



# FRESCHÉ SOLUTIONS

## Introduction to Python for IBM i

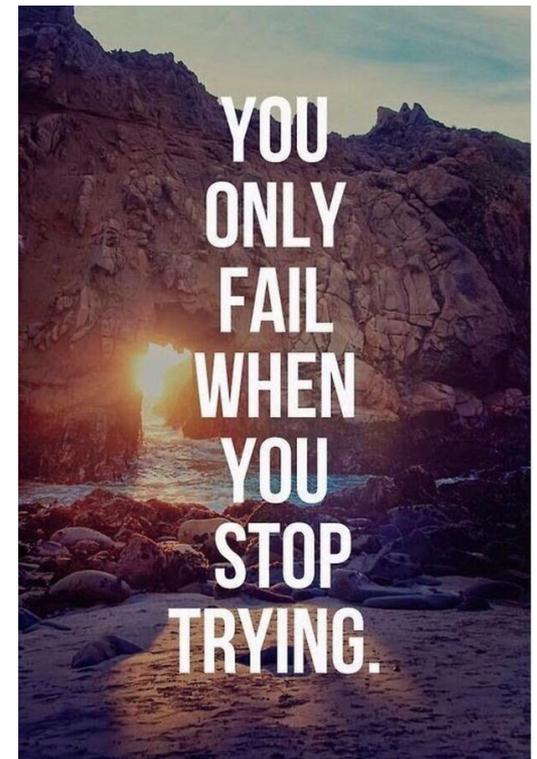
Mike Pavlak – IT Strategist

[mike.pavlak@freschesolutions.com](mailto:mike.pavlak@freschesolutions.com)

# Agenda

---

- A little about Python
- Why use Python
- How to install/determine if installed
  - ▣ IDE
- Syntax 101
  - ▣ Variables
  - ▣ Strings
  - ▣ Functions
- Database
- Toolkit



# Acknowledgements

---

- Kevin Adler
- Tony Cairns
- Jesse Gorzinski
- Google
- Memegenerator
- Corn chips and salsa
- Parrots
- And, of course,  
spam

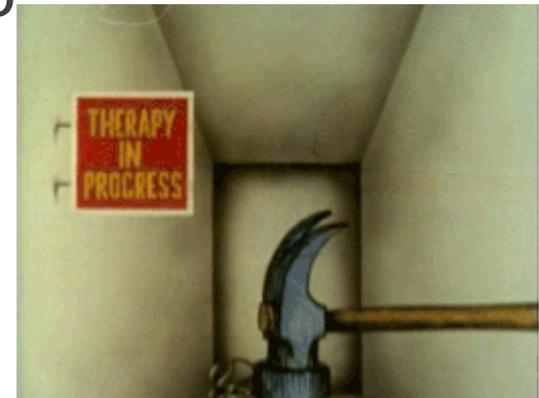


# A little about Python

# What is it, really?

---

- General purpose language
- Easy to get started
- Simple syntax
- Great for integrations (glue between systems)
- Access to C and other APIs
- Infrastructure first, but applications, too



# Historically...

- Python was conceptualized by **Guido Van Rossum** in the late 1980's
- Rossum published the first version of Python code (0.9.0) in February of 1991 at the CWI(Centrum Wiskunde & Informatica) in the Netherlands, Amsterdam
- Python is derived from the ABC programming language, which is a general purpose language that was also developed at CWI.
- Rossum chose the name “Python” since he was a 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 and as leading “commitor”.



The Python programming language <https://www.python.org/>

99,953 commits    9 branches    331 releases    356 contributors

Branch: master    New pull request    Find file    Close

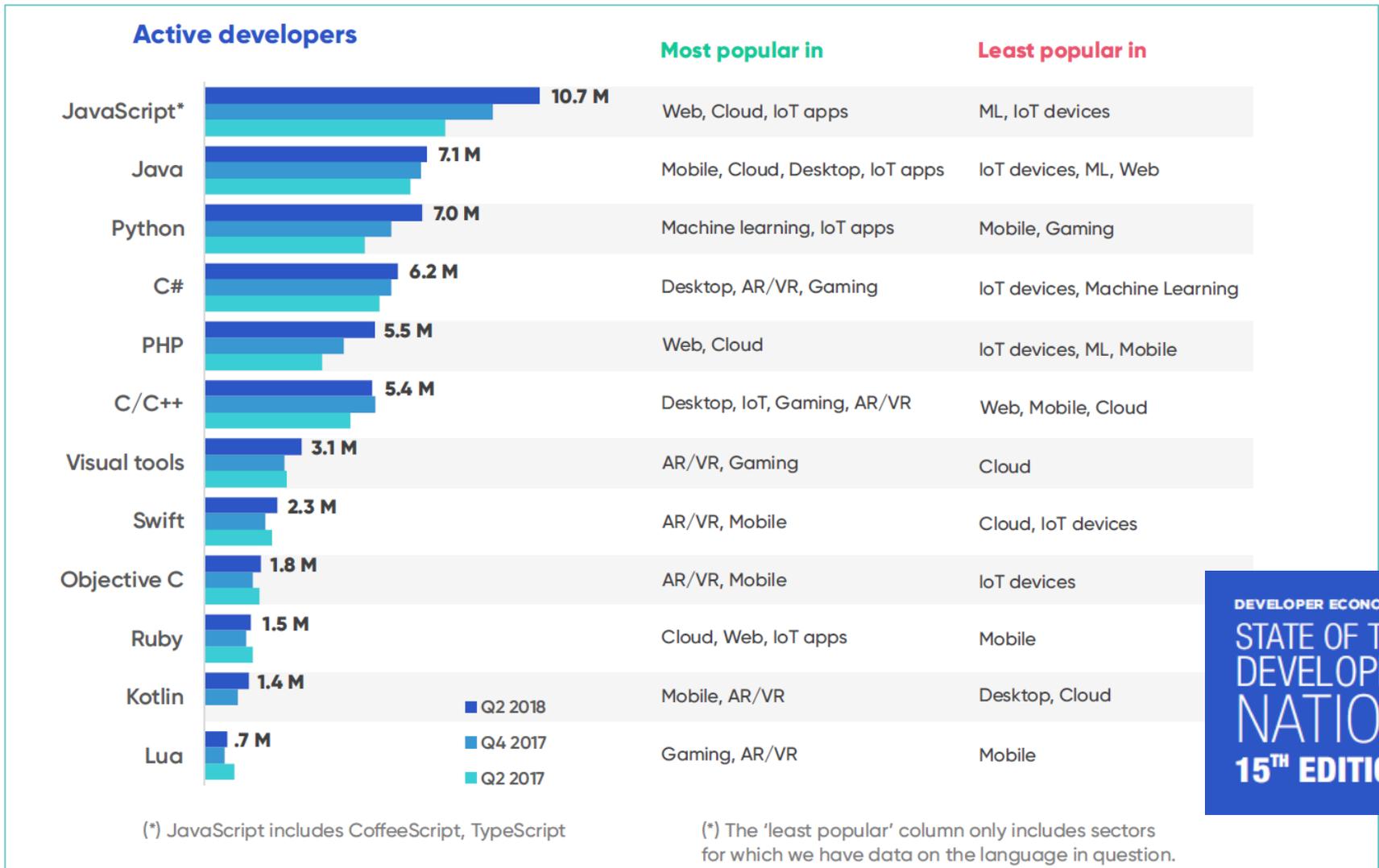
haypo committed on GitHub bpo-31234: Enhance test\_thread.test\_forkinthread() (#3516)    Latest commit a15d155 5 hours ago

.github	Create PULL_REQUEST_TEMPLATE.md (GH-3404)	6 days ago
Doc	bpo-31421: Document how IDLE runs tkinter programs. (#3513)	10 hours ago

Look 4 this

# Why Python?

# / Data survey 40,000 developers, worldwide



# Got Python?

# Details at Developerworks



The sun is setting on OPS!

## Open Source Technologies on IBM i

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

### 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	Description
ibm_db	DB2 for i connector - Allows native access to DB2 for i.
itoolkit	Toolkit for IBM i - allows access to system resources through program calls, SQL queries, CL commands, shell commands, and more.
flipflop	FastCGI gateway
bottle	Lightweight web framework.

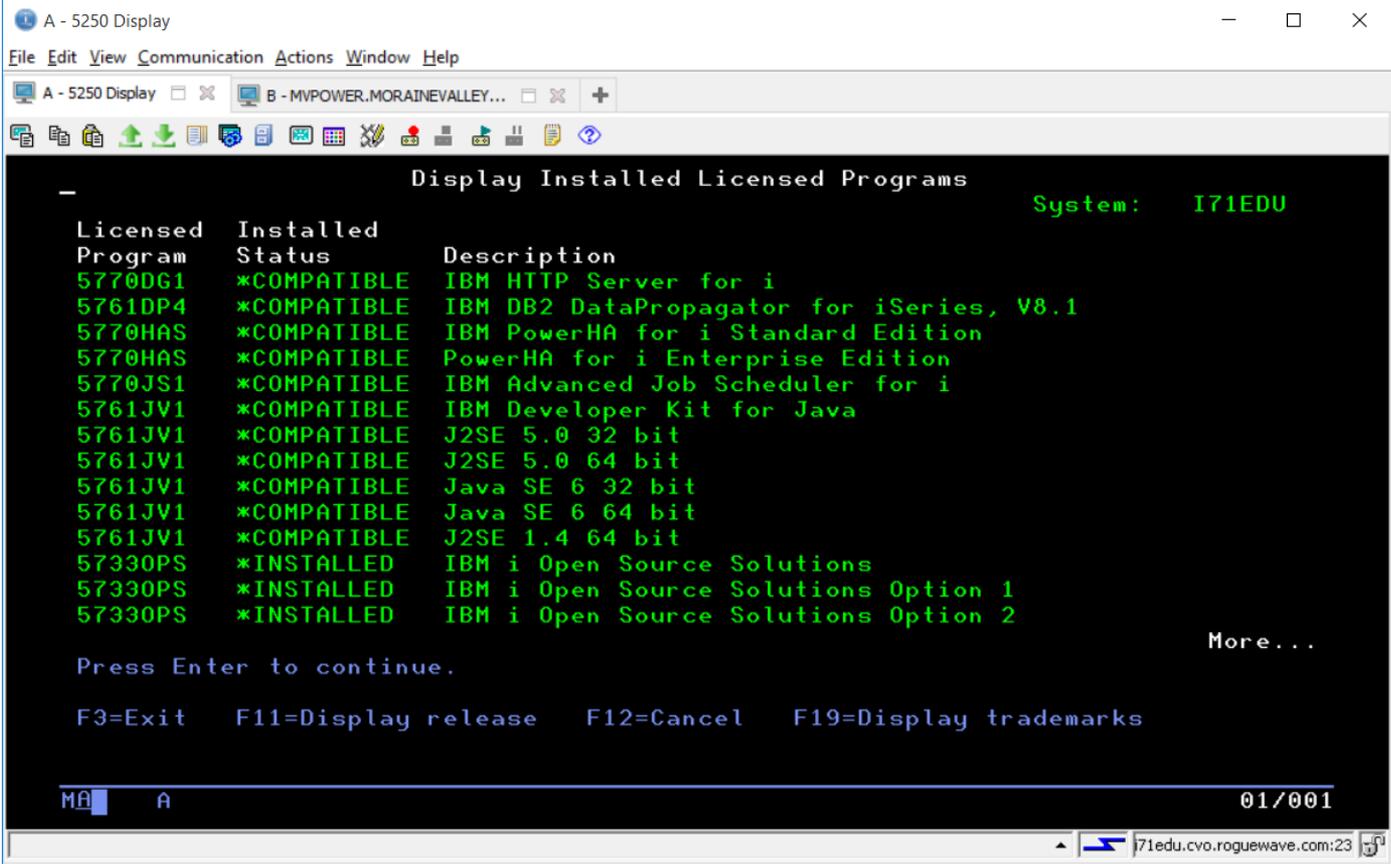
	SAMBA on IBM i
5733-OPS Option 1	Node.js v1
5733-OPS Option 2	Python 3
5733-OPS Option 3	CHROOT
5733-OPS Option 4	Python 2
5733-OPS Option 5	Node.js v4
5733-OPS Option 6	Git
5733-OPS Option 7	Tools
5733-OPS Option 8	Orion
5733-OPS Option 9	cloud-init
5733-OPS Option 10	Node.js v6
5733-OPS Option 11	Nginx
5733-OPS Option 12	TBD
5733-OPS Option 13	TBD
5733-OPS Option 14	TBD
5733-OPS Option 15	TBD

### Open Source Solutions for i Group PTF

IBM i	Group PTF	Level
7.3	SF99225	5
7.2	SF99223	5
7.1	SF99123	5

# Need licensed program

- 57330PS Base and option 2 or 4



A screenshot of a terminal window titled "A - 5250 Display". The window shows a list of installed licensed programs. The list includes various IBM products such as HTTP Server, DB2 DataPropagator, PowerHA, Job Scheduler, and Developer Kit for Java. The status of each program is listed as either \*COMPATIBLE or \*INSTALLED. The system ID is I71EDU. The terminal also shows instructions for pressing Enter to continue and function keys for Exit, Display release, Cancel, and Display trademarks.

```
A - 5250 Display
File Edit View Communication Actions Window Help
A - 5250 Display B - MVPOWER.MORAINEVALLEY...
Display Installed Licensed Programs
System: I71EDU

Licensed Program   Installed Status   Description
5770DG1            *COMPATIBLE  IBM HTTP Server for i
5761DP4            *COMPATIBLE  IBM DB2 DataPropagator for iSeries, V8.1
5770HAS            *COMPATIBLE  IBM PowerHA for i Standard Edition
5770HAS            *COMPATIBLE  PowerHA for i Enterprise Edition
5770JS1            *COMPATIBLE  IBM Advanced Job Scheduler for i
5761JV1            *COMPATIBLE  IBM Developer Kit for Java
5761JV1            *COMPATIBLE  J2SE 5.0 32 bit
5761JV1            *COMPATIBLE  J2SE 5.0 64 bit
5761JV1            *COMPATIBLE  Java SE 6 32 bit
5761JV1            *COMPATIBLE  Java SE 6 64 bit
5761JV1            *COMPATIBLE  J2SE 1.4 64 bit
57330PS            *INSTALLED   IBM i Open Source Solutions
57330PS            *INSTALLED   IBM i Open Source Solutions Option 1
57330PS            *INSTALLED   IBM i Open Source Solutions Option 2

More...

Press Enter to continue.

F3=Exit  F11=Display release  F12=Cancel  F19=Display trademarks

MA A 01/001
```

Times change...

---

*That was then...*

*...this is now*

# Leverage RPM's

- Use ACS to do the heavy lifting! But fire up SSH first.

```

A - 192.168.21.124
File Edit View Communication Actions Window Help
Start TCP/IP Server (STRTCPSVR)

Type choices, press Enter.

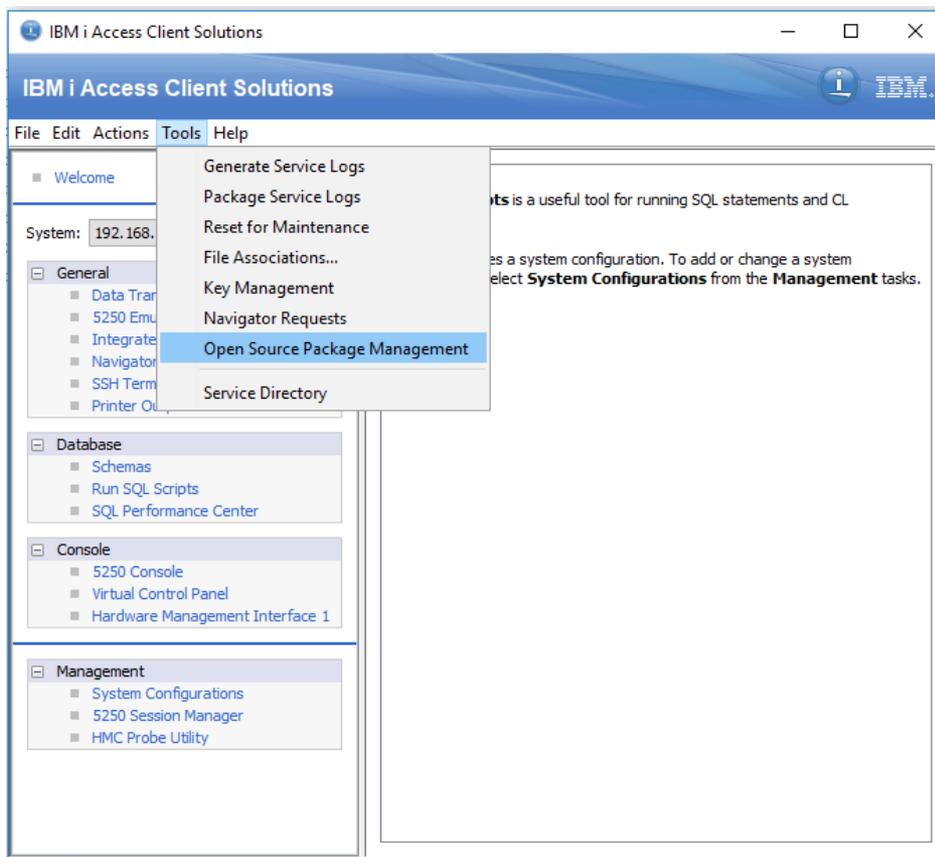
Server application . . . . . > *SSH*          *ALL, *AUTOSTART...
      + for more values
Instance:
  Server instance . . . . . *DFT
  Instance startup values . . .

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys

MA  A  MW  05/037
192.168.21.124:23
  
```

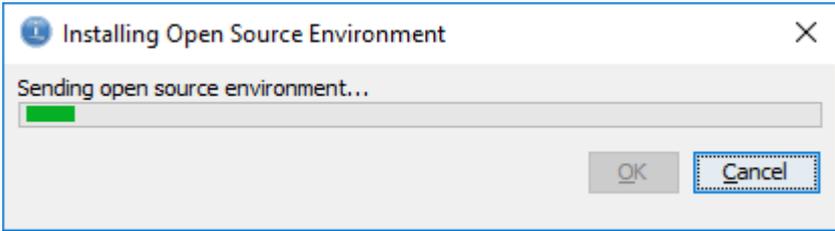
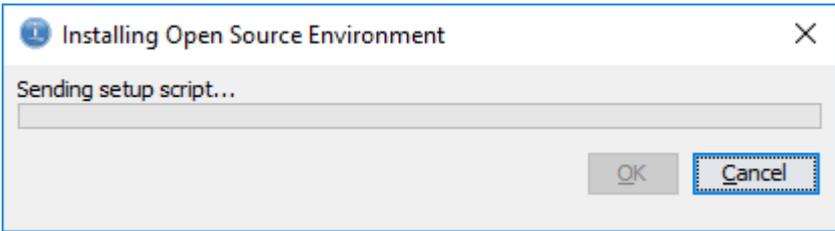
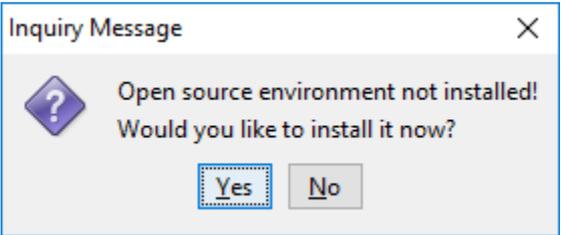
# Insert RPM stuff here...

- Use ACS to do the heavy lifting!



# Select system and credentials

The screenshot displays the 'Open Source Package Management' application window. The main window has a menu bar with 'File', 'View', 'Connection', and 'Utilities'. Below the menu bar, it shows '<NOT CONNECTED>' and three tabs: 'Installed packages', 'Updates available', and 'Available packages'. The 'Installed packages' tab is active. In the foreground, there is a 'Connect to SSH' dialog box with the following fields: 'System' (a dropdown menu showing '192.168.21.124'), 'User' (a text box with 'mpavlak'), 'Password' (a text box with masked characters), and 'Container' (a text box with '/'). There are 'OK' and 'Cancel' buttons at the bottom of the dialog. Overlaid on top of the SSH dialog is an 'Inquiry Message' dialog box. The message text reads: 'The authenticity of host '192.168.21.124' can't be established. RSA key fingerprint is fd:4e:58:9c:10:db:7a:d1:a4:b4:cd:6a:0b:bc:d5:88. Are you sure you want to continue connecting?'. There are 'Yes' and 'No' buttons at the bottom of the inquiry message dialog.





# Now install Python3

Open Source Package Management

File View Connection Utilities

Connection: mpavlak@192.168.21.124:/

Installed packages Updates available Available packages

Package	Version	Repository
lib-devel	3.33-3	ibm
openssh	6.9p1-1	ibm
openssl	1.1.1-2	ibm
openssl-devel	1.1.1-2	ibm
pase-utf8-locale	7.1-0	ibm
patch-gnu	2.7.5-1	ibm
pcre-devel	8.40-0	ibm
pcre-tools	8.40-0	ibm
pkg-config	0.29.2-0	ibm
popt-devel	1.16-1	ibm
python2-devel	2.7.14-0	ibm
python2-iniparse	0.4-0	ibm
python2-pip	9.0.1-0	ibm
python2-setuptools	36.0.1-0	ibm
python2-wheel	0.29.0-0	ibm
<b>python3</b>	<b>3.6.5-1</b>	<b>ibm</b>
python3-devel	3.6.5-1	ibm
python3-ibm_db	2.0.5.7-0	ibm
python3-itookit	1.4.0-0	ibm
python3-pip	9.0.1-0	ibm
python3-setuptools	36.0.1-0	ibm
python3-six	1.10.0-0	ibm
python3-wheel	0.29.0-0	ibm
readline-devel	6.3-2	ibm
readline-doc	6.3-2	ibm
rpm-build	4.13.0.1-7	ibm
rpm-devel	4.13.0.1-7	ibm

Done: 148 rows retrieved.

Information Install

# Info command

```
Package Information
clear;exec /Q0penSys/pkgs/bin/yum info 'python3' 'python3-devel' 'python3-ibm_db' 'python3-itoolkit' 'python3-ibm_db'
Could not chdir to home directory /home/MPAVLAK: No such file or directory
$ Available Packages
Name      : python3
Arch      : ppc64
Version   : 3.6.5
Release   : 1
Size      : 37 M
Repo      : ibm
Summary   : Python 3 Programming Language
URL       : https://www.python.org
License   : Python
Description: Python is a programming language that lets you work more quickly
          : and integrate your systems more effectively.
          :
          : You can learn to use Python and see almost immediate gains in
          : productivity and lower maintenance costs.

Name      : python3-devel
Arch      : ppc64
Version   : 3.6.5
Release   : 1
Size      : 159 k
Repo      : ibm
Summary   : Python 3 development files
URL       : https://www.python.org
License   : Python
Description: Python 3 development files

Name      : python3-ibm_db
Arch      : ppc64
Version   : 2.0.5.7
Release   : 0
Size      : 191 k
Repo      : ibm
Summary   : IBM i toolkit for Python
```

# Confirm “Is this ok [y/n]”

```

Package Installation
---> Package python3-pip.noarch 0:9.0.1-0 will be installed
---> Package python3-setuptools.noarch 0:36.0.1-0 will be installed
---> Package python3-six.noarch 0:1.10.0-0 will be installed
---> Package python3-wheel.noarch 0:0.29.0-0 will be installed
--> Running transaction check
---> Package libncurses6.ppc64 0:6.0-2 will be installed
--> Processing Dependency: ncurses-terminfo for package: libncurses6-6.0-2.ppc64
--> Running transaction check
---> Package ncurses-terminfo.ppc64 0:6.0-2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package                Arch          Version        Repository     Size
=====
Installing:
python3                 ppc64         3.6.5-1        ibm            37 M
python3-devel          ppc64         3.6.5-1        ibm            159 k
python3-ibm_db         ppc64         2.0.5.7-0      ibm            191 k
python3-itoolkit       ppc64         1.4.0-0        ibm            63 k
python3-pip            noarch        9.0.1-0        ibm            1.9 M
python3-setuptools     noarch        36.0.1-0       ibm            526 k
python3-six            noarch        1.10.0-0       ibm            15 k
python3-wheel         noarch        0.29.0-0       ibm            118 k
Installing for dependencies:
libncurses6           ppc64         6.0-2          ibm            318 k
ncurses-terminfo     ppc64         6.0-2          ibm            582 k

Transaction Summary
=====
Install      10 Packages

Total download size: 41 M
Installed size: 152 M
Is this ok [y/N]:

```

# Watch the installation until complete

```
Package Installation
(3/10): python3-3.6.5-1.ibmi7.1.ppc64.rpm | 37 MB 00:04
(4/10): python3-devel-3.6.5-1.ibmi7.1.ppc64.rpm | 159 kB 00:00
(5/10): python3-ibm_db-2.0.5.7-0.ibmi7.1.ppc64.rpm | 191 kB 00:00
(6/10): python3-itoolkkit-1.4.0-0.ibmi7.1.ppc64.rpm | 63 kB 00:00
(7/10): python3-pip-9.0.1-0.ibmi7.1.noarch.rpm | 1.9 MB 00:00
(8/10): python3-setuptools-36.0.1-0.ibmi7.1.noarch.rpm | 526 kB 00:00
(9/10): python3-six-1.10.0-0.ibmi7.1.noarch.rpm | 15 kB 00:00
(10/10): python3-wheel-0.29.0-0.ibmi7.1.noarch.rpm | 118 kB 00:00
-----
Total | 2.7 MB/s | 41 MB 00:15
Running Transaction Check
Running Transaction Test
Transaction Test Succeeded
Running Transaction
Installing : python3-six-1.10.0-0.noarch 1/10
Installing : ncurses-terminfo-6.0-2.ppc64 2/10
Installing : libncurses6-6.0-2.ppc64 3/10
Installing : python3-3.6.5-1.ppc64 4/10
Installing : python3-wheel-0.29.0-0.noarch 5/10
Installing : python3-devel-3.6.5-1.ppc64 6/10
Installing : python3-setuptools-36.0.1-0.noarch 7/10
Installing : python3-pip-9.0.1-0.noarch 8/10
Installing : python3-itoolkkit-1.4.0-0.ppc64 9/10
Installing : python3-ibm_db-2.0.5.7-0.ppc64 10/10

Installed:
python3.ppc64 0:3.6.5-1 python3-devel.ppc64 0:3.6.5-1
python3-ibm_db.ppc64 0:2.0.5.7-0 python3-itoolkkit.ppc64 0:1.4.0-0
python3-pip.noarch 0:9.0.1-0 python3-setuptools.noarch 0:36.0.1-0
python3-six.noarch 0:1.10.0-0 python3-wheel.noarch 0:0.29.0-0

Dependency Installed:
libncurses6.ppc64 0:6.0-2 ncurses-terminfo.ppc64 0:6.0-2

Complete!
```

# Looking back at ACS in the installed tab...

Open Source Package Management

File View Connection Utilities

Connection: mpavlak@192.168.21.124:/

Installed packages Updates available Available packages

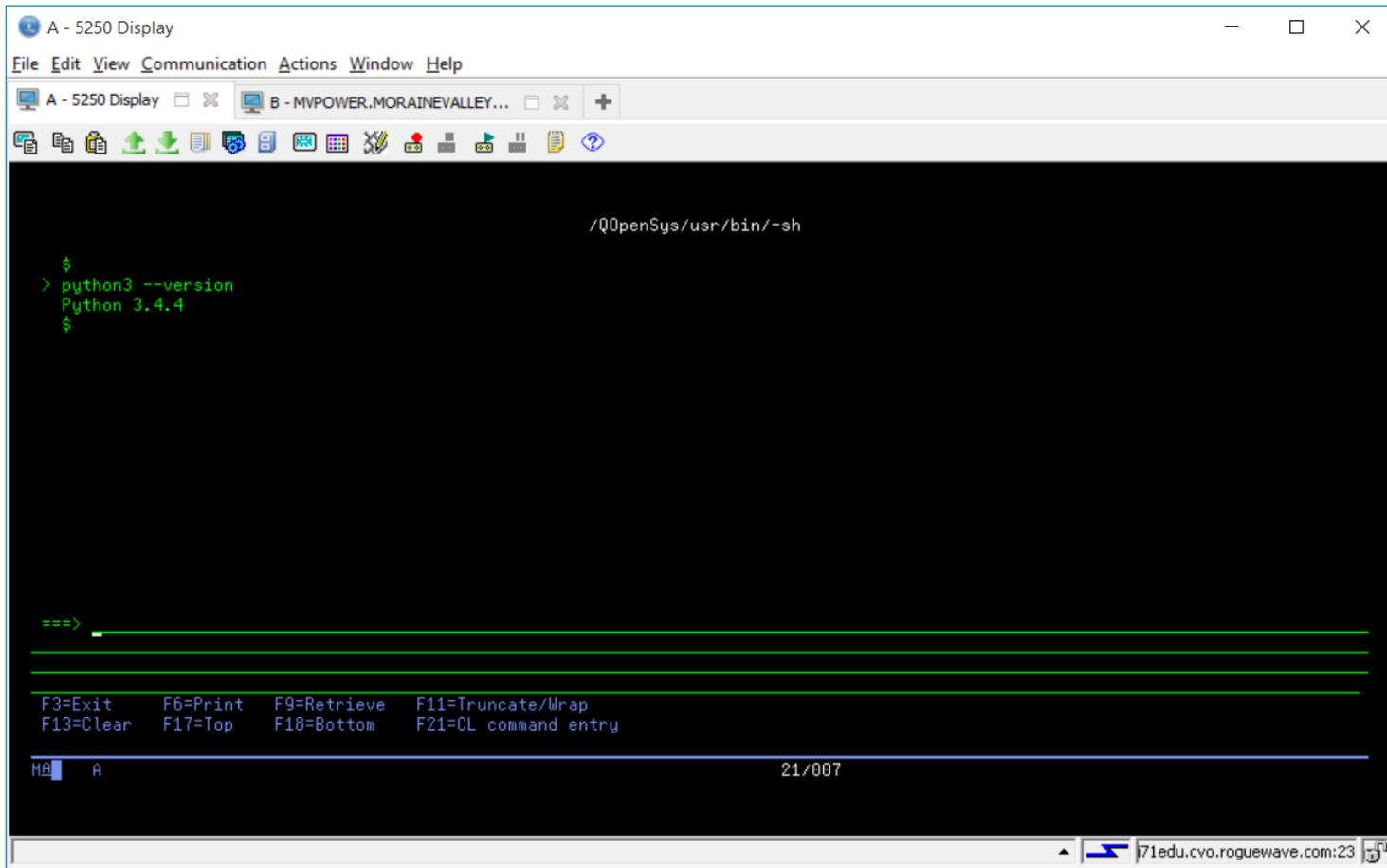
Package	Version	Repository
libpcre1	8.40-0	installed
libpopt0	1.16-1	installed
libreadline6	6.3-1	installed
libsquid3-0	3.19.3-0	installed
libutil1	0.3-0	installed
libxml2-2	2.9.4-2	installed
libz1	1.2.11-1	installed
ncurses-terminfo	6.0-2	@ibm
nspr	4.13.1-3	installed
nss	3.30-5	installed
pase-libs-dummy	7.1-0	installed
perl	5.24.1-0	installed
python2	2.7.14-0	installed
python2-pycurl	7.43.0-0	installed
python2-rpm	4.13.0.1-7	installed
python2-urlgrabber	3.10.2-0	installed
python3	3.6.5-1	@ibm
python3-devel	3.6.5-1	@ibm
python3-ibm_db	2.0.5.7-0	@ibm
python3-itookit	1.4.0-0	@ibm
python3-pip	9.0.1-0	@ibm
python3-setuptools	36.0.1-0	@ibm
python3-six	1.10.0-0	@ibm
python3-wheel	0.29.0-0	@ibm
rpm	4.13.0.1-7	installed
yum	3.4.3-6	installed
yum-metadata-parser	1.1.4-0	installed

Done: 44 rows retrieved.

Information Reinstall Remove

# Python in action

## ■ Command line via green screen



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 terminal content is as follows:

```
/QopenSys/usr/bin/-sh

$
> python3 --version
Python 3.4.4
$

===>

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

MA A                                     21/007
```

At the bottom right of the terminal window, there is a status bar showing a cursor icon, a blue arrow icon, and the text "71edu.cvo.roguewave.com:23".



# Most prefer SSH

- Command line via SSH terminal
  - ▣ Recommended strongly by Jesse!



<http://ibmsystemsmag.com/blogs/open-your-i/>

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

## Eight Reasons to Embrace SSH

In my previous post, I gave a brief introduction to the concept of a shell and focused on SSH connectivity. Often, when we think of a command-entry interface to our IBM i system, we think of a 5250 emulator. Perhaps we also know QSHELL as an interface to run open source or other commands in the root (/) or /QOpenSys filesystems.

[Read More](#)

Posted: August 29, 2017 | 0 Comments



# Shell available in ACS (shortcut to



PuTTY



IBM i Access Client Solutions

IBM i Access Client Solutions

File Edit Actions Tools Help

System: 192.168.21.124

- General
  - Data Transfer
  - 5250 Emulator
  - Integrated File System
  - Navigator for i
  - SSH Terminal**
  - Printer Output
- Database
  - Schemas
  - Run SQL Scripts
  - SQL Performance Center
- Console
  - 5250 Console
  - Virtual Control Panel
  - Hardware Management Interface 1
- Management
  - System Configurations
  - 5250 Session Manager
  - HMC Probe Utility

**SSH Terminal** will launch an already-installed SSH client (terminal emulator). With an SSH terminal, it is easy to run IBM i commands, invoke things in the Portable Application Solutions Environment (PASE), and access open source technologies (e.g. Python, Node.js, Git, etc).

If you receive a connection error within the launched SSH client (eg "connection refused"), or if a window pops up but immediately vanishes, you may need to start the SSH daemon by running this CL command:

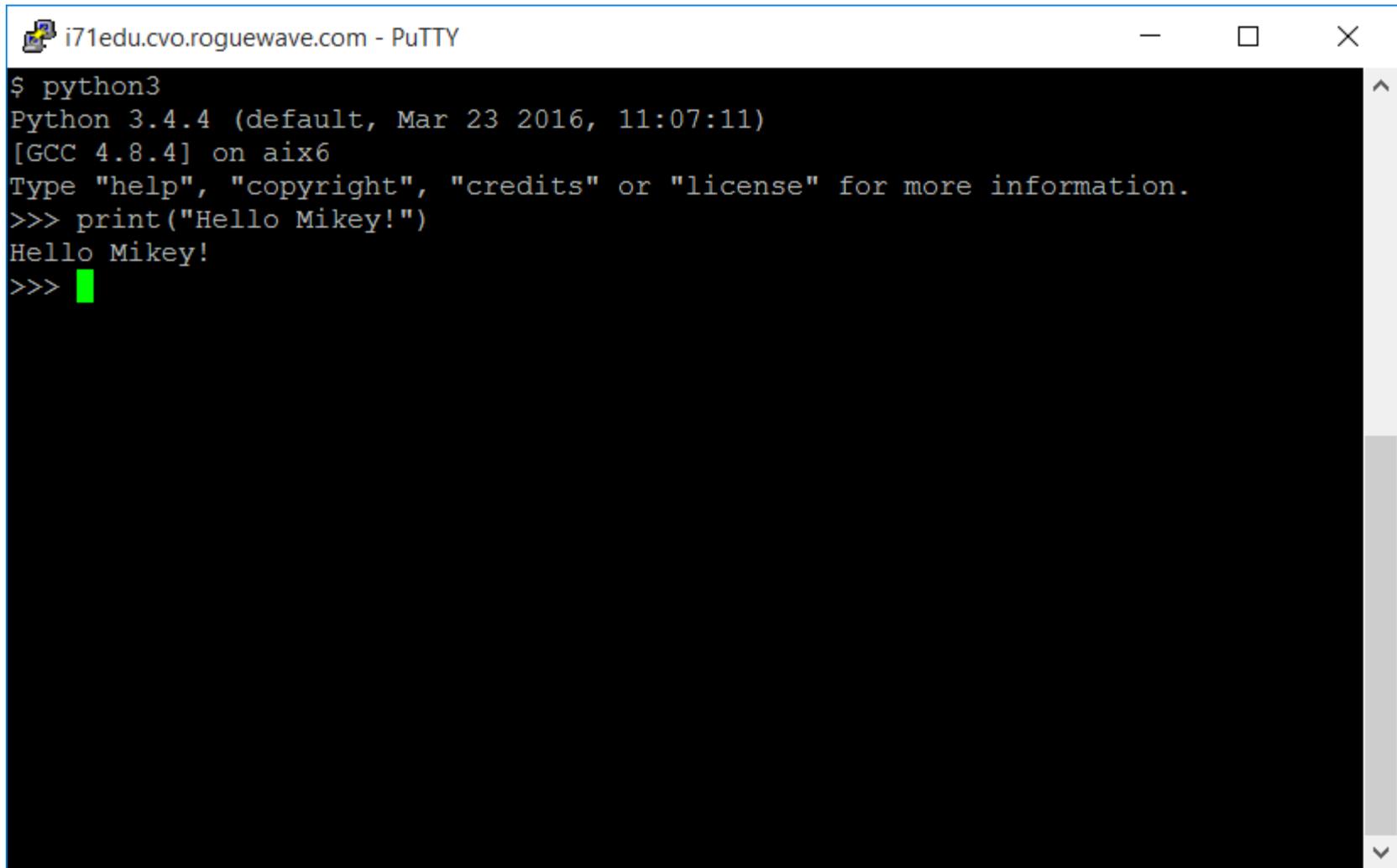
● STRICPSVR \*SSHD (This requires the 5733-SC1 product)

- Encrypted
- BASH, etc.
- Linux alignment

mvpower.morainevalley.edu - PuTTY

```
Using username "mpavlak".
mpavlak@mvpower.morainevalley.edu's password:
$
$
```

# 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](http://pydev.org)

### Getting Started:

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

### Feature Matrix:

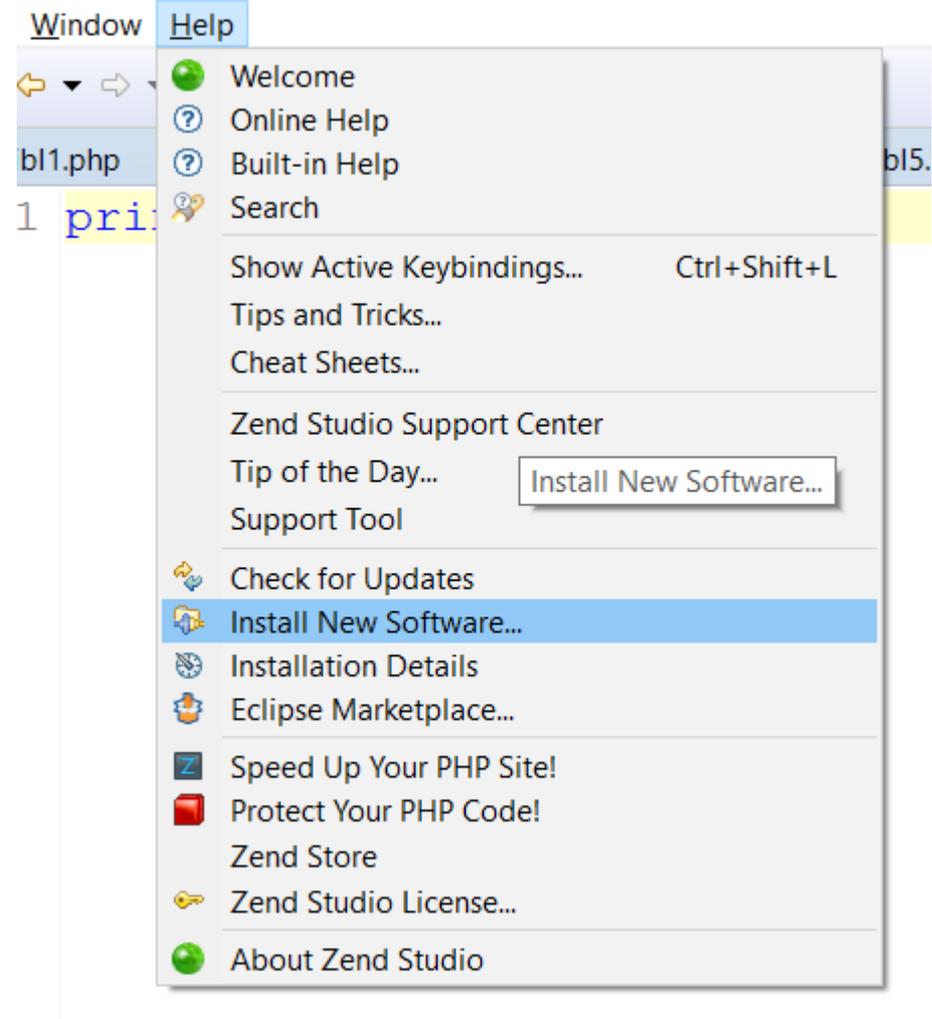
[pydev.org/manual\\_adv\\_features.html](http://pydev.org/manual_adv_features.html)

# Download PyDev from Eclipse

The screenshot shows the Eclipse Marketplace website. The URL in the browser is <https://marketplace.eclipse.org/content/pydev-python-ide-eclipse>. The page features the Eclipse Marketplace logo and navigation links like 'MY MARKETPLACE', 'ADD CONTENT', and 'MORE'. A search bar is visible on the left. The main content area is titled 'PyDev - Python IDE for Eclipse' and includes a search bar, an 'Install' button, a star rating of 472, and a description of the plugin. 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 the description, 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:'. The page also includes a 'MARKETS' sidebar and a 'MORE LIKE THIS' section with links to 'LiClipseText', 'LiClipse', and 'Design and Verification Tools (DVT) IDE for e, SystemVerilog, and VHDL'.

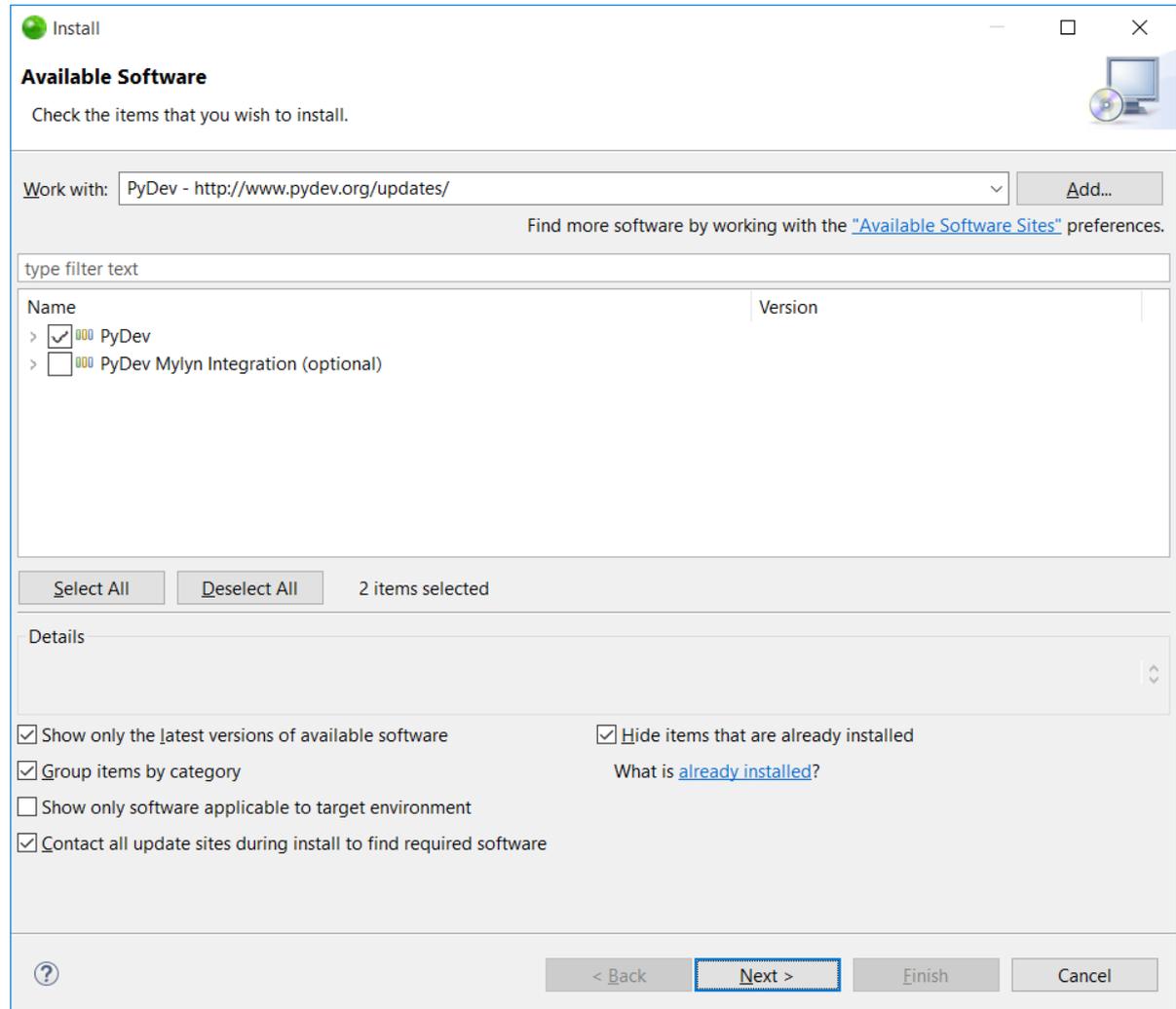
# Capture URL

- Help →
  - ▣ Install New Software
  - ▣ Follow prompts



# Editor for Eclipse

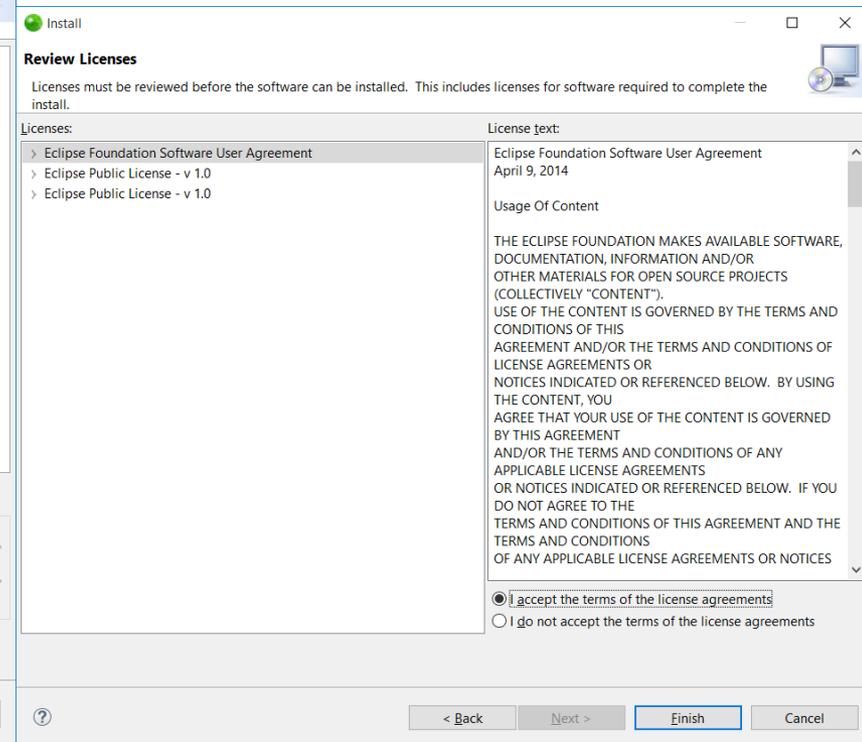
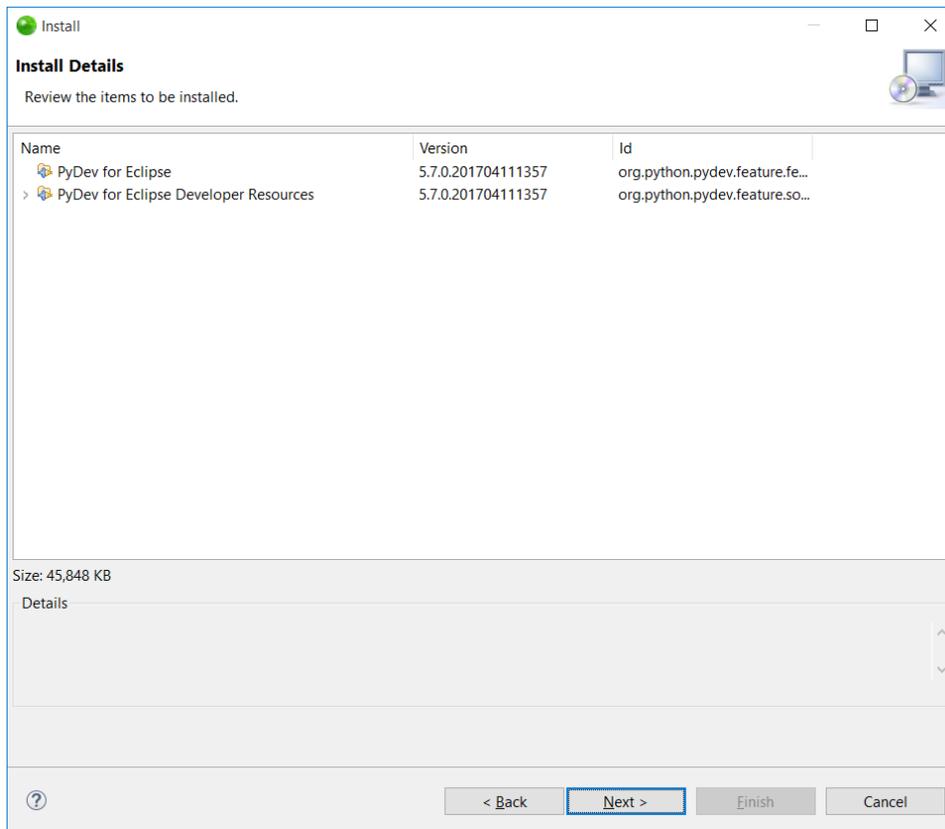
- Select what you like
- ▶ Click next



# Confirm versions

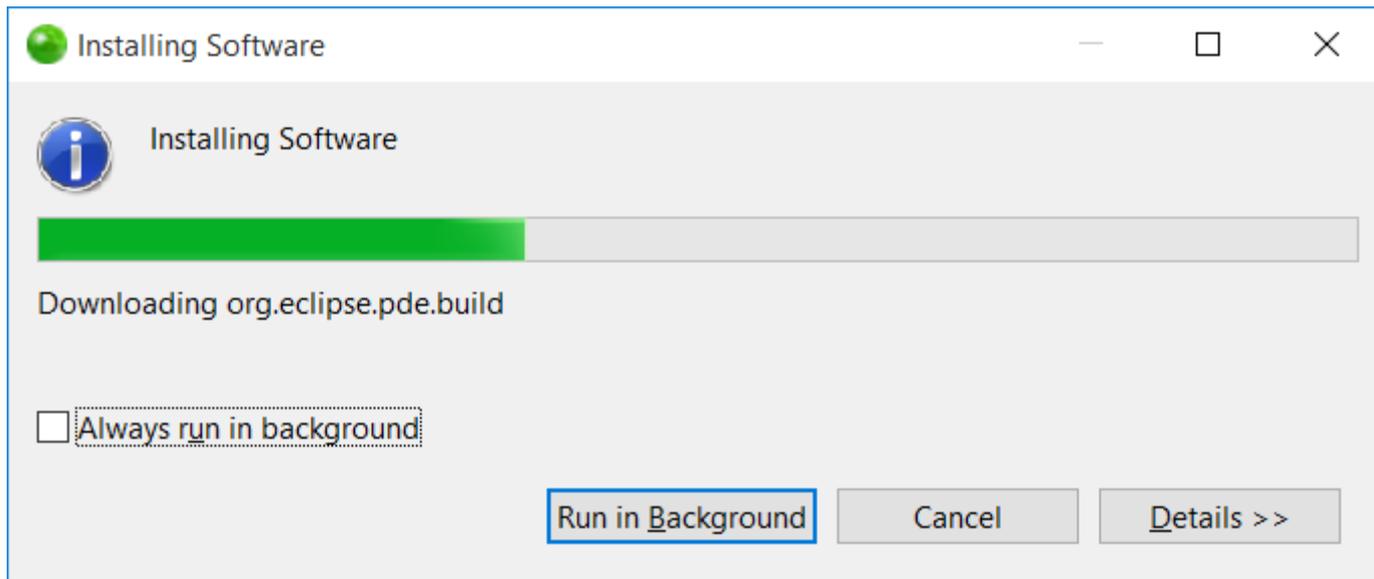
■ Click next again

▶ Then accept EULA



# Watch the pretty status bar

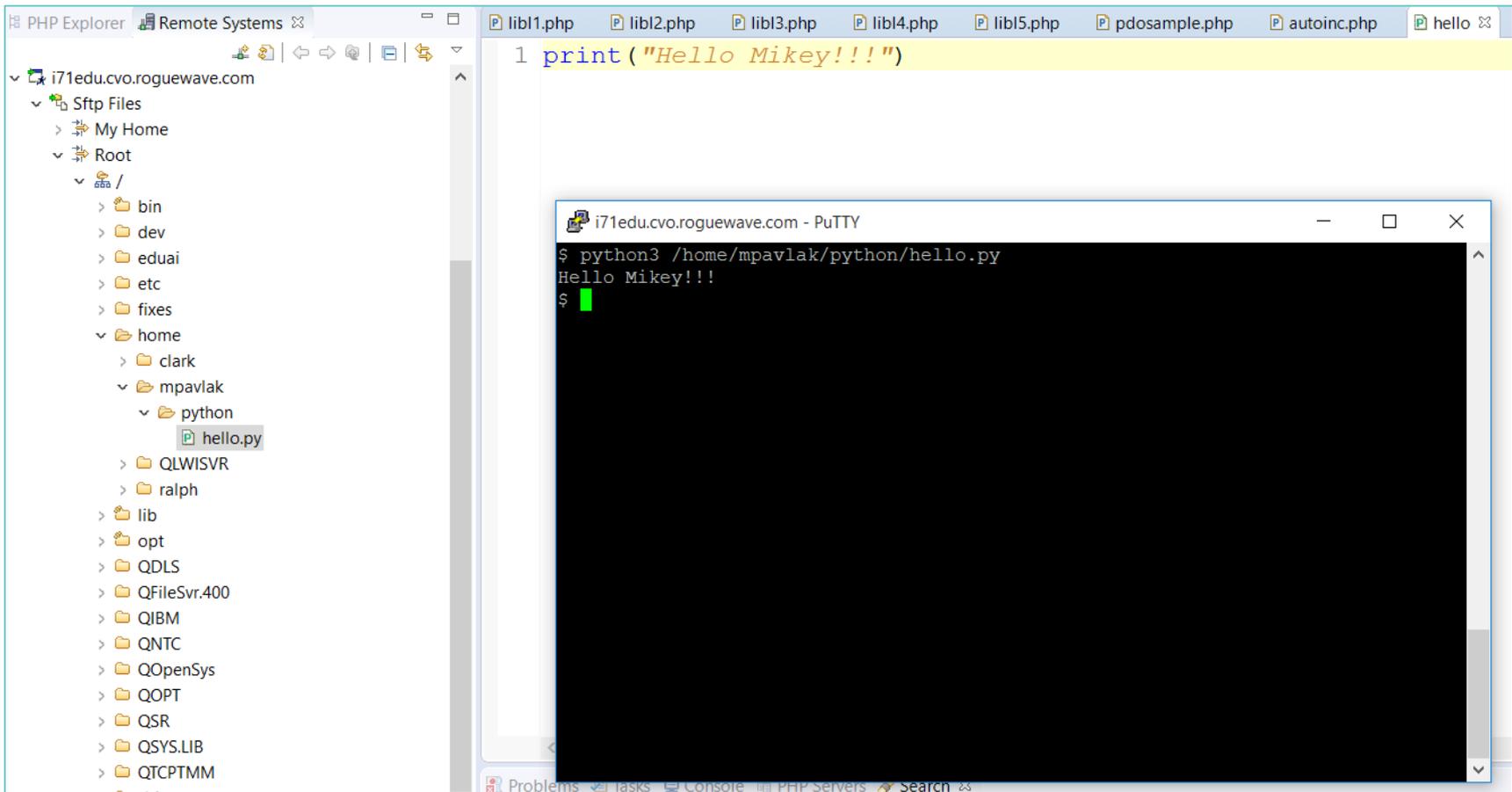
---



skwib.com

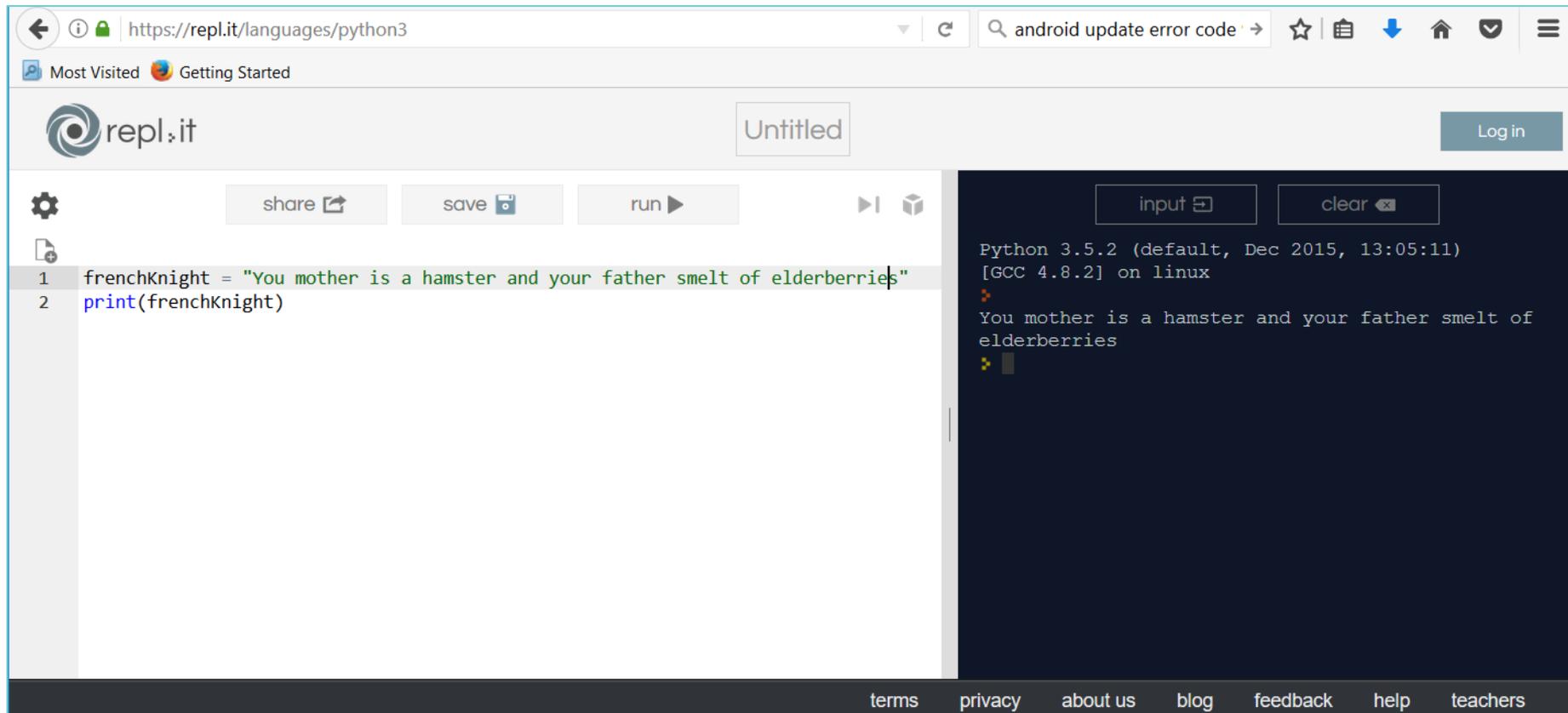
# Python in Eclipse (i.e. Zend Studio)

- I bet RDi works, too!



# Alternatives to IBM i when learning

- What's that? The boss won't let you install Python on the IBM i?



The screenshot shows a web browser window with the URL `https://repl.it/languages/python3`. The browser's address bar contains a search query `android update error code`. The page header includes the `repl.it` logo, a tab titled `Untitled`, and a `Login` button. Below the header, there are buttons for `share`, `save`, and `run`. The main area is a code editor with two lines of Python code:

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

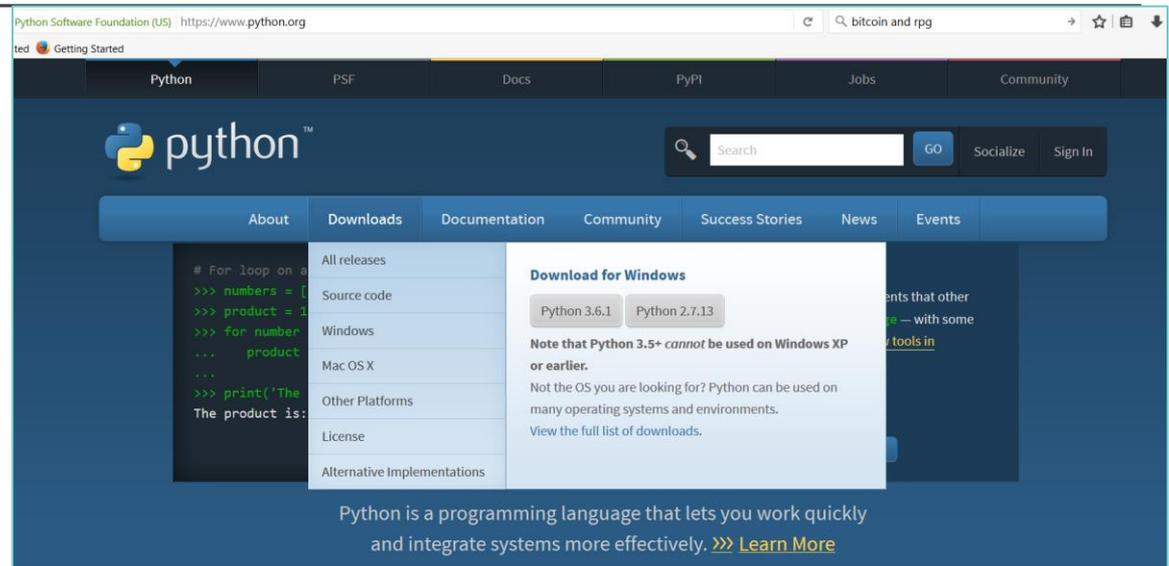
To the right of the code editor is a terminal window with the following output:

```
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 page, there is a navigation bar with links for `terms`, `privacy`, `about us`, `blog`, `feedback`, `help`, and `teachers`.

# Desktop education at it's finest

- How about your PC?
- Head to Python.org 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.
>>>
```

# Python Script in IFS

---

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

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

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

# Python Syntax Fundamentals

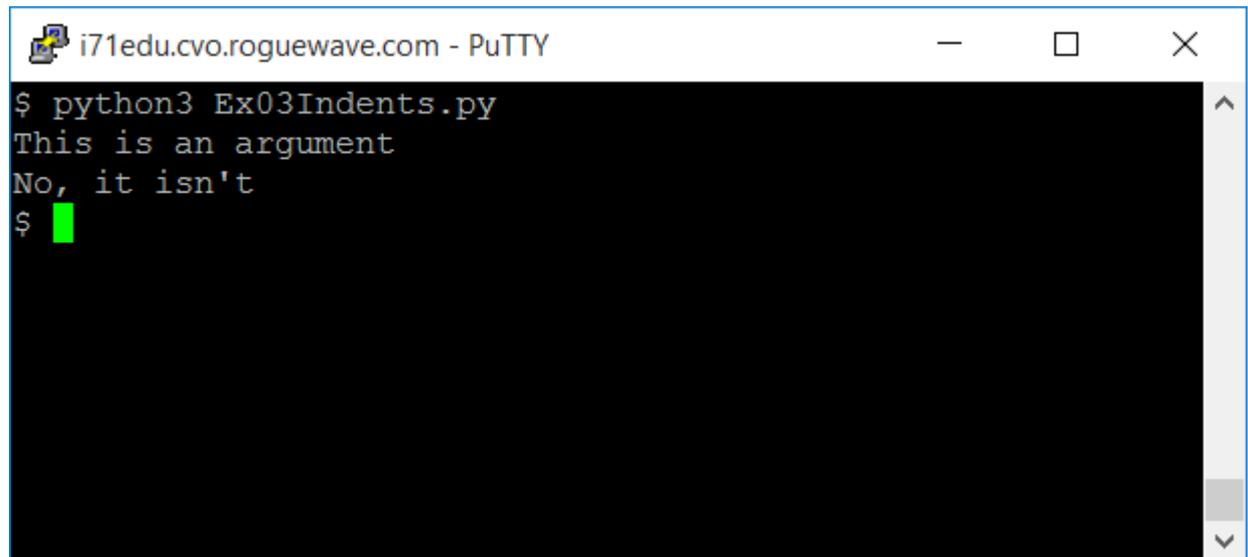
# How it is written

---

- Indentation means EVERYTHING
  - ▣ Don't use tab
  - ▣ 4 spaces – No more, no less
  - ▣ Mismatched indents can cause failures. Good luck finding...
- No 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 other C derivatives

## ■ 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

# Syntax

## Variables

# Data types – yeah...about that...

---

## ■ Int

- ▶ Integer of unlimited size

## ■ Float

- ▶ System defined precision

## ■ Complex

- ▶ Complex with real and imaginary parts

## ■ Strings

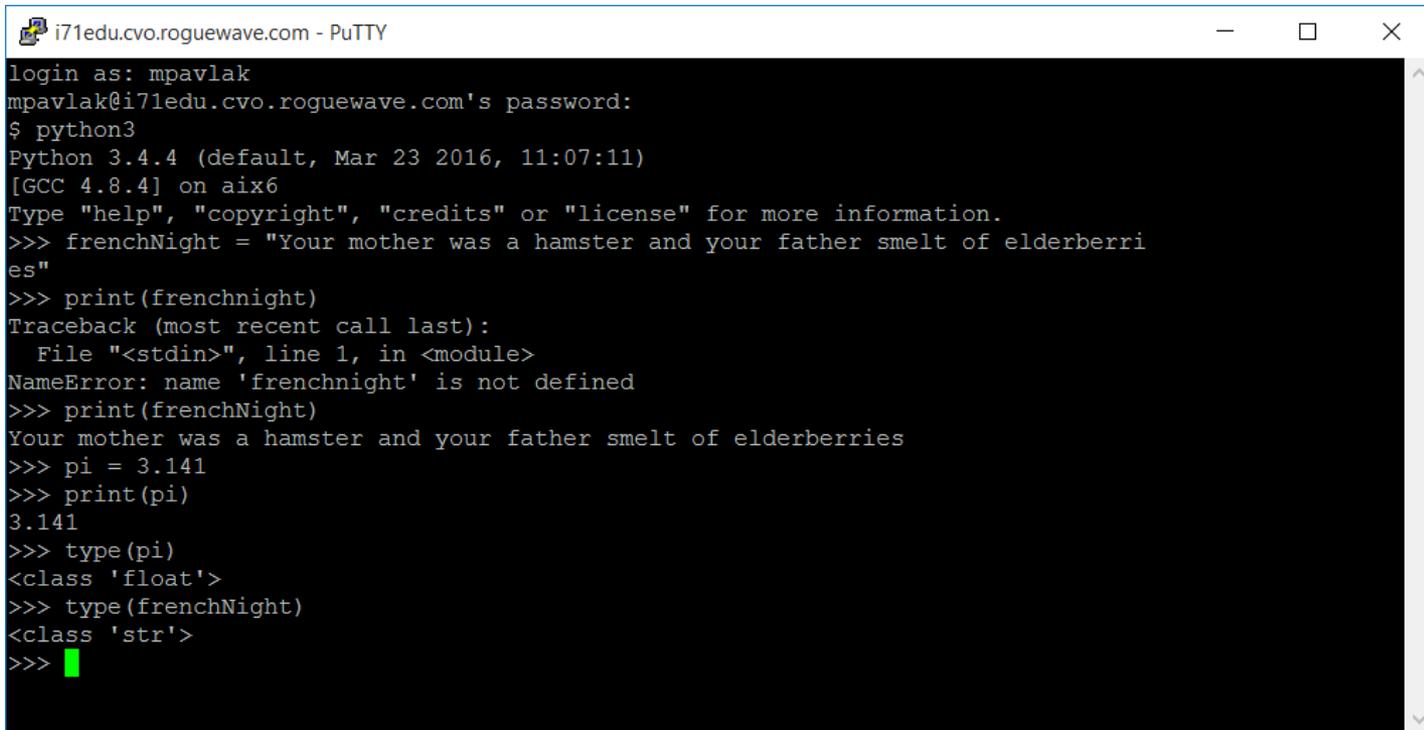
- ▶ Sequences of character data

## ■ Bool

- ▶ TRUE & FALSE

# Variables on the fly

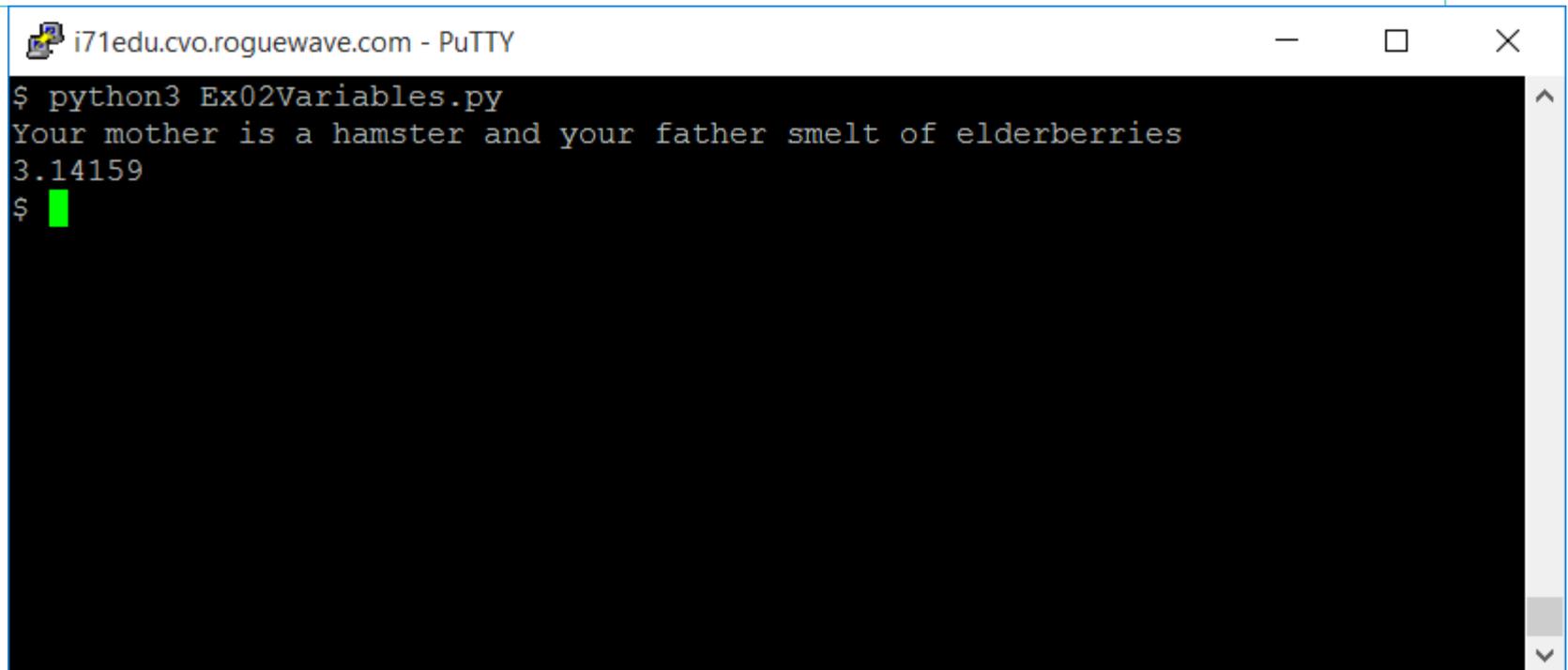
- Case sensitive
- camelCase
- Who are you? `type()`



```
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)
```

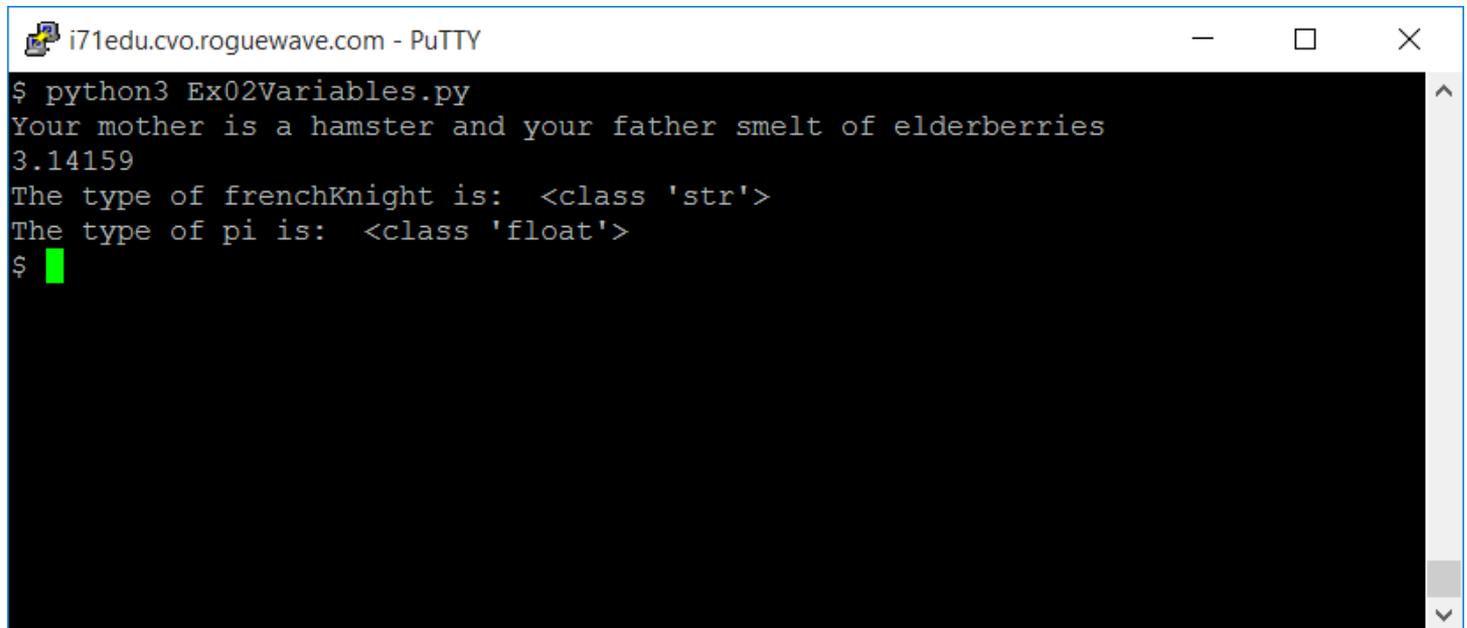


The screenshot shows a PuTTY terminal window titled "i71edu.cvo.roguewave.com - PuTTY". The terminal displays the following output:

```
$ 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'>
$
```



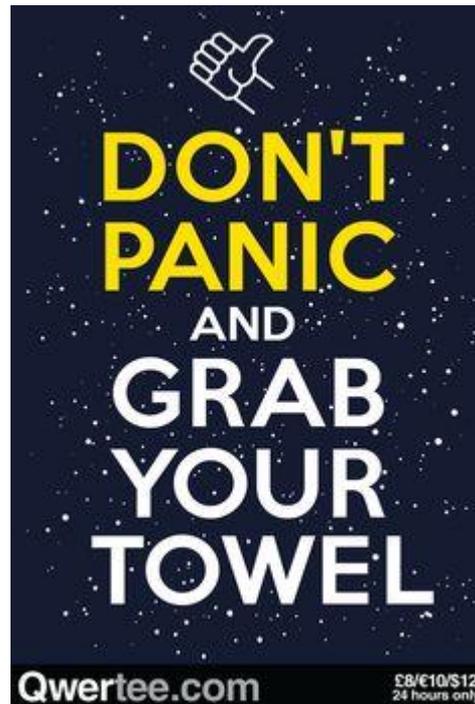
# Every variable is implemented as a class

---



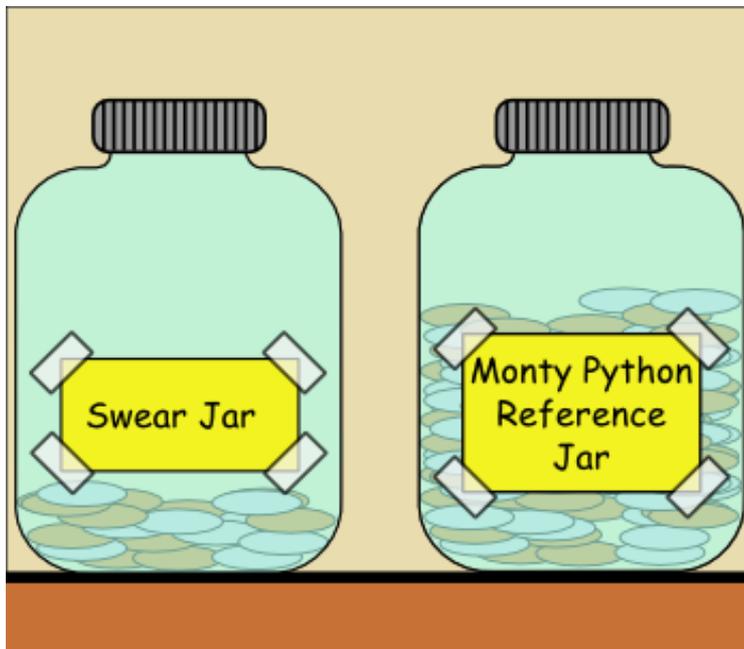
# And now for something completely different

---



# 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

---

## ■ Numbers – 3 Data types

▶ Integer      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 (of course...)



# 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!  
$ █
```

# Set Processing

# 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)
```

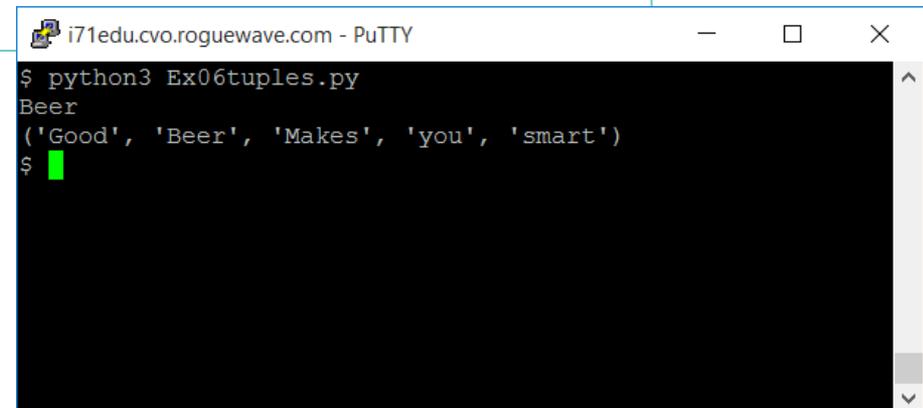


```
i71edu.cvo.roguewave.com - PuTTY  
$ python3 Ex05Lists.py  
Gordon Biersch  
['Rock Bottom', 'Gordon Biersch']  
['Rock Bottom', 'Gordon Biersch', 'BJ's', 'Granite City']  
$
```

# 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)
```

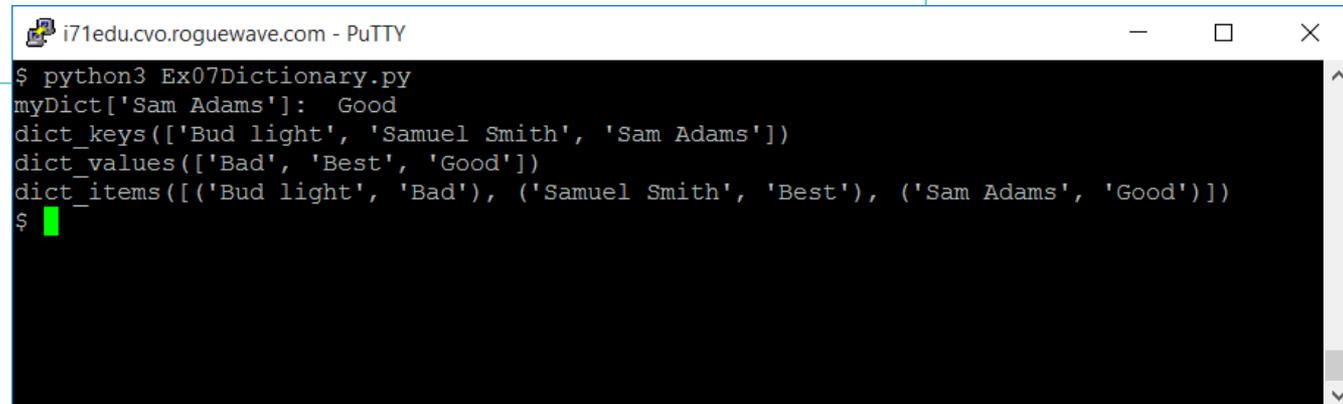


```
i71edu.cvo.roguewave.com - PuTTY  
$ python3 Ex06tuples.py  
Beer  
( 'Good', 'Beer', 'Makes', 'you', 'smart' )  
$
```

# Dictionary

- Again, like lists but more like hash or PHP Assoc. Array
- 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())
```

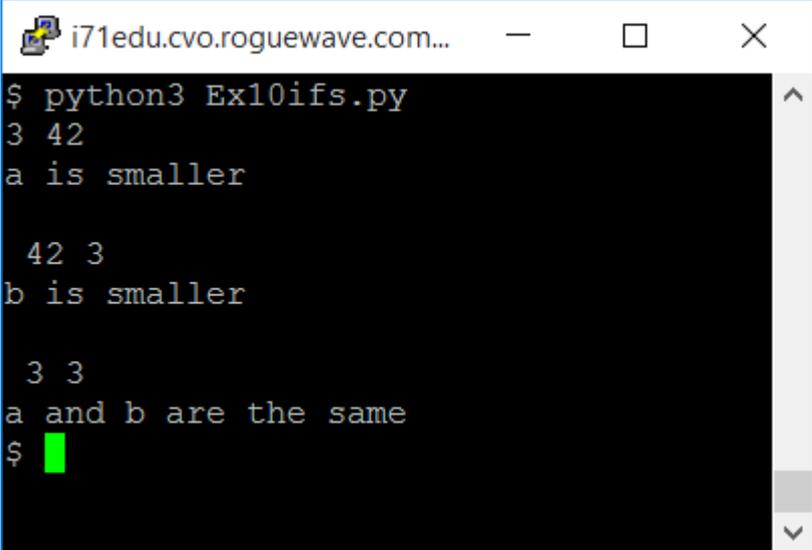


```
i71edu.cvo.roguewave.com - PuTTY  
$ python3 Ex07Dictionary.py  
myDict['Sam Adams']: Good  
dict_keys(['Bud light', 'Samuel Smith', 'Sam Adams'])  
dict_values(['Bad', 'Best', 'Good'])  
dict_items([('Bud light', 'Bad'), ('Samuel Smith', 'Best'), ('Sam Adams', 'Good')])  
$
```

# Control Structures

# ifs

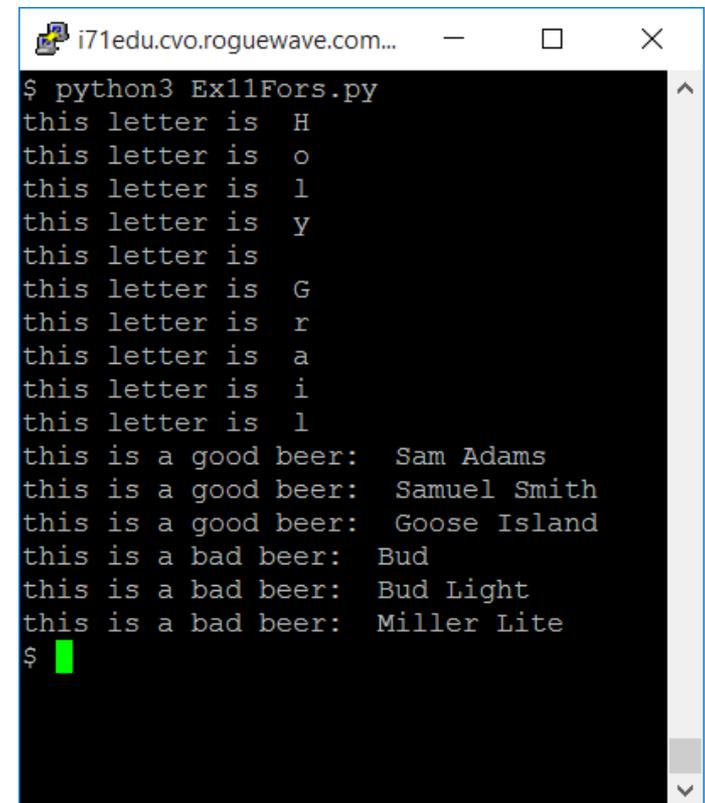
```
1 #  
2 # If examples  
3 #  
4 a,b = 3,42  
5 print(a,b)  
6 if a < b:  
7     print("a is smaller")  
8  
9 a,b = 42,3  
10 print("\n",a,b)  
11 if a < b:  
12     print("a is smaller")  
13 else:  
14     print("b is smaller")  
15  
16 a,b = 3,3  
17 print("\n",a,b)  
18 if a < b:  
19     print("a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

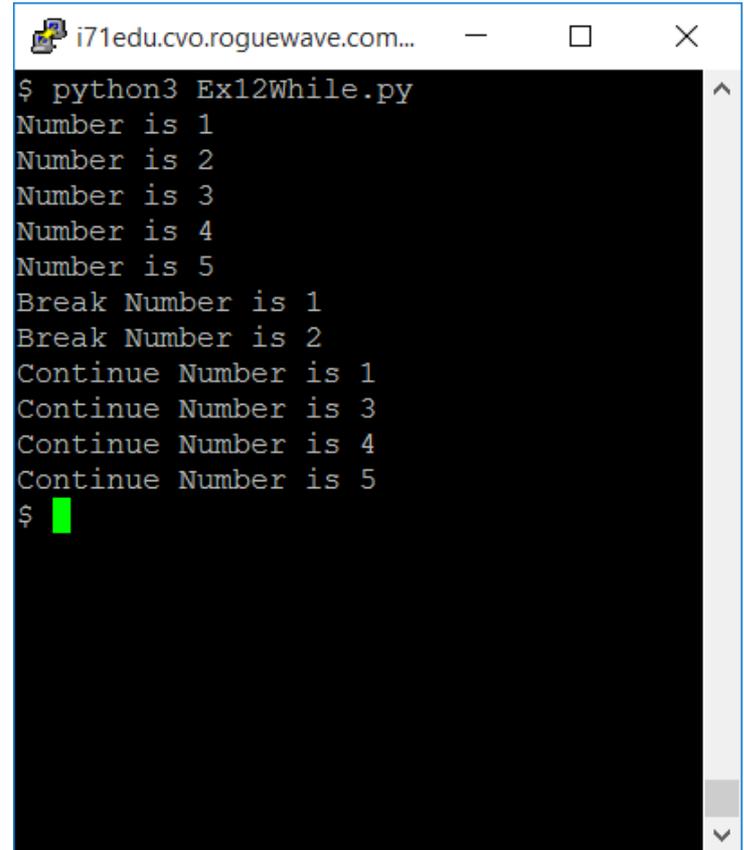
```
1 #
2 # For Loop Examples
3 #
4
5 myString = "Holy Grail"
6 for letter in myString:
7     print("this letter is ", letter)
8
9 beers = ["Sam Adams", "Samuel Smith", "Goose Island"]
10 for beer in beers:
11     print("this is a good beer: ", beer)
12
13 badBeers = ["Bud", "Bud Light", "Miller Lite"]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

# Built in functions

---

- About 68
  - ▶ Math
  - ▶ Type
  - ▶ Iterables
  - ▶ Composite
  - ▶ Classes
  - ▶ Variables/Reference Scope
  - ▶ Misc...

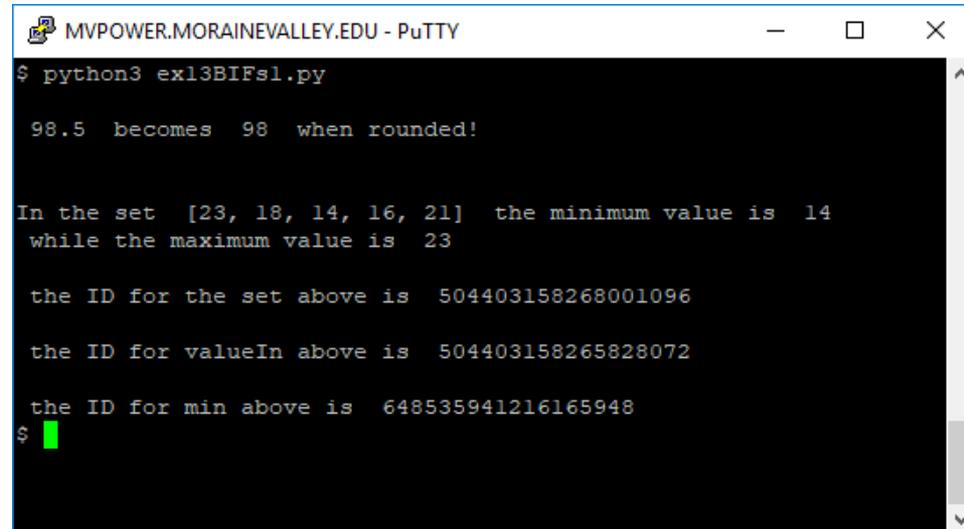
# Built in's are straight forward

---

```
#rounding...
valueIn = 98.5
valueOut = round(valueIn)
print("\n", valueIn, " becomes ", valueOut, " when rounded!\n")

#set processing
MySet1 = [23, 18, 14, 16, 21]
minset = min(MySet1)
print("\nIn the set ", MySet1, " the minimum value is ", minset)
print(" while the maximum value is ", max(MySet1))

#How about that object ID?
print("\n the ID for the set above is ", id(MySet1))
print("\n the ID for valueIn above is ", id(valueIn))
print("\n the ID for min above is ", id(minset))
```



```
MVPOWER.MORAINEVALLEY.EDU - PuTTY
$ python3 ex13BIFs1.py

98.5 becomes 98 when rounded!

In the set [23, 18, 14, 16, 21] the minimum value is 14
while the maximum value is 23

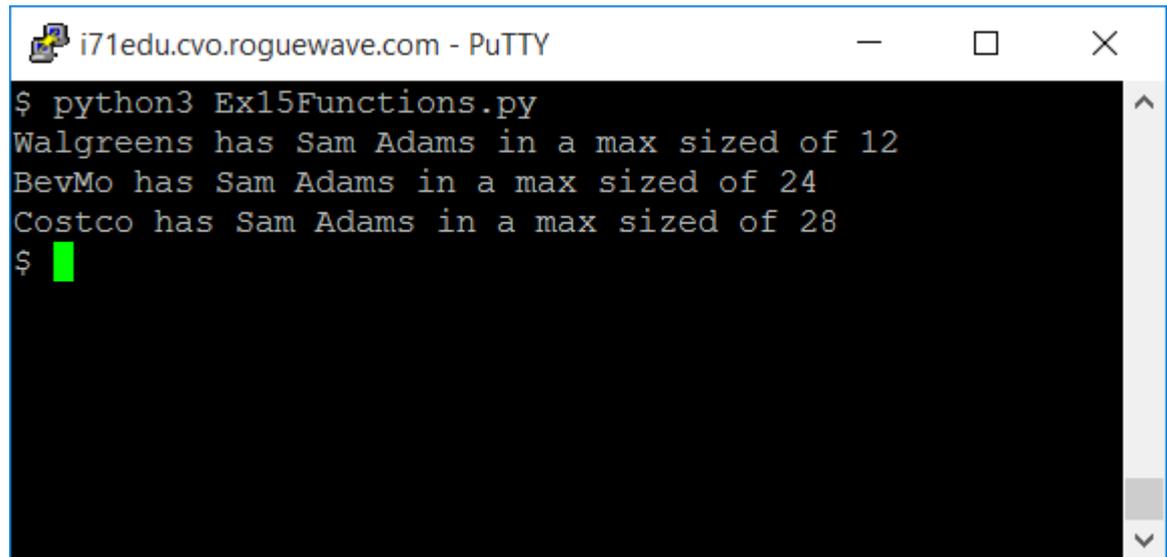
the ID for the set above is 504403158268001096

the ID for valueIn above is 504403158265828072

the ID for min above is 648535941216165948
$
```

# User Defined Functions

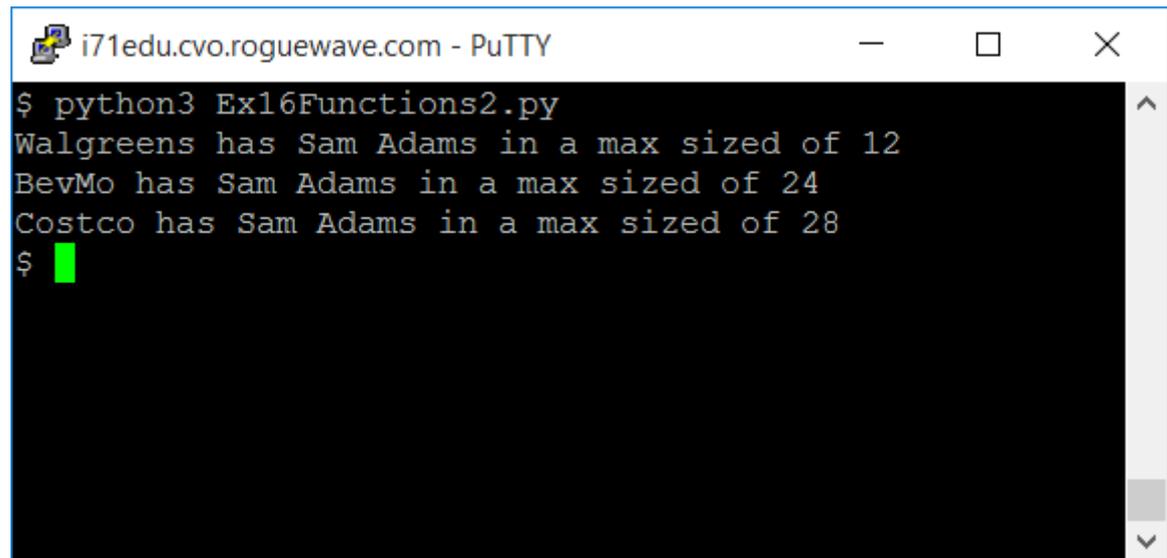
```
1 #  
2 # Function Examples  
3 #  
4  
5 def printBeer(store, beer, size):  
6     print(store + " has " + beer + " in a max sized of " + str(size) )  
7  
8 myBeer = "Sam Adams"  
9 printBeer("Walgreens", myBeer, 12)  
10 printBeer("BevMo", myBeer, 24)  
11 printBeer("Costco", myBeer, 28)
```



```
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

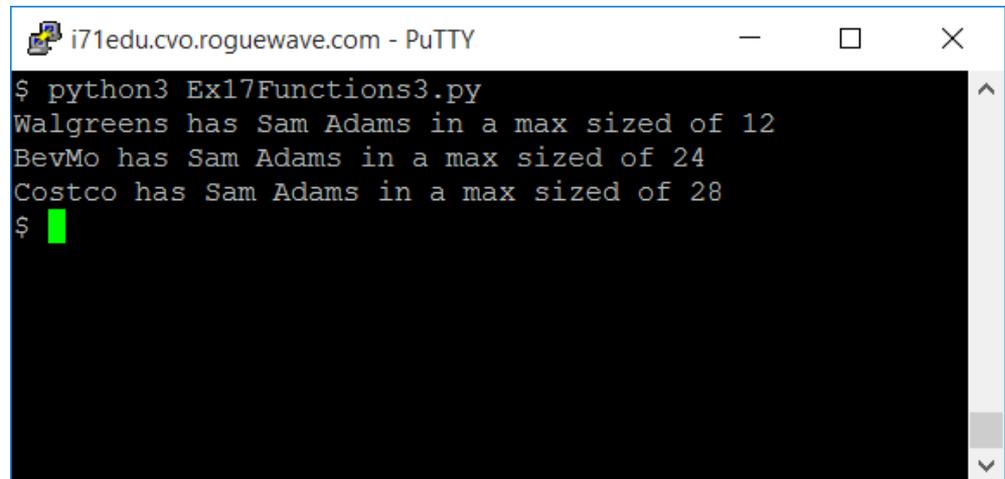
```
1 #  
2 # Function Examples  
3 #  
4  
5 def printBeer(store, beer, size=24):  
6     print(store + " has " + beer + " in a max sized of " + str(size) )  
7  
8 myBeer = "Sam Adams"  
9 printBeer("Walgreens", myBeer, 12)  
10 printBeer("BevMo", myBeer)  
11 printBeer("Costco", myBeer, 28)
```



```
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)
```



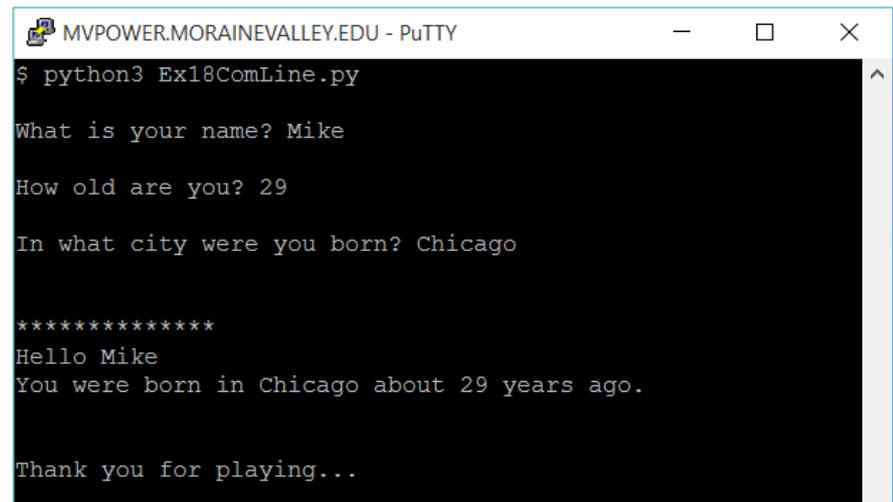
```
i71edu.cvo.roguewave.com - PuTTY
$ python3 Ex17Functions3.py
Walgreens has Sam Adams in a max sized of 12
BevMo has Sam Adams in a max sized of 24
Costco has Sam Adams in a max sized of 28
$
```

# Command Line

# Input from command line

- “Talk” to 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”

```

A - MVPOWER.MORAINEVALLEY.EDU
File Edit View Communication Actions Window Help
[Icons]
Display Attributes
Object . . . . . : /Q0penSys/QIBM/ProdData/OPS/Python-pkgs/ibm_db/i >
-----
Object link
- /Q0penSys/QIBM/ProdData/OPS/Python-pkgs/ibm_db/ibm_db-2.0.5.5-cp34-c
  p34m-os400_powerpc.whl
Bottom
F12=Cancel
Need to archive (PC) . . . . . : Yes
Need to archive (System) . . . . . : Yes
More...
Press Enter to continue.
F3=Exit F12=Cancel F22=Display entire field
MA A MW 06/004
[mvpower.morainevalley.edu:23]
  
```

# 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:**

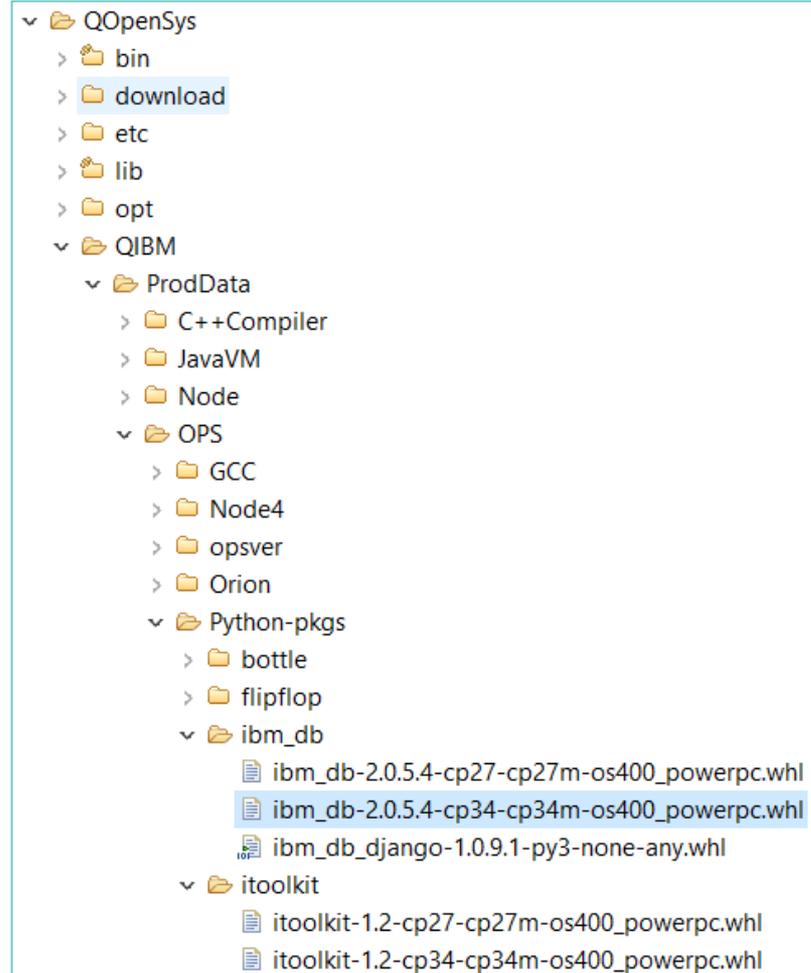
```
pip3 install /QOpenSys/QIBM/ProdData/OPS/Python-pkgs/itoolkit/itoolkit-*-cp34m-*.whl
```

**To install FastCGI gateway support:**

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

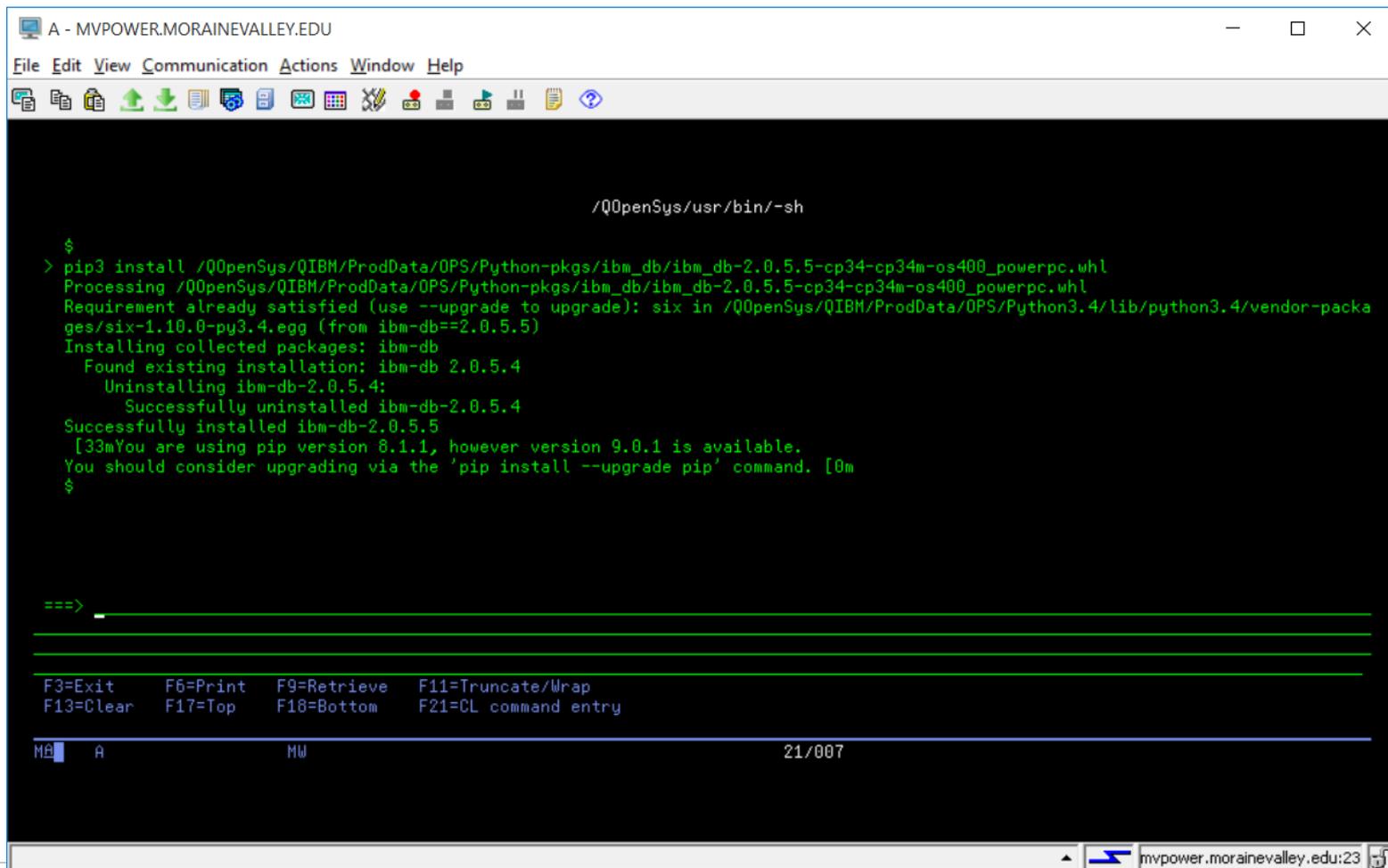
# Find the connector

- YMMV
- With wheels



# Run the pip install

- pip == Python installer program



```
A - MVPOWER.MORAINEVALLEY.EDU
File Edit View Communication Actions Window Help
/qqopensys/usr/bin/-sh

$
> pip3 install /qqopensys/qibm/proddata/ops/python-pkgs/ibm_db/ibm_db-2.0.5.5-cp34-cp34m-os400_powerpc.whl
Processing /qqopensys/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 /qqopensys/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   F10=Bottom F21=CL command entry

MA  A                               MW                               21/007
mvpower.morainevalley.edu:23
```

# 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
$
```

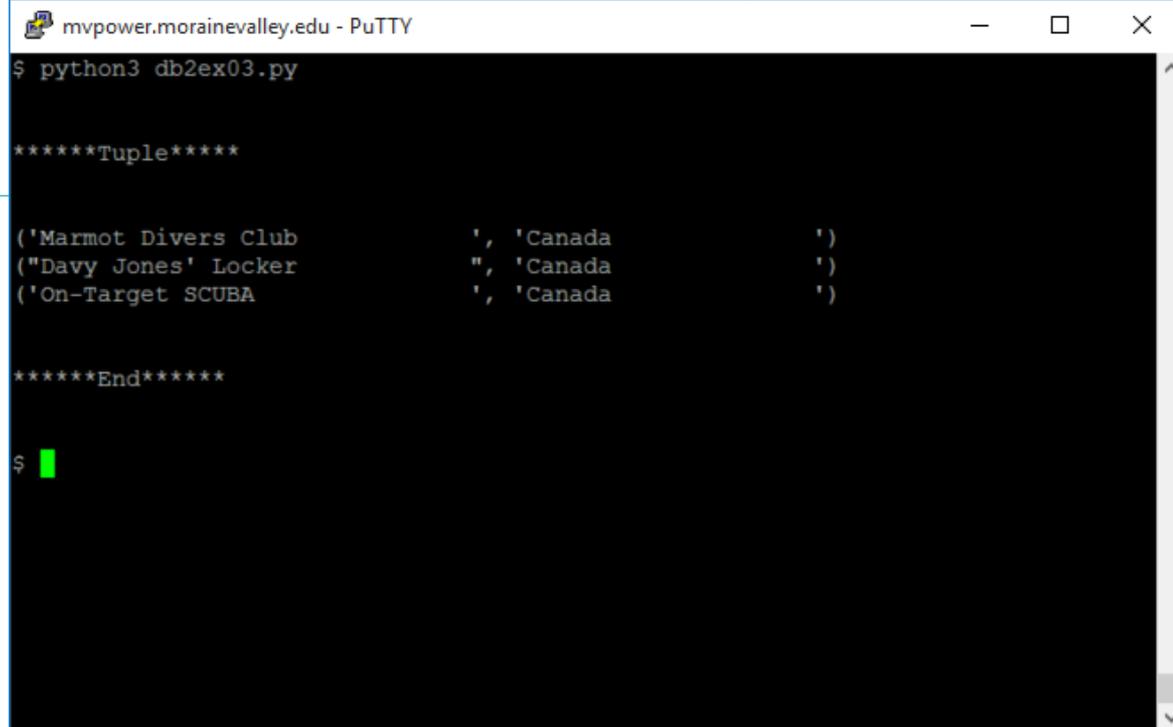
# Steps for simple database Access

---

- 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 info

```
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*****
```

# IBM i Toolkit

# A few ways to access toolkit

---

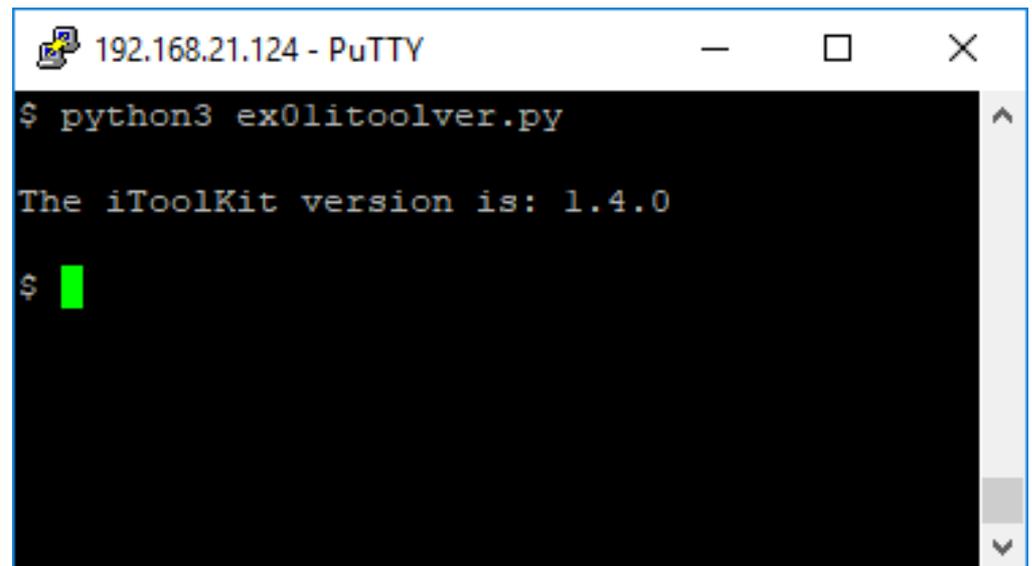
- Transition from Python2 to Python3 and RPM
  - ▣ Broke a few things
  - ▣ Under construction
  - ▣ This will work

# Toolkit Version

---

- IBM is maintaining version numbers

```
#  
import itoolkit  
  
#Version?  
print("\nThe iToolKit version is: " + itoolkit.__version__ + "\n")
```



The screenshot shows a PuTTY terminal window titled "192.168.21.124 - PuTTY". The terminal displays the following text:

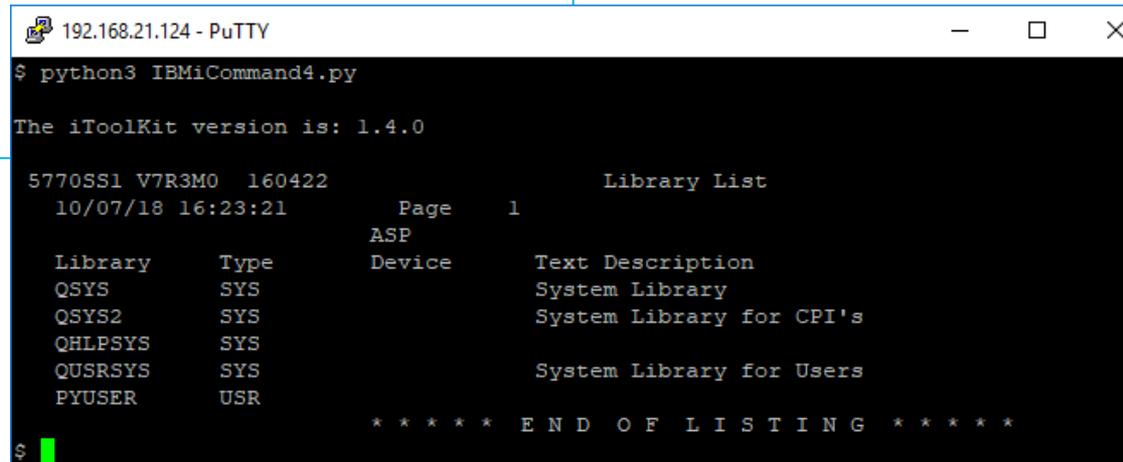
```
$ python3 ex01toolver.py  
  
The iToolKit version is: 1.4.0  
  
$ █
```

# CL Command

```
import toolkit as tkt
import toolkit.db2.idb2call as Db2tkt

itool = tkt.iToolKit()
print("\nThe iToolKit version is: " + tkt.__version__ + "\n")
itransport = Db2tkt.iDB2Call('PYUSER1', 'pyuser1')
itool.add(tkt.iCmd5250('dsplLibl', 'dsplLibl'))
itool.call(itransport)

# output
commandOutput = itool.dict_out('dsplLibl')
if 'error' in commandOutput:
    print (commandOutput['error'])
    exit()
else:
    print (commandOutput['dsplLibl'])
```



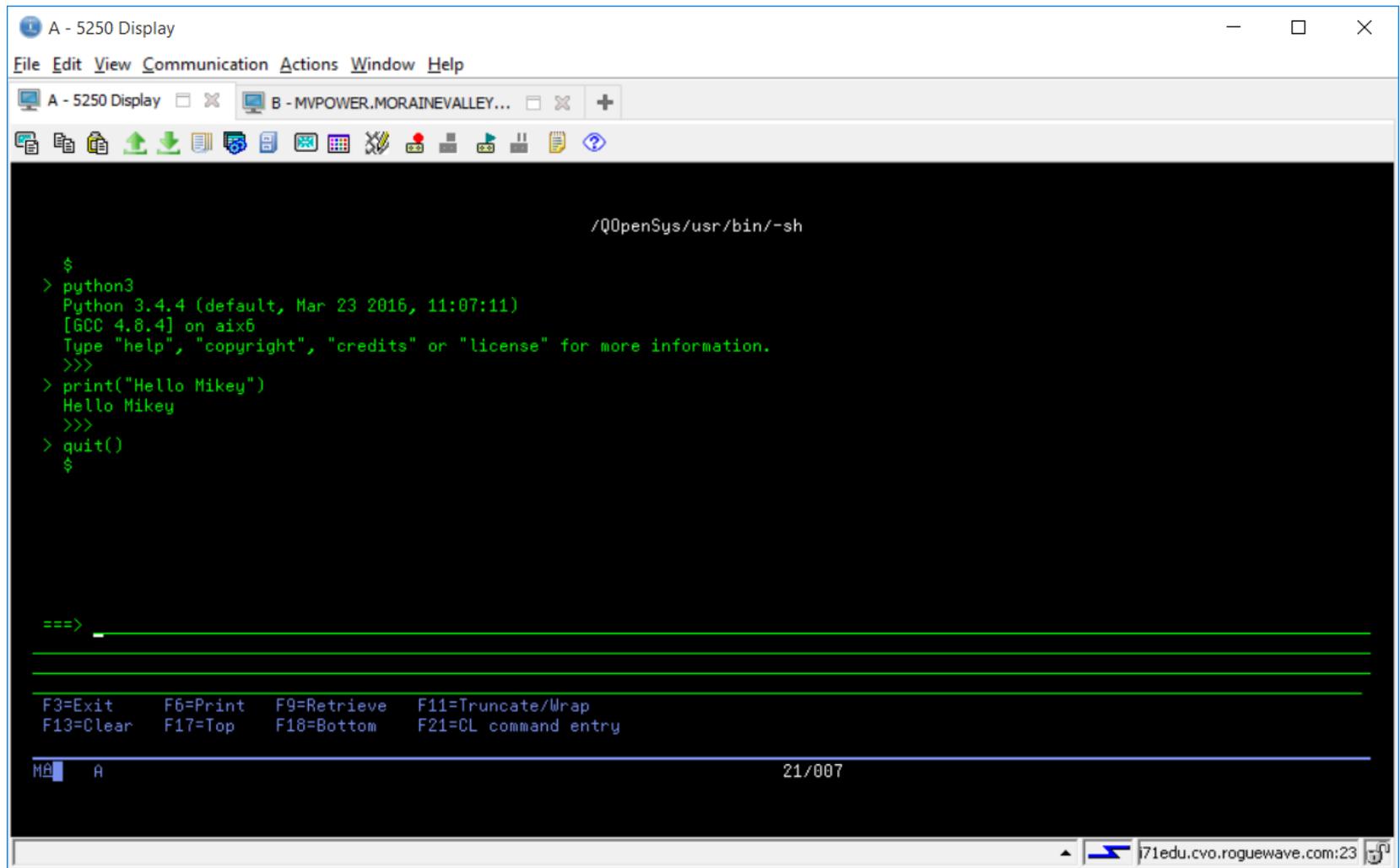
```
192.168.21.124 - PuTTY
$ python3 IBMiCommand4.py
The iToolKit version is: 1.4.0
5770SS1 V7R3M0 160422 Library List
10/07/18 16:23:21 Page 1
ASP
Library Type Device Text Description
QSYS SYS System Library
QSYS2 SYS System Library for CPI's
QHLPSYS SYS
QUSRSYS SYS System Library for Users
PYUSER USR
***** END OF LISTING *****
$
```

# Summary – Why Python

---

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

# End the session



The screenshot shows a remote 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 terminal content is as follows:

```
/QOpenSys/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
>>>
> quit()
$

===>
_____
_____
_____
_____
F3=Exit   F6=Print  F9=Retrieve  F11=Truncate/Wrap
F13=Clear  F17=Top   F18=Bottom  F21=CL command entry
M6 A                                           21/007
```

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



**"I AM SPEECHLESS, WITHOUT WORDS,  
AS SILENT AS A EUROPEAN SPARROW."**

**THANK YOU. THANK YOU. THANK YOU  
VERY MUCH.**

# THANK YOU

[Mike.Pavlak@freschesolutions.com](mailto:Mike.Pavlak@freschesolutions.com)



**FRESCHÉ** SOLUTIONS