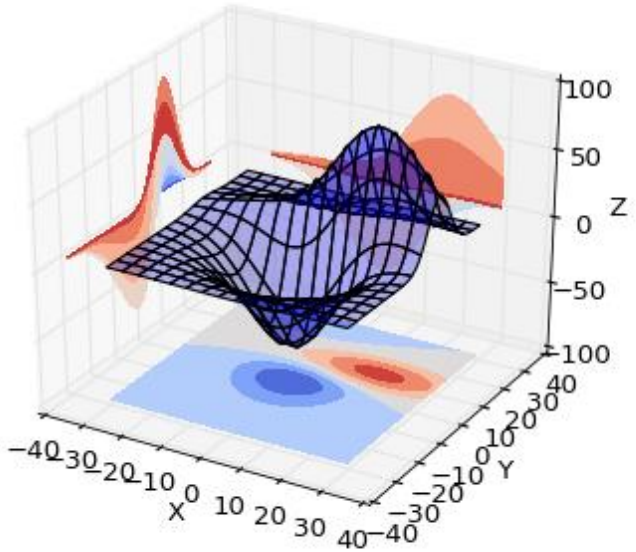
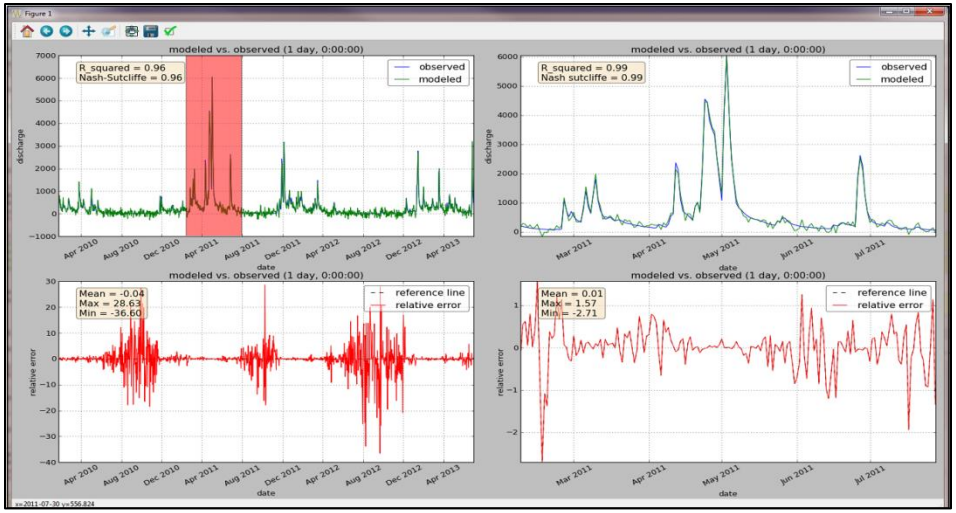
# Introduction to Scientific Computing
# Meeting 13
# Programming with Python




```
# Write Fibonacci series up to n
>>> def fib(n):
>>>     a, b = 0, 1
>>>     while a < n:
>>>         print(a, end=' ')
>>>         a, b = b, a+b
>>>     print()
>>> fib(1000)
0 1 1 2 3 5 8 13 21 34 55 89 144 233 377 610
```


Jeremiah Lant, Hydrologist
USGS Kentucky Water Science Center
jlant@usgs.gov

# General Overview of Topics

1. **Unix shell** – with **Git Bash** for learning basic commands/language to communicate with a computer; basis for learning how to program.
2. **Version control** – with **Git**, **GitHub**, **Bitbucket** for learning how to track changes, reproduce, and share work efficiently.
3. **Programming** – with **Python** for learning how to program, and to write programs in a modular and testable way.
   - scientific libraries; **numpy** and **matplotlib** for learning how to work with arrays and matrices and how to create plots.

# Meeting Structure

- Briefly review material from previous meeting.
- Discuss objective(s) for the meeting.
  - ~ 1 - 2 minutes
- Give introduction to topic
  - ~ 5 – 10 minutes
- Watch a related brief video that covers the topic.
  - ~ 10 minutes
- Work through hands-on examples together.
  - ~ 20 – 30 minutes
- Discuss topic covered and answer any outstanding questions.
  - ~ 5 minutes

# Last meeting

- Learned about version control with Git
- Learned how to create and use remote repositories.
- Learned about how to start collaborating with others using the same repository

# Today's Objectives

- Learn what Python is and why it is used
- Learn some basics of Python
  - How to use Python interactively vs. writing a script
  - Creating variables through assignment; **=**
  - Basic data types; **integers, floats, booleans, strings**
  - Simple arithmetic operations; **+, -, %, *, ****
  - Comparisons; **True or False**

# A Few Recommended Resources

- Software Carpentry, Greg Wilson
  - http://software-carpentry.org/
- Learn Python the Hard Way, Zed Shaw
  - http://learnpythonthehardway.org/
- Pro Git, Scott Chacon
  - http://git-scm.com/book

# Video – Python Basics


- Software Carpentry, Greg Wilson
  - Python: Basics

    http://software-carpentry.org/v4/python/basics.html

# Try out Python

- Use python interactively as a calculator

- Create “Hello World” program in python

# Review – page 1

1. What are variables?

2. What is the difference between “=“ and “==“ operators?

3. What is the output of the following program?

```
>>> x = 2
>>> y = x
>>> x = 3
>>> print(“x is”), x
>>> print(“y is”), y
```

# Review – page 2

1. Is is a **int, float, boolean, or string**
    1. 2.5
    2. 10
    3. True
    4. “Hello World”
    5. False
    6. ‘discharge’

2. Is there a function to check the type of a value? Is so, what is it?

3. What is the difference between “*” and “**”

# Review – page 3

```
>>> x = “2”
>>> y = 3
>>> x * y
```

1. What is the output of x * y?
2. What is the output of x + y?

3. What is the output for each of the following
    1. 5 == 5
    2. 5 > 6
    3. 3 < = 3
    4. a = “Area”
       a == ‘area’
    5. 8 != 8
    6. 3 **2 == 9

# Next meeting

- Python - controlling the flow of a program using **while loop** and **if, elif,** and **else**