



# Python Modules for Profitability

MIKE PAVLAK



[mikepavlak@gmail.com](mailto:mikepavlak@gmail.com)



# Today's schedule

---

12:00 - Introduction to Python for RPGers (Mike)

1:30 - Python and Data Access (Mike)

3:00 - RPG Makes Friends with Open Source Apps (Richard)

4:30 - Python Modules for Profitability (Mike)

7:00 - Pass the Flask & Quickly Pour IBM I Python Web Apps & Services (Richard)



# Agenda

---

Recent updates and announcements

Module Basics

- Passing arguments
- Paths

PSL – Python Standard Library

Examples

- Web service
- 3<sup>rd</sup> Party
- PDF & Spreadsheet
- Stocks

Flask

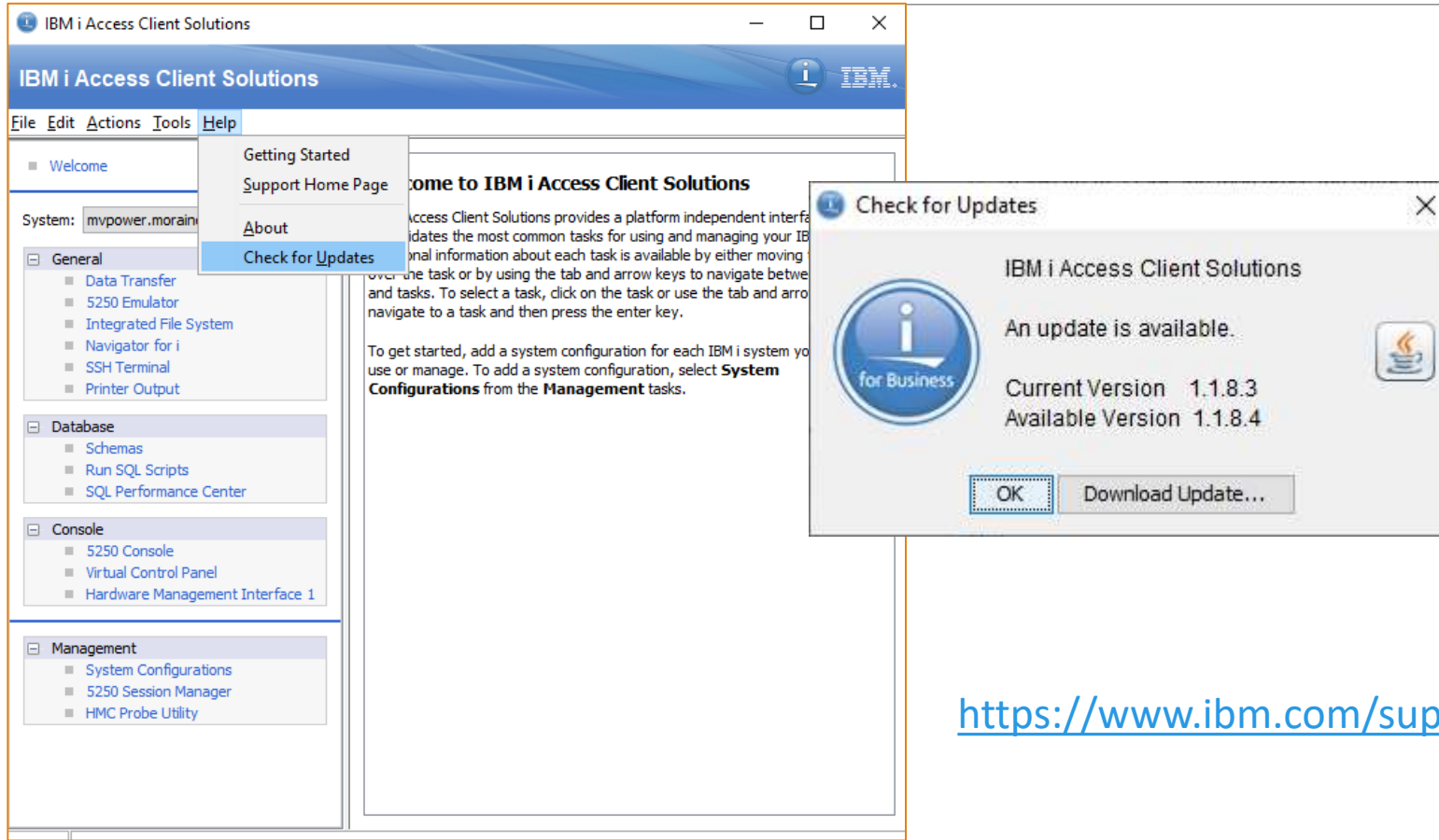


# Recent Updates

---

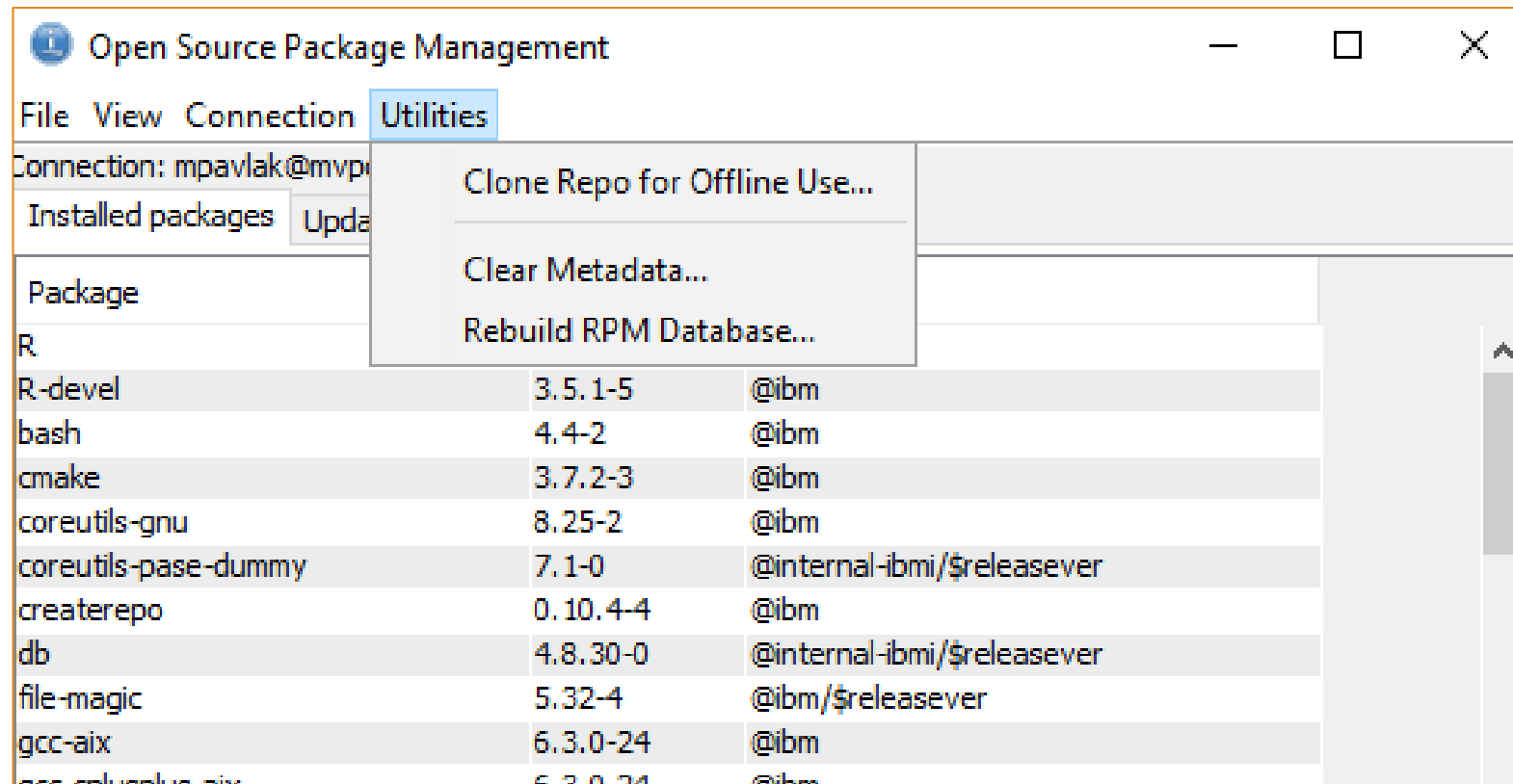
ACCESS CLIENT SOLUTIONS

# PSA: Update ACS

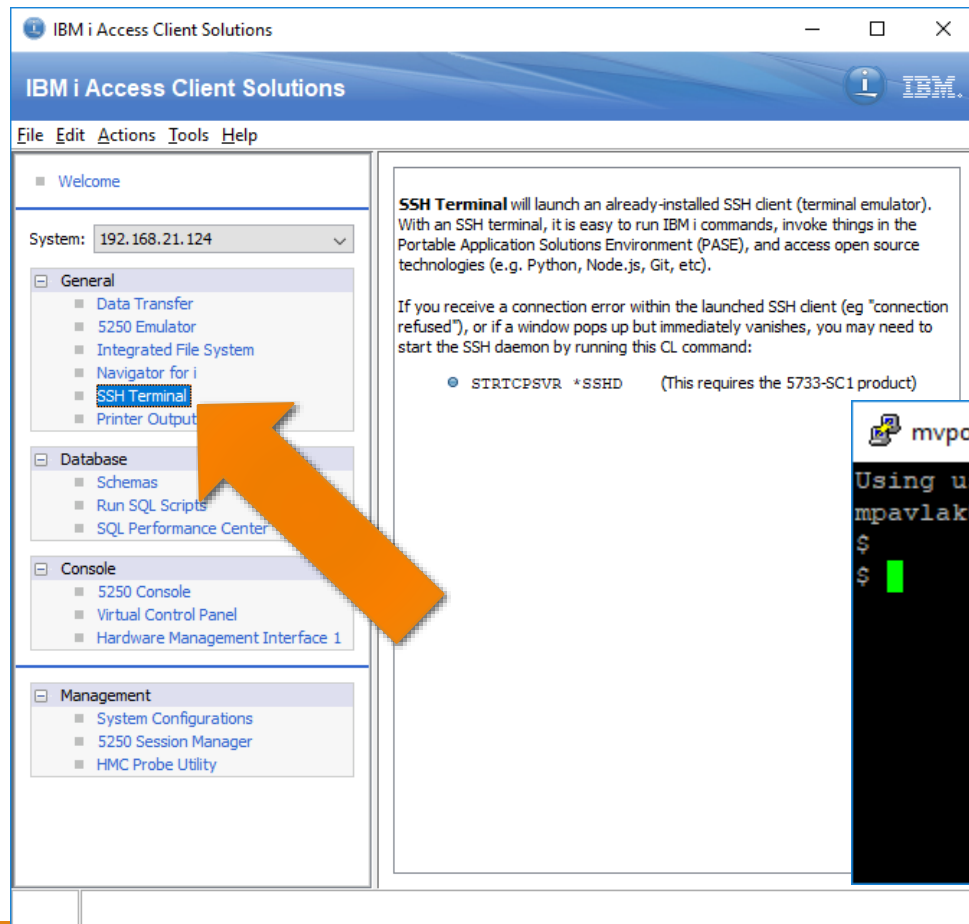


<https://www.ibm.com/support/pages/ibm-i-access-client-solutions>

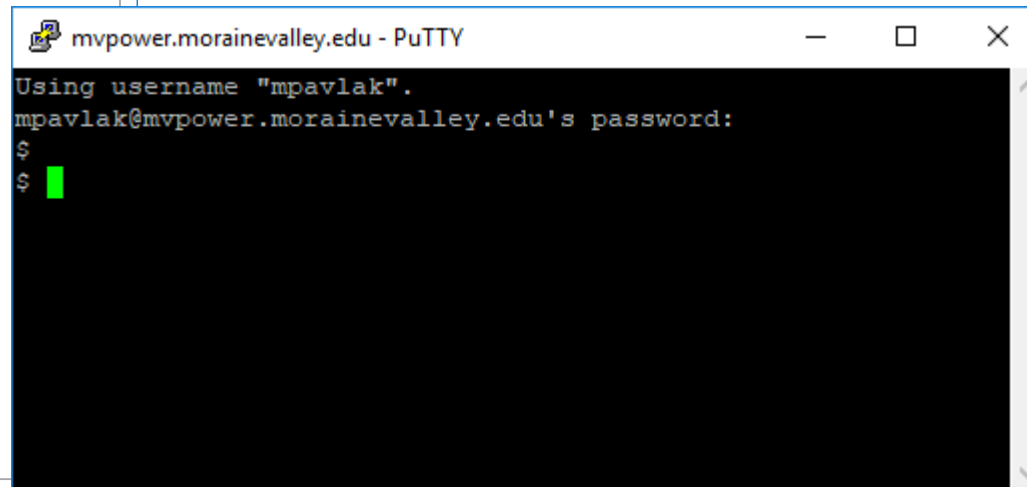
# ACS features in 1.1.8.4, 5



# Shell available in ACS (short PuTTY )



Encrypted  
BASH, etc.  
Linux alignment



# Module basics

---

PASSING ARGUMENTS, ETC.



# sys module list: argv

---

Need to import

First argument is ALWAYS the script name

```
from sys import argv  
  
print(argv)
```

```
bash-4.4$ python3 adv03argv.py  
['adv03argv.py']  
bash-4.4$
```

Something a little more practical

```
bash-4.4$ python3 adv03argv.py "Mikey" 1234 "IBM i"  
['adv03argv.py', 'Mikey', '1234', 'IBM i']  
bash-4.4$
```

<https://docs.python.org/3/library/sys.html>

# Use the values...

---

```
from sys import argv

script, name, number, system = argv

print("\nThe script name is", script)
print("Your name is", name)
print("Your number is", number)
print("Your system is", system)
```

```
The script name is adv03argv.py
Your name is Mikey
Your number is 1234
Your system is IBM i
bash-4.4$ █
```

```
from sys import argv

script = argv.pop(0)

print("\nThe script name is", script)

for name in argv:
    print("\nHello", name)
```

# Use many values...

---

```
from sys import argv

script = argv.pop(0)

print("\nThe script name is", script)

for name in argv:
    print("\nHeLLo", name)

print("\n\nYou gave me", Len(argv), "names")
```

```
bash-4.4$ python3 adv04argvmany.py "Pete" "Keith" "John" "Roger"

The script name is adv04argvmany.py

Hello Pete

Hello Keith

Hello John

Hello Roger

You gave me 4 names
bash-4.4$ █
```

# Module Basics

---

PYTHON PATHING



# Where does Python find stuff?

---

## Current Directory

- `os.getcwd`
- Can navigate

## PYTHONPATH

- Augment the search path for importing Python modules

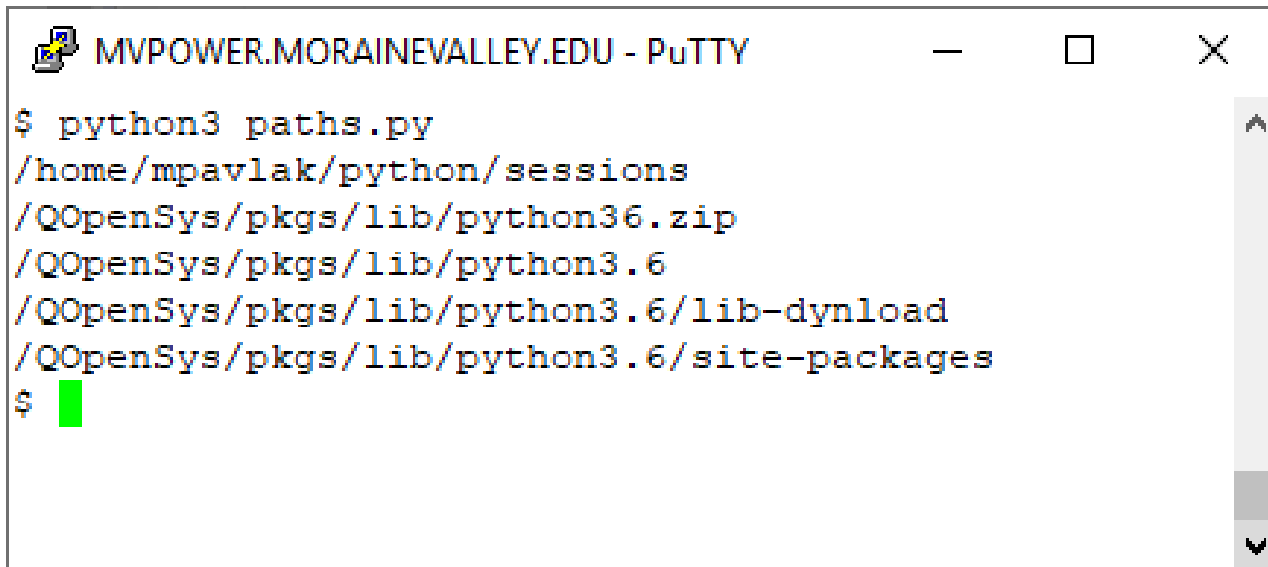
## Default Directory

- Where the script resides

# Where does Python find stuff?

---

Dump the paths



```
MVPOWER.MORAINEVALLEY.EDU - PuTTY
$ python3 paths.py
/home/mpavlak/python/sessions
/QOpenSys/pkgs/lib/python36.zip
/QOpenSys/pkgs/lib/python3.6
/QOpenSys/pkgs/lib/python3.6/lib-dynload
/QOpenSys/pkgs/lib/python3.6/site-packages
$
```

```
import sys

pathList = sys.path

for path in pathList:
    print(path)
```

# Modules and functions

---

## os.path module

- Built in module for system related stuff

## Pathlib

- Object Oriented path manipulation

## os.getcwd

- Display current working directory

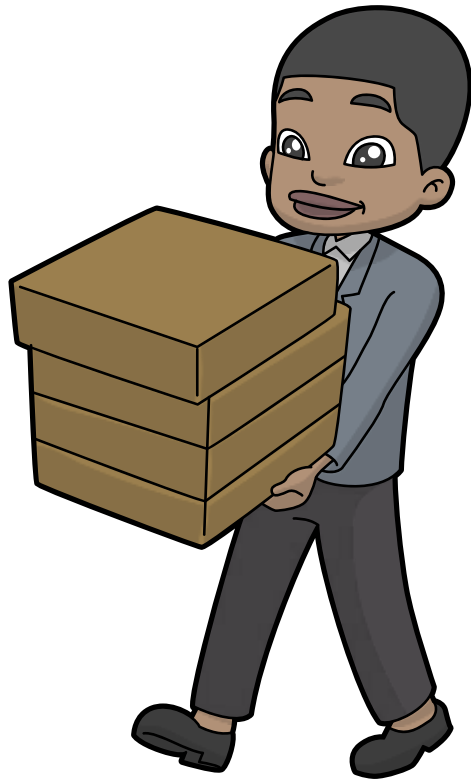
---

# Modules: The Secret Sauce of Python



# If you invite Python to a party...

---



**PHP**



**Python**

# What is a module

---

Code library – hunks of Python code

They come from the Python community

- PSL: Python Standard Library
- Home grown
- Third party

Usually Python, but not necessarily...

Designed to augment, not replace

Think of them as copybooks

Tested, trusted, useful...



OPC

# Base Python

---

Python tends to be lightweight

- Only about 60 built in functions
- `print()`, `str()`, `round()`, `set()`, etc.
- Keeps performance in check
- Minimalist or least access concept

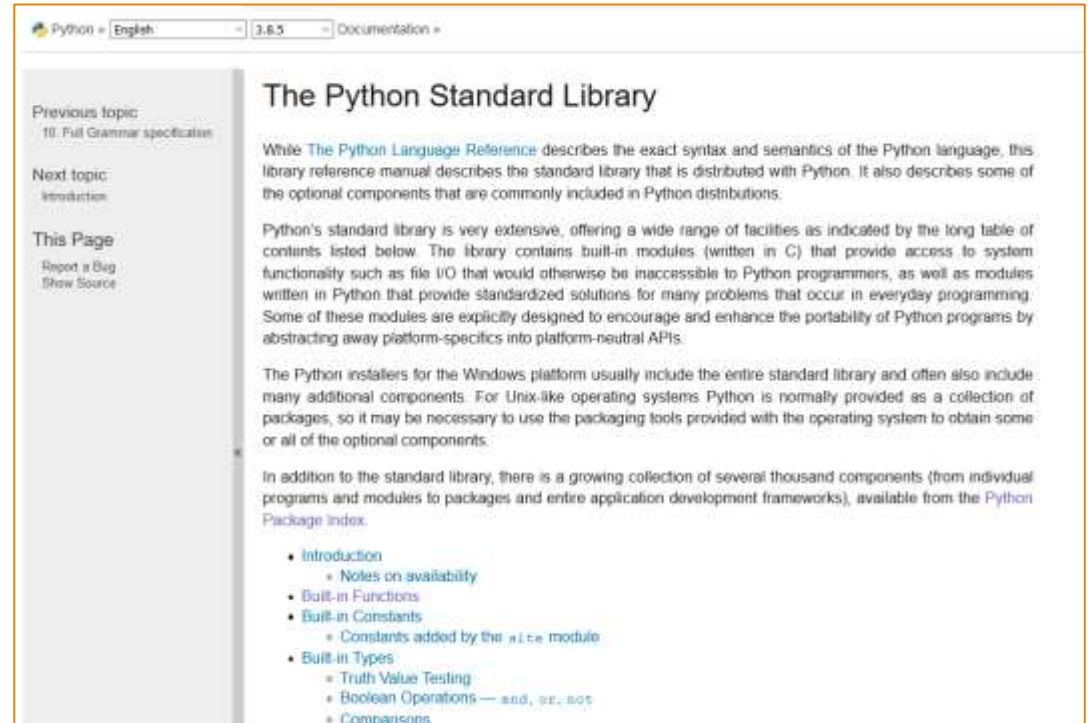
If you need something else...use a module & bring it in only if/when you need it!



# It all begins with the PSL

## Python Standard Library

- Collection of over 200 sets of functions
- Math, os, decimal, random, etc.
- Shipped with Python
- Must be “imported” in order to use
- Simple, powerful, accepted...
- Assumed to be available



The screenshot shows the Python Standard Library documentation page for version 3.8.5. The page title is "The Python Standard Library". The left sidebar contains navigation links: "Previous topic: 10. Full Grammar specification", "Next topic: Introduction", and "This Page" with sub-links "Report a Bug" and "Show Source". The main content area starts with an introduction paragraph, followed by a paragraph describing the library's scope and portability. It then mentions that the Python installer for Windows includes the entire standard library, while Unix-like systems provide it as a collection of packages. The page concludes by mentioning the Python Package Index and a list of sub-topics: Introduction (with a sub-link for Notes on availability), Built-in Functions, Built-in Constants (with a sub-link for Constants added by the sys module), and Built-in Types (with sub-links for Truth Value Testing, Boolean Operations — and, or, not, and Comparisons).

<https://docs.python.org/3/library/>

# Math module example

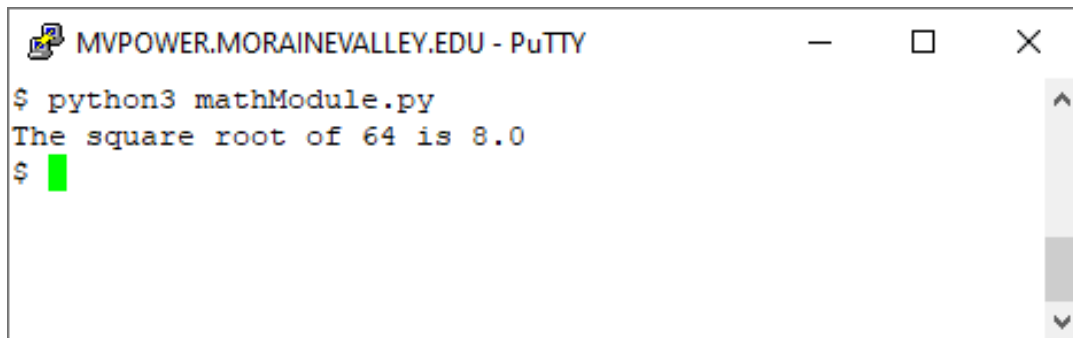
---

Let's say we want to do a square root or use pi

```
import math

x = 64
y = math.sqrt(x)

print("The square root of", x, "is", y)
```



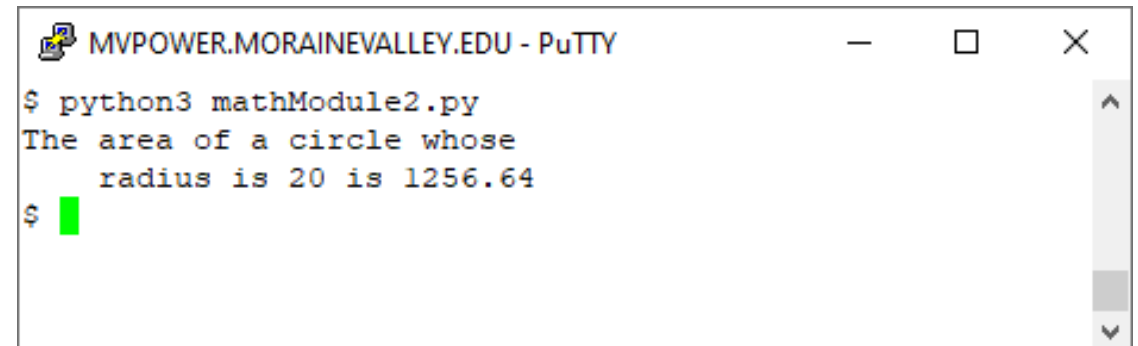
A terminal window titled "MVPOWER.MORAIN VALLEY.EDU - PuTTY" showing the execution of a Python script. The prompt "\$" is followed by the command "python3 mathModule.py". The output is "The square root of 64 is 8.0". The prompt "\$" is followed by a green cursor.

<https://docs.python.org/3/library/math.html>

```
import math

# Calculate the area of circle
radius = 20
area = round((math.pi * radius**2),2)

print("""The area of a circle whose
radius is""", radius, "is", area)
```



A terminal window titled "MVPOWER.MORAIN VALLEY.EDU - PuTTY" showing the execution of a Python script. The prompt "\$" is followed by the command "python3 mathModule2.py". The output is "The area of a circle whose radius is 20 is 1256.64". The prompt "\$" is followed by a green cursor.

# Other solid PSL candidates

---

OS

Sqlite3

Datetime

Decimal

Copy

Pprint

Random

Logging

Email

XML & JSON

Gettext

Doctest

Locale

Turtle

Pydoc

Sys

Site

Cmd

http

And many more

# Module Examples

---

WEB SERVICES

# Requests module

---

*“Apache2 HTTP library written in Python for human beings.”*

Make REST calls from a single line in Python script

```
>>> r = requests.get('https://api.github.com', auth=('user', 'pass'))
>>> r.status_code
204
>>> r.headers['content-type']
'application/json'
>>> r.text
...
```

<https://requests.readthedocs.io/en/master/>



# Example Access Watson

---

```
import requests  
import json
```

{ Bring in some functionality }

```
textinput='Hello'  
model='en-es'  
apikey= 'NOT A REAL API KEY...'  
url='https://gateway.watsonplatform.net/Language-  
translator/api/v3/translate?version=2018-05-01'  
data={'text':textinput, 'model_id':model}
```

{ Set some values }

```
r = requests.post(url, auth=('apikey', apikey), json=data)
```

{ Call service using requests }

```
jdata = json.loads(r.content)
```

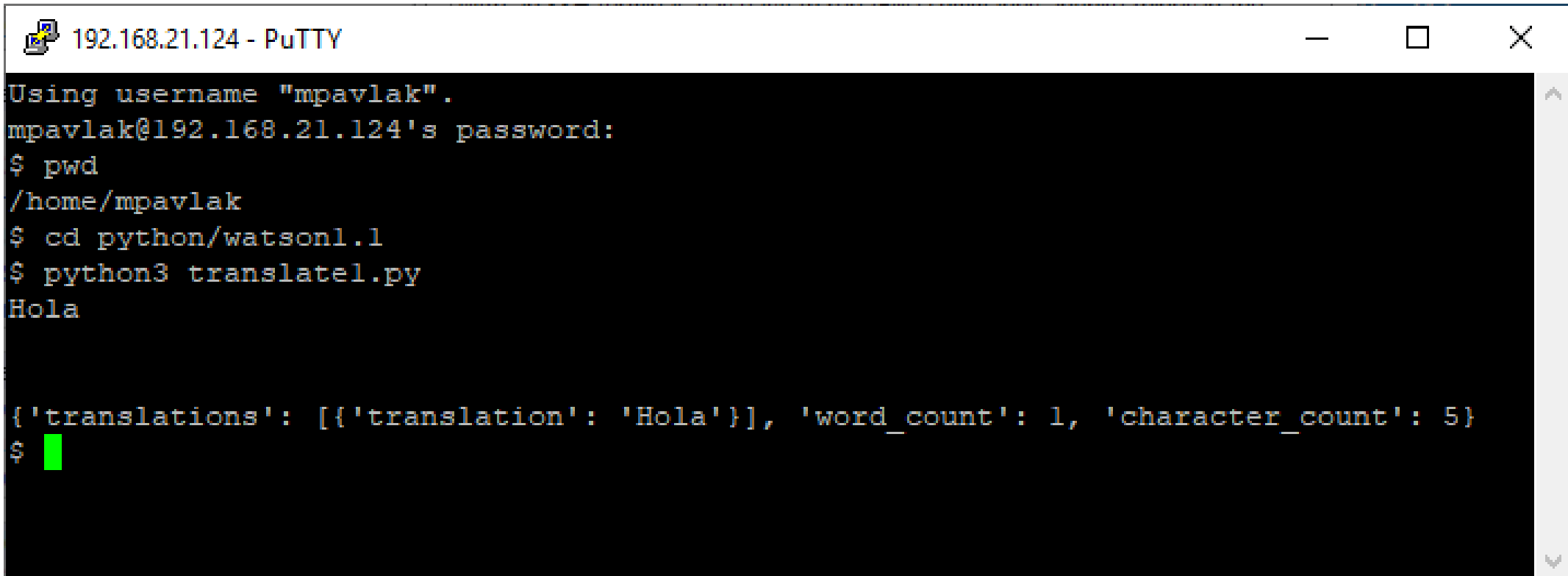
{ Call JSON to format data }

```
translatedtext=jdata['translations'][0]['translation']  
print(translatedtext, '\n', jdata)
```

{ Display something }

# Output

---



```
192.168.21.124 - PuTTY
Using username "mpavlak".
mpavlak@192.168.21.124's password:
$ pwd
/home/mpavlak
$ cd python/watson1.1
$ python3 translater.py
Hola

{'translations': [{'translation': 'Hola'}], 'word_count': 1, 'character_count': 5}
$ █
```

# Popular third party modules

---

## Data Science

- Mumpy, Pandas, maPlotLib, nltk

## Machine Learning

- Tensor Flow, Keras, Pytorch

## Web

- Django, Flask, Bottle

## Other

- Selenium, xlsxwriter,



TensorFlow



Keras



Flask



Bottle



# Third parties require an extra step

---

PSL is shipped with python, already installed

Third parties need to be installed in the Python directories

pip == Python Installer Program

*\*\*Note pip3 on IBM i*

Reads from repositories

- Pypi.org == public repository (Python Package Index)
- Others can be added to pip.conf (PerForce with PHP)

Can point to private repositories at time of install

<https://docs.python.org/3/installing/index.html>

# fpdf

---

Simple example of installing with fpdf.

- Create PDF files

Command: **pip3 install fpdf**



```
MVPOWER.MORAINEVALLEY.EDU - PuTTY
$ pip3 install fpdf
Collecting fpdf
  Downloading https://files.pythonhosted.org/packages/37/c6/608a9e6c172bf9124aa687ec8b9f0e8e5d697d59a5f4fad0e2d5ec2a7556/fpdf-1.7.2.tar.gz
Building wheels for collected packages: fpdf
  Running setup.py bdist_wheel for fpdf ... done
  Stored in directory: /home/mpavlak/.cache/pip/wheels/9a/e9/77/4554ff5c99bc3f487c8d69620d8c41d99d54e9c54ab20ef4c9
Successfully built fpdf
Installing collected packages: fpdf
Successfully installed fpdf-1.7.2
$ █
```

<https://pyfpdf.readthedocs.io/en/latest/>

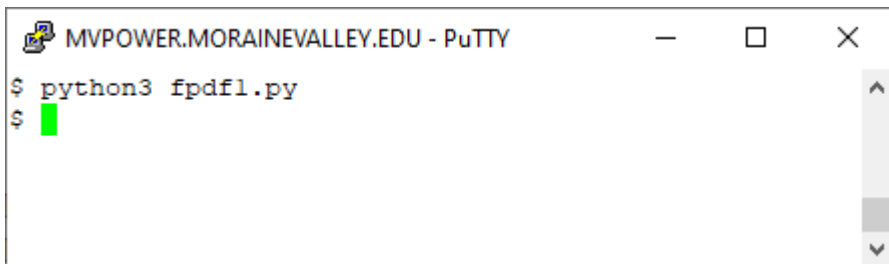
# Code & output

```
from fpdf import FPDF

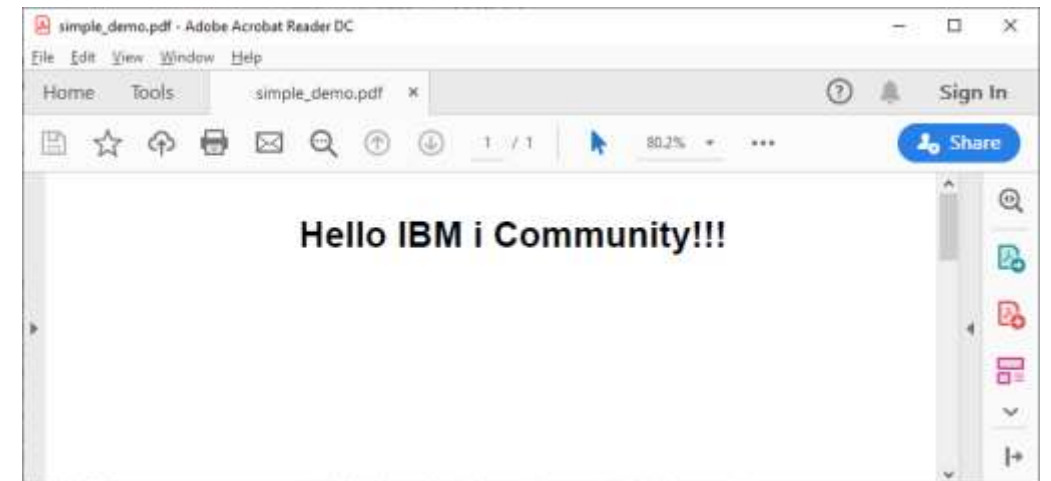
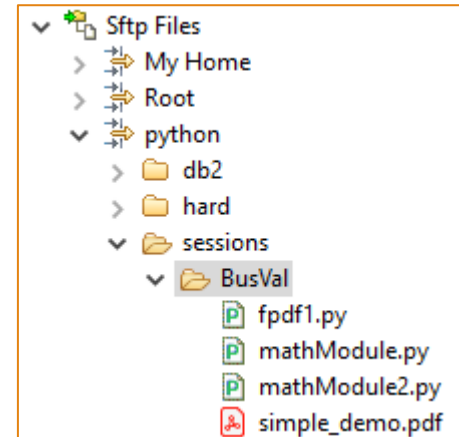
pdf = FPDF()
pdf.add_page()
pdf.set_font("Arial", size=24)

pdf.cell(200, 10, txt="Hello IBM i Community!!!",
        ln=1, align="C")

pdf.output("simple_demo.pdf")
```



```
MVPOWER.MORAINEVALLEY.EDU - PuTTY
$ python3 fpdf1.py
$
```



# Wikipedia

Access from a command line...

- pip3 install wikipedia

```
mvpower.morainevalley.edu - PuTTY
bash-4.4$ pip3 install wikipedia
Collecting wikipedia
  Downloading https://files.pythonhosted.org/packages/67/35/25e68fbc99e672127cc6fbb14b8ec1ba3dfef035bf1e4c90f78f24a80b7d/wikipedia-1.4.0.tar.gz
Collecting beautifulsoup4 (from wikipedia)
  Downloading https://files.pythonhosted.org/packages/66/25/ff030e2437265616a1e9b25ccc864e0371a0bc3adb7c5a404fd661c6f4f6/beautifulsoup4-4.9.1-py3-none-any.whl (115kB)
100% |#####| 122kB 3.7MB/s
Requirement already satisfied: requests<3.0.0,>=2.0.0 in /QOpenSys/pkgs/lib/python3.6/site-packages (from wikipedia)
Collecting soupsieve>1.2 (from beautifulsoup4->wikipedia)
  Downloading https://files.pythonhosted.org/packages/6f/8f/457f4a5390eeae1cc3aeab89deb7724c965be841ffca6cfca9197482e470/soupsieve-2.0.1-py3-none-any.whl
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in /QOpenSys/pkgs/lib/python3.6/site-packages (from requests<3.0.0,>=2.0.0->wikipedia)
Requirement already satisfied: urllib3!=1.25.0,!<1.25.1,<1.26,>=1.21.1 in /QOpenSys/pkgs/lib/python3.6/site-packages (from requests<3.0.0,>=2.0.0->wikipedia)
Requirement already satisfied: certifi>=2017.4.17 in /QOpenSys/pkgs/lib/python3.6/site-packages (from requests<3.0.0,>=2.0.0->wikipedia)
Requirement already satisfied: idna<2.9,>=2.5 in /QOpenSys/pkgs/lib/python3.6/site-packages (from requests<3.0.0,>=2.0.0->wikipedia)
Building wheels for collected packages: wikipedia
  Running setup.py bdist_wheel for wikipedia ... done
  Stored in directory: /home/mpavlak/.cache/pip/wheels/87/2a/18/4e471fd96d12114d16fe4a446d00c3b38fb9efcb744bd31f4a
Successfully built wikipedia
Installing collected packages: soupsieve, beautifulsoup4, wikipedia
Successfully installed beautifulsoup4-4.9.1 soupsieve-2.0.1 wikipedia-1.4.0
bash-4.4$
```

```
import wikipedia
```

```
searchSubject = "IBM i"
```

```
response = wikipedia.page(searchSubject)
```

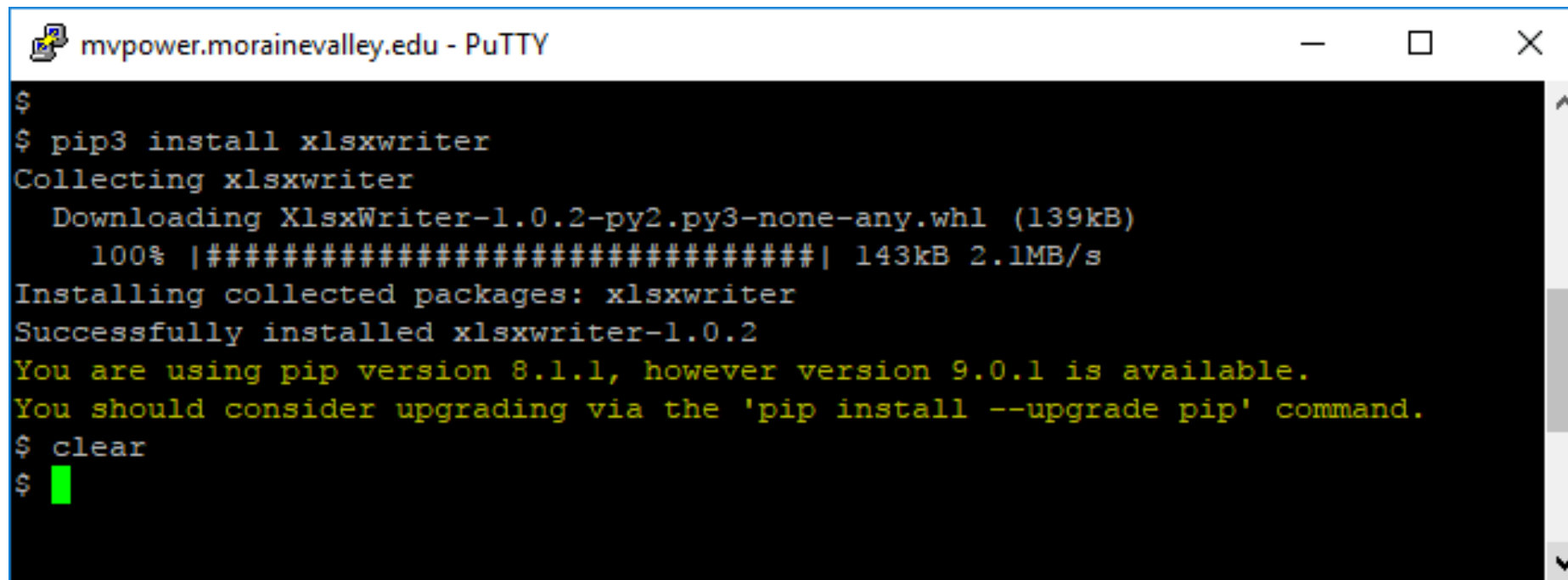
```
print(response.summary)
```

```
mvpower.morainevalley.edu - PuTTY
bash-4.4$ python3 wikipedia.py
IBM i is an operating system or operating environment providing an abstract interface to IBM Power Systems hardware via layers of low-level machine interface code (MI) or Microcode that reside above the Technology Independent Machine Interface (TIMI) and the System Licensed Internal Code (SLIC) or kernel, described by IBM i chief architect Steve Will, and ensures application compatibility across multiple technology generations, and is a topic of discussion.
bash-4.4$ █
```

<https://pypi.org/project/wikipedia/>

# Spreadsheet? Install xlswriter

---



```
mvpower.morainevalley.edu - PuTTY
$
$ pip3 install xlswriter
Collecting xlswriter
  Downloading XlsxWriter-1.0.2-py2.py3-none-any.whl (139kB)
    100% |#####| 143kB 2.1MB/s
Installing collected packages: xlswriter
Successfully installed xlswriter-1.0.2
You 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.
$ clear
$ █
```



# Got doc?

<http://xlsxwriter.readthedocs.io/chart.html>

XlsxWriter

Contents ▾

Page ▾

« The Format Class

The Chartsheet Class »

Search

## The Chart Class

The `Chart` module is a base class for modules that implement charts in XlsxWriter. The information in this section is applicable to all of the available chart subclasses, such as Area, Bar, Column, Doughnut, Line, Pie, Scatter, Stock and Radar.

A chart object is created via the Workbook `add_chart()` method where the chart type is specified:

```
chart = workbook.add_chart({'type': 'column'})
```

It is then inserted into a worksheet as an embedded chart using the `insert_chart()` Worksheet method:

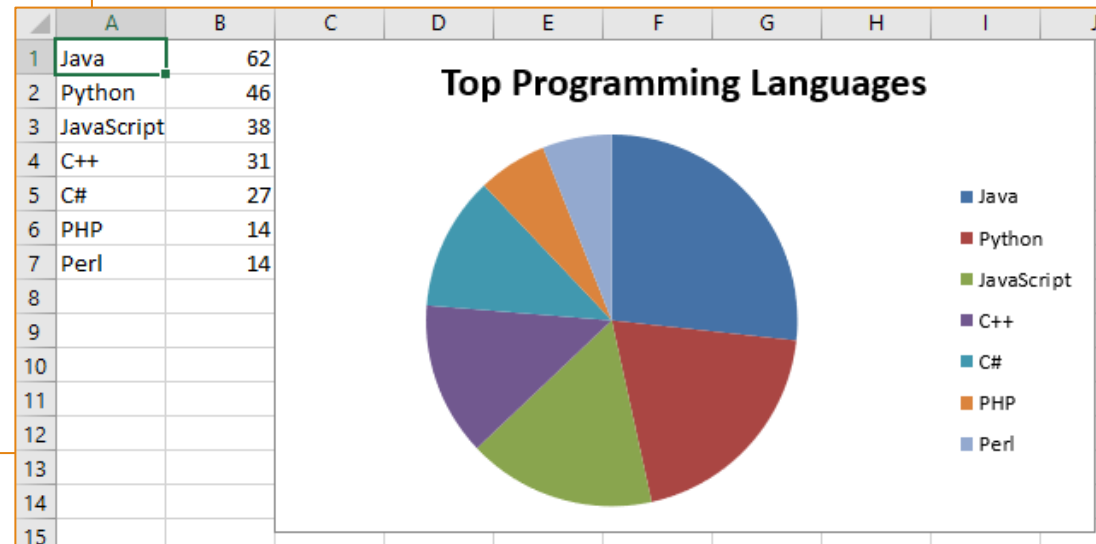
```
worksheet.insert_chart('A7', chart)
```

# Now let's do something cool

```
from xlswriter import Workbook

with Workbook('test.xlsx') as workbook:
    ws = workbook.add_worksheet()
    ws.write_column('A1', ['Java', 'Python', 'JavaScript',
                          'C++', 'C#', 'PHP', 'Perl'])
    ws.write_column('B1', [62, 46, 38, 31, 27, 14, 14])
    #chart = workbook.add_chart({'type': 'line'})
    chart = workbook.add_chart({'type': 'pie'})
    chart.set_title({'name': 'Top Programming Languages'})
    chart.add_series({'categories': '=Sheet1!$A$1:$A$7',
                     'values': '=Sheet1!$B$1:$B$7'})

    ws.insert_chart('C1', chart)
```



# Stock prices



<https://pypi.org/project/yahoo-finance/>

```
#yFinance...
import yfinance as yf

#List of stocks

stocks = ['msft', 'aapl', 'amzn']

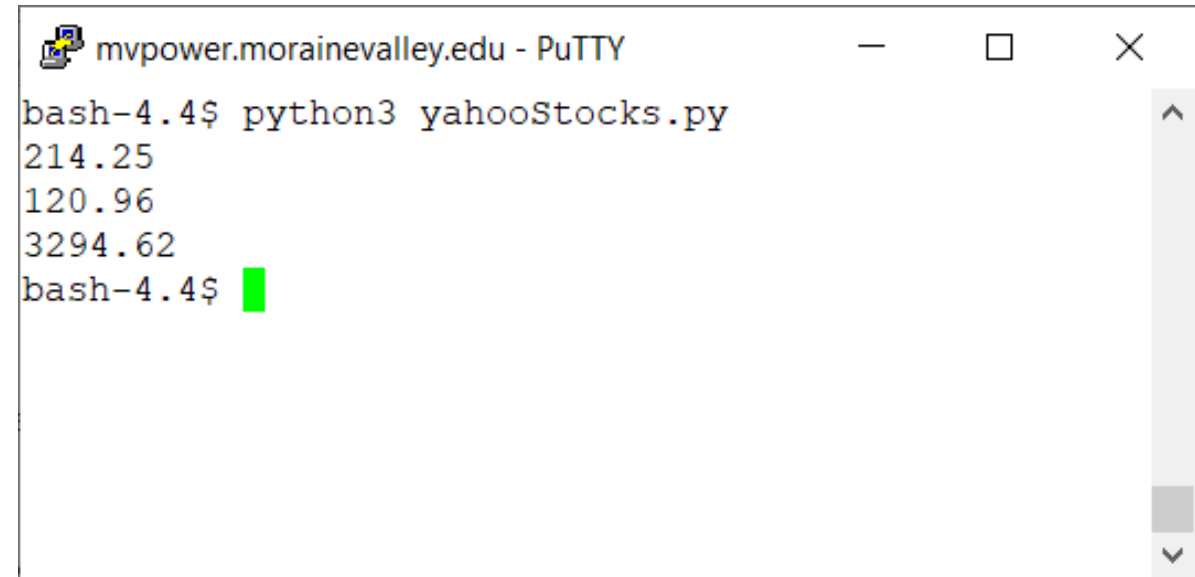
stocksParm = ' '.join(stocks)

tickers = yf.Tickers(stocksParm)

print(tickers.tickers.MSFT.history("1d")["Close"].values[0])

print(tickers.tickers.AAPL.history("1d")["Close"].values[0])

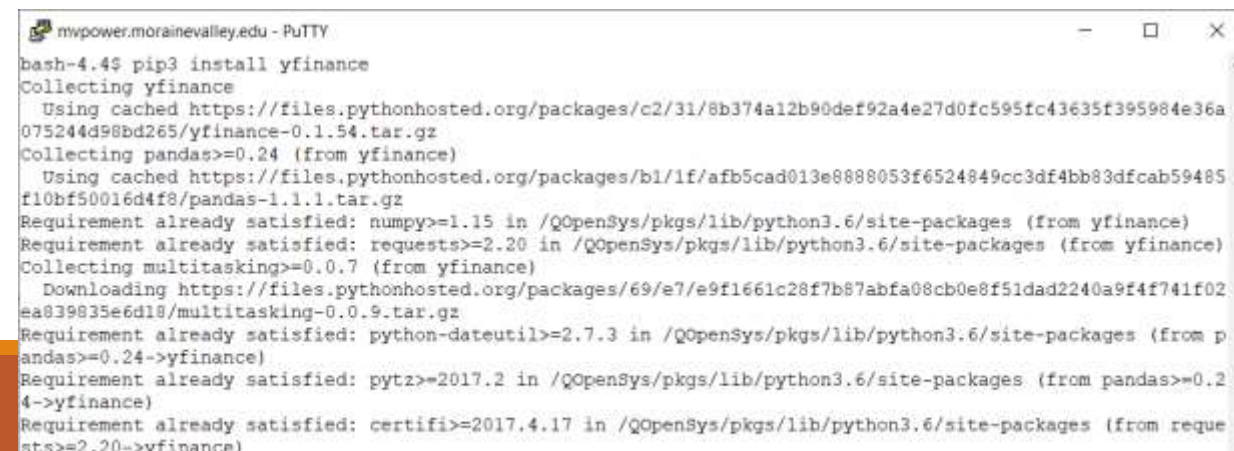
print(tickers.tickers.AMZN.history("1d")["Close"].values[0])
```



```
mvpower.morainevalley.edu - PuTTY
bash-4.4$ python3 yahooStocks.py
214.25
120.96
3294.62
bash-4.4$ █
```



```
mvpower.morainevalley.edu - PuTTY
bash-4.4$ pip3 install Cython
Collecting Cython
  Downloading https://files.pythonhosted.org/packages/ad/4b/9e53bcce3c959fd0
.py3-none-any.whl (974kB)
    100% |#####| 983kB 918kB/s
Installing collected packages: Cython
Successfully installed Cython-0.29.21
bash-4.4$ █
```



```
mvpower.morainevalley.edu - PuTTY
bash-4.4$ pip3 install yfinance
Collecting yfinance
  Using cached https://files.pythonhosted.org/packages/c2/31/8b374a12b90def92a4e27d0fc595fc43635f395984e36a
075244d98bd265/yfinance-0.1.54.tar.gz
Collecting pandas>=0.24 (from yfinance)
  Using cached https://files.pythonhosted.org/packages/b1/1f/afb5cad013e888053f6524849cc3df4bb83dfcab59485
f10bf50016d4f8/pandas-1.1.1.tar.gz
Requirement already satisfied: numpy>=1.15 in /QOpenSys/pkgs/lib/python3.6/site-packages (from yfinance)
Requirement already satisfied: requests>=2.20 in /QOpenSys/pkgs/lib/python3.6/site-packages (from yfinance)
Collecting multitasking>=0.0.7 (from yfinance)
  Downloading https://files.pythonhosted.org/packages/69/e7/e9f1661c28f7b87abfa08cb0e8f51dad2240a9f4f741f02
ea839835e6d18/multitasking-0.0.9.tar.gz
Requirement already satisfied: python-dateutil>=2.7.3 in /QOpenSys/pkgs/lib/python3.6/site-packages (from p
andas>=0.24->yfinance)
Requirement already satisfied: pytz>=2017.2 in /QOpenSys/pkgs/lib/python3.6/site-packages (from pandas>=0.2
4->yfinance)
Requirement already satisfied: certifi>=2017.4.17 in /QOpenSys/pkgs/lib/python3.6/site-packages (from requ
sts>=2.20->yfinance)
```

# PYODBC

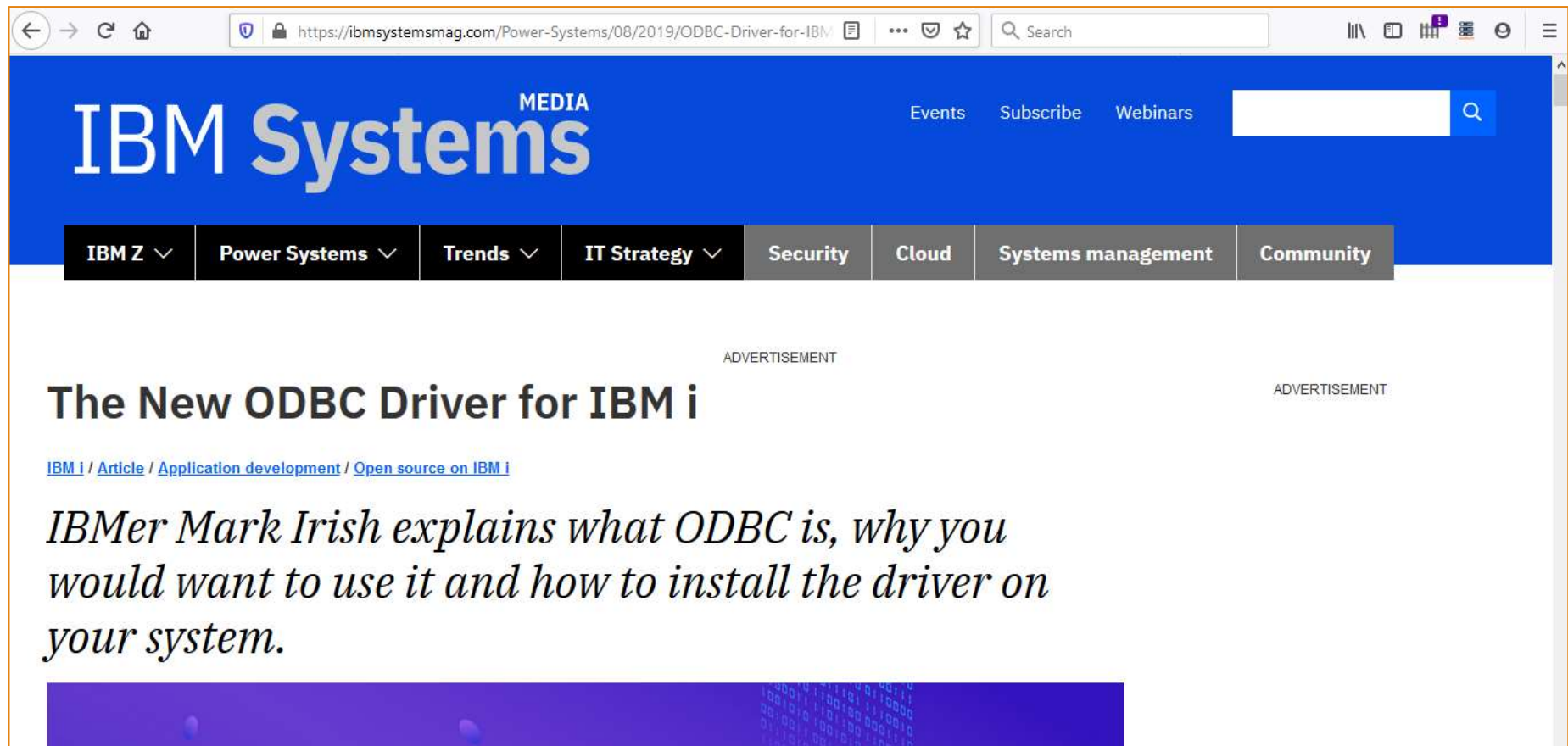
---

DATABASE

# ODBC is where it's at...

Read the article by Mark Irish

<https://ibmsystemsmag.com/Power-Systems/08/2019/ODBC-Driver-for-IBM-i>



The screenshot shows a web browser displaying the IBM Systems Media website. The page features a blue header with the IBM Systems logo and navigation links for Events, Subscribe, and Webinars. Below the header is a dark navigation bar with categories like IBM Z, Power Systems, Trends, IT Strategy, Security, Cloud, Systems management, and Community. The main content area has a white background with a large heading 'The New ODBC Driver for IBM i' and a sub-heading 'ADVERTISEMENT'. The article text is in italics, stating: 'IBMer Mark Irish explains what ODBC is, why you would want to use it and how to install the driver on your system.' The page also includes a breadcrumb trail: 'IBM i / Article / Application development / Open source on IBM i'.

ADVERTISEMENT

## The New ODBC Driver for IBM i

[IBM i](#) / [Article](#) / [Application development](#) / [Open source on IBM i](#)

*IBMer Mark Irish explains what ODBC is, why you would want to use it and how to install the driver on your system.*

ADVERTISEMENT

# ODBC Needs 3 pieces to work

ODBC Driver

ODBC Driver Manager

Application specific connectors (adapter)

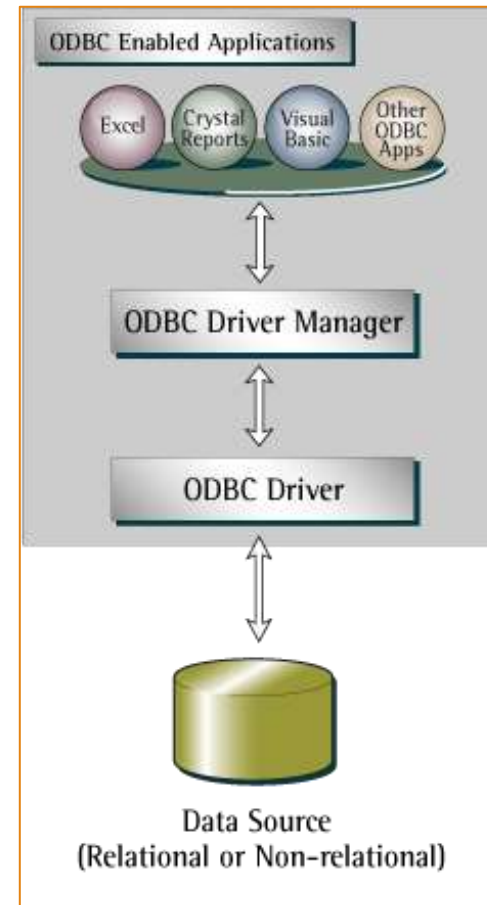
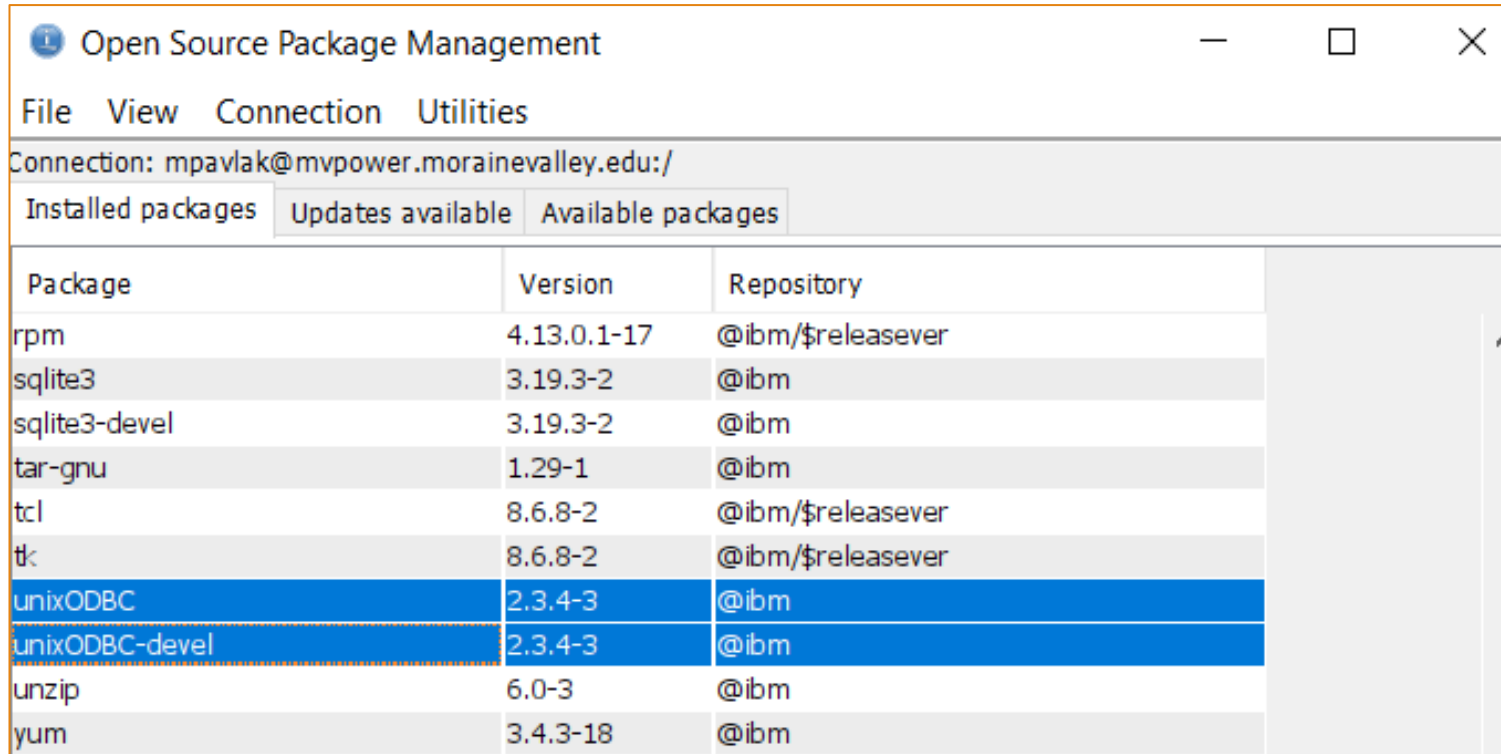


Image: Wikimedia Commons

# ODBC: Driver Manager

## Step 1

- Command line: **yum install unixODBC unixODBC-devel**
- GUI:



The screenshot shows the 'Open Source Package Management' window. The connection is 'mpavlak@mvpower.morainevalley.edu/'. The 'Available packages' tab is selected. A table lists the following packages:

Package	Version	Repository
rpm	4.13.0.1-17	@ibm/\$releasever
sqlite3	3.19.3-2	@ibm
sqlite3-devel	3.19.3-2	@ibm
tar-gnu	1.29-1	@ibm
tcl	8.6.8-2	@ibm/\$releasever
tk	8.6.8-2	@ibm/\$releasever
unixODBC	2.3.4-3	@ibm
unixODBC-devel	2.3.4-3	@ibm
unzip	6.0-3	@ibm
yum	3.4.3-18	@ibm

# ODBC Driver is on ACS site

## Step 2:

- Dowload
- Install

Description	Filename	Size	Action
IBM i Access Client Solutions	IBMiAccess_v1r1.zip	134107575 B	<a href="#">Download</a> ↓
Readme file for 5733-XJ1	Readme.txt	5 KB	<a href="#">Download</a> ↓
Quick Start Guide	QuickStartGuide_en.html	10 KB	<a href="#">Download</a> ↓
Getting Started	GettingStarted_en.html	119 KB	<a href="#">Download</a> ↓
ACS EHLAPI - Version 8.20.104	acshllapi.exe	986862	<a href="#">Download</a> ↓
ACS Windows App Pkg English (64bit)	IBMiAccess_v1r1_WindowsAP_English.zip	53311376 B	<a href="#">Download</a> ↓
ACS Linux App Pkg	IBMiAccess_v1r1_LinuxAP.zip	20797233 B	<a href="#">Download</a> ↓
ACS PASE App Pkg	IBMiAccess_v1r1_PASE_AP.zip	8577229 B	<a href="#">Download</a> ↓



# PyODBC: Application Adapter for Python

## Step 3:

- Use IBM Open Source package Manager in ACS
  - Highlight
  - Click install
  - Type “y”
- Driver is strategic to python data access.
- Indirectly supported

python3	3.6.10-1	@ibm
python3-Pillow	5.0.0-5	@ibm
python3-asn1crypto	0.24.0-1	@ibm
python3-bcrypt	3.1.4-6	@ibm
python3-cffi	1.11.5-3	@ibm
python3-cryptography	2.8-0	@ibm
python3-dateutil	2.7.5-1	@ibm
python3-devel	3.6.10-1	@ibm
python3-ibm_db	2.0.5.12-0	@ibm
python3-idna	2.8-1	@ibm
python3-itoolkit	1.6.1-1	@ibm
python3-lxml	4.2.1-4	@ibm
python3-numpy	1.15.4-0	@ibm/\$releasever
python3-pandas	0.22.0-5	@ibm
python3-paramiko	2.6.0-1	@ibm
python3-pip	9.0.1-3	@ibm
python3-psutil	5.5.1-0	@ibm
python3-pyparser	2.19-2	@ibm
python3-pynacl	1.2.1-4	@ibm
python3-pyodbc	4.0.27-0	@ibm
python3-pytz	2018.5-3	@ibm
python3-pyzmq	17.1.2-0	@ibm
python3-rpm	4.13.0.1-18	@ibm
python3-scikit-learn	0.19.1-7	@ibm
python3-scipy	1.1.0-1	@ibm
python3-setuptools	36.0.1-3	@ibm
python3-six	1.10.0-1	@ibm
python3-tkinter	3.6.10-1	@ibm
python3-wheel	0.29.0-3	@ibm

# Steps for simple database Access

---

Import the class

Connect (with or without options)

Open the cursor

Set the SQL

Read

# Simple script

```
import pyodbc

cnxn = pyodbc.connect('DSN=*LOCAL')

cursor = cnxn.cursor()

#Sample select query
cursor.execute('''SELECT CUST_ID, COMPANY, FIRSTNAME, LASTNAME, COUNTRY
                from zendphp7.sp_cust''')

row = cursor.fetchone()
while row:
    print(row)
    row = cursor.fetchone()
```



```
mvpower.morainevalley.edu - PuTTY
bash-4.4$ python3 pyodbc03select.py
(Decimal('1221'), 'Kauai Dive Shoppe', 'LINA', 'Norman', 'US')
(Decimal('1231'), 'Unisco', 'George', 'Weathers', 'Bahamas')
(Decimal('1351'), 'Sight Diver', 'Phyllis', 'Spooner', 'Cyprus')
(Decimal('1354'), 'Cayman Divers World Unlimited', 'Joe', 'Bailey', 'British West Indies')
(Decimal('1356'), 'Tom Sawyer Diving Centre', 'Chris', 'Thomas', 'US Virgin Islands')
(Decimal('1380'), 'Blue Jack Aqua Center', 'Ernest', 'Barratt', 'US')
(Decimal('1384'), 'VIP Divers Club', 'Russell', 'Christopher', 'US Virgin Islands')
(Decimal('1510'), 'Ocean Paradise', 'Paul', 'Gardner', 'US')
(Decimal('1513'), 'Fantastique Aquatica', 'Susan', 'Wong', 'Columbia')
(Decimal('1551'), 'Marmot Divers Club', 'Joyce', 'Marsh', 'Canada')
(Decimal('1560'), 'The Depth Charge', 'Sam', 'Witherspoon', 'US')
(Decimal('1563'), 'Blue Sports', 'Theresa', 'Kunec', 'US')
(Decimal('1624'), 'Makai SCUBA Club', 'Donna', 'Siaus', 'US')
(Decimal('1645'), 'Action Club', 'Michael', 'Spurling', 'US')
(Decimal('1651'), 'Jamaica SCUBA Centre', 'Barbara', 'Harvey', 'West Indies')
(Decimal('1680'), 'Island Finders', 'Desmond', 'Ortega', 'US')
(Decimal('1984'), 'Adventure Undersea', 'Gloria', 'Gonzales', 'Belize')
(Decimal('2118'), 'Blue Sports Club', 'Harry', 'Bathbone', 'US')
(Decimal('2135'), 'Frank's Divers Supply', 'Lloyd', 'Fellows', 'US')
(Decimal('2156'), 'Davy Jones' Locker', 'Tanya', 'Wagner', 'Canada')
(Decimal('2163'), 'SCUBA Heaven', 'Robert', 'Michelind', 'Bahamas')
(Decimal('2165'), 'Shangri-La Sports Center', 'Frank', 'Paniagua', 'Bahamas')
(Decimal('2315'), 'Divers of Corfu, Inc.', 'Charles', 'Lopez', 'Greece')
```

# For DML type transactions

```
import pyodbc

cnxn = pyodbc.connect('DSN=*LOCAL;CommitMode=0') #Turn off commitment control!

cursor = cnxn.cursor()

#Sample select query
cursor.execute("""update zendphp7.sp_cust
                set FIRSTNAME = 'Mike' where CUST_ID = 9841""")
```

Before:

CUST_ID	COMPANY	FIRSTNAME	LASTNAME	CIVIL	ADDRESS
9841	Neptune's Trident Supply	MIKE	Franks	2	PO Box 129

After:

CUST_ID	COMPANY	FIRSTNAME	LASTNAME	CIVIL	ADDRESS
9841	Neptune's Trident Supply	Mike	Franks	2	PO Box 129

---

FLASK

# What is Flask?

---

Popular Micro web framework

Automates the boring “webby” stuff

Fast method to the web pages

Productivity == \$\$\$

Community and documentation

Very explicit code style, no large learning curve



<https://pypi.org/project/Flask/>



# Installation on Windows

---

```
PowerShell 6 (x64)
PS C:\python\samples\flask> pip install Flask
Collecting Flask
  Downloading Flask-1.1.2-py2.py3-none-any.whl (94 kB)
    |████████████████████████████████████████| 94 kB 787 kB/s
Collecting itsdangerous>=0.24
  Downloading itsdangerous-1.1.0-py2.py3-none-any.whl (16 kB)
Collecting Werkzeug>=0.15
  Downloading Werkzeug-1.0.1-py2.py3-none-any.whl (298 kB)
    |████████████████████████████████████████| 298 kB 6.8 MB/s
Requirement already satisfied: Jinja2>=2.10.1 in c:\users\mike.pavlak\appdata\roaming\python\python37\site-packages (from
m Flask) (2.10.1)
Collecting click>=5.1
  Downloading click-7.1.2-py2.py3-none-any.whl (82 kB)
    |████████████████████████████████████████| 82 kB 446 kB/s
Requirement already satisfied: MarkupSafe>=0.23 in c:\users\mike.pavlak\appdata\roaming\python\python37\site-packages (f
rom Jinja2>=2.10.1->Flask) (1.1.1)
Could not build wheels for Jinja2, since package 'wheel' is not installed.
Could not build wheels for MarkupSafe, since package 'wheel' is not installed.
Installing collected packages: itsdangerous, Werkzeug, click, Flask
Successfully installed Flask-1.1.2 Werkzeug-1.0.1 click-7.1.2 itsdangerous-1.1.0
PS C:\python\samples\flask>
```

# Installation on IBM i: pip<sub>3</sub> install Flask

---

```
bash-4.4$ pip3 install Flask
Collecting Flask
  Using cached https://files.pythonhosted.org/packages/f2/28/2a03252dfb9ebf377f40fba6a7841b47083260bf8bd8e737b0c6952df83f/Flask-1.1.2-py2.py3-none-any.whl
Collecting Werkzeug>=0.15 (from Flask)
  Using cached https://files.pythonhosted.org/packages/cc/94/5f7079a0e00bd6863ef8f1da638721e9da21e5bacee597595b318f71d62e/Werkzeug-1.0.1-py2.py3-none-any.whl
Collecting click>=5.1 (from Flask)
  Using cached https://files.pythonhosted.org/packages/d2/3d/fa76db83bf75c4f8d338c2fd15c8d33fdd7ad23a9b5e57eb6c5de26b430e/click-7.1.2-py2.py3-none-any.whl
Collecting itsdangerous>=0.24 (from Flask)
  Using cached https://files.pythonhosted.org/packages/76/ae/44b03b253d6fade317f32c24d100b3b35c2239807046a4c953c7b89fa49e/itsdangerous-1.1.0-py2.py3-none-any.whl
Collecting Jinja2>=2.10.1 (from Flask)
  Using cached https://files.pythonhosted.org/packages/30/9e/f663a2aa66a09d838042a1a2c5659828bb9b41ea3a6efa20a20fd92b121/Jinja2-2.11.2-py2.py3-none-any.whl
Collecting MarkupSafe>=0.23 (from Jinja2>=2.10.1->Flask)
  Using cached https://files.pythonhosted.org/packages/b9/2e/64db92e53b86efccfaea71321f597fa2e1b2bd3853d8ce658568f7a13094/MarkupSafe-1.1.1.tar.gz
Building wheels for collected packages: MarkupSafe
  Running setup.py bdist_wheel for MarkupSafe ... done
  Stored in directory: /home/mpavlak/.cache/pip/wheels/f2/aa/04/0edf07alb8a5f5flaed7580fffb69ce8972edc16a505916a77
Successfully built MarkupSafe
Installing collected packages: Werkzeug, click, itsdangerous, MarkupSafe, Jinja2, Flask
Successfully installed Flask-1.1.2 Jinja2-2.11.2 MarkupSafe-1.1.1 Werkzeug-1.0.1 click-7.1.2 itsdangerous-1.1.0
bash-4.4$ █
```



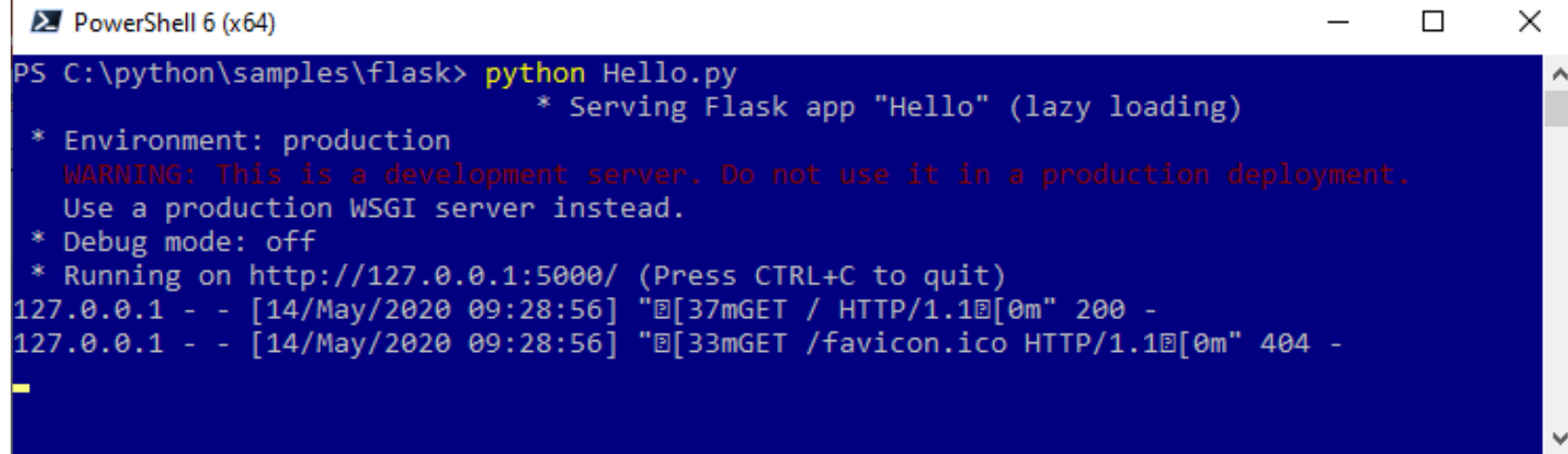
# Hello on Windows

---

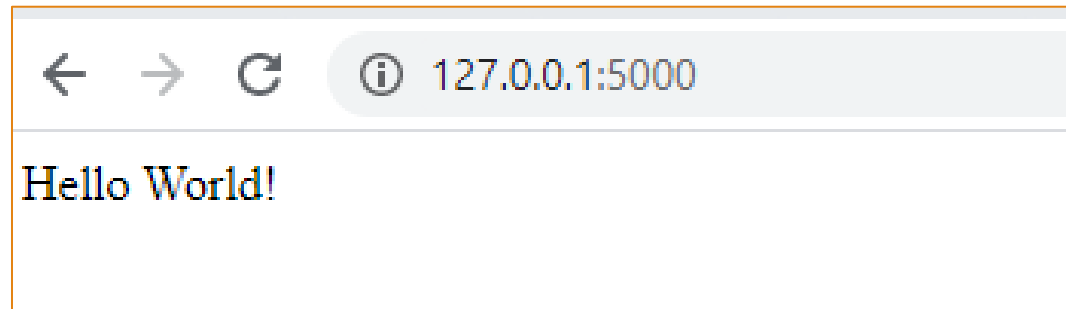
```
from flask import Flask
app = Flask(__name__)

@app.route("/")
def hello():
    return "Hello World!"

if __name__ == "__main__":
    app.run()
```



```
PowerShell 6 (x64)
PS C:\python\samples\flask> python Hello.py
* Serving Flask app "Hello" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [14/May/2020 09:28:56] "[37mGET / HTTP/1.1[0m" 200 -
127.0.0.1 - - [14/May/2020 09:28:56] "[33mGET /favicon.ico HTTP/1.1[0m" 404 -
```



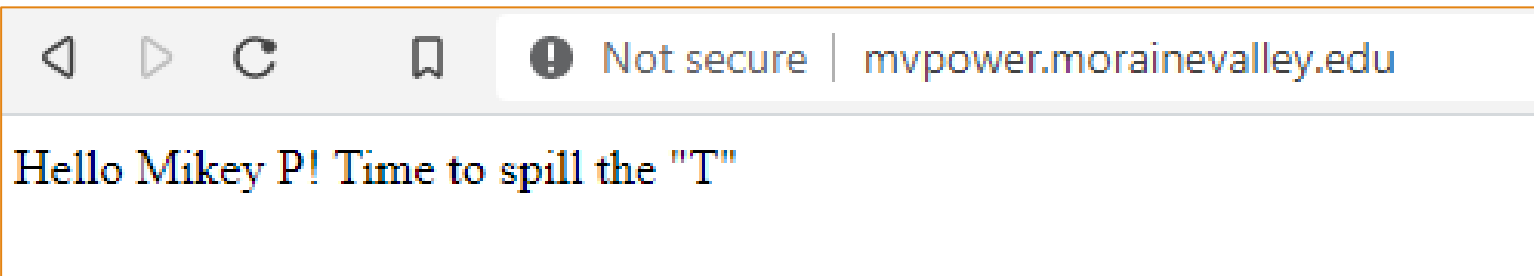
# Hello on IBM i

```
from flask import Flask
app = Flask(__name__)

@app.route("/")
def hello():
    return 'Hello Mikey P! Time to spill the "T"'

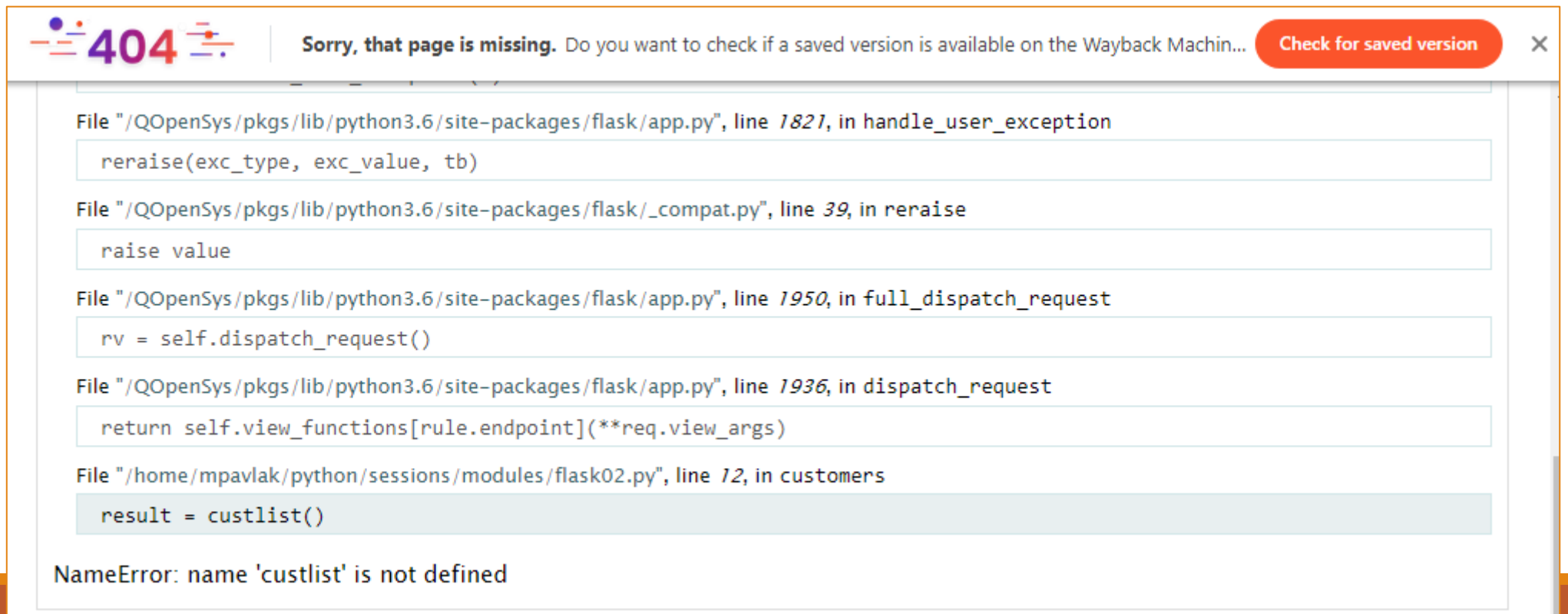
if __name__ == "__main__":
    app.run(host="10.1.1.131", port=int("80"))
```

```
mvpower.morainevalley.edu - PuTTY
bash-4.4$ python3 flask01.py
* Serving Flask app "flask01" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production
  environment.
  Use a production WSGI server instead.
* Debug mode: off
* Running on http://10.1.1.131:80/ (Press CTRL+C to quit)
76.221.92.140 - - [14/May/2020 09:55:40] "GET / HTTP/1.1" 200 -
```



# Consider debug...

```
app.run(host="10.1.1.131", port=int("80"), debug=True)
```



The screenshot shows a web browser window with a 404 error message: "Sorry, that page is missing. Do you want to check if a saved version is available on the Wayback Machin...". Below the message is a traceback for a NameError. The traceback shows the following stack of frames:

- File "/QOpenSys/pkgs/lib/python3.6/site-packages/flask/app.py", line 1821, in handle\_user\_exception: `reraise(exc_type, exc_value, tb)`
- File "/QOpenSys/pkgs/lib/python3.6/site-packages/flask/\_compat.py", line 39, in reraise: `raise value`
- File "/QOpenSys/pkgs/lib/python3.6/site-packages/flask/app.py", line 1950, in full\_dispatch\_request: `rv = self.dispatch_request()`
- File "/QOpenSys/pkgs/lib/python3.6/site-packages/flask/app.py", line 1936, in dispatch\_request: `return self.view_functions[rule.endpoint](**req.view_args)`
- File "/home/mpavlak/python/sessions/modules/flask02.py", line 12, in customers: `result = custlist()`

The final error message is: `NameError: name 'custlist' is not defined`

# Troubleshooting note...

---

Process runs in PASE

If you lose shell connection, you may need to go into PASE to “kill” the PID.

```
559539      - A      0:00 bsh bsh bsh
559540      - A      0:00 /QOpenSys/QIBM/ProdData/SC1/OpenSSH/libexec/sftp-serv
559755      - A      0:00 /QOpenSys/QIBM/ProdData/SC1/OpenSSH/sbin/sshd -R
559757 pts/0 A      0:00 -bsh
559758 pts/0 A      0:00 bash
559898 pts/0 A      0:00 python3 flask01.py
560414      - A      0:00 /QOpenSys/QIBM/ProdData/SC1/OpenSSH/sbin/sshd -R
560416 pts/1 A      0:00 -bsh
560417 pts/1 A      0:00 bash
560418      - A      0:00 /QOpenSys/QIBM/ProdData/SC1/OpenSSH/sbin/sshd -R
560419      - A      0:00 /QOpenSys/QIBM/ProdData/SC1/OpenSSH/sbin/sshd -R
560420      - A      0:00 /QOpenSys/QIBM/ProdData/SC1/OpenSSH/sbin/sshd -R
560421      - A      0:00 /QOpenSys/QIBM/ProdData/SC1/OpenSSH/sbin/sshd -R
560422 pts/1 A      0:00 ps ax
bash-4.4$ kill 559898
```

```
from flask import Flask
import pyodbcfunc as pof
```

```
app = Flask(__name__)
```

```
@app.route("/")
```

```
def hello():
```

```
    return 'Hello Mikey P! Time to spill the "T"
```

```
@app.route("/customers")
```

```
def customers():
```

```
    result = list(pof.custlist())
```

```
    response = '''<table border="2"><tr><th>Number</th><th>Company</th>
    <th>First Name</th><th>Last
    Name</th><th>Country</th></tr>\n'''
```

```
    for row in result:
```

```
        response += '<tr>'
```

```
        for col in row:
```

```
            data = str(col)
```

```
            response += '<td>' + data.strip() + '</td>'
```

```
        response += '</tr>\n'
```

```
    response += '</table>'
```

```
    return response
```

```
if __name__ == "__main__":
```

```
    app.run(host="10.1.1.131", port=int("80"), debug=True)
```

```
bash-4.4$ python3 flask02.py
```

```
* Serving Flask app "flask02" (lazy loading)
```

```
* Environment: production
```

```
WARNING: This is a development server. Do not use it in a production deployment.
```

```
Use a production WSGI server instead.
```

```
* Debug mode: on
```

```
* Running on http://10.1.1.131:80/ (Press CTRL+C to quit)
```

```
* Restarting with stat
```

```
* Debugger is active!
```

```
* Debugger PIN: 276-042-664
```



Number	Company	First Name	Last Name	Country
1221	Kauni Dive Shoppe	LINA	Norman	US
1231	Unisco	George	Weathers	Bahamas
1351	Sight Diver	Phyllis	Spooner	Cyprus
1354	Cayman Divers World Unlimited	Joe	Bailey	British West Indies
1356	Tom Sawyer Diving Centre	Chris	Thomas	US Virgin Islands
1380	Blue Jack Aqua Center	Ernest	Barratt	US
1384	VIP Divers Club	Russell	Christopher	US Virgin Islands
1510	Ocean Paradise	Paul	Gardner	US
1513	Fantastique Aquatica	Susan	Wong	Columbia
1551	Marmot Divers Club	Joyce	Marsh	Canada
1560	The Depth Charge	Sam	Witherspoon	US
1563	Blue Sports	Theresa	Kunec	US
1624	Makai SCUBA Club	Donna	Siaus	US
1645	Action Club	Michael	Spurling	US
1651	Jamaica SCUBA Centre	Barbara	Harvey	West Indies
1680	Island Finders	Desmond	Ortega	US
1984	Adventure Undersea	Gloria	Gonzales	Belize
2118	Blue Sports Club	Harry	Bathbone	US
2135	Frank's Divers Supply	Lloyd	Fellows	US
2156	Davy Jones' Locker	Tanya	Wagner	Canada
2163	SCUBA Heaven	Robert	Micheliné	Bahamas
2165	Shangri-La Sports Center	Frank	Paniagua	Bahamas
2315	Divers of Corfu, Inc.	Charles	Lopez	Greece
2354	Kirk Enterprises	Rudolph	Claus	US

# Some formidable candidates

---

Django



Pyramid



TurboGears



Dash

Microframeworks

- Flask
- Bottle
- CherryPy

Asynchronous Frameworks

- Sanic
- Tornado



# Where to get more info and help?

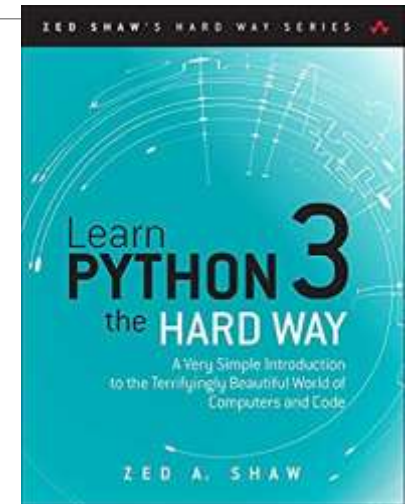
Club Seiden

Learn Python 3 The Hard Way

SoloLearn: Learn Python (Android & Apple App)

<https://ibmioss.ryver.com> (send me a note for invite...)

Pythonweekly.com weekly newsletter



	Up and Running	Pyth-online	Personal Trainer	Djump Into Django
	Installation and Concepts (Online—Live)	Installation and Basic Web App (Online—Live)	In-Depth with Hands-on Help (Onsite 3 Days)	Django Framework for Professional Apps (Onsite 5 Days)

**Learn Python**  
SoloLearn Education ★★★★★ 79,502  
Everyone  
This app is compatible with all of your devices.  
Installed

# Python Modules

---

MISC





# TQDM

Progress bar

## Project description



tqdm

python 2.7 | 3.5 | 3.6 | 3.7 | 3.8 | 3.9 | pypi v4.60.0 | conda-forge v4.60.0 | docker pull | snap install

build failing | coverage 90% | codecov 90% | code quality A | sourcerank 26 | pypi downloads 23M/month

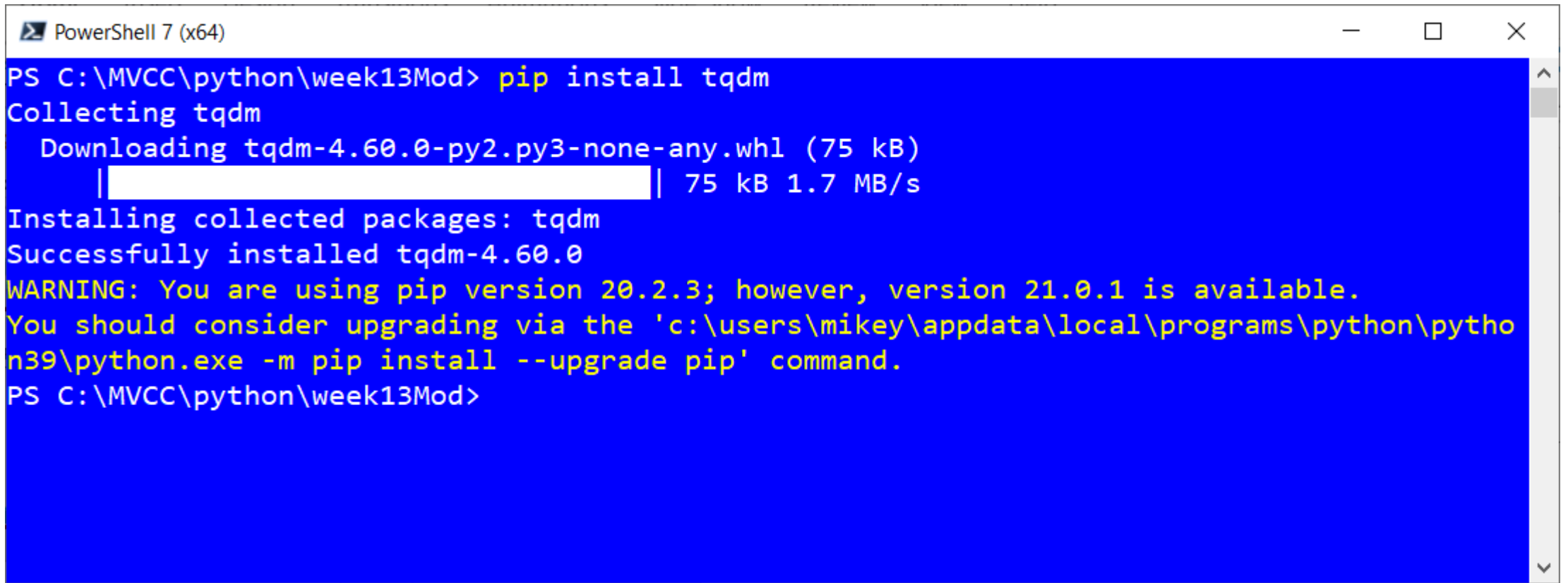
license MPLv2.0, MIT Licences | Open Hub tqdm | launch binder | mentioned in awesome

`tqdm` derives from the Arabic word *taqaddum* (تَقَدَّمَ) which can mean “progress,” and is an abbreviation for “I love you so much” in Spanish (*te quiero demasiado*).

Instantly make your loops show a smart progress meter - just wrap any iterable with `tqdm(iterable)`, and you’re done!

# Progress bar

---



```
PowerShell 7 (x64)
PS C:\MVCC\python\week13Mod> pip install tqdm
Collecting tqdm
  Downloading tqdm-4.60.0-py2.py3-none-any.whl (75 kB)
    |████████████████████████████████████████| 75 kB 1.7 MB/s
Installing collected packages: tqdm
Successfully installed tqdm-4.60.0
WARNING: You are using pip version 20.2.3; however, version 21.0.1 is available.
You should consider upgrading via the 'c:\users\mikey\appdata\local\programs\python\python39\python.exe -m pip install --upgrade pip' command.
PS C:\MVCC\python\week13Mod>
```



# Needed to load data

---

```
for sequence in range(1,100):
    count = 0
    print("\n")
    print(datetime.datetime.now())
    for claim in range(1826, 1930):
        urlnew = url + str(claim)
        res = requests.get(urlnew)
        count +=1

    print("\nnumber of claims processed for claimcost is:",count)
    print(datetime.datetime.now())

    count = 0
    for claim in range(1826, 1930):
        urlnew = url2 + str(claim)
        res = requests.get(urlnew)
        count +=1

    print("\nnumber of claims processed for claims is:",count)
    print(datetime.datetime.now())
```

# Site connectivity checker

---

```
import urllib.request

def connect(host) :
    try:
        urllib.request.urlopen(host)
        return True
    except:
        return False

print()
host = 'http://google.com'
print(('Found ' + host) if connect(host) else ('could not find' + host))

print()
host = 'http://googleapilove.com'
print(('Found ' + host) if connect(host) else ('could not find' + host))

print()
host = 'http://ibm.com'
print(('Found ' + host) if connect(host) else ('could not find' + host))
```

# Summary

---

Python is the CL of Open Source

Useful beyond infrastructure

Virtually ANY root file system task

Supported and strategic

Full data access

Framework add productivity

# Python Modules for Profitability

---

*Questions?*



[MikePavlak@gmail.com](mailto:MikePavlak@gmail.com)



@MikeyPEI

