

Introduction to Python on IBM i



Agenda

- A little about Python
- Why use Python?
- How to install/determine if installed
- Syntax101
 - Variables
 - Strings
 - Functions
 - Command Line
- DB2 Data Access

Acknowledgements

- Kevin Adler
- Tony Cairns
- Jesse Gorzinski
- Google
- Memegenerator
- Corn chips & salsa
- Clean socks
- and, of course,
 - spam



Before you freak out

- Why isn't Mike talking about PHP?
 - Zend WAS the PHP company
 - Rogue Wave IS the Open Source company
 - Support for 300+ Open Source projects
 - And, of course, PHP



A little about Python

What is it really?

- General purpose programming language
- Easy to get started
- Simple syntax
- Great for integrations
- Access to C and other API
- Infrastructure first, but applications too.

Thanks: Tahani Alamanie

Historically...

- Python was conceptualized by **Guido Van Rossum** in the late **1980s**.
- Rossum published the first version of Python code (0.9.0) in February **1991** at the CWI (Centrum Wiskunde & Informatica) in the Netherlands , Amsterdam.
- Python is derived from **ABC** programming language, which is a general-purpose programming language that had been developed at the CWI.
- Rossum chose the name "**Python**", since he was a big fan of Monty Python's Flying Circus.
- Python is now maintained by a core development team at the institute, although Rossum still holds a vital role in directing its progress.

Thanks: Tahani Alamanie

Python lineage

- Python 1 – 1994
- Python 2 – 2000 (Not dead yet...)
 - 2.7 - 2010
- Python 3 – 2008
 - 3.5 – 2015
 - 3.6.1 – March 2017



Python 2 or 3?



The screenshot shows a web browser window with the URL <https://pythonclock.org>. The page features a large green countdown timer for Python 2.7 retirement. The timer is displayed as follows:

2	9	22	7	34	16
Years	Months	Days	Hours	Minutes	Seconds

Below the timer, there is a section titled "What's all this, then?" which explains that Python 2.7 will not be maintained past 2020 and suggests April 12th, 2020, as the end-of-life date. It also provides an email address pythonclockorg@gmail.com for those interested. Another section titled "How do I get started?" provides advice on transitioning from Python 2 to Python 3, mentioning the [official porting guide](#).

What's the diff?

- Example:
- Python 2 print statement replaced by function:
 - Python2 – `print "Hello World!"`
 - Python3 – `print("Hello World!")`
- *Many more differences, tho...*

Why use it?

Python==hot

TIOBE Index for May 2017

May Headline: the pack is closing in on Java and C

May 2017	May 2016	Change	Programming Language	Ratings	Change
1	1		Java	14.639%	-6.32%
2	2		C	7.002%	-6.22%
3	3		C++	4.751%	-1.95%
4	5	▲	Python	3.548%	-0.24%
5	4	▼	C#	3.457%	-1.02%
6	10	▲▲	Visual Basic .NET	3.391%	+1.07%
7	7		JavaScript	3.071%	+0.73%
8	12	▲▲	Assembly language	2.859%	+0.98%
9	6	▼	PHP	2.693%	-0.30%
10	9	▼	Perl	2.602%	+0.28%

- RPG is 45, COBOL is 25, ugh...

Some reasons folks use Python

- Open source is free, right?
 - Yes, no...well – sort of...
 - No charge LP on IBM i – 5733OPS
- Only scripting language to support multiple inheritance
- Multi-threading
 - Stateful, not Async like Node
 - But same motivation as Node, non-blocking
- Applications develop faster
 - Extensive standard library that is constantly evolving
 - Frameworks such as Bottle

```
from multiprocessing.dummy import Pool as ThreadPool
pool = ThreadPool(4)
results = pool.map(my_function, my_array)
```

Industries using Python

- Technology
 - Geospatial, Machine Learning
 - Honeywell - Build automation
- Manufacturing
 - D-Link updates – Multithreaded
 - Phillips – Robot programming and sequencing
- Entertainment
 - ILM – Scripting CGI Intensive films
 - Video gaming
- Transportation
 - Airports – Frequentis for weather info
 - US, Denmark, Iceland, Hong Kong, etc.

Companies using Python

- YouTube - Backend
- Facebook – (see next slide)
- Dropbox
- NASA – Weather and more
- IBM
- Mozilla
- Instagram
- Red
- Quora

– Why python?



Adam D'Angelo, wrote a lot of Python for Quora

Updated Sep 12, 2014 · Upvoted by Charlie Cheever, [One of the founders of Quora](#) and Scott Danzig, avid Quora user since 2012, and a Quora Top Writer

Python was a language that Charlie and I both knew reasonably well (though I know it a lot better now than I did when we started). We also briefly considered C#, Java, and Scala. The biggest issues with Python are speed and the lack of typechecking.

Facebook in detail

- Top three languages

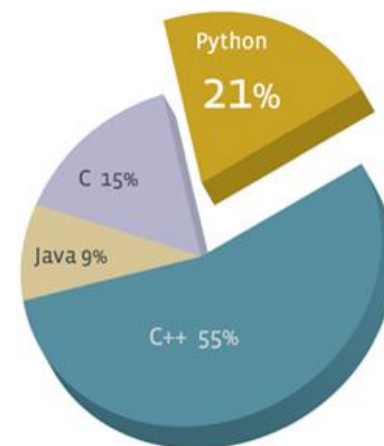
- Hack (fork of PHP)
- C++
- Python

- Infrastructure management

- Network switch setup
- Core services (DNS, etc,)
- Server imaging, burnin-in.

- Platform services: Job Engine, team workflow management

- Millions of lines of code, thousands of libraries and binaries
- 2016 to date: average 5,000 commits per month, 1,000+ committers
- 5 percent Py3 (as of May 2016)



<https://code.facebook.com/posts/1040181199381023/python-in-production-engineering/>

Got Python?

Details at Developerworks...

- <https://www.ibm.com/developerworks/community/wikis/home?lang=en#!/wiki/IBM%20i%20Technology%20Updates/page/Open%20Source%20Technologies>

You are in: [IBM i Technology Updates](#) > [IBM i Technology Updates](#) > [Open Source Technologies](#) > [Python](#)

Python
 2 | Updated November 3, 2016 by kadler | Tags: None

Page Actions -

Python

Python is a popular high-level programming language. It is easily extensible through the use of third-party packages and often allows powerful function to be written with few lines of code. Python caters to multiple programming styles (object oriented, procedural, etc) and the code tends to be readable and maintainable.

Python is now being delivered and packaged for IBM i. It is available through the following options:

- Option 2 - Python 3.4
- Option 4 - Python 2.7

The following add-ons are also available via separate PTFs:

Package	Option 2 PTF	Option 4 PTF	Description
ibm_db	SI57253	SI60567	DB2 for i connector - Allows native access to DB2 for i
itoolkit	SI57254	SI60568	Toolkit for IBM i - allows access to system resources through program calls, SQL queries, CL commands, shell commands, and more.
flipflop	SI57255		FastCGI gateway
bottle	SI57256		Lightweight web framework.

Add-on packages are located in `/QOpenSys/QIBM/ProdData/OPS/Python-pkgs`. They must first be installed in order to be available. See [Installing shipped add-ons](#).

Open Source Technologies on IBM i

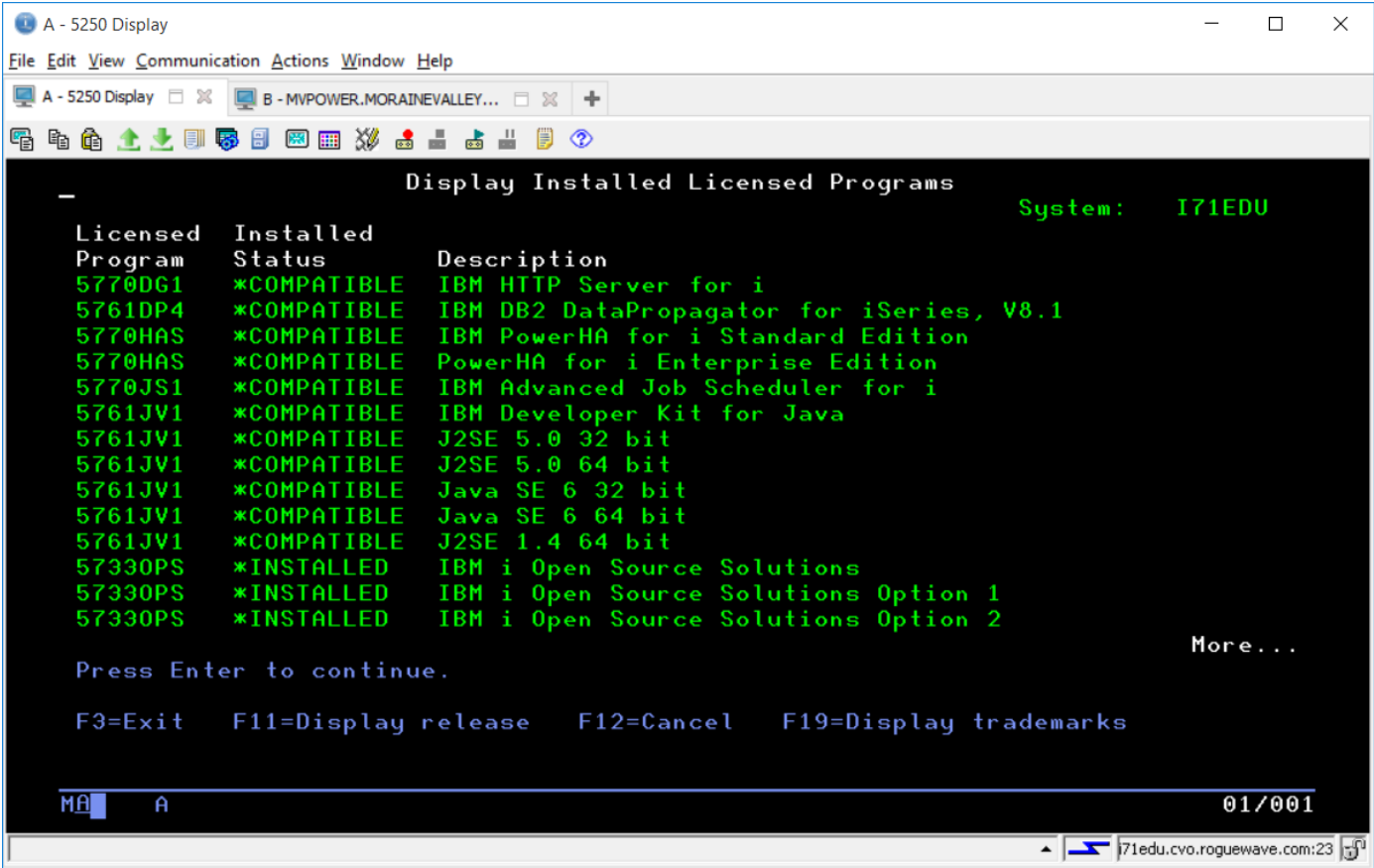
	SAMBA on IBM i
5733-OPS Option 1	Node.js v1
5733-OPS Option 2	Python 3
5733-OPS Option 3	CHROOT
5733-OPS Option 4	Python 2
5733-OPS Option 5	Node.js v4
5733-OPS Option 6	Git
5733-OPS Option 7	Tools
5733-OPS Option 8	Orion
5733-OPS Option 9	cloud-init
5733-OPS Option 10	Node.js v6
5733-OPS Option 11	TBD
5733-OPS Option 12	TBD
5733-OPS Option 13	TBD

Which one?

- Python 3 is LPP option 2
- Python 2 is LPP option 4
- Correct answer: It depends...
 - Many existing libraries are Python 2
 - But 90%+ are also Python 3 compliant, or on their way
- Python 2.7 will retire in less than 3 years (EOL==2020)

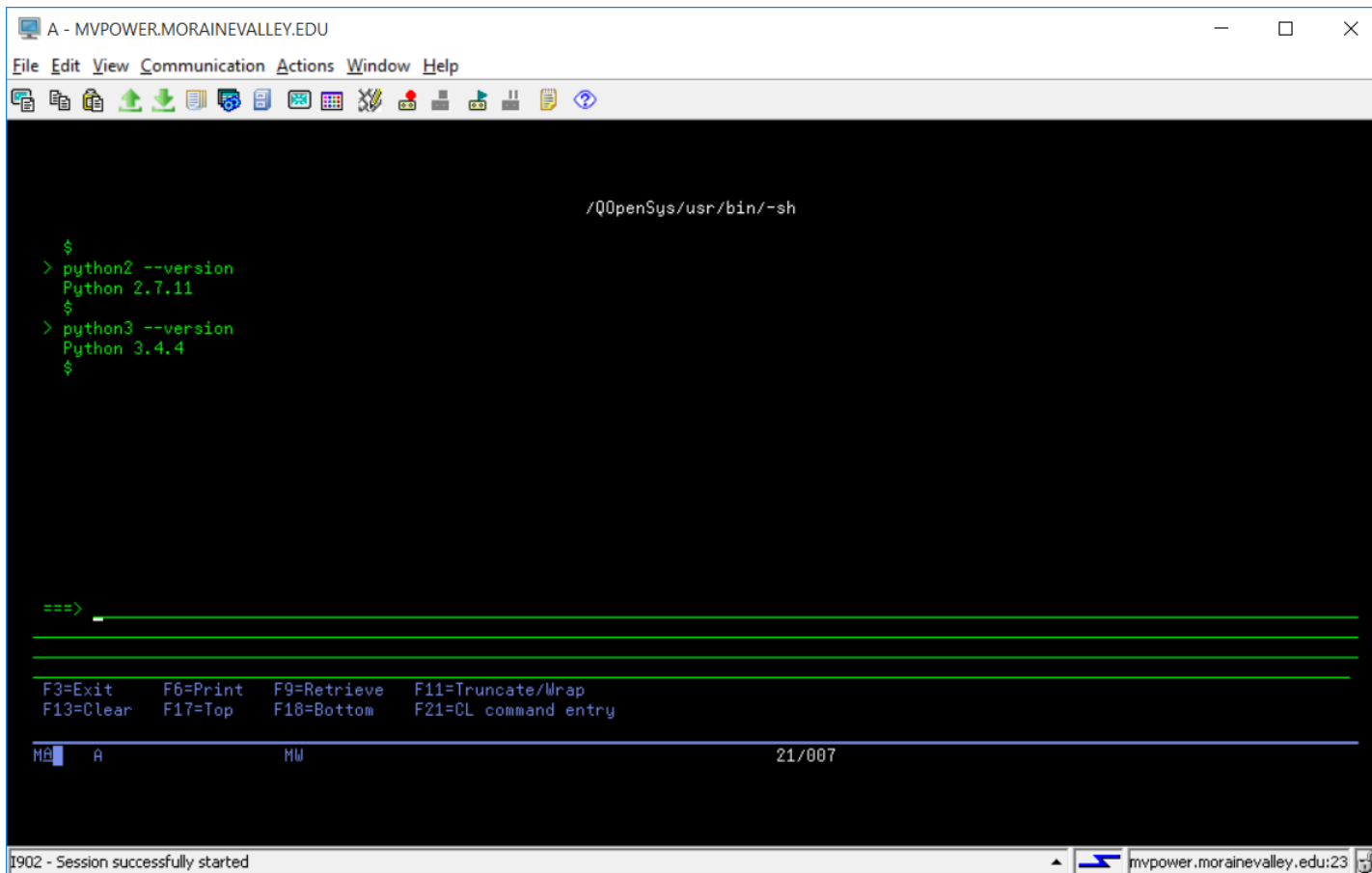
Need Licensed Programs

- 5733OPS Base and option 2 or 4



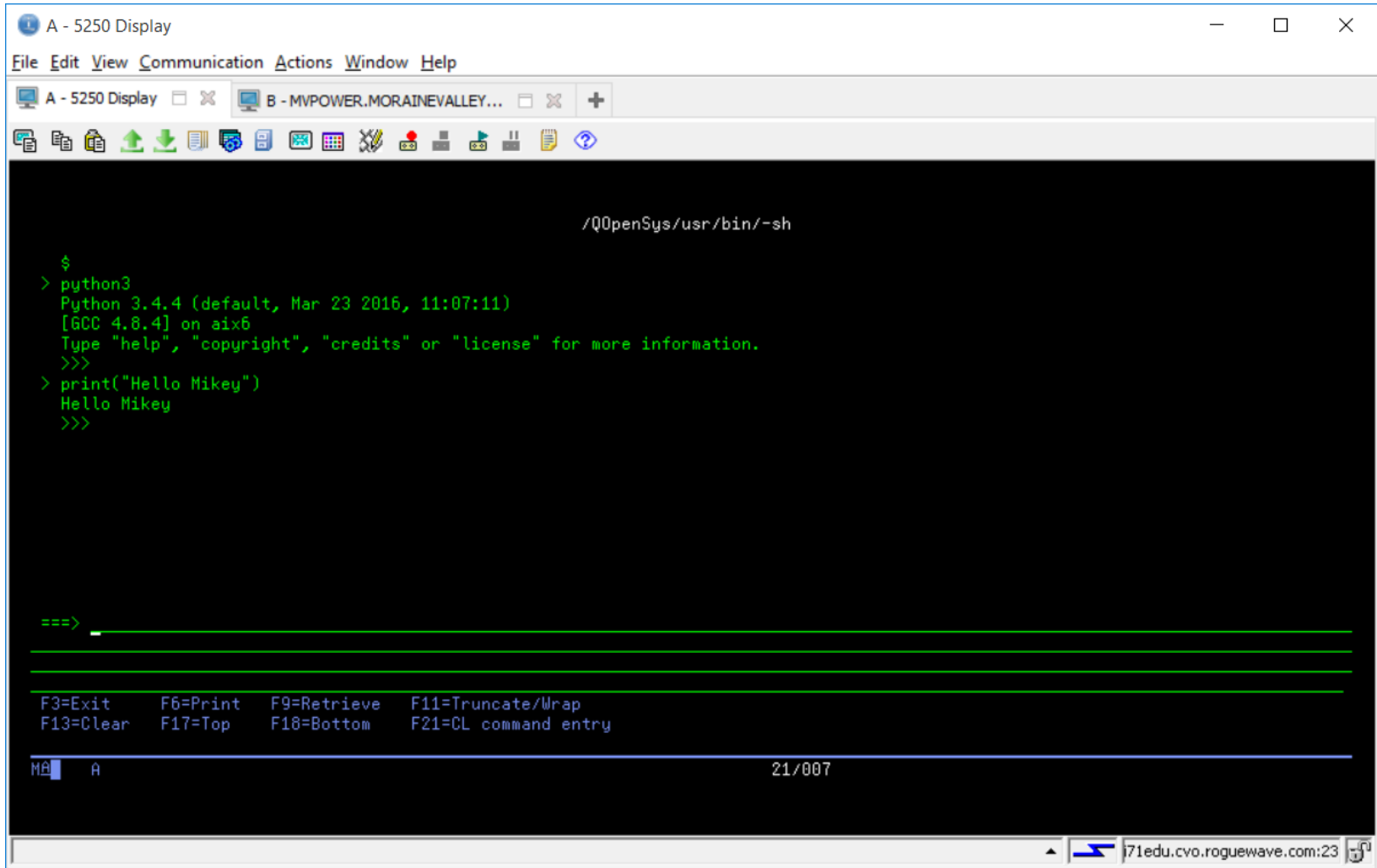
Python in action

- Command line via green screen (CALL QP2TERM)



The screenshot shows a terminal window titled "A - MVPOWER.MORAINEVALLEY.EDU". The terminal prompt is `/Q0penSys/usr/bin/-sh`. The user has entered two commands: `> python2 --version` which outputs `Python 2.7.11`, and `> python3 --version` which outputs `Python 3.4.4`. Below the terminal output, there is a green screen interface with a green prompt `===>` and a horizontal line. At the bottom of the green screen, there is a list of function key shortcuts: `F3=Exit`, `F6=Print`, `F9=Retrieve`, `F11=Truncate/Wrap`, `F13=Clear`, `F17=Top`, `F18=Bottom`, and `F21=CL command entry`. The bottom status bar of the terminal shows `MA A MW 21/007`. At the very bottom of the window, a status bar indicates `T902 - Session successfully started` and the URL `mvpower.morainevalley.edu:23`.

Hello world?



The image shows a terminal window titled "A - 5250 Display" with a menu bar (File, Edit, View, Communication, Actions, Window, Help) and a toolbar. The terminal content is as follows:

```
/Q0penSys/usr/bin/-sh

$
> python3
Python 3.4.4 (default, Mar 23 2016, 11:07:11)
[GCC 4.8.4] on aix6
Type "help", "copyright", "credits" or "license" for more information.
>>>
> print("Hello Mikey")
Hello Mikey
>>>

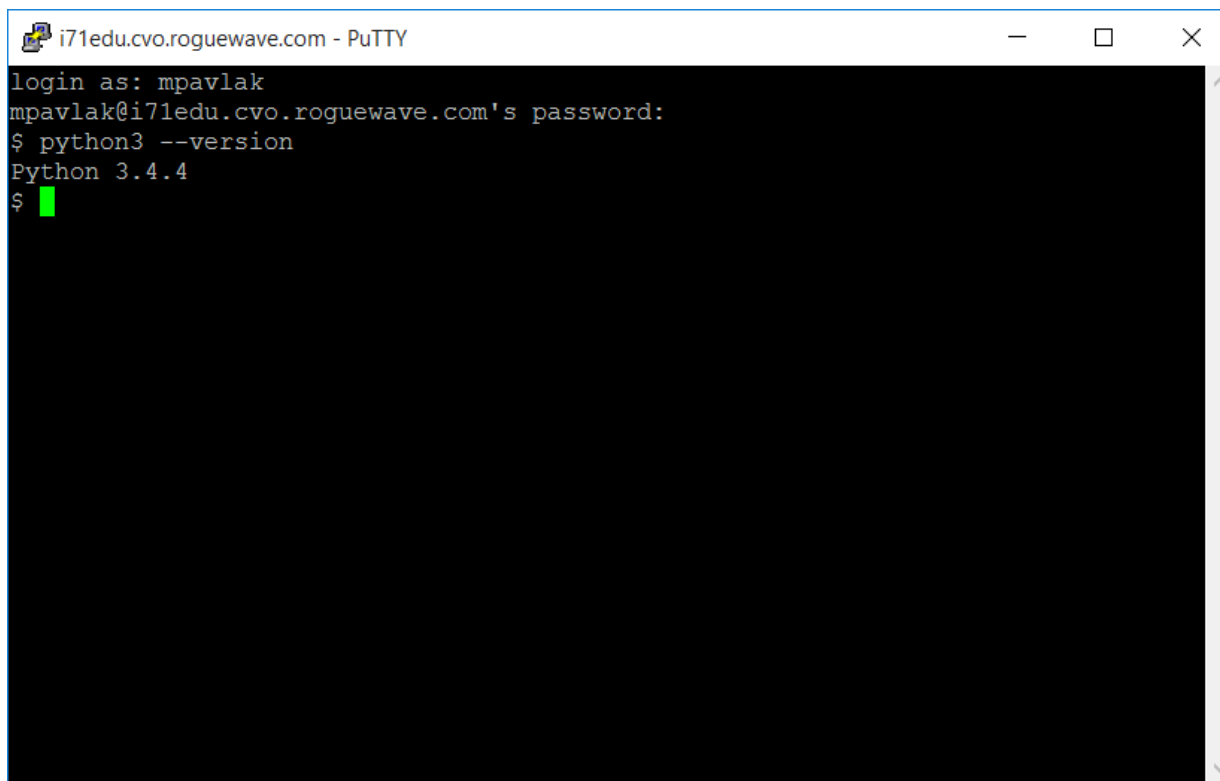
===> _____
_____
_____
_____
F3=Exit    F6=Print  F9=Retrieve  F11=Truncate/Wrap
F13=Clear  F17=Top   F18=Bottom  F21=CL command entry

MBA A 21/007
```

The terminal window also shows a status bar at the bottom with the URL "71edu.cvo.roguewave.com:23" and a system tray icon.

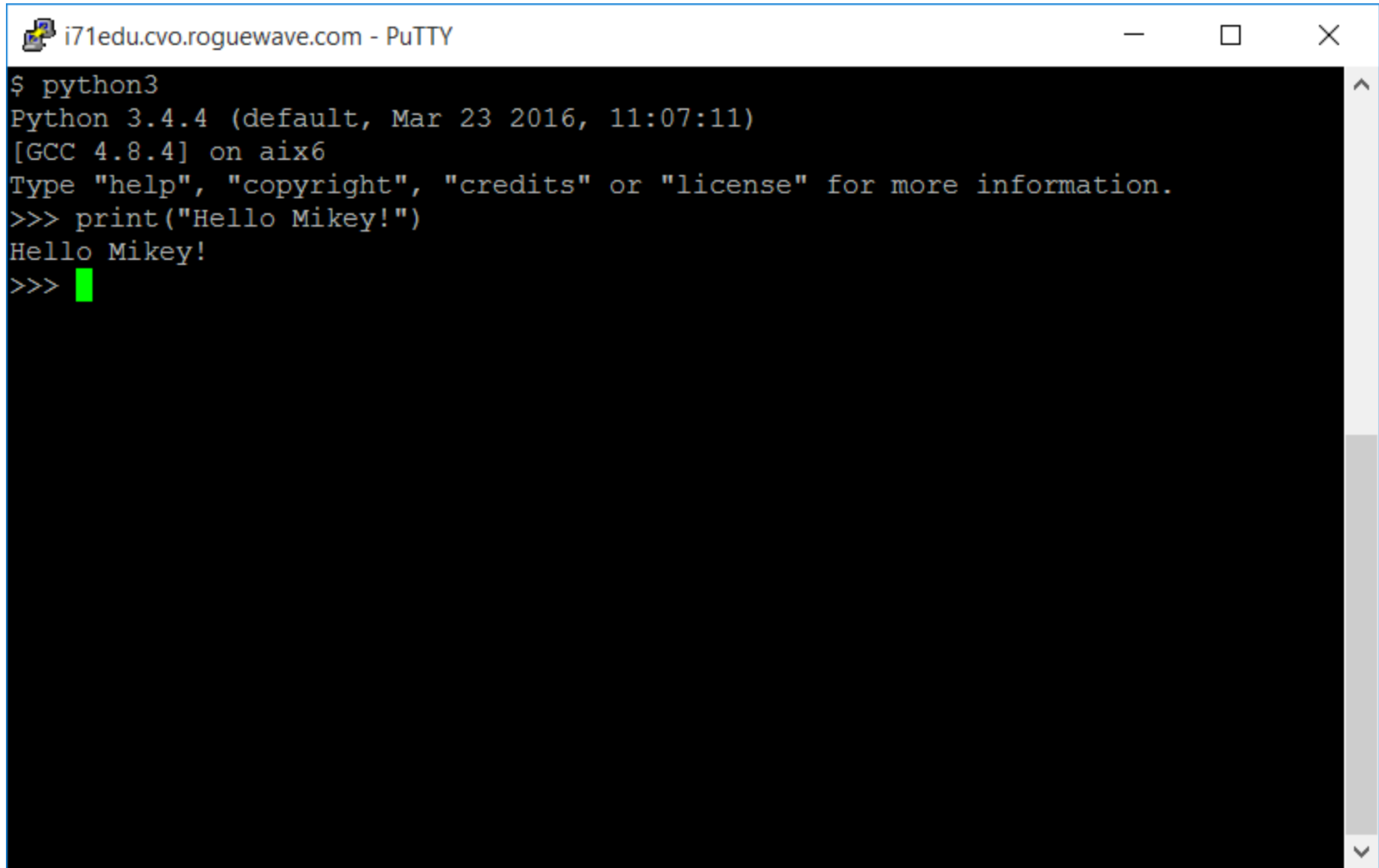
Most prefer SSH

- Command line via SSH terminal

A screenshot of a PuTTY terminal window titled "i71edu.cvo.roguewave.com - PuTTY". The terminal shows a login session for user "mpavlak" on the host "i71edu.cvo.roguewave.com". The user enters their password, and the prompt changes to "\$". They then run the command "python3 --version", which outputs "Python 3.4.4". The prompt returns to "\$" with a green cursor.

```
i71edu.cvo.roguewave.com - PuTTY
login as: mpavlak
mpavlak@i71edu.cvo.roguewave.com's password:
$ python3 --version
Python 3.4.4
$
```

Hello World, again?



```
i71edu.cvo.roguewave.com - PuTTY
$ python3
Python 3.4.4 (default, Mar 23 2016, 11:07:11)
[GCC 4.8.4] on aix6
Type "help", "copyright", "credits" or "license" for more information.
>>> print("Hello Mikey!")
Hello Mikey!
>>> █
```


IDE?

Zend Studio

- No, you don't need to buy Zend Studio
- Use Orion, etc.
- But if you have Studio or RDi...
 - Consider something from Eclipse.org
 - I grabbed PyDev

Eclipse

PyDev - Python IDE for Eclipse



☆ 474 💬 27

 **Install**



Details

Metrics

Errors

External Install Button

PyDev is a plugin that enables Eclipse to be used as a Python IDE (supporting also Jython and IronPython).

It uses advanced type inference techniques which allow it to provide things such as code completion and code analysis, besides providing a debugger, interactive console, refactoring, tokens browser, django integration, etc.

Homepage:

pydev.org

Getting Started:

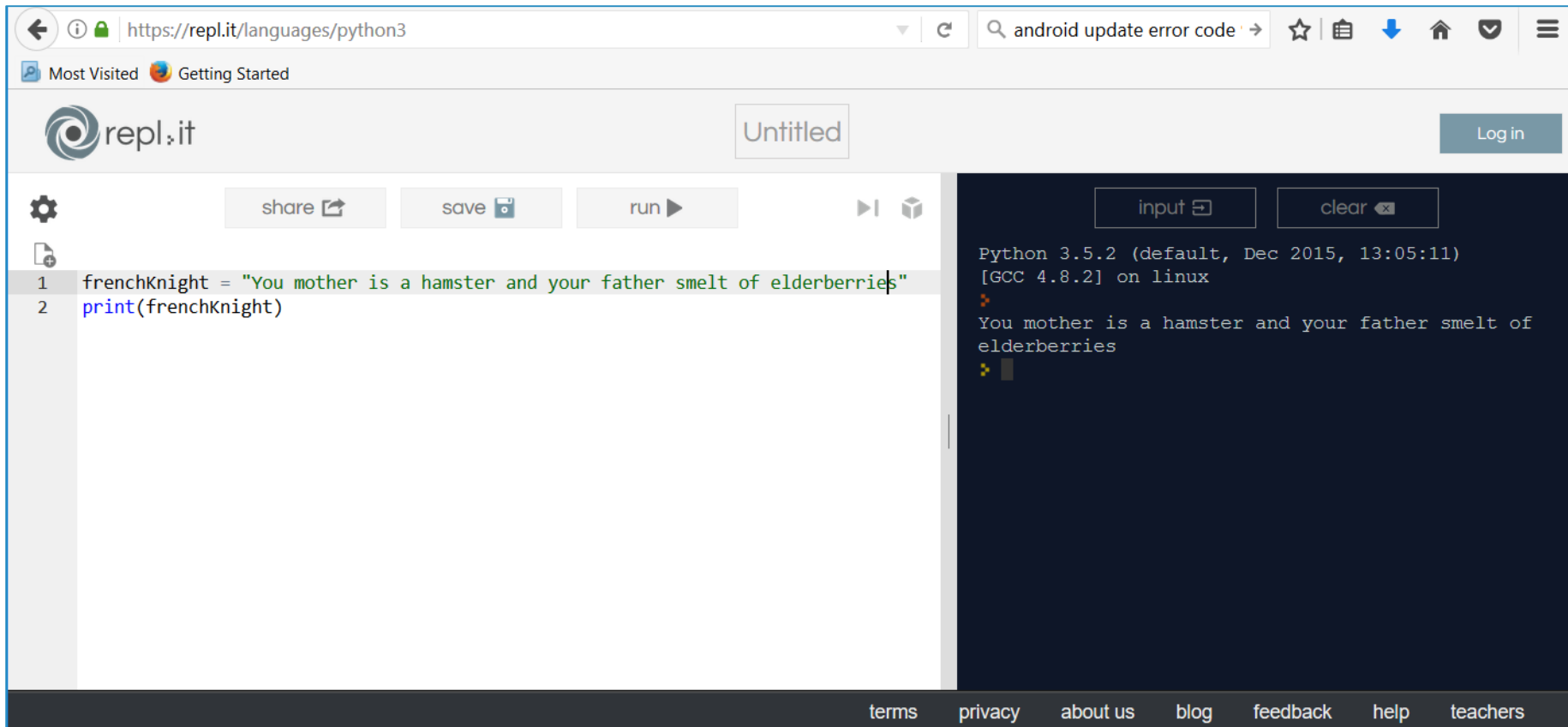
[Getting Started \(read to make sure you can get most out of PyDev\)](#)

Feature Matrix:

pydev.org/manual_adv_features.html

Alternatives to IBM i when learning

- What's that? The boss won't let you install Python?
 - Consider repl.it



The screenshot shows the repl.it web interface for Python3. The browser address bar displays `https://repl.it/languages/python3`. The page header includes the repl.it logo, a tab titled "Untitled", and a "Log in" button. Below the header, there are buttons for "share", "save", and "run". The code editor on the left contains two lines of Python code:

```
1 frenchKnight = "You mother is a hamster and your father smelt of elderberries"  
2 print(frenchKnight)
```

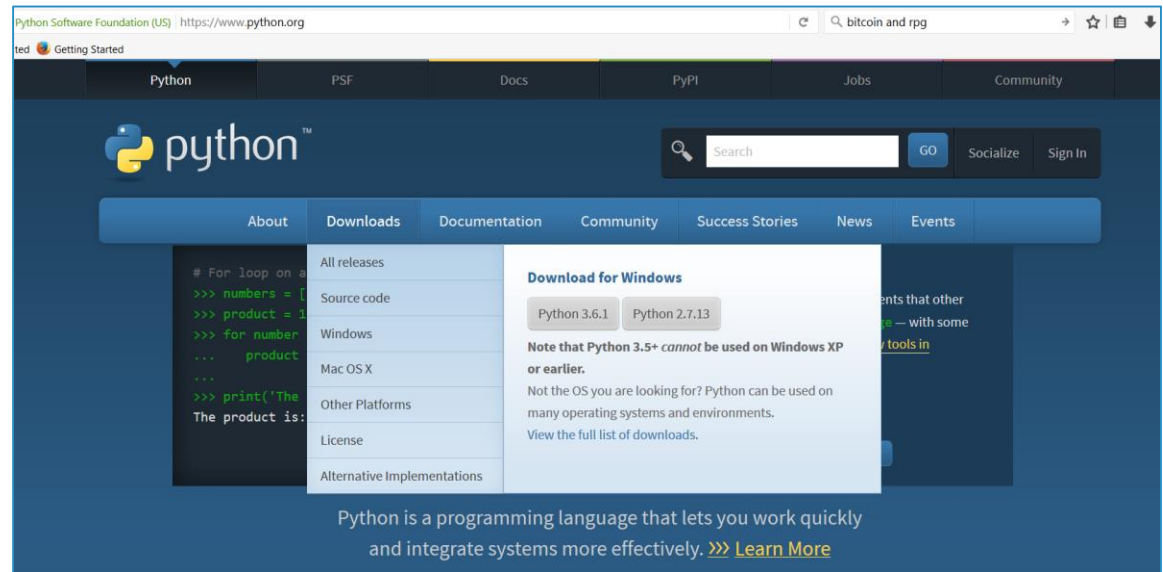
The output terminal on the right shows the following text:

```
Python 3.5.2 (default, Dec 2015, 13:05:11)  
[GCC 4.8.2] on linux  
You mother is a hamster and your father smelt of  
elderberries
```

At the bottom of the interface, there are links for "terms", "privacy", "about us", "blog", "feedback", "help", and "teachers".

Alternatives to IBM i when learning

- How about your PC?
- Goto the Python site:
 - Download
 - Install
 - Viola!



Python 3.6 (32-bit)

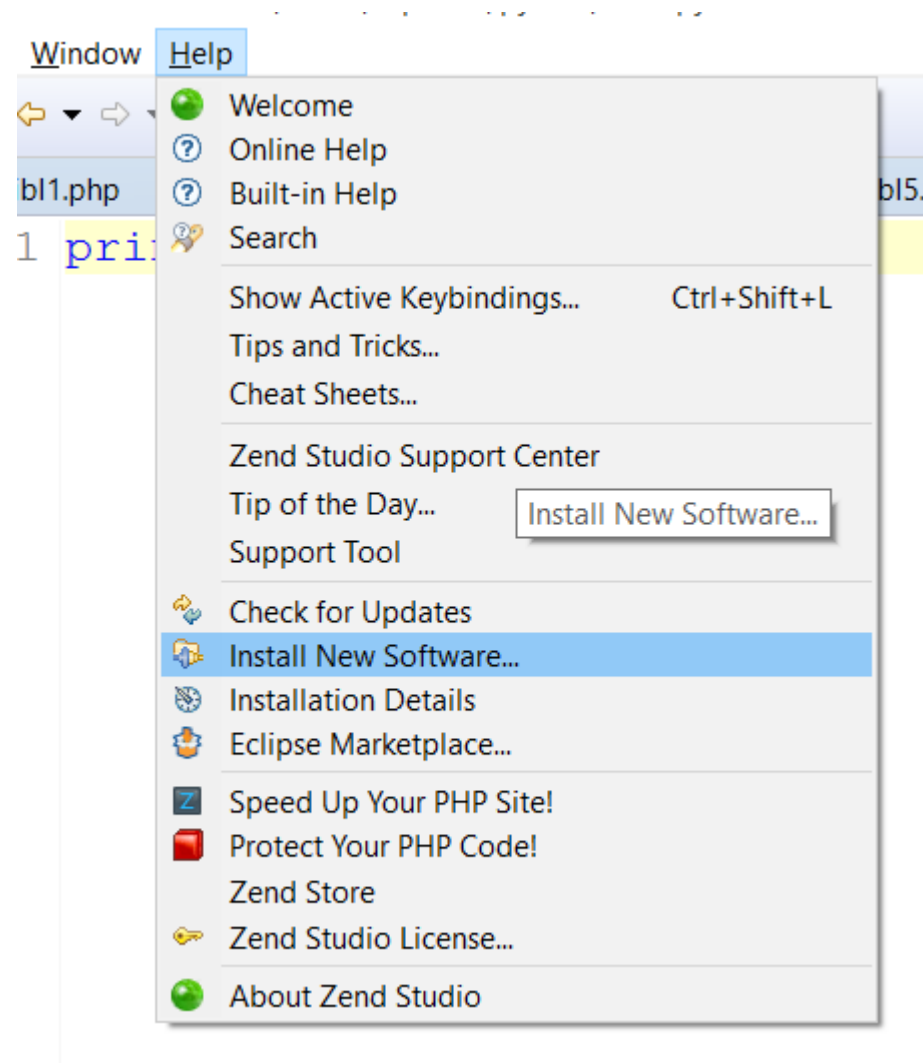
```
Python 3.6.1 (v3.6.1:69c0db5, Mar 21 2017, 17:54:52) [MSC v.1900 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print("I unclg my nose in your direction, sons of a window dresser.")
I unclg my nose in your direction, sons of a window dresser.
>>>
```

Download PyDev from Eclipse

The screenshot shows the Eclipse Marketplace website for the PyDev plugin. The browser address bar displays <https://marketplace.eclipse.org/content/pydev-python-ide-eclipse>. The page header includes the Eclipse Marketplace logo, navigation links for 'MY MARKETPLACE', 'ADD CONTENT', and 'MORE', and user options for 'Create account' and 'Log in'. The breadcrumb trail reads 'HOME / MARKETPLACE / TOOLS (1644) / PYDEV - PYTHON IDE FOR ECLIPSE'. The main content area features a 'MARKETS' sidebar, a search bar with an 'ADVANCED SEARCH' link and a 'SEARCH' button, and a 'MORE LIKE THIS' section listing related plugins like LiClipseText and LiClipse. The main product card for PyDev shows 472 stars and 27 comments, an 'Install' button, and an 'MPC DOWNLOADS Top 10' badge. The details section includes tabs for 'Details', 'Metrics', 'Errors', and 'External Install Button'. The description states: 'PyDev is a plugin that enables Eclipse to be used as a Python IDE (supporting also Jython and IronPython). It uses advanced type inference techniques which allow it to provide things such as code completion and code analysis, besides providing a debugger, interactive console, refactoring, tokens browser, django integration, etc.' Below this, there are sections for 'Homepage: pydev.org', 'Getting Started: Getting Started (read to make sure you can get most out of PyDev)', and 'Feature Matrix:'.

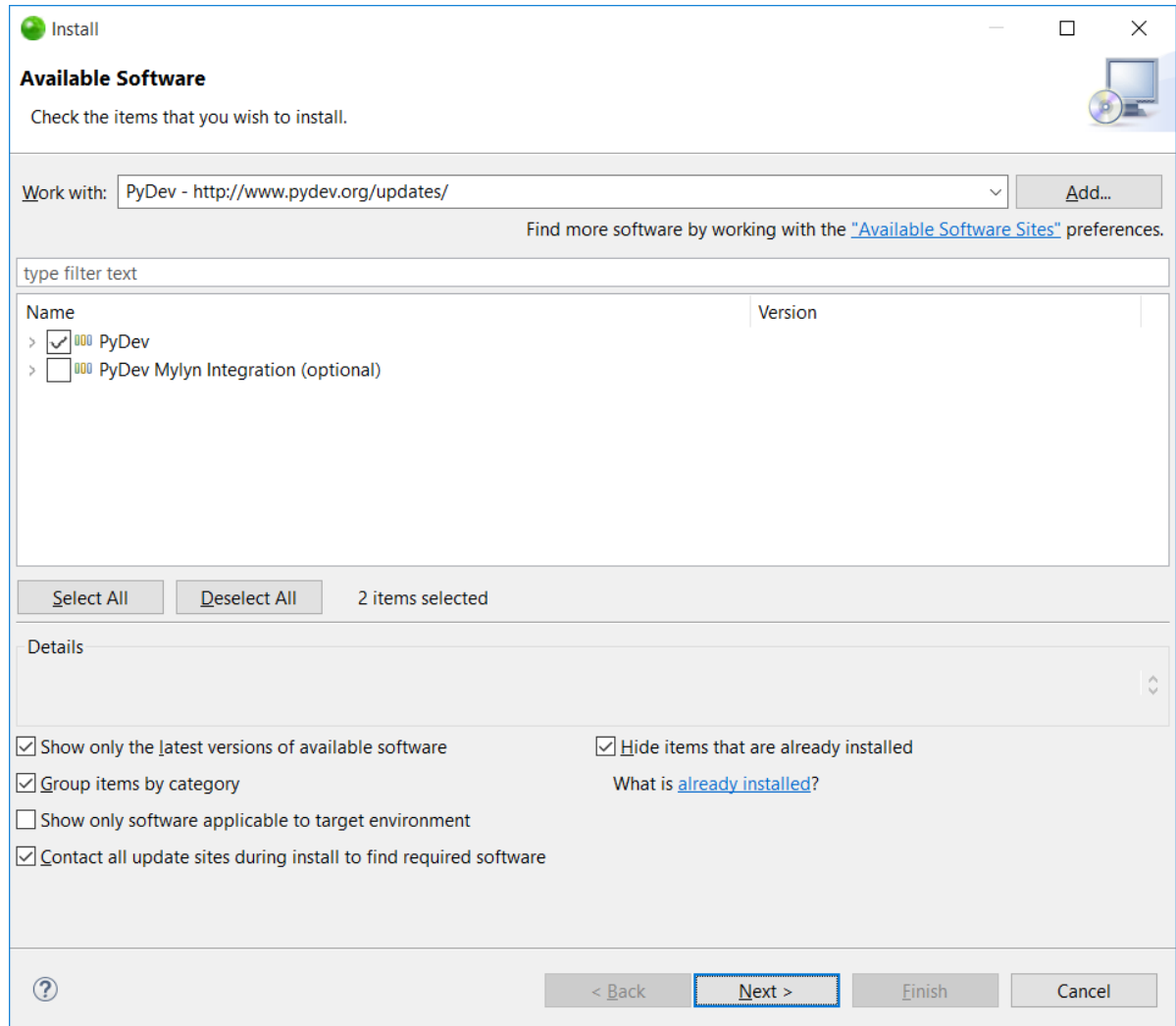
Capture URL

- Follow prompts



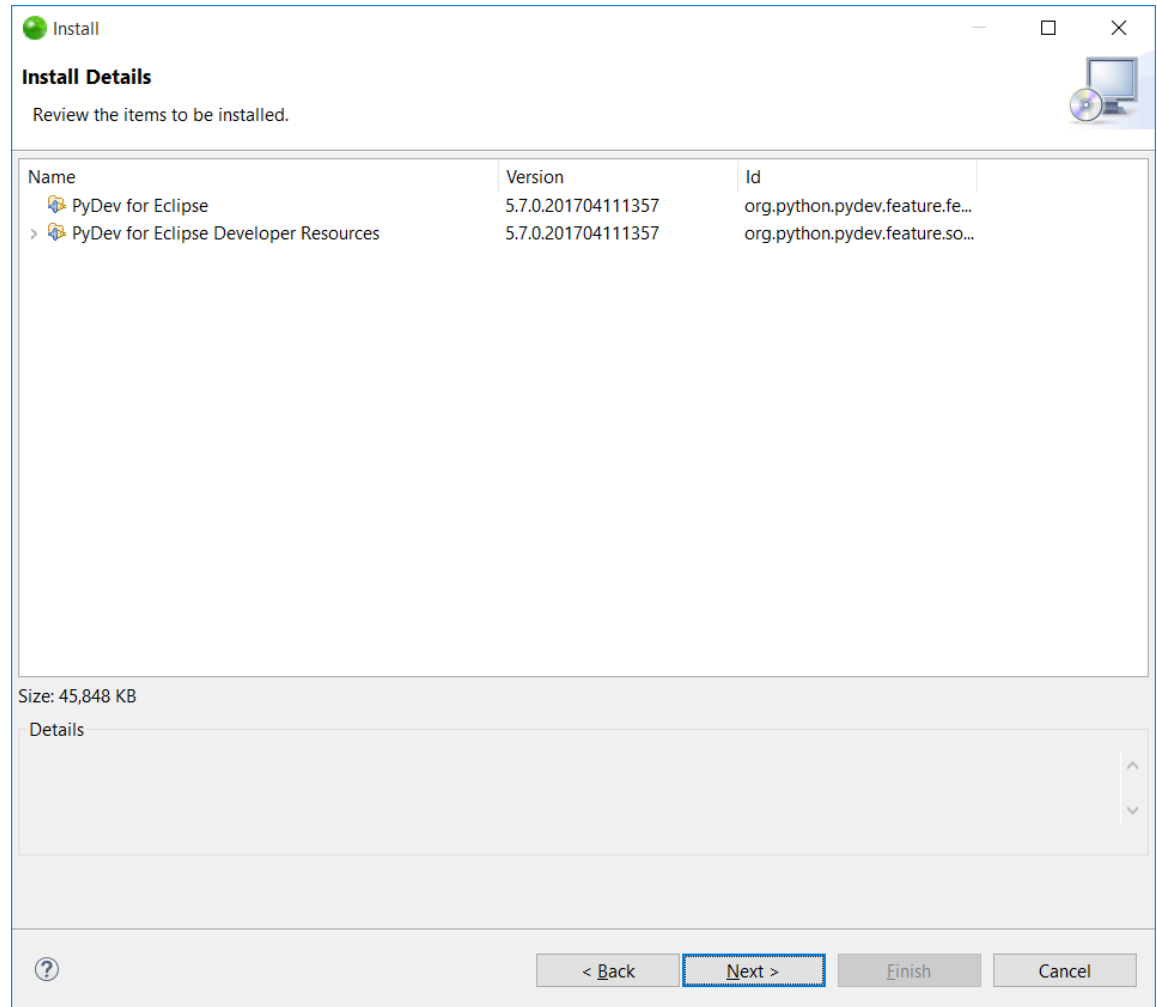
Editor for Eclipse

- Select what you like
- Next



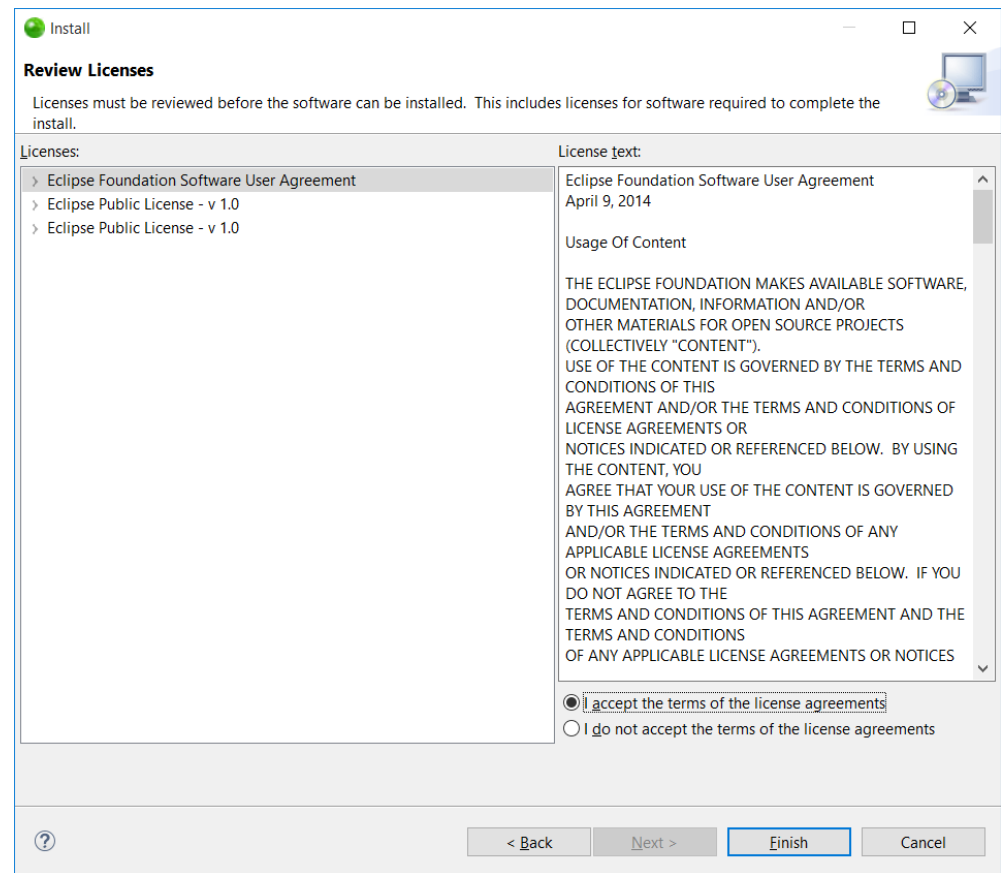
Confirm versions

- Next



Accept terms and EULA

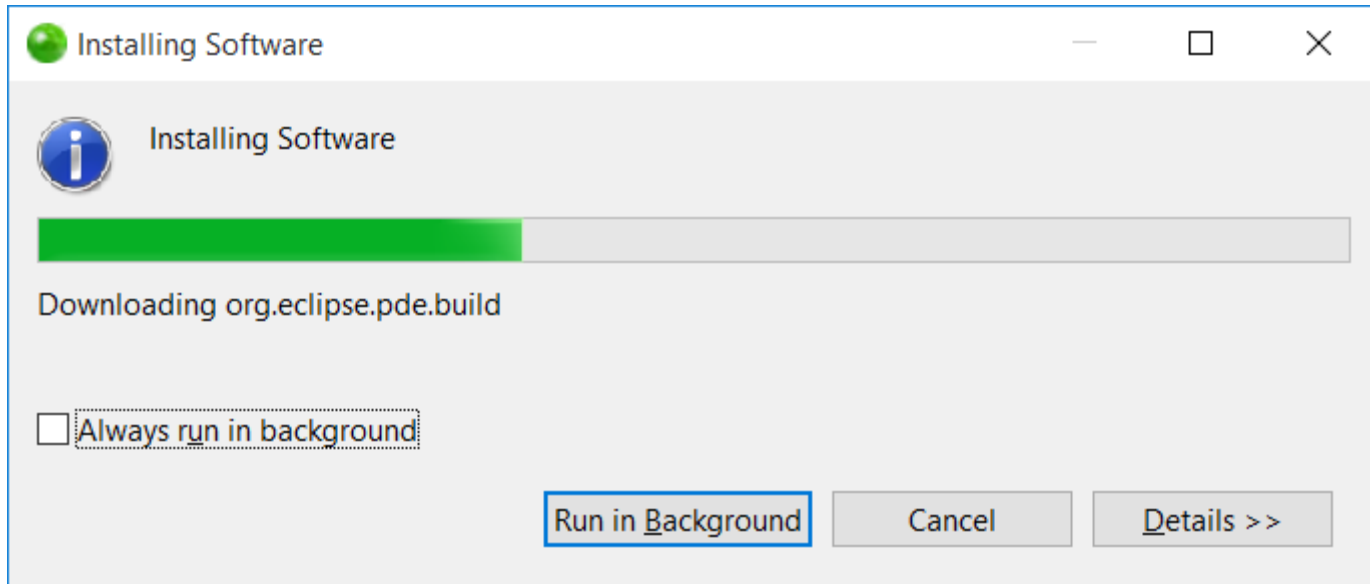
- Finish



Watch the pretty status bar

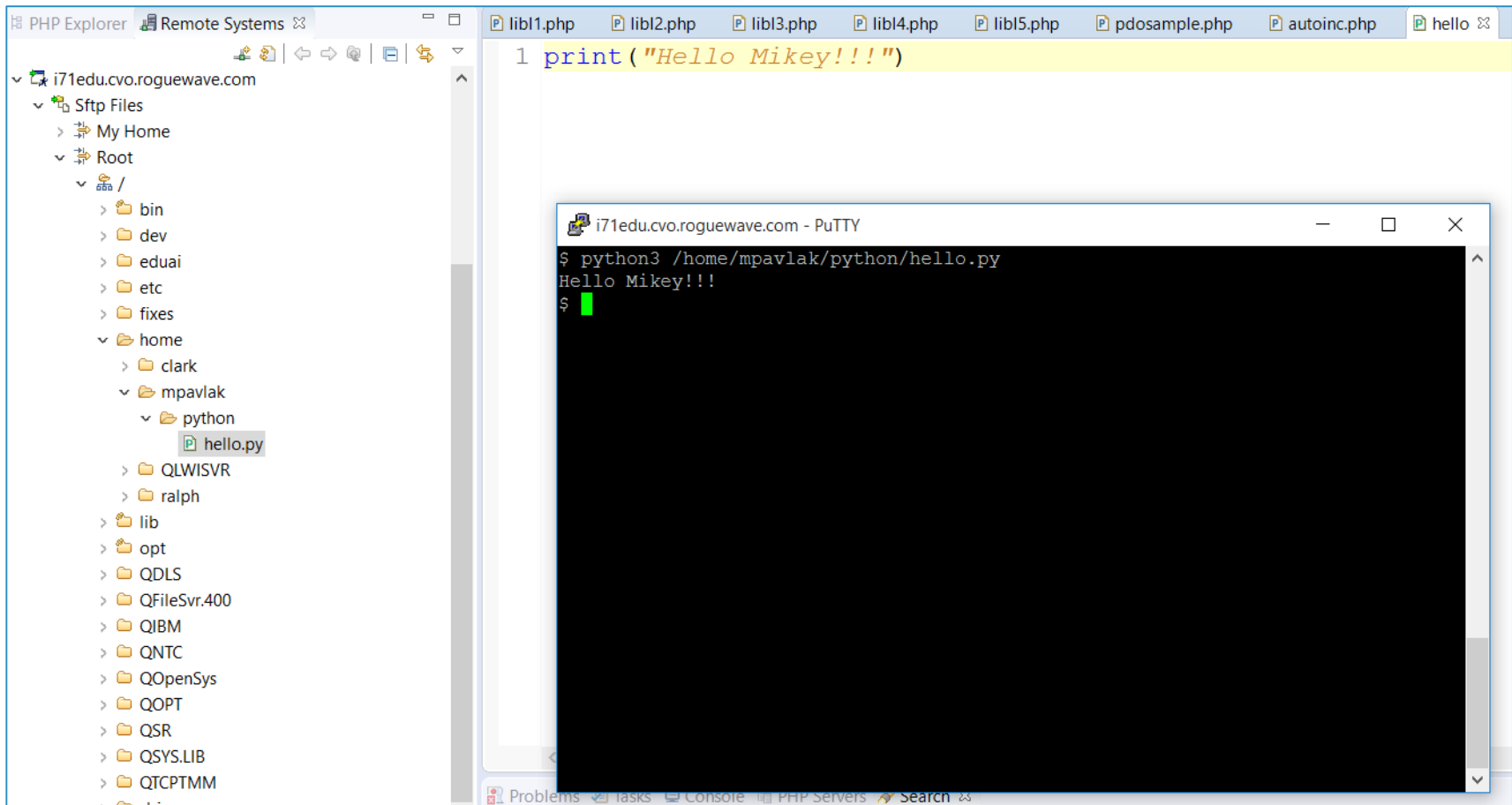


skwib.com



Python in Eclipse (i.e. Zend Studio)

- I bet RDi works, too!



Hello world...again...

- Create a file like Ex01hello.py
- Open the file
- Key up some code and click save

```
1 #  
2 # Hello World???  
3 #  
4 print("Hello Mikey!!!")
```

```
$  
> python3 /home/mpavlak/python/Ex01hello.py  
Hello Mikey!!!  
$
```

Hello world...again...

- Change the file
- Click save
- Back to qp2term & F9

```
1 # |
2 # Hello World???
3 #
4 #print("Hello Mikey!!!")
5 #
6 print("\n\nHello Mikey!!!\nTry the spam!\n\n")
```

```
> python3 /home/mpavlak/python/Ex01hello.py
Hello Mikey!!!
$
> python3 /home/mpavlak/python/Ex01hello.py

Hello Mikey!!!
Try the spam!

$
```

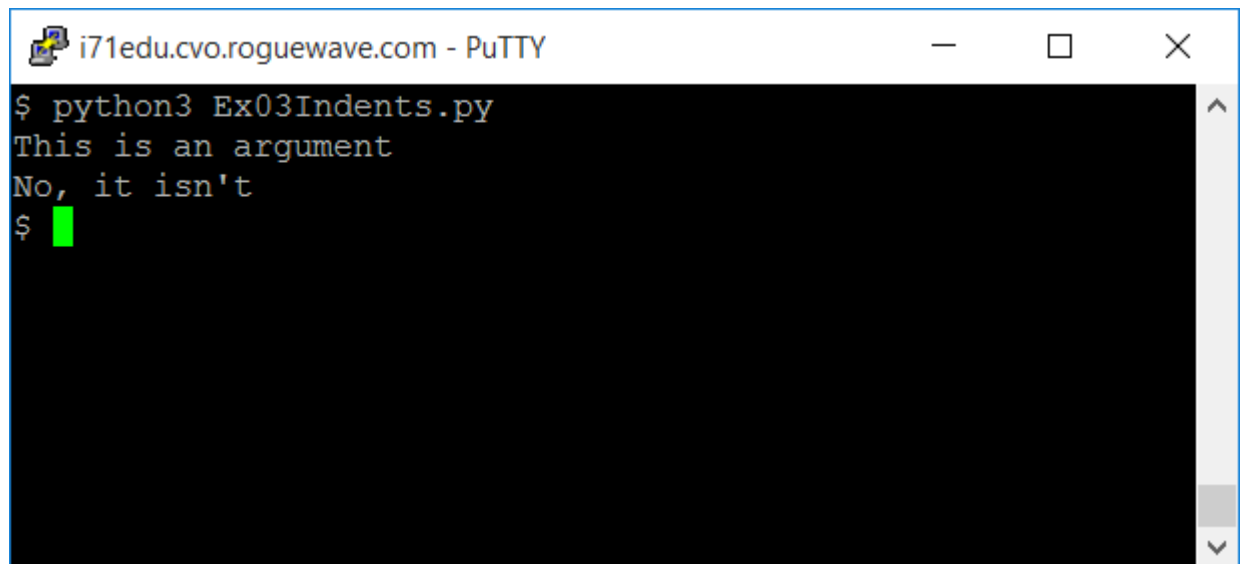
Syntax

How is it written

- Indentation means EVERYTHING
 - Don't use tab
 - 4 spaces is the best practice
 - Mismatched indents can cause failures. Good luck finding...
 - Mismatched spaces and tabs will cause failures
- No need for scope terminators like other languages
- Colon introduces start block, then indent
- Much more readable than other languages
- Get a good editor!!!

Indentation

```
1 #
2 #Indentation example
3 #
4 count = 0
5 argument = True
6 while count < 2:
7     if argument:
8         print ("This is an argument")
9     else:
10        print ("No, it isn't ")
11    argument = False
12    count = count+1
```



```
i71edu.cvo.roguewave.com - PuTTY
$ python3 Ex03Indents.py
This is an argument
No, it isn't
$
```

Operators – Similar to Java, PHP, C, etc.

- Comparison
 - Assignment =
 - Comparison ==
 - Inequality !=
 - Less than <
 - Greater than >
 - Less than or equal to <=
 - Greater than or equal to >=
- Mathematical
 - Addition +
 - Multiplication *
 - Division /
 - Floor division //
 - Modulus %
 - Exponentiation **
- Booleans
 - And
 - Or
 - Not

The air speed velocity

$$V_i = A_o \sqrt{5 \left[\left(\frac{Q_c}{P_o} + 1 \right)^{\frac{2}{7}} - 1 \right]}$$

Of an unladen swallow.

bigredbubbles5

Variables

Data Types – yeah...about that

- Int
 - Integer of unlimited size
- Float
 - System defined precision
- Complex
 - Complex with real and imaginary parts
- Bool
 - TRUE & FALSE

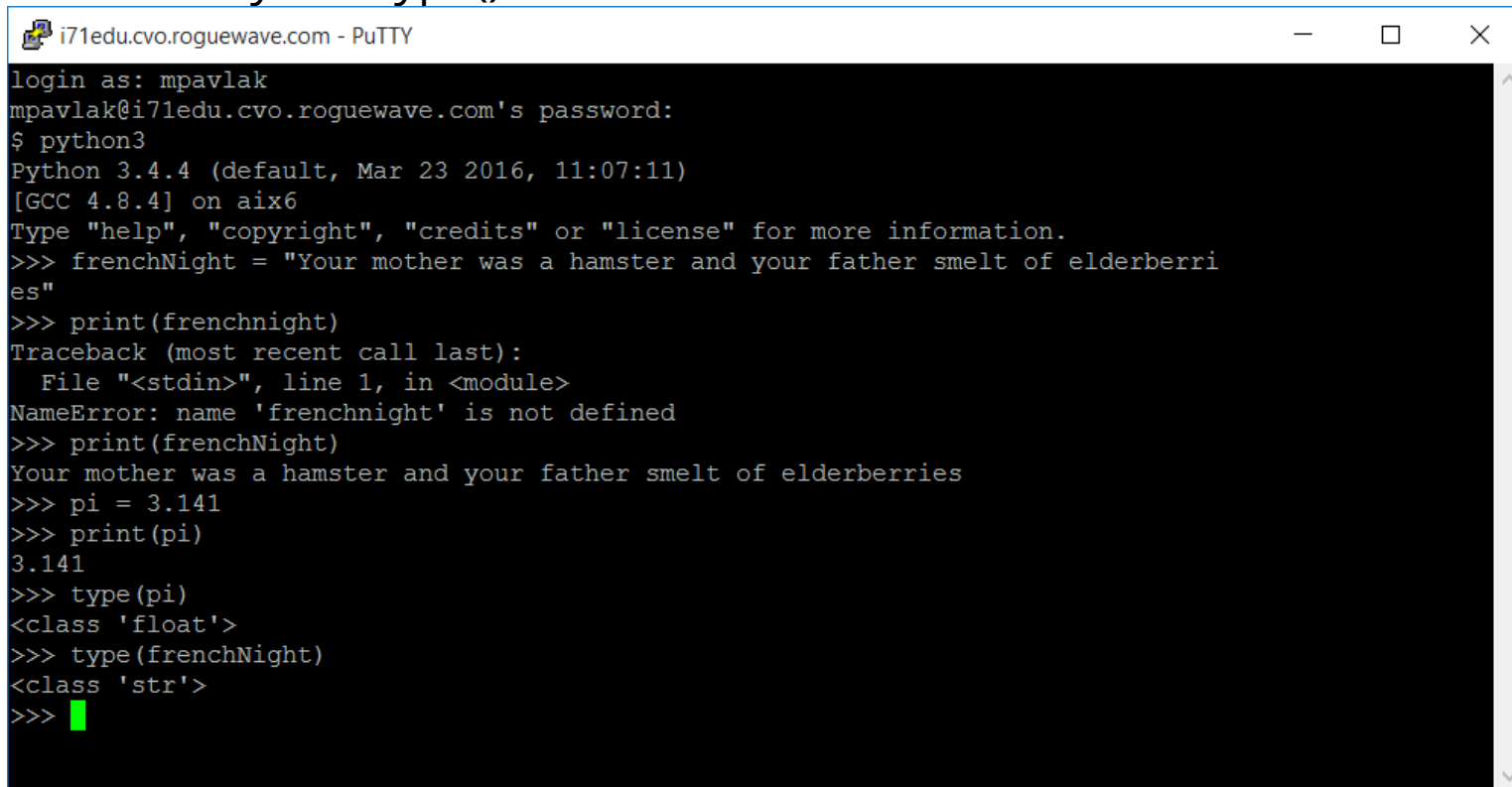
Built in types

- Str
 - Character string composed of Unicode
- Bytes and bytearray
 - Sequences of bytes
- List and tuple (list/array/data structure)

- Range
 - Start, end, step
- Set & frozenset
 - Unordered set of terms
- Dict
 - Associative array (dictionary, hash map)

Variables on the fly

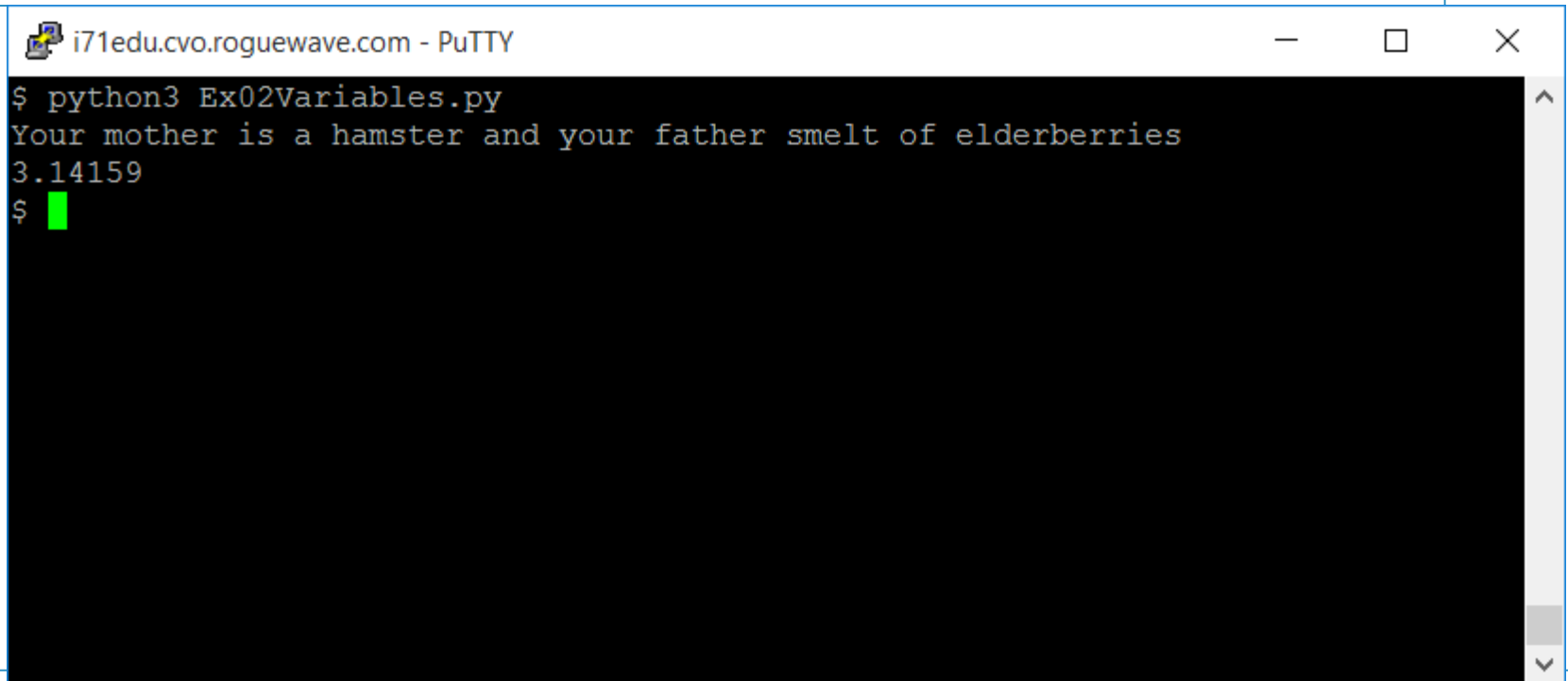
- Case sensitive
- camelCase
- Who are you? type()



```
i71edu.cvo.roguewave.com - PuTTY
login as: mpavlak
mpavlak@i71edu.cvo.roguewave.com's password:
$ python3
Python 3.4.4 (default, Mar 23 2016, 11:07:11)
[GCC 4.8.4] on aix6
Type "help", "copyright", "credits" or "license" for more information.
>>> frenchNight = "Your mother was a hamster and your father smelt of elderberries"
>>> print(frenchnight)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'frenchnight' is not defined
>>> print(frenchNight)
Your mother was a hamster and your father smelt of elderberries
>>> pi = 3.141
>>> print(pi)
3.141
>>> type(pi)
<class 'float'>
>>> type(frenchNight)
<class 'str'>
>>>
```

Variables in a file

```
1 #  
2 # Variables are defined on the fly...  
3 #  
4 frenchKnight = "Your mother is a hamster and your father smelt of elderberries"  
5 pi = 3.14159  
6  
7 print(frenchKnight)  
8 print(pi)
```

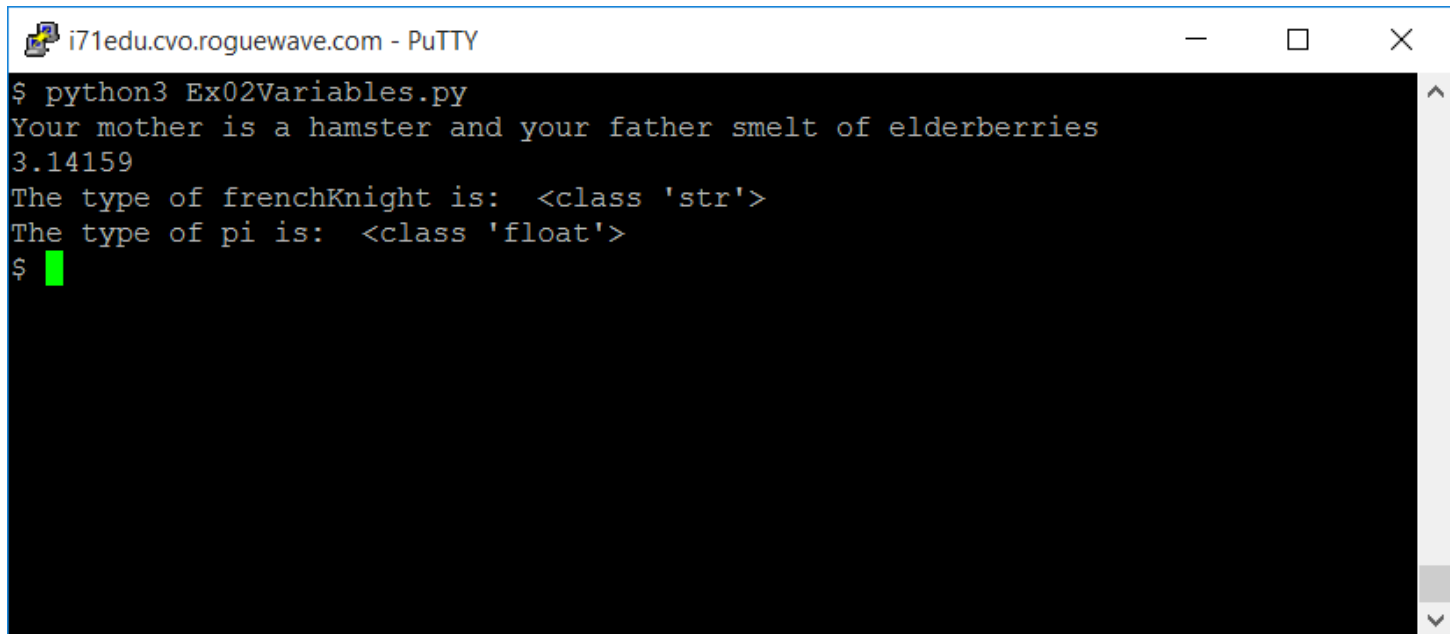


The screenshot shows a PuTTY terminal window titled "i71edu.cvo.roguewave.com - PuTTY". The terminal displays the following output:

```
$ python3 Ex02Variables.py  
Your mother is a hamster and your father smelt of elderberries  
3.14159  
$
```

Data Type?

```
1 #  
2 # Variables are defined on the fly...  
3 #  
4 frenchKnight = "Your mother is a hamster and your father smelt of elderberries"  
5 pi = 3.14159  
6  
7 print(frenchKnight)  
8 print(pi)  
9  
10 print("The type of frenchKnight is: ", type(frenchKnight))  
11 print("The type of pi is: ", type(pi))
```



```
i71edu.cvo.roguewave.com - PuTTY  
$ python3 Ex02Variables.py  
Your mother is a hamster and your father smelt of elderberries  
3.14159  
The type of frenchKnight is: <class 'str'>  
The type of pi is: <class 'float'>  
$
```


Every variable is implemented as a class!



And now for something completely different



It's OK....

- Monty Python references are not only acceptable...
 - They are encouraged!
- Documentation is littered with references
- Examples are well covered



Back to Variables

- Numbrs - 3 Data Types
 - Int 1,2,42
 - Float 3.14159
 - Complex: <real> + <imaginary> (not used much...)

Strings

- Immutable objects, cannot change value
- Can reassign. (dynamic typing)
- Single or Double quotes, OK (even triple...)
- Index starts at 0



String formatting

- Interpolation, of sorts

```
1 #
2 # String example
3 #
4
5 count = 0
6 while count < 6:
7     string1 = "I have {} dead parrots!".format(count)
8     print(string1)
9     count = count+1
10 print("\nThank you for shopping!")
```



```
i71edu.cvo.roguewave.com - PuTTY
$ python3 Ex04Strings.py
I have 0 dead parrots!
I have 1 dead parrots!
I have 2 dead parrots!
I have 3 dead parrots!
I have 4 dead parrots!
I have 5 dead parrots!

Thank you for shopping!
$
```

Lists

- Ordered group, similar to array
- Different data types, ok
- Multi-dimensional (sub lists)
- Mutable (changeable)

```
1 #  
2 # List ExampleService  
3 #  
4 mylist = ["Rock Bottom", "Gordon Biersch", "BJ's", "Granite City"]  
5 print(mylist[1])  
6  
7 print(mylist[0:2])  
8  
9 print(mylist)
```

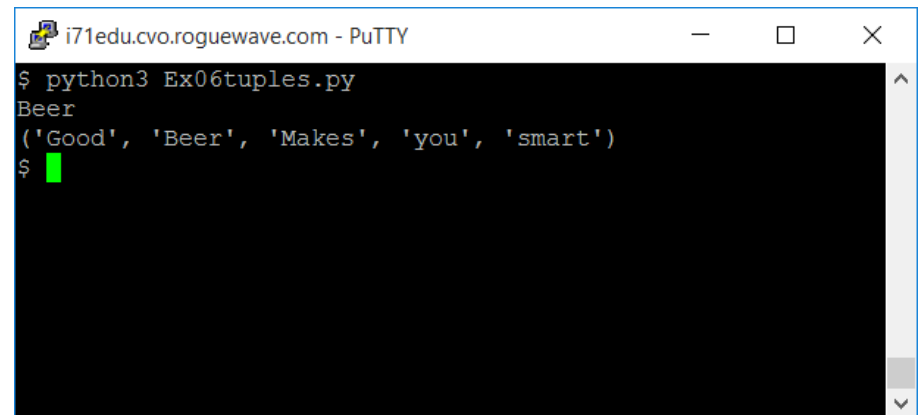


```
i71edu.cvo.roguewave.com - PuTTY  
$ python3 Ex05Lists.py  
Gordon Biersch  
['Rock Bottom', 'Gordon Biersch']  
['Rock Bottom', 'Gordon Biersch', 'BJ's', 'Granite City']  
$
```

Tuples

- Similar to lists
- Immutable (don't change once created)
- Use parenthesis instead of brackets

```
1 #  
2 # Tuples Examples  
3 #  
4  
5 mytuple = ("Good", "Beer", "Makes", "you", "smart")  
6 print(mytuple[1])  
7 print(mytuple)
```

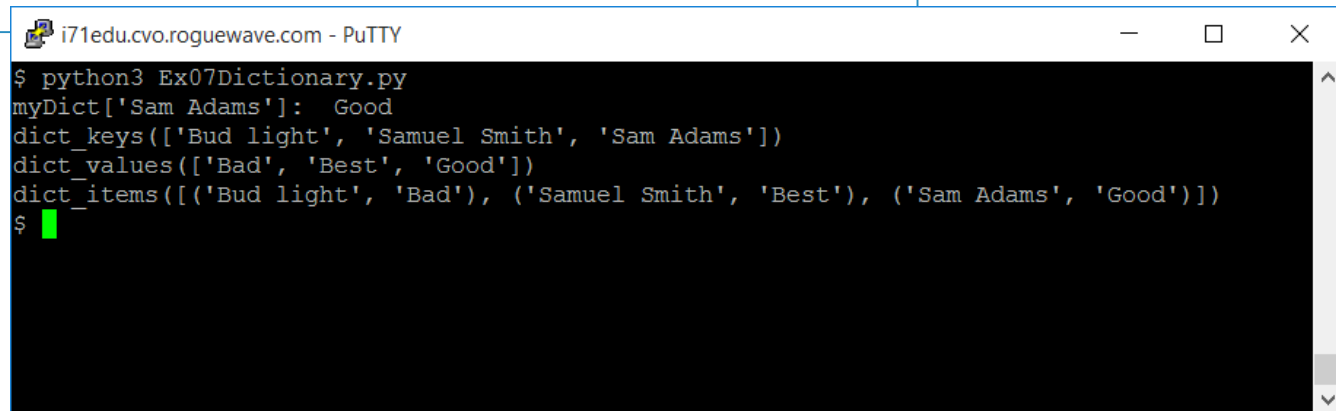


```
i71edu.cvo.roguewave.com - PuTTY  
$ python3 Ex06tuples.py  
Beer  
( 'Good', 'Beer', 'Makes', 'you', 'smart' )  
$
```


Dictionary

- Again, like lists but more like hash table
- Mutable
- Key value pairs

```
1 #
2 # Dictionary Examples
3 #
4
5 myDict = {"Sam Adams": "Good", "Samuel Smith": "Best", "Bud light": "Bad"}
6
7 print("myDict['Sam Adams']: ", myDict["Sam Adams"])
8
9 print(myDict.keys())
10 print(myDict.values())
11 print(myDict.items())
```

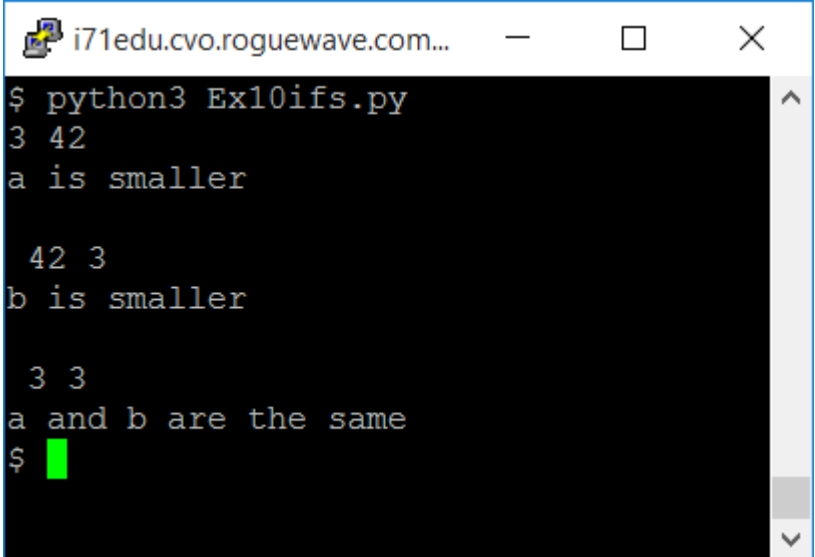


```
i71edu.cvo.roguewave.com - PuTTY
$ python3 Ex07Dictionary.py
myDict['Sam Adams']: Good
dict_keys(['Bud light', 'Samuel Smith', 'Sam Adams'])
dict_values(['Bad', 'Best', 'Good'])
dict_items([('Bud light', 'Bad'), ('Samuel Smith', 'Best'), ('Sam Adams', 'Good')])
$
```

Control Structures

Iifs

```
1 #
2 # If examples
3 #
4 a,b = 3,42
5 print(a,b)
6 if a < b:
7     print("a is smaller")
8
9 a,b = 42,3
10 print("\n",a,b)
11 if a < b:
12     print("a is smaller")
13 else:
14     print("b is smaller")
15
16 a,b = 3,3
17 print("\n",a,b)
18 if a < b:
19     print("a is smaller")
20 elif a > b:
21     print("b is smaller")
22 else:
23     print("a and b are the same")
```



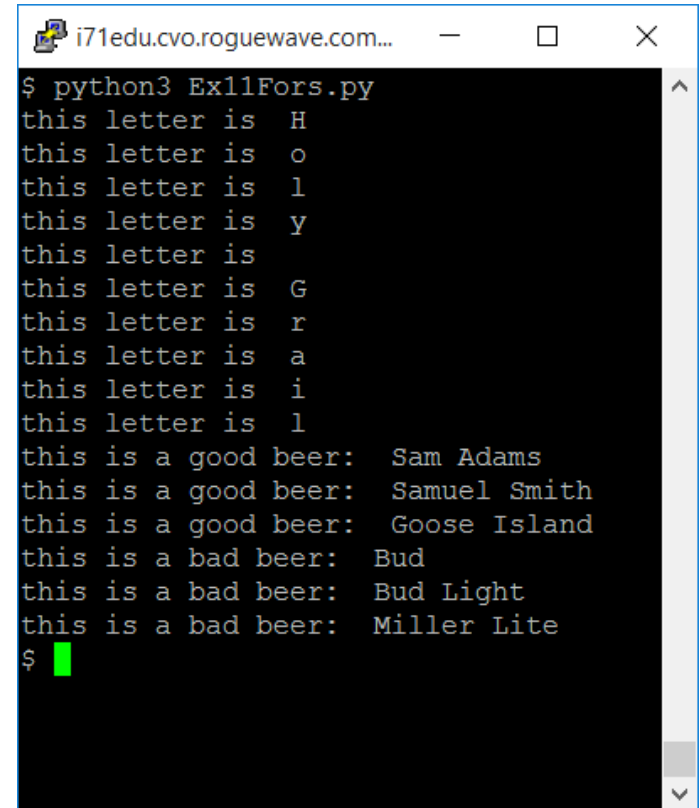
```
i71edu.cvo.roguewave.com...
$ python3 Ex10ifs.py
3 42
a is smaller

42 3
b is smaller

3 3
a and b are the same
$
```

For loop

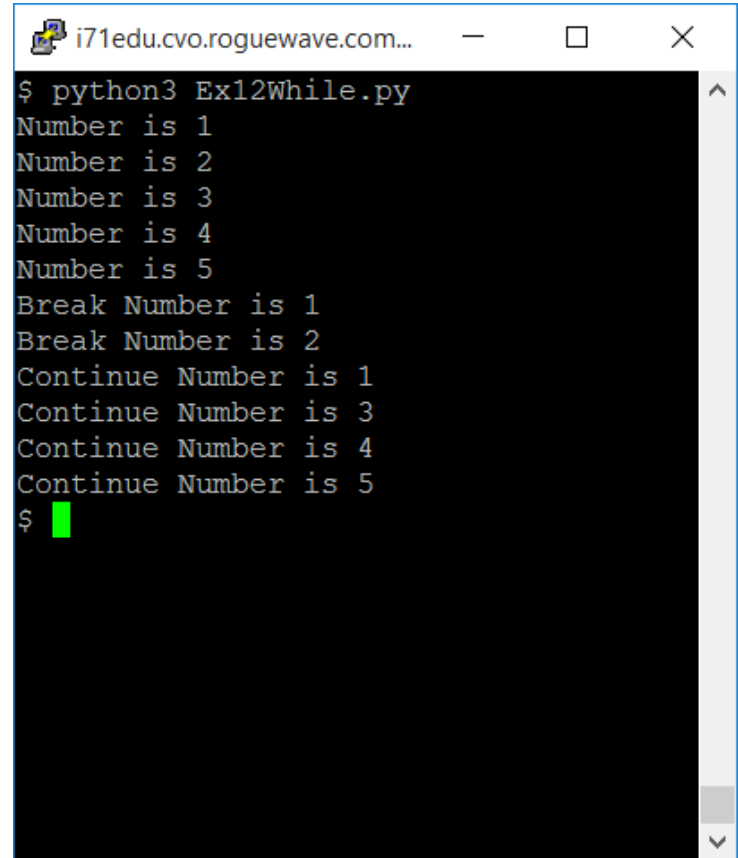
```
1 #
2 # For Loop Examples
3 #
4
5 myString = "Holy Grail"
6 for letter in myString:
7     print("this letter is ", letter)
8
9 beers = ["Sam Adams", "Samuel Smith", "Goose Island"]
10 for beer in beers:
11     print("this is a good beer: ", beer)
12
13 badBeers = ["Bud", "Bud Light", "Miller Lite"]
14 for index in range(len(beers)): #iterates 0 thru 2
15     print("this is a bad beer: ", badBeers[index])
```



```
i71edu.cvo.roguewave.com...
$ python3 Ex11Fors.py
this letter is H
this letter is o
this letter is l
this letter is y
this letter is G
this letter is r
this letter is a
this letter is i
this letter is l
this is a good beer: Sam Adams
this is a good beer: Samuel Smith
this is a good beer: Goose Island
this is a bad beer: Bud
this is a bad beer: Bud Light
this is a bad beer: Miller Lite
$
```

While loop

```
1 #
2 # While Loop Examples
3 #
4
5 count, limit = 0,5
6 while count < limit:
7     count = count+1
8     print("Number is", count)
9
10 count = 0
11 while count < limit:
12     count = count+1
13     if count==3:
14         break
15     print("Break Number is", count)
16
17
18 count = 0
19 while count < limit:
20     count = count+1
21     if count==2:
22         continue
23     print("Continue Number is", count)
```

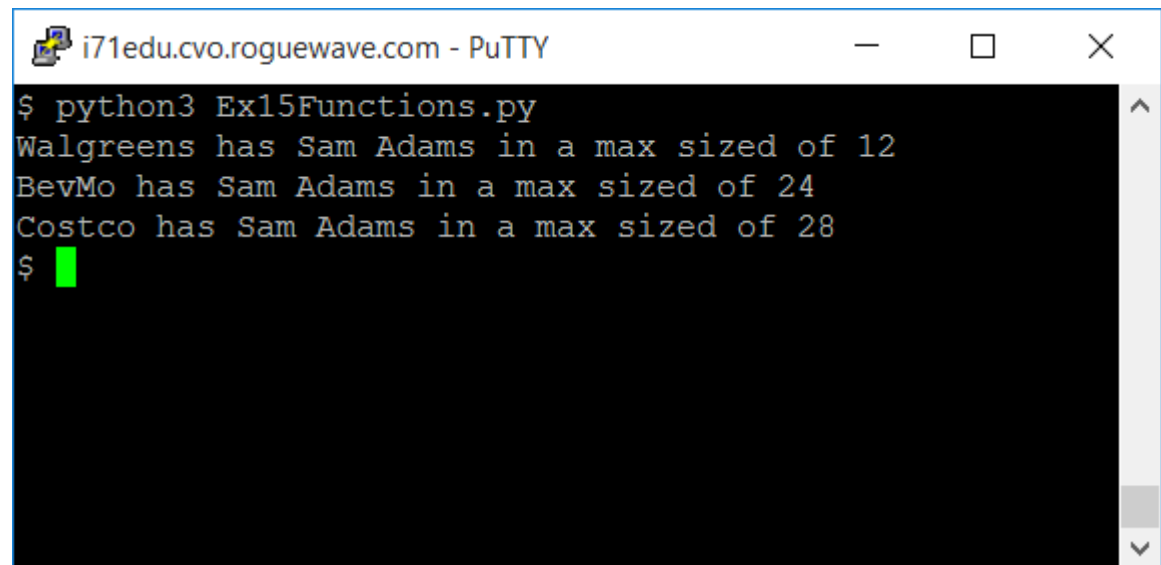


```
i71edu.cvo.roguewave.com...
$ python3 Ex12While.py
Number is 1
Number is 2
Number is 3
Number is 4
Number is 5
Break Number is 1
Break Number is 2
Continue Number is 1
Continue Number is 3
Continue Number is 4
Continue Number is 5
$
```

Functions

Basic functions

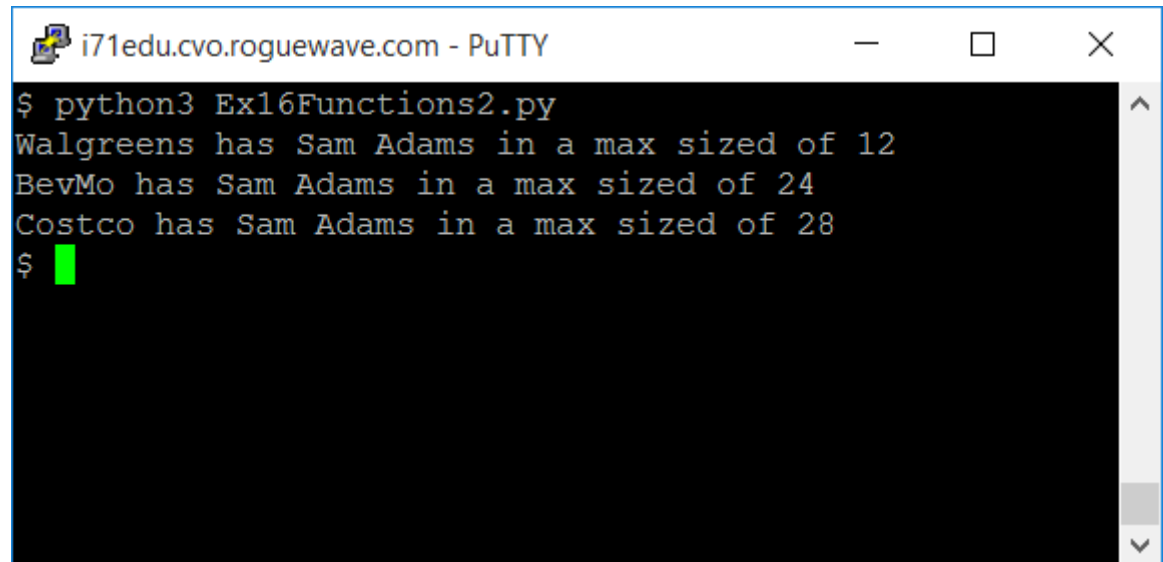
```
1 #
2 # Function Examples
3 #
4
5 def printBeer(store, beer, size):
6     print(store + " has " + beer + " in a max sized of " + str(size) )
7
8 myBeer = "Sam Adams"
9 printBeer("Walgreens", myBeer, 12)
10 printBeer("BevMo", myBeer, 24)
11 printBeer("Costco", myBeer, 28)
```



```
i71edu.cvo.roguewave.com - PuTTY
$ python3 Ex15Functions.py
Walgreens has Sam Adams in a max sized of 12
BevMo has Sam Adams in a max sized of 24
Costco has Sam Adams in a max sized of 28
$
```

Functions with defaults

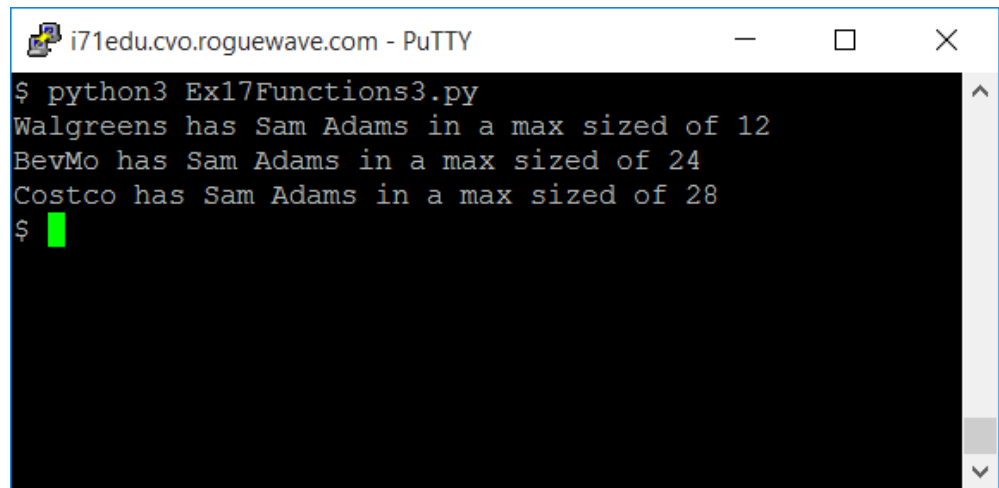
```
1 #
2 # Function Examples
3 #
4
5 def printBeer(store, beer, size=24):
6     print(store + " has " + beer + " in a max sized of " + str(size) )
7
8 myBeer = "Sam Adams"
9 printBeer("Walgreens", myBeer, 12)
10 printBeer("BevMo", myBeer)
11 printBeer("Costco", myBeer, 28)
```



```
i71edu.cvo.roguewave.com - PuTTY
$ python3 Ex16Functions2.py
Walgreens has Sam Adams in a max sized of 12
BevMo has Sam Adams in a max sized of 24
Costco has Sam Adams in a max sized of 28
$
```


Functions with keyword arguments

```
1 #
2 # Function Examples
3 #
4
5 def printBeer(store, beer, size):
6     print(store + " has " + beer + " in a max sized of " + str(size) )
7
8 myBeer = "Sam Adams"
9 printBeer("Walgreens", myBeer, 12)
10 printBeer(beer=myBeer, size=24, store="BevMo")
11 printBeer(beer=myBeer, store="Costco", size=28)
```



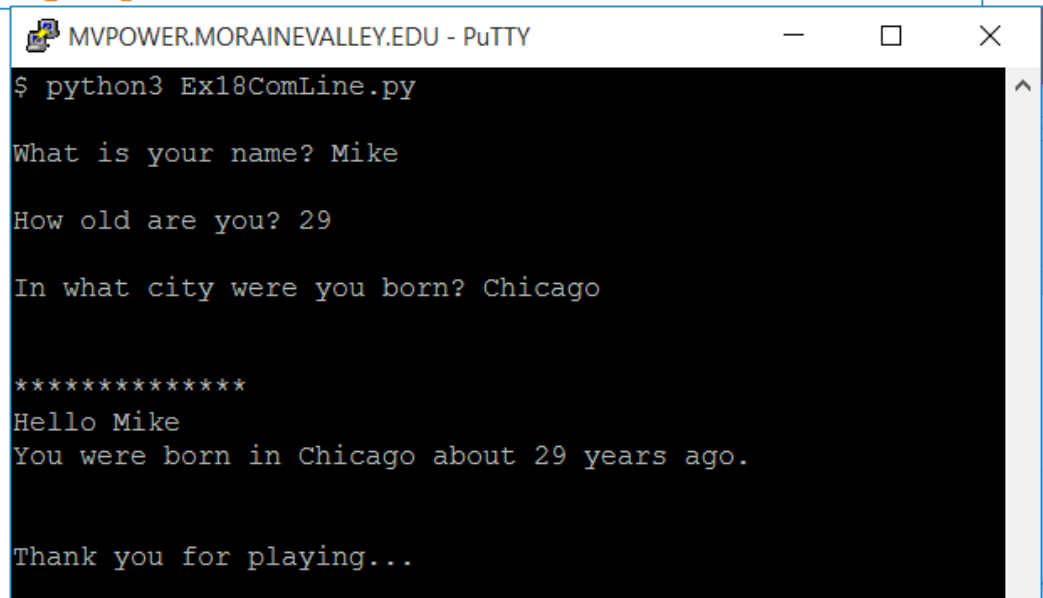
```
i71edu.cvo.roguewave.com - PuTTY
$ python3 Ex17Functions3.py
Walgreens has Sam Adams in a max sized of 12
BevMo has Sam Adams in a max sized of 24
Costco has Sam Adams in a max sized of 28
$
```

Command Line

Input from command line

- Talk with the script

```
1 # Get input from user and then embed in string
2 from pip._vendor.distlib.compat import raw_input
3
4 name = raw_input("\nWhat is your name? ")
5 age = raw_input("\nHow old are you? ")
6 city = raw_input("\nIn what city were you born? ")
7 print("\n\n*****")
8 print("Hello %s" % (name))
9 print("You were born in %s about %s years ago." % (city, str(age)))
10 print("\n\nThank you for playing...\n\n")
```



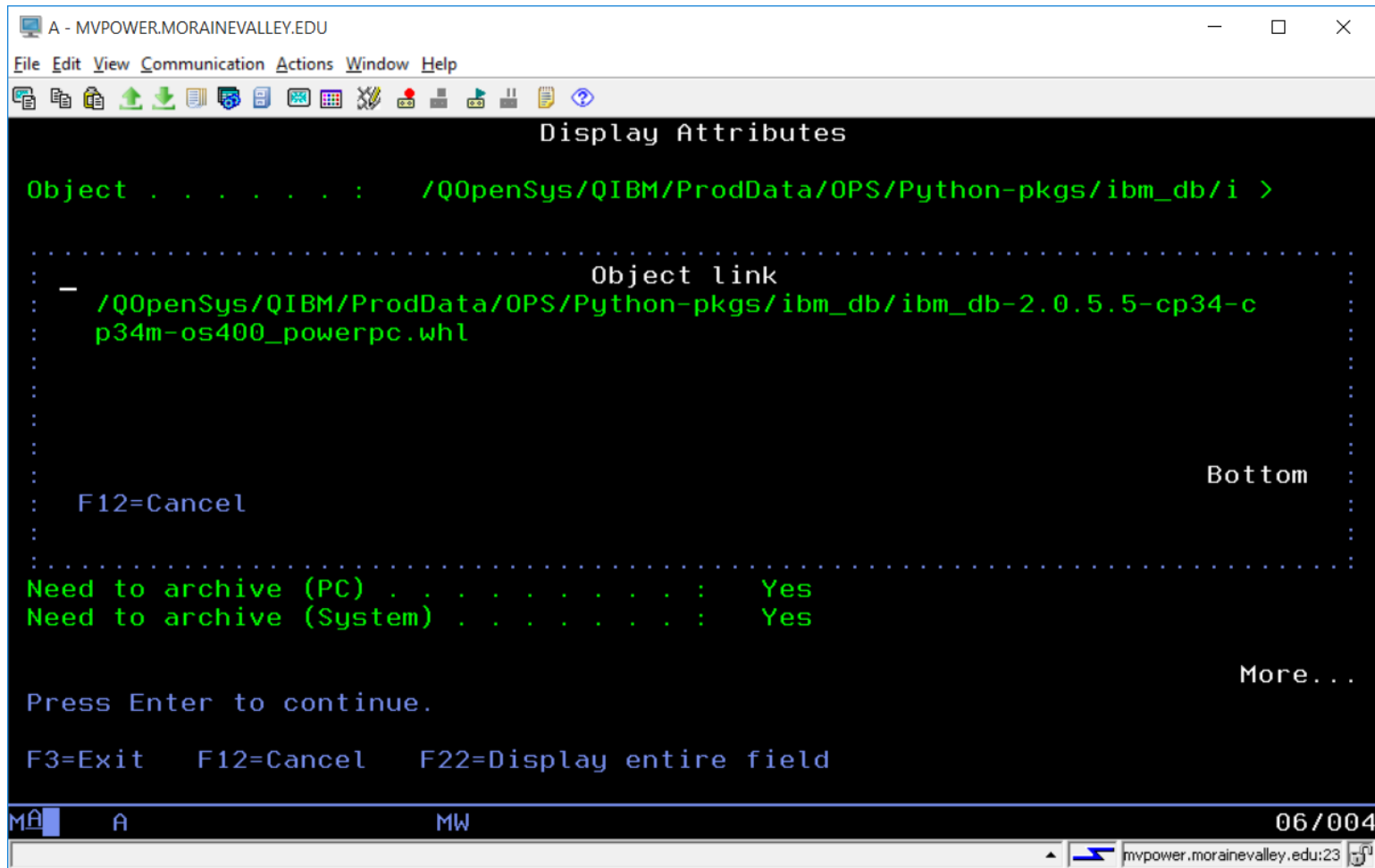
```
MVPOWER.MORAINEVALLEY.EDU - PuTTY
$ python3 Ex18ComLine.py
What is your name? Mike
How old are you? 29
In what city were you born? Chicago

*****
Hello Mike
You were born in Chicago about 29 years ago.

Thank you for playing...
```

Database

Locate the package or “wheel”



Install commands

Installing shipped add-ons

5733-OPS Option 2 and Option 4 come with several add-on packages (shipped via separate [PTFs](#)). Installation of these add-ons is easy, just use the applicable command.

If you're on a recent PTF level, all the packages should now be in wheel format (*.whl). Previous versions used egg format (*.egg). If you want to know the nitty-gritty details of why wheels are better than eggs and why we switched, click [this link](#). Otherwise, just know that wheels are better in every way except name.

New way, with wheels:

(for Python 3)

To install the native DB2 connector:

```
pip3 install /QOpenSys/QIBM/ProdData/OPS/Python-pkgs/ibm_db/ibm_db-*-cp34m-*.whl
```

To install the DB2 Django interface:

```
pip3 install --no-deps /QOpenSys/QIBM/ProdData/OPS/Python-pkgs/ibm_db/ibm_db_django-*-py3-*.whl
```

To install the Toolkit for IBM i:

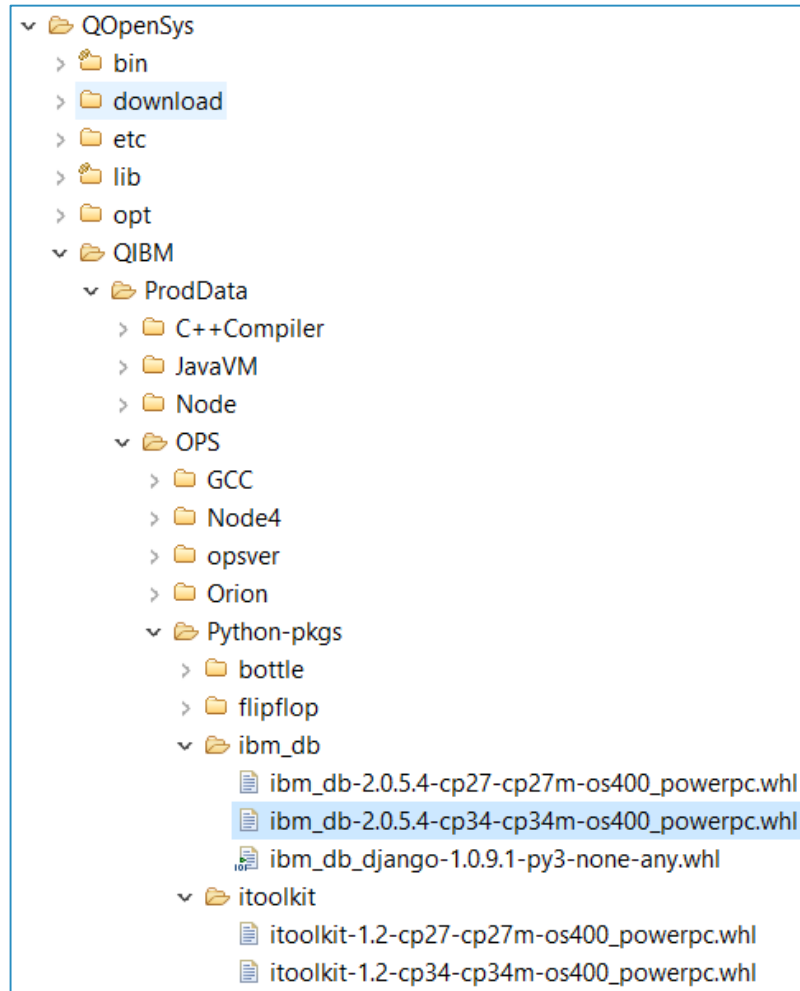
```
pip3 install /QOpenSys/QIBM/ProdData/OPS/Python-pkgs/itoolkit/itoolkit-*-cp34m-*.whl
```

To install FastCGI gateway support:

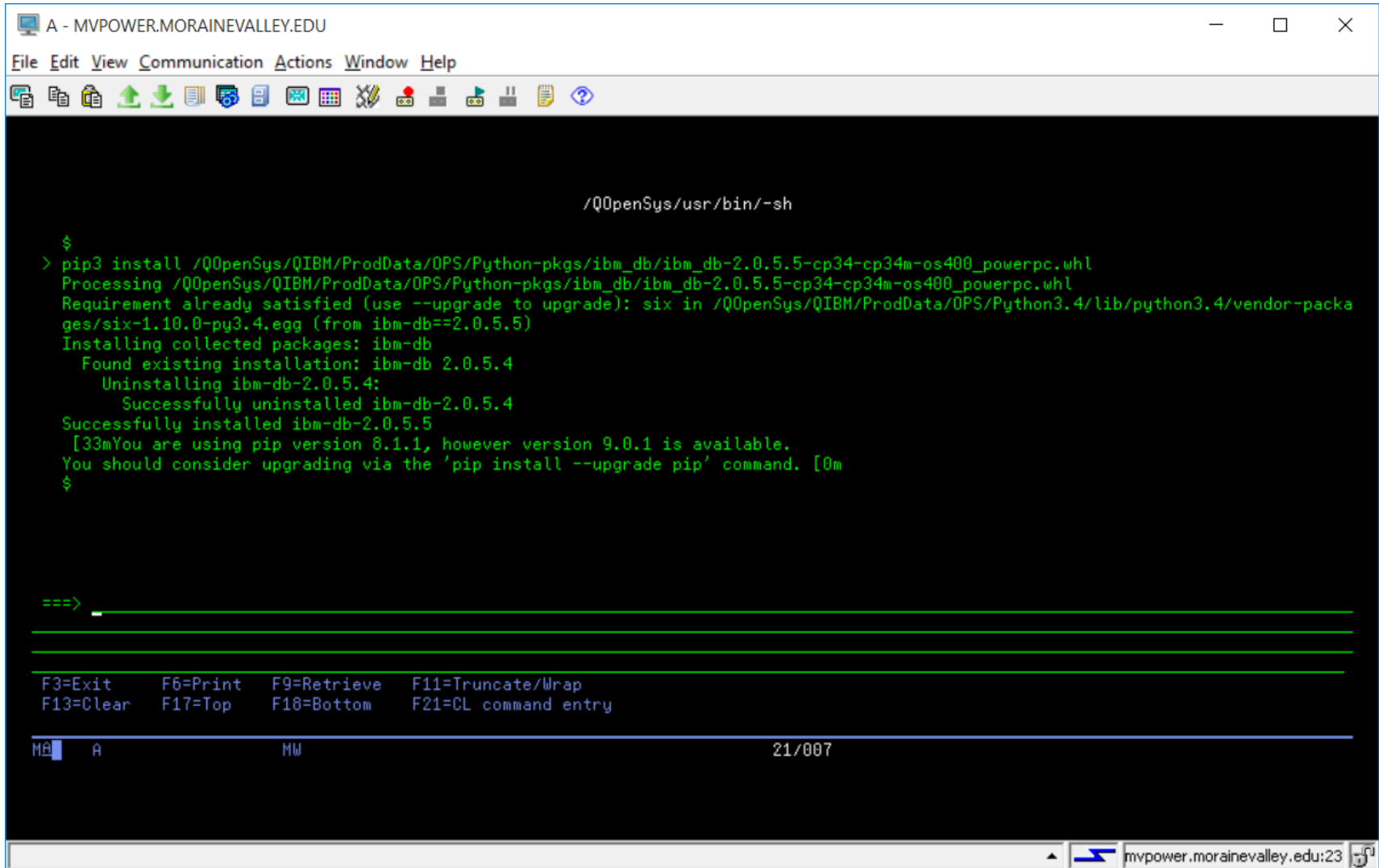
```
pip3 install /QOpenSys/QIBM/ProdData/OPS/Python-pkgs/flipflop/flipflop-*-py34-*.whl
```

Find the connector

- YMMV
- With wheels



Run the pip install...



The screenshot shows a terminal window titled "A - MVPOWER.MORAINEVALLEY.EDU". The terminal prompt is `/QOpenSys/usr/bin/-sh`. The user enters the command `pip3 install /QOpenSys/QIBM/ProdData/OPS/Python-pkgs/ibm_db/ibm_db-2.0.5.5-cp34-cp34m-os400_powerpc.whl`. The output shows that the requirement for `six` is already satisfied. The `ibm-db` package is installed, and the previous version `2.0.5.4` is uninstalled. A message indicates that `pip` version `8.1.1` is being used, but `9.0.1` is available. The terminal also displays a list of function key shortcuts and a status bar at the bottom.

```
$
> pip3 install /QOpenSys/QIBM/ProdData/OPS/Python-pkgs/ibm_db/ibm_db-2.0.5.5-cp34-cp34m-os400_powerpc.whl
Processing /QOpenSys/QIBM/ProdData/OPS/Python-pkgs/ibm_db/ibm_db-2.0.5.5-cp34-cp34m-os400_powerpc.whl
Requirement already satisfied (use --upgrade to upgrade): six in /QOpenSys/QIBM/ProdData/OPS/Python3.4/lib/python3.4/vendor-packa
ges/six-1.10.0-py3.4.egg (from ibm-db==2.0.5.5)
Installing collected packages: ibm-db
  Found existing installation: ibm-db 2.0.5.4
  Uninstalling ibm-db-2.0.5.4:
    Successfully uninstalled ibm-db-2.0.5.4
  Successfully installed ibm-db-2.0.5.5
[33mYou are using pip version 8.1.1, however version 9.0.1 is available.
You should consider upgrading via the 'pip install --upgrade pip' command. [0m
$

===>

F3=Exit      F6=Print     F9=Retrieve  F11=Truncate/Wrap
F13=Clear    F17=Top      F18=Bottom   F21=CL command entry

MB A MW 21/007
```


What version of the DB2 Extension?

```
1 import ibm_db_dbi as dbi
2
3 print(dbi.__version__)
```

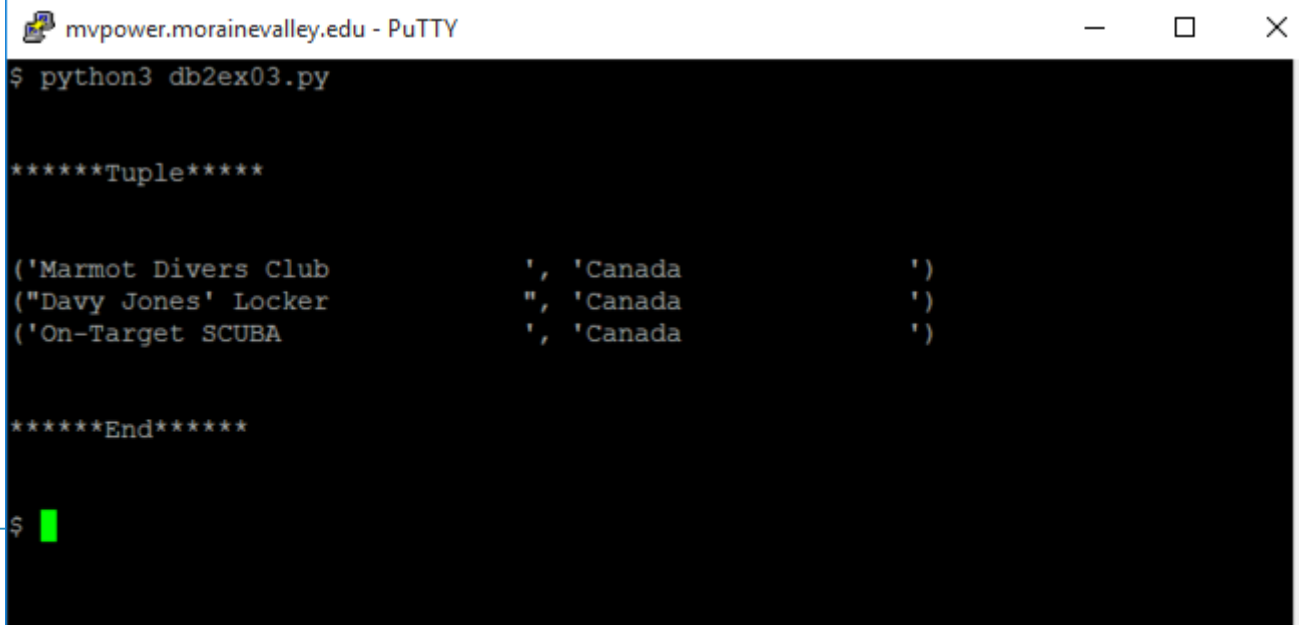
```
$
> python3 /home/mpavlak/python/db2/db2ex01.py
2.0.5.5
$
```

Simple database access

- Import the class
- Connect (with or without options)
- Open the cursor
- Set the SQL
- Read

Simple database access

```
1 import ibm_db_dbi as dbi
2 conn = dbi.connect()
3 sql = "SELECT COMPANY, COUNTRY FROM samples.SP_CUST where country = 'US'"
4 c01 = conn.cursor()
5 c01.execute(sql)
6 #Bring it in as tuple
7 print("\n\n*****Tuple*****\n\n")
8
9 for row in c01.fetchall():
10     print(row)
11 c01.close()
12 conn.close()
13 print("\n\n*****End*****\n\n")
```



```
mvpower.morainevalley.edu - PuTTY
$ python3 db2ex03.py

*****Tuple*****


('Marmot Divers Club', 'Canada')
('Davy Jones' Locker', 'Canada')
('On-Target SCUBA', 'Canada')

*****End*****

$
```

Table information

```
1 import ibm_db_dbi as dbi
2 conn = dbi.connect()
3 sql = "SELECT COMPANY, COUNTRY FROM ZENDSVR6.SP_CUST where country = 'Canada'"
4 c01 = conn.cursor()
5 c01.execute(sql)
6 desc = c01.description
7 print(desc[0][0], desc[0][4], "\n")
8 print(desc[1][0], desc[1][4], "\n")
9
10 #Bring it in as tuple
11 print("\n\n*****Tuple*****\n\n")
12 for row in c01.fetchall():
13     print(row)
14 c01.close()
15 conn.close()
16 print("\n\n*****End*****\n\n")
```

 mvpower.morainevalley.edu - PuTTY

```
$ python3 db2ex04.py
```

```
COMPANY 30
```

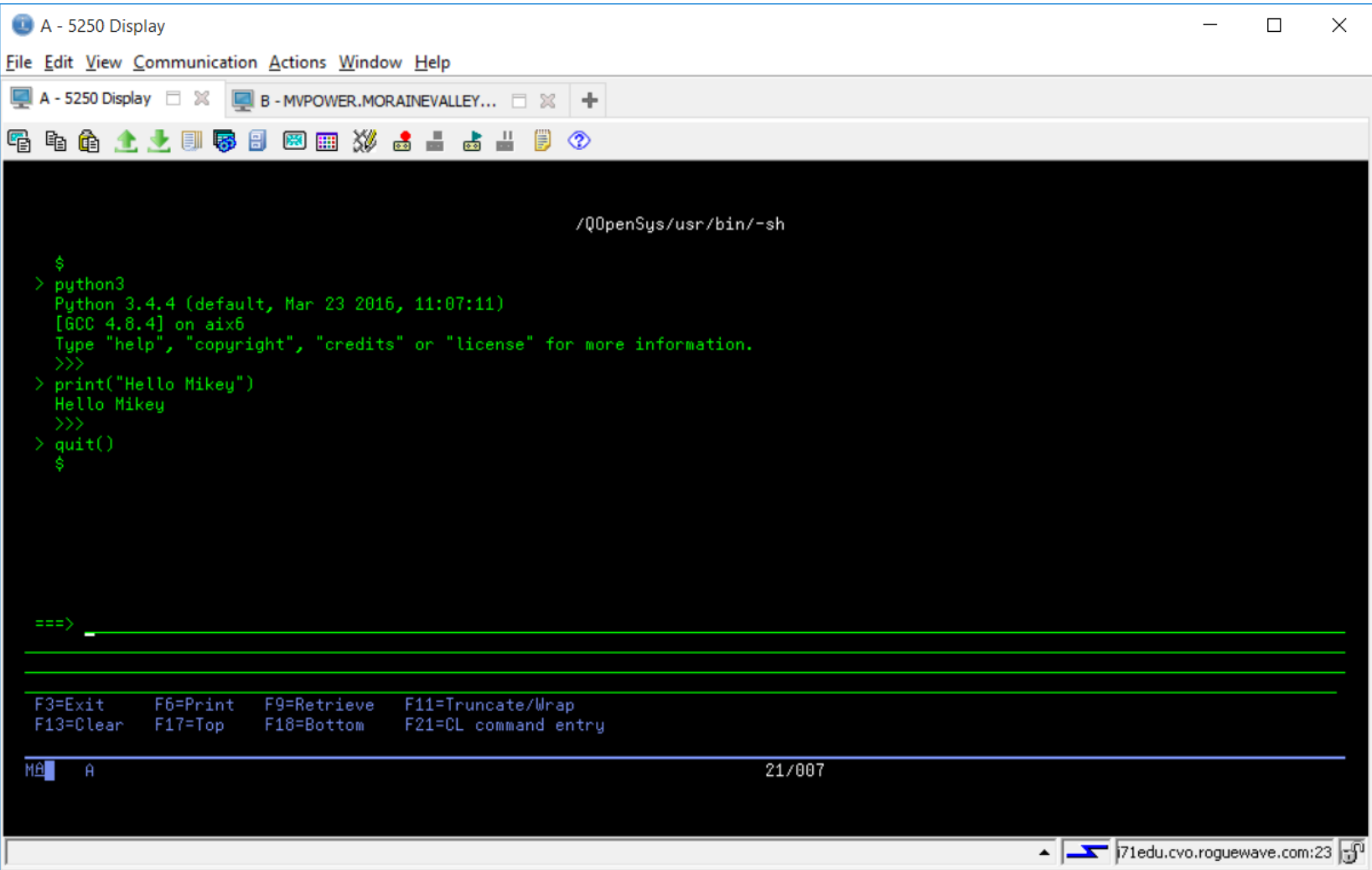
```
COUNTRY 20
```

```
*****Tuple*****
```

Summary – Why Python?

- Lot's of libraries
- Make it easy to do stuff
- OPC / OPO
- Education

End the session



The screenshot shows a terminal window titled "A - 5250 Display". The window has a menu bar with "File", "Edit", "View", "Communication", "Actions", "Window", and "Help". Below the menu bar is a toolbar with various icons. The main area of the terminal is black with green text. The prompt is "/QOpenSys/usr/bin/-sh". The user has entered the following commands and received the following output:

```
/$
> python3
Python 3.4.4 (default, Mar 23 2016, 11:07:11)
[GCC 4.8.4] on aix6
Type "help", "copyright", "credits" or "license" for more information.
>>>
> print("Hello Mikey")
Hello Mikey
>>>
> quit()
/$
```

Below the terminal output, there is a red prompt "===>" followed by a horizontal line. At the bottom of the terminal, there is a status bar with the following text:

```
F3=Exit    F6=Print  F9=Retrieve  F11=Truncate/Wrap
F13=Clear  F17=Top   F18=Bottom  F21=CL command entry
```

At the very bottom of the window, there is a taskbar with the text "M& A" and "21/007".

zendcon2017

Accelerating PHP & Open Source Oct 23-26, Las Vegas



ATTEND

Connect with experts.
Become an authority.
Register now:

zendcon.com/register-now



EXPLORE

Share, inspire, educate.
Network



SPONSOR

Spotlight your best in
enterprise PHP, open
source & frameworks:

sponsors@zendcon.com

Visit zendcon.com



Who was that masked man?

Mike Pavlak – Solution Consultant, Rogue Wave Software

Mike.Pavlak@roguewave.com

Thank You!



 **RogueWave**
S O F T W A R E
Accelerating Great Code