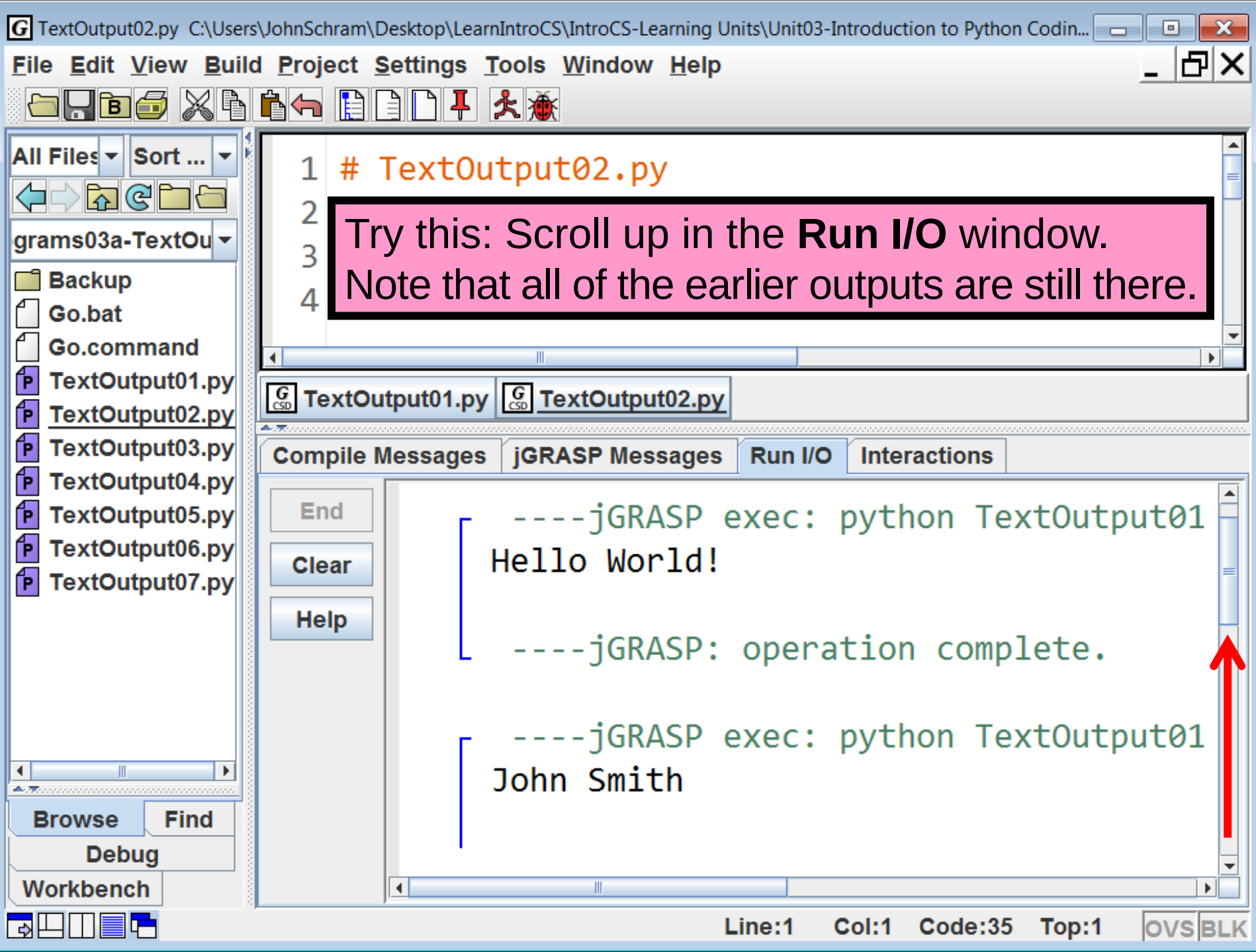

TextOutput01.py C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Codin...
File Edit View Build Project Settings Tools Window Help
All Files
Sort B...
grams03a-TextOutpu
Backup
Go.bat
Go.command
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
Browse
Find
Debug
Workbench
1 # TextOutput01.py
2
3
4
5
6 print("John Smith")
7
When you click the Save icon,
The asterisks (*) go away.
TextOutput01.py
Compile Messages
jGRASP Messages
Run I/O
Interactions
End
Clear
Help
----jGRASP exec: python TextOutput
Hello World!
----jGRASP: operation complete.
Line:3 Col:13 Code:116 Top:1
OVS BLK

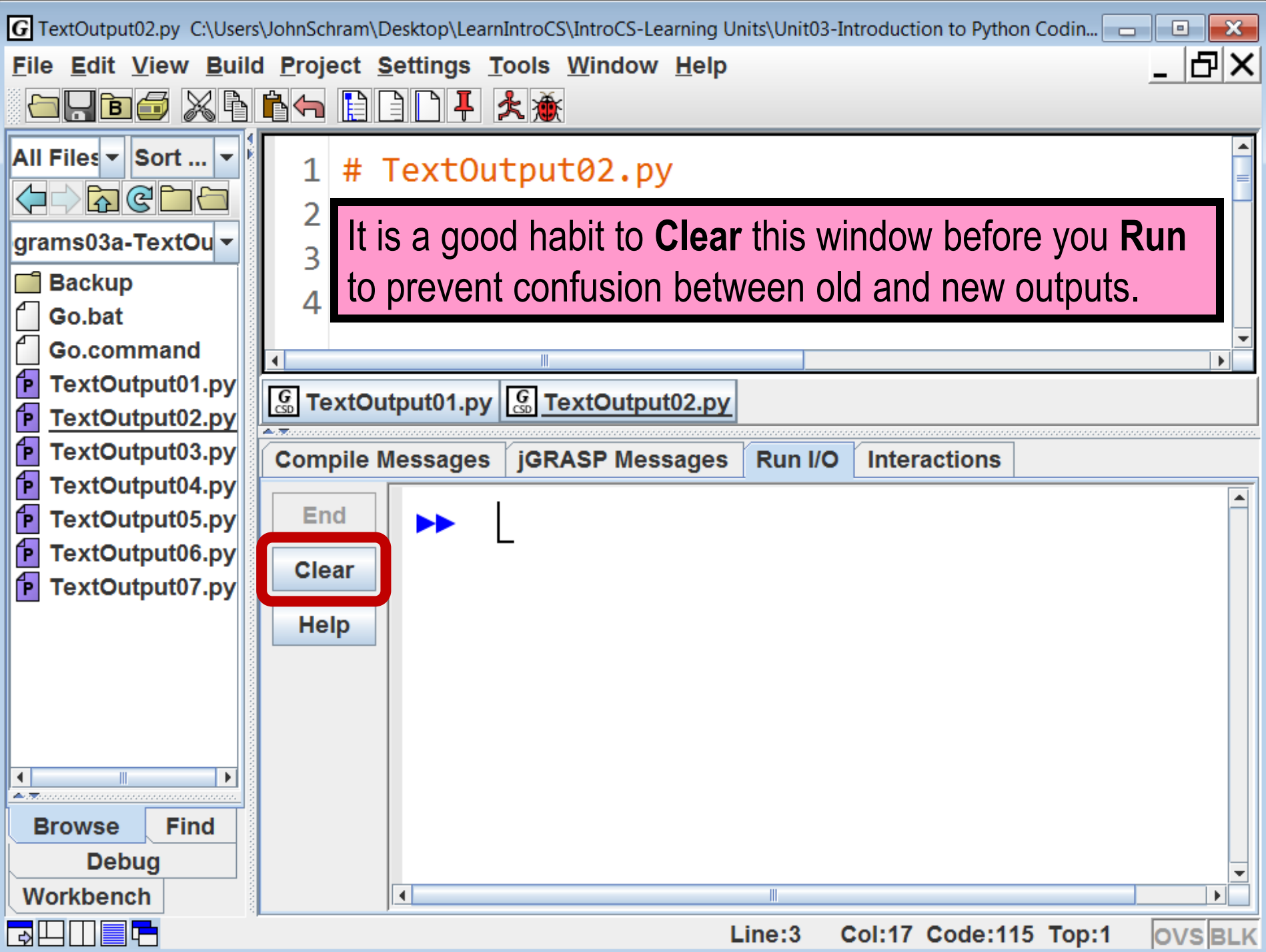

TextOutput01.py C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Codin...
File Edit View Build Project Settings Tools Window Help
All Files
Sort B...
grams03a-TextOutpu
Backup
Go.bat
Go.command
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
Browse
Find
Debug
Workbench
1
2
3
4
5
6
7
print("John Smith")
Note that even though we have changed and saved the program, we still see the old output.
TextOutput01.py
Compile Messages
jGRASP Messages
Run I/O
Interactions
End
Clear
Help
----jGRASP exec: python TextOutput
Hello World!
----jGRASP: operation complete.
Line:3 Col:13 Code:116 Top:1
OVS BLK

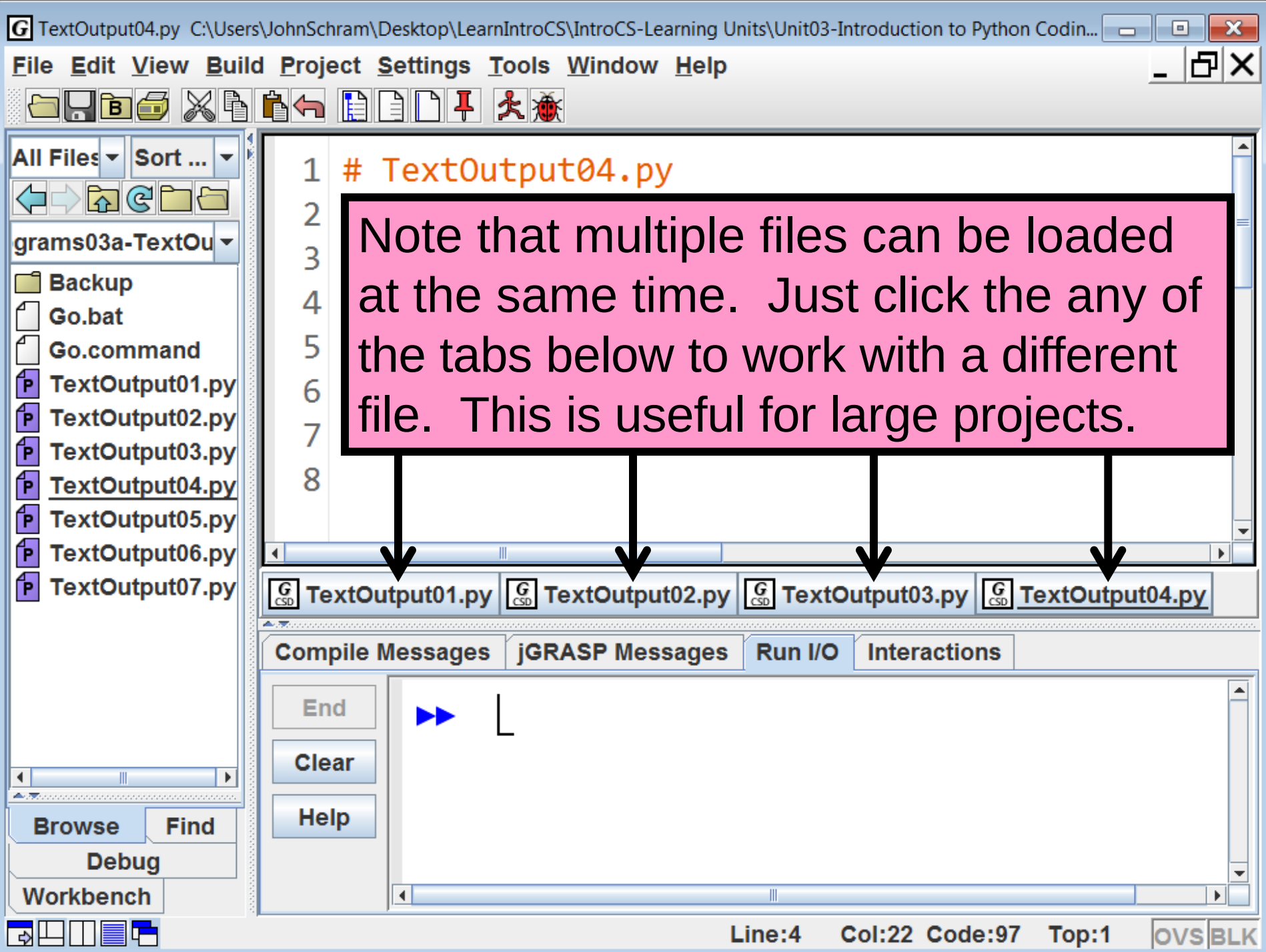

TextOutput01.py C:\Users\JohnSchram\Desktop\Learn ntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Codin...
File Edit View Build Project Settings Tools Window Help
All Files
Sort B...
grams03a-TextOutpu
Backup
Go.bat
Go.command
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
Browse
Find
Debug
Workbench
1 # TextOutput01.py
2
3
4
5
6 print("John Smith")
7
Execute or “run” the program again to see the new output.
TextOutput01.py
Compile Messages
jGRASP Messages
Run I/O
Interactions
End
Clear
Help
----jGRASP exec: python TextOutput
John Smith
----jGRASP: operation complete.
Line:4 Col:1 Code:0 Top:1
OVS BLK

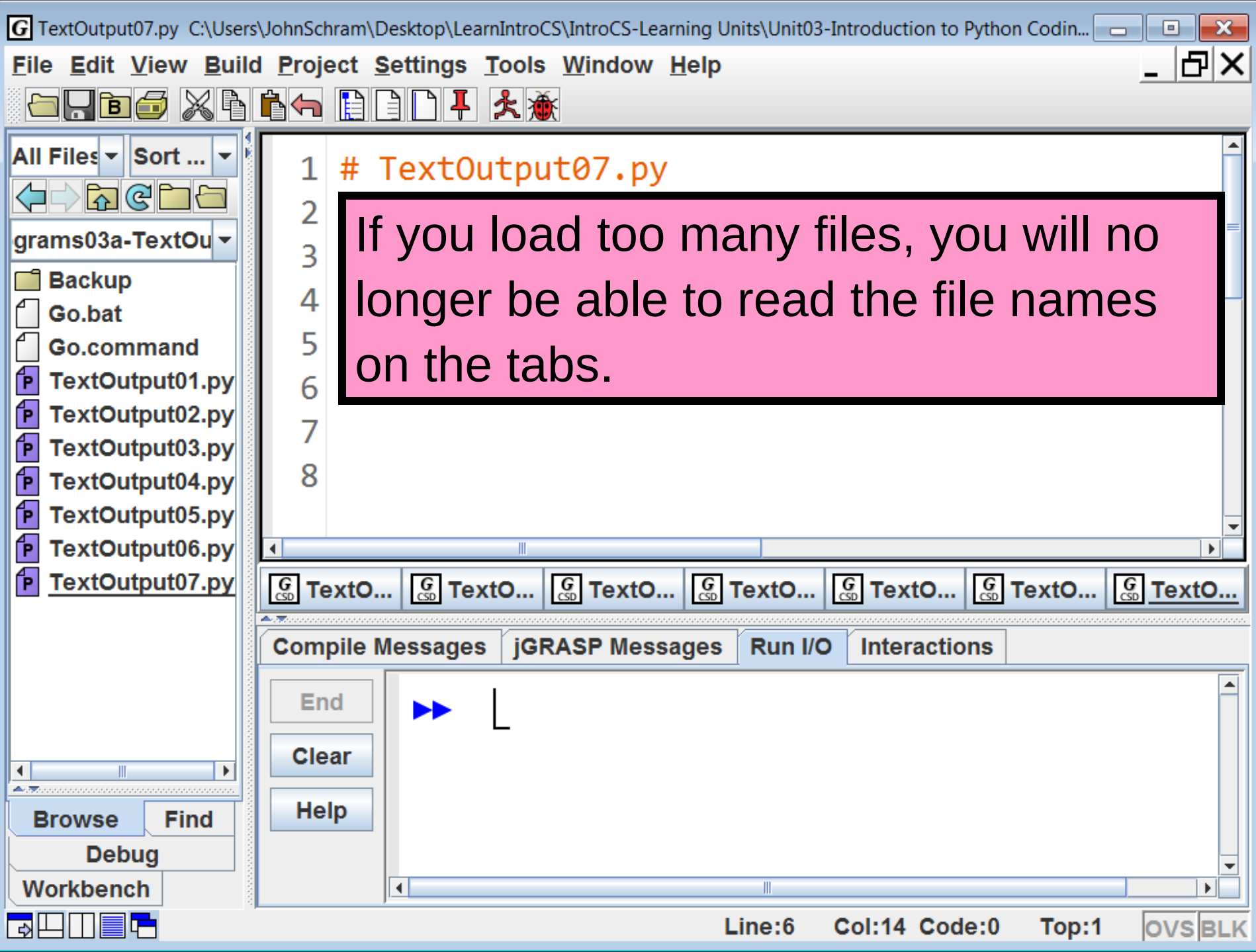

TextOutput02.py C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Codin...
File Edit View Build Project Settings Tools Window Help
All Files
Sort B...
grams03a-TextOutpu
Backup
Go.bat
Go.command
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
1 # TextOutput02.py
2 # This program demonstrates how to
3 # display 4 lines of text using 4
4 # separate <print> commands.
5
6
7 print("Line1")
8 print("Line2")
9 print("Line3")
10 print("Line4")
11
TextOutput01.py
TextOutput02.py
ASP Messages
Run I/O
Interactions
-jGRASP exec: python TextOutput
You can double-click any of these files to load them in the **Edit Window**…
Line:1 Col:1 Code:35 Top:1
OVS BLK

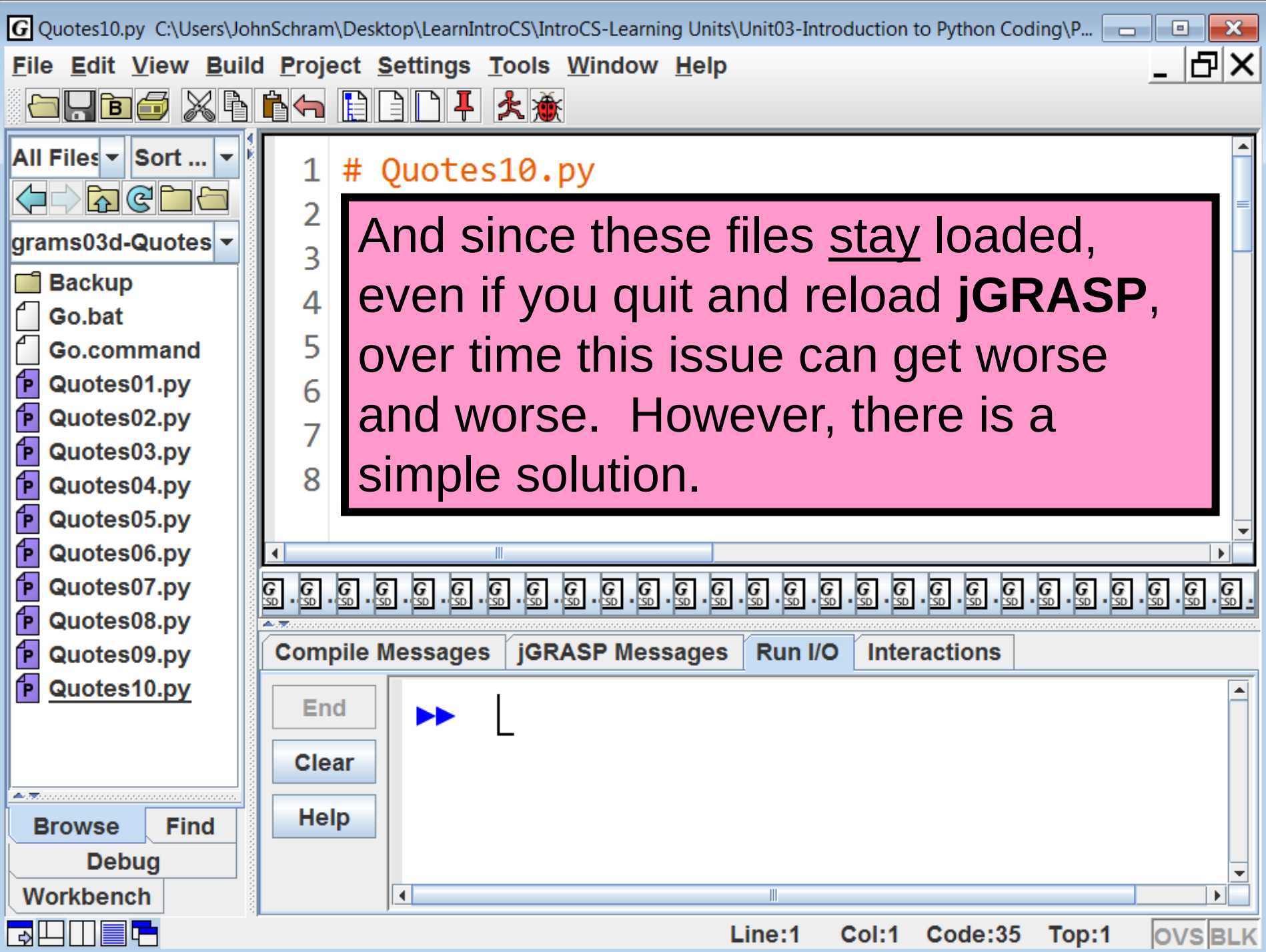

TextOutput02.py C:\Users\JohnSchram\Desktop\Learn ntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Codin...
File Edit View Build Project Settings Tools Window Help
All Files
Sort ...
grams03a-TextOu
Backup
Go.bat
Go.command
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
1 # TextOutput02.py
2 # This program demonstrates how to
3 # display 4 lines of text using 4
4 # separate <print> commands.
TextOutput01.py
TextOutput02.
Compile Messages
jGRASP Messages
Run I/O
Interactions
End
Clear
Help
----jGRASP exec: python TextOutput02
Line1
Line2
Line3
Line4
----jGRASP: operation complete.
… and Run them as well.
Browse
Find
Debug
Workbench
Line:1
Col:1
Code:35
Top:1
OVS
BLK

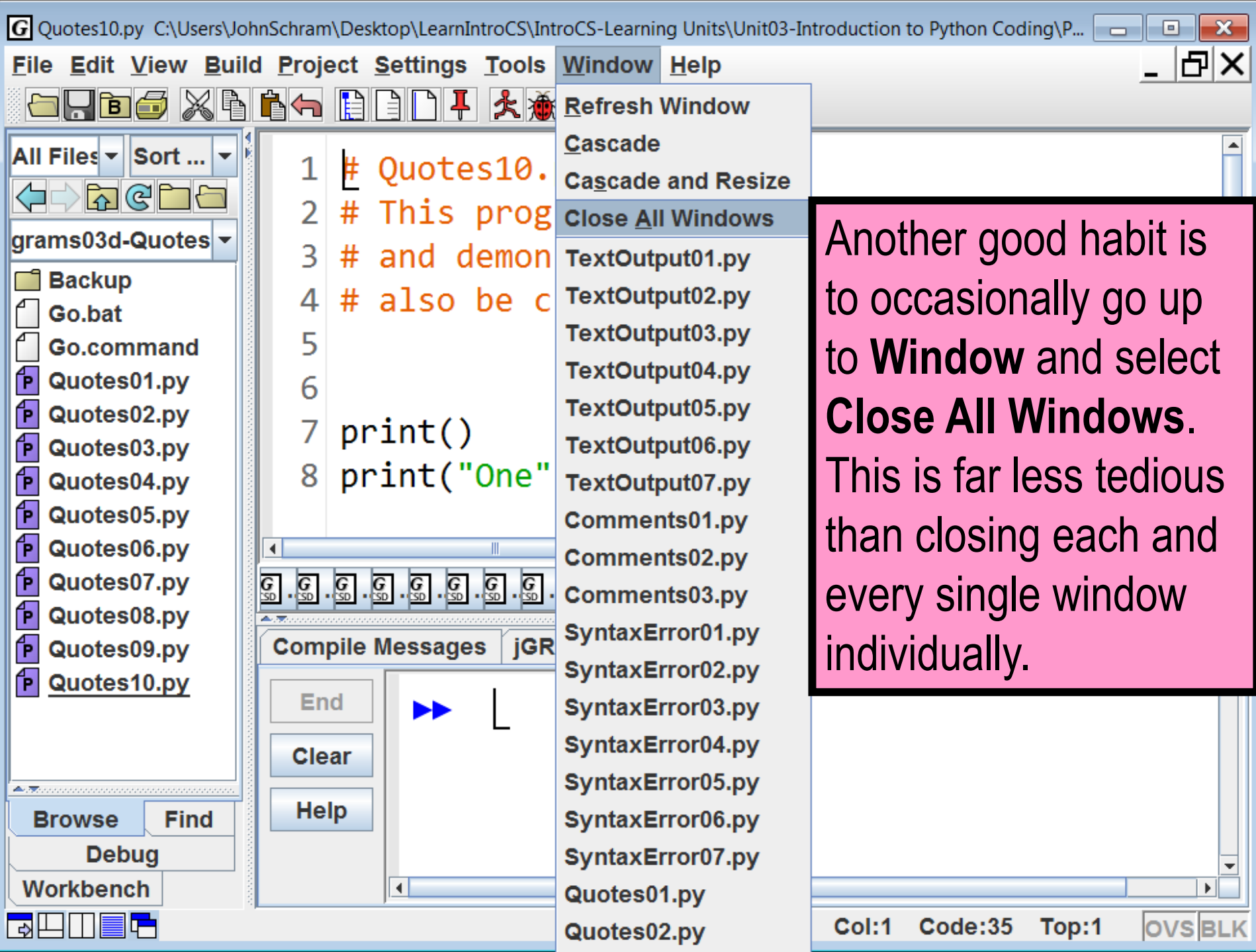

TextOutput02.py C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Codin...
File Edit View Build Project Settings Tools Window Help
All Files
Sort ...
grams03a-TextOu
Backup
Go.bat
Go.command
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
Browse
Find
Debug
Workbench
1 # TextOutput02.py
2
3
4
Try this: Scroll up in the Run I/O window.
Note that all of the earlier outputs are still there.
TextOutput01.py
TextOutput02.py
Compile Messages
jGRASP Messages
Run I/O
Interactions
End
Clear
Help
----jGRASP exec: python TextOutput01
Hello World!
----jGRASP: operation complete.
----jGRASP exec: python TextOutput01
John Smith
Line:1 Col:1 Code:35 Top:1
OVS BLK

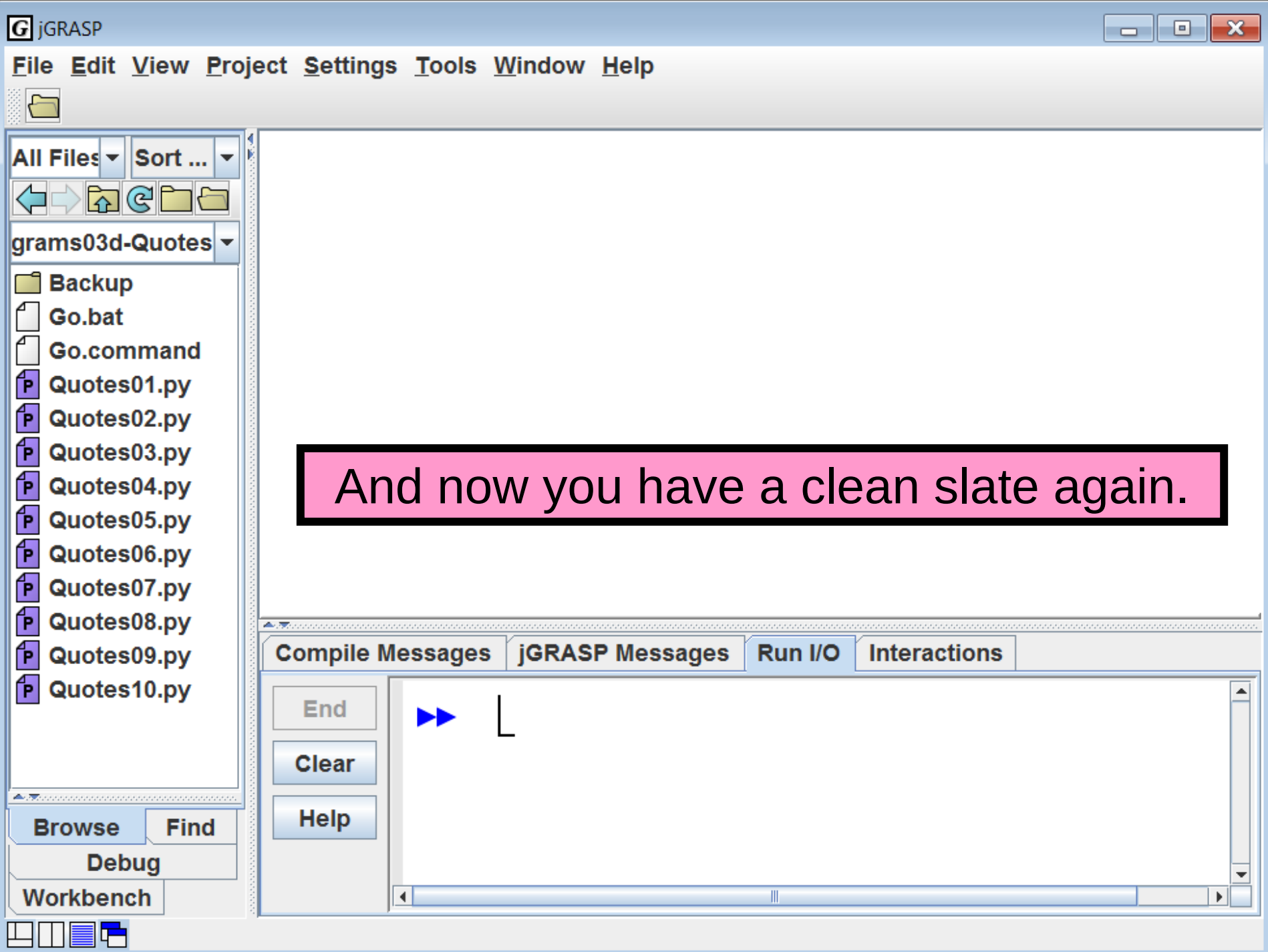

TextOutput02.py C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Codin...
File Edit View Build Project Settings Tools Window Help
All Files
Sort ...
grams03a-TextOu
Backup
Go.bat
Go.command
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
Browse
Find
Debug
Workbench
1 # TextOutput02.py
2
3
4
It is a good habit to Clear this window before you Run to prevent confusion between old and new outputs.
TextOutput01.py
TextOutput02.py
Compile Messages
jGRASP Messages
Run I/O
Interactions
End
Clear
Help
Line:3 Col:17 Code:115 Top:1
OVS
BLK


TextOutput04.py C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Codin...
File Edit View Build Project Settings Tools Window Help
All Files
Sort ...
grams03a-TextOu
Backup
Go.bat
Go.command
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
Browse
Find
Debug
Workbench
1 # TextOutput04.py
Note that multiple files can be loaded at the same time. Just click the any of the tabs below to work with a different file. This is useful for large projects.
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
Compile Messages
jGRASP Messages
Run I/O
Interactions
End
Clear
Help
Line:4 Col:22 Code:97 Top:1
OVS BLK


TextOutput07.py C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Codin...
File Edit View Build Project Settings Tools Window Help
All Files
Sort ...
grams03a-TextOu
Backup
Go.bat
Go.command
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
Browse
Find
Debug
Workbench
1 # TextOutput07.py
2
3
4
5
6
7
8
If you load too many files, you will no longer be able to read the file names on the tabs.
TextO...
TextO...
TextO...
TextO...
TextO...
TextO...
TextO...
Compile Messages
jGRASP Messages
Run I/O
Interactions
End
Clear
Help
Line:6 Col:14 Code:0 Top:1
OVS BLK


Quotes10.py C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Coding\P...
File Edit View Build Project Settings Tools Window Help
All Files
Sort ...
grams03d-Quotes
Backup
Go.bat
Go.command
Quotes01.py
Quotes02.py
Quotes03.py
Quotes04.py
Quotes05.py
Quotes06.py
Quotes07.py
Quotes08.py
Quotes09.py
Quotes10.py
Browse
Find
Debug
Workbench
1 # Quotes10.py
2
3
4
5
6
7
8
And since these files stay loaded, even if you quit and reload jGRASP, over time this issue can get worse and worse. However, there is a simple solution.
Compile Messages
jGRASP Messages
Run I/O
Interactions
End
Clear
Help
Line:1 Col:1 Code:35 Top:1
OVS BLK


Quotes10.py C:\Users\JohnSchram\Desktop\LearnIntroCS\IntroCS-Learning Units\Unit03-Introduction to Python Coding\P...
File Edit View Build Project Settings Tools Window Help
All Files Sort ...
grams03d-Quotes
Backup
Go.bat
Go.command
Quotes01.py
Quotes02.py
Quotes03.py
Quotes04.py
Quotes05.py
Quotes06.py
Quotes07.py
Quotes08.py
Quotes09.py
Quotes10.py
Browse Find
Debug
Workbench
1 # Quotes10.
2 # This prog
3 # and demon
4 # also be c
5
6
7 print()
8 print("One"
Compile Messages jGR
End
Clear
Help
Refresh Window
Cascade
Cascade and Resize
Close All Windows
TextOutput01.py
TextOutput02.py
TextOutput03.py
TextOutput04.py
TextOutput05.py
TextOutput06.py
TextOutput07.py
Comments01.py
Comments02.py
Comments03.py
SyntaxError01.py
SyntaxError02.py
SyntaxError03.py
SyntaxError04.py
SyntaxError05.py
SyntaxError06.py
SyntaxError07.py
Quotes01.py
Quotes02.py
Another good habit is to occasionally go up to **Window** and select **Close All Windows**. This is far less tedious than closing each and every single window individually.
Col:1 Code:35 Top:1
OVS BLK


jGRASP
File Edit View Project Settings Tools Window Help
All Files
Sort ...
grams03d-Quotes
Backup
Go.bat
Go.command
Quotes01.py
Quotes02.py
Quotes03.py
Quotes04.py
Quotes05.py
Quotes06.py
Quotes07.py
Quotes08.py
Quotes09.py
Quotes10.py
Browse
Find
Debug
Workbench
And now you have a clean slate again.
Compile Messages
jGRASP Messages
Run I/O
Interactions
End
Clear
Help

Section 3.3

Text Output With print

```
1 # TextOutput01.py
2 # This program demonstrates
3 # text output with <print>.
4
5
6 print("Hello World!")
7
8
```

```
 ----jGRASP exec: python TextOutput01.py
Hello World!


 ----jGRASP: operation complete.
```

Python Keywords & Program Statements

A Python *keyword* is a word that has a special meaning in a program or performs a special function.

A *program statement* usually contains one or more *keywords*.

Keywords in Python are case-sensitive.
For example:
print is a Python keyword, while **PRINT** is not.

```
1  # TextOutput02.py
2  # This program demonstrates how to
3  # display 4 lines of text using 4
4  # separate <print> commands.
5
6
7  print("Line1")
8  print("Line2")
9  print("Line3")
10 print("Line4")
11
```

```
  ----jGRASP exec: python TextOutput02.py
Line1
Line2
Line3
Line4

  ----jGRASP: operation complete.
```

```
 1 # TextOutput03.py
 2 # By default, the <print> command "ends" by going to
 3 # the next line -- as if it pressed the <enter> key.
 4 # However, you can change what happens at the <end>
 5 # of a <print> command.  This allows multiple outputs
 6 # to be on the same line.  In the particular example,
 7 # the <end> values are spaces, so the first 3 outputs
 8 # are all on the same line, separated by spaces.
 9 # Note that "Line4" is displayed on its own line.
10 # This is because "Line3" was displayed with a normal
11 # <print> command without an <end>.
12
13
14 print("Line1",end = " ")
15 print("Line2",end = " ")
16 print("Line3")
17 print("Line4")
18
```

```
----jGRASP exec: python TextOutput03.py
Line1 Line2 Line3
Line4


----jGRASP: operation complete.
```

```
1  # TextOutput04.py
2  # This program is very similar to the previous
3  # program.  The only difference is that instead
4  # of ending with a space, the first 2 <print>
5  # commands <end> with an "empty string".  While
6  # the first 3 outputs are still on the same line,
7  # this time there is nothing in-between them.
8
9
10 print("Line1",end = "")
11 print("Line2",end = "")
12 print("Line3")
13 print("Line4")
14
```

```
  ----jGRASP exec: python TextOutput04.py
Line1Line2Line3
Line4


  ----jGRASP: operation complete.
```

```
1  # TextOutput05.py
2  # This program demonstrates that you can put any
3  # character, or even several characters, inside
4  # the quotes of an <end> keyword, which can lead
5  # to some weird looking output.
6
7
8  print("Line1",end = "$")
9  print("Line2",end = "???")
10 print("Line3")
11 print("Line4")
```

```
  ----jGRASP exec: python TextOutput05.py
Line1$Line2???Line3
Line4


  ----jGRASP: operation complete.
```

```python
# TextOutput06.py
# This program shows how to skip one or more
# lines when displaying text.  Using <print>
# with empty parentheses will generate a
# crlf (carriage-return/line-feed).


print("Line1")
print()
print("Line3")
print()
print()
print()
print("Line7")

```

```
----jGRASP exec: python
Line1

Line3




Line7

----jGRASP: operation co
```

```
 1 # TextOutput07.py
 2 # This program has the exact same output
 3 # as TextOutput06.py. It shows another
 4 # way to skip one or more lines by using
 5 # the "Escape Sequence" <\n> which means
 6 # "New Line".
 7
 8
 9 print("Line1\n")
10 print("Line3\n\n\n")
11 print("Line7")
12
```

```
----jGRASP
Line1

Line3



Line7

----jGRASP:
```

print & end

The **print** command generates an output display of the characters contained between double quotes.

If the **end** keyword is not used, **print** will also generate a *carriage-return/line-feed* (crlf).

If the **end** keyword is used, **print** will not generate a crlf and instead display the specified text in the second set of quotes.

Examples:

print("Hello") **print("World")**	**print("Hello", end = " ")** **print("World")**
will display:	will display:
Hello World	Hello World

Creating Blank Lines

Creating a blank line of output can be done 2 different ways. One is to use **print()** with absolutely nothing in the parentheses. The other is to add the **\n** *Escape Sequence* to an existing **print** statement:

Examples:

print("Hello") **print()** **print("World")**	**print("Hello\n")** **print("World")**
will display:	will also display:

`Hello`
 	
`World`
	`Hello`
	
	`World`

Section 3.4

Program Comments

```
 1 # Comments01.py
 2 # This program displays several number words.
 3 # The focus now is on program comments.
 4 # Program comments aid in "program documentation"
 5 # and makes your program more readable.  Every line
 6 # that begins with a "hashtag" is considered a
 7 # "comment" by the Python interpreter.
 8 # The Python interpreter ignores all comments.
 9 # They are not executed.
10 # If a line begins with a hashtag, it is simply
11 # ignored by Python. That is precisely what is
12 # happening with the first 19 lines of this program.
13 # Note below that a comment can also be placed in
14 # the middle of the program.  They can even be placed
15 # right after a program statement on the same line.
16 # You will also see that the word "Thirteen" is not
17 # displayed because it has been "commented-out"
18 # possibly by someone suffering from "Triskaidekaphobia"
19 # (the fear of the number 13).
```

```
20
21
22 print()
23 print("One")
24 print("Two")
25 print("Three")
26 print("Four")
27 print("Five")
28 print("Six")        # half a dozen
29 print("Seven")
30 print("Eight")
31 print("Nine")
32 print("Ten")
33 print("Eleven")
34 print("Twelve")     # one dozen
35 #print("Thirteen")  # one baker's dozen
36
37
38
```

```
20
21
22 print()
23 print("One")
24 print("Two")
25 print("Three")
26 print("Four")
27 print("Five")
28 print("Six")        # half a dozen
29 print("Seven")
30 print("Eight")
31 print("Nine")
32 print("Ten")
33 print("Eleven")
34 print("Twelve")     # one dozen
35 #print("Thirteen")  # one baker's
36
37
38
```

```
----jGRASP

One
Two
Three
Four
Five
Six
Seven
Eight
Nine
Ten
Eleven
Twelve

----jGRASP:
```

```
1  # Comments02.py
2  # This program demonstrates that a section of
3  # code can essentially be "commented-out" by
4  # using triple-double-quotes.
5
6
7  print()
8  print("One")
9  print("Two")
10 """
11 print("Three")
12 print("Four")
13 print("Five")
14 print("Six")          # half a dozen
15 print("Seven")
16 print("Eight")
17 print("Nine")
18 print("Ten")
19 print("Eleven")
20 """
21 print("Twelve")      # one dozen
22 print("Thirteen")    # one baker's dozen
```

```
----jGRASP

One
Two
Twelve
Thirteen

----jGRASP:
```

String Literal Definition

A *string literal* is any text in-between a set of quotes.

The “text” can be a name, letter, group of words, or basically anything that you can type inside a set of quotes.

Examples:
"computer"
"John Smith"
"Hello all you happy people."
"Q"
"?"
"811 Fleming Trail"
"Richardson, Texas"
"75081"

Most **print** statements contain a *string literal* in their parentheses.

Python Multi-Line Comment Disclaimer

Technically, Python does not have *Multi-Line Comments*.

The “Triple-Double-Quote” is actually used to create a multi-line *string literal*.

This is why they are **green** and not **red** like the *Single-Line Comments*.

However, multi-line string literals which are not part of a **print** statement are simply ignored by the Python interpreter.

This essentially makes them multi-line comments.

```
1  ###################################
2  #                                 #
3  #          Comments03.py          #
4  #        Numbers from 1-13        #
5  #         By: John Schram         #
6  #             11/8/17             #
7  #                                 #
8  #  This program is similar to     #
9  #  the previous two and shows     #
10 #  that a comment can be used     #
11 #  to create a heading.           #
12 #                                 #
13 ###################################
14
15
16 print()
17 print("One")
18 print("Two")
19 print("Three")
20 print("Four")
21 print("Five")
22 print("Six")        # half a dozen
23 print("Seven")
24 print("Eight")
25 print("Nine")
26 print("Ten")
27 print("Eleven")
28 print("Twelve")     # one dozen
29 print("Thirteen")   # one baker's dozen
```

```
----jGRASP

One
Two
Three
Four
Five
Six
Seven
Eight
Nine
Ten
Eleven
Twelve
Thirteen

----jGRASP:
```

2 Different Types of Comments

```
# This is a single-line comment.

print("Good morning!")   # So is this.
```

```
"""
This is a
multi-line
comment.
"""
```