Assignments and Loops

Rose-Hulman Institute of Technology

Computer Science and Software Engineering

ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

Outline

- Basic number types: **int** and **float**
- Variables and assignments
- Definite loops
- Math library
- Accumulator problem

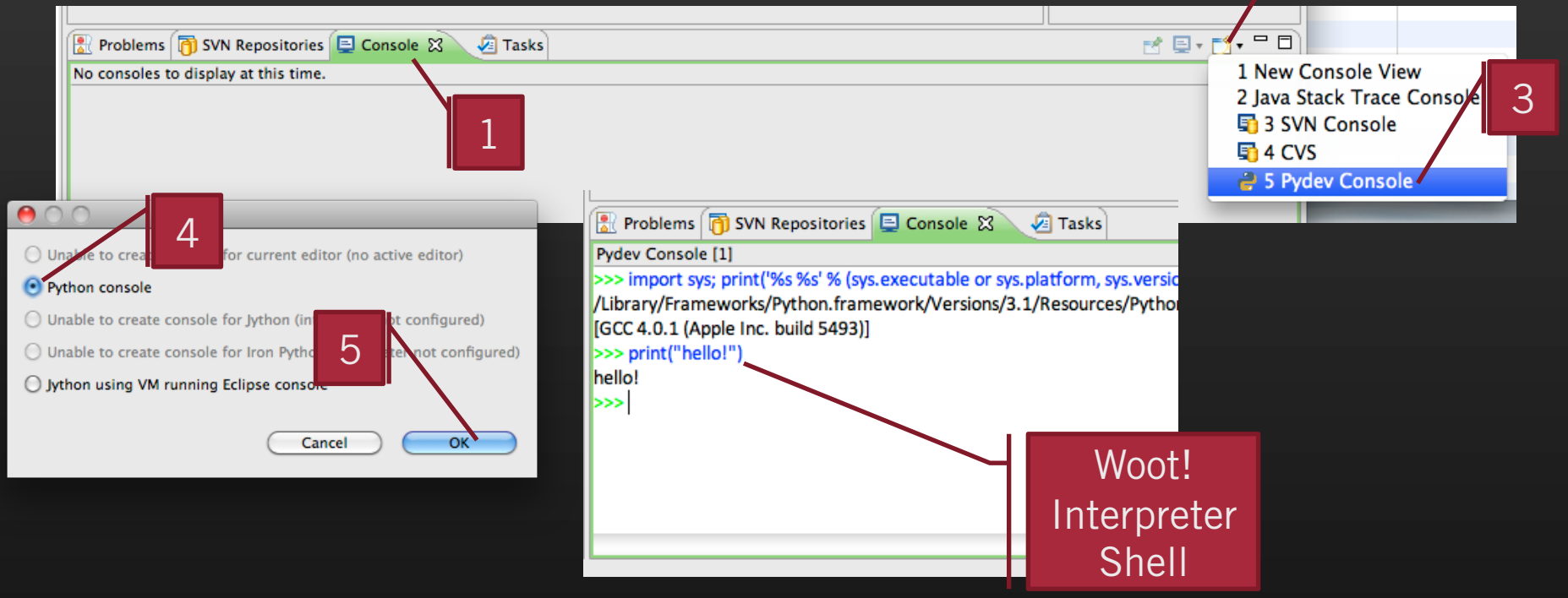
ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

Check out 03-AssignmentsAndLoops

- Go to SVN Repository view at bottom of the workbench
  - Missing? Add it back:
    Window→Show View→Other→SVN → SVN Repositories
- Browse SVN Repository view for **03-AssignmentsAndLoops**
- Right-click it and choose **Checkout**
- Accept defaults in the dialog
- Expand the **03-AssignmentsAndLoops** project that appears in **Package Explorer** (on the left-hand-side)

ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

PyDev Interpreter Console



ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

Some numeric operations

Operator	Operation
+	Addition
-	Subtraction
*	Multiplication
/	Division
**	Exponentiation
%	Remainder
//	Integer division

Function	Operation
abs(x)	Absolute value of x
round(x, y)	Round x to y decimal places
int(x)	Convert x to the int data type
float(x)	Convert x to the float data type

ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

Variables

- Identifiers that refer to values stored in memory
- Values can be of any type

```
width = 4
temperature = 98.6

dogName = "fido"
lost = [4, 8, 15, 16, 23, 42]

triangleArea = width * height / 2
xyPoint = (r * cos(theta), r * sin(theta))
```

ROSE-HULMAN INSTITUTE OF TECHNOLOGY

Variables and Assignment

- Assignment gives a variable a value

  x = 6 * 7

  – Python evaluates right-hand side (**42**)

  – Then variable on left "gets" the value

- "Gets" not "Equals"

  – **x = 3.9 * x * (1 – x)**

ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

How to Think About Variables

- Variables as sticky notes
- Example on board…

```
x = 10
x = x + 1
```

ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

Three Kinds of Assignment

- Simple
- Compound
- Multiple (or simultaneous)

ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

Simple Assignment

- **<variable> = <expr>**

- Note:
  - **input(<string>)** is an expression
  - input statements are just assignment statements

Q1,2

ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

Compound Assignment

- **<var> <op=> <expr>** means
  <var> = <var> <op> <expr>
  - where **<op=>** is **+=**, **-=**, ***=**, **/=**, **//=**, or **%=**
- Example:
  - **total += 5** is the same as
    total = total + 5

ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

Simultaneous Assignment

- **<var>, <var>, ... = <expr>, <expr>, ...**
- Example:
  - **sum, diff = x + y, x - y**

Q3

Assignment Assignment

- See **assignmentsAndLoops.py** module
- Do the TODOs inside the **assignmentStatements()** function

ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

Summary: Assignment Statements

- Simple assignments: **<variable> = <expr>**
- Compound assignments
  - **<var> <op=> <expr>** means
    <var> = <var> <op> <expr>
    where **<op=>** is **+=**, **-=**, ***=**, **/=**, **//=**, or **%=**
- Simultaneous (multiple) assignments
  - **<var>, <var>, … = <expr>, <expr>, …**

ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

Sequence

- A list of things
- For example:
  - **[2, 3, 5, 7]**
  - **[“My”, “dog”, “has”, “fleas”]**
- Every for loop uses a list

ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

Definite Loops

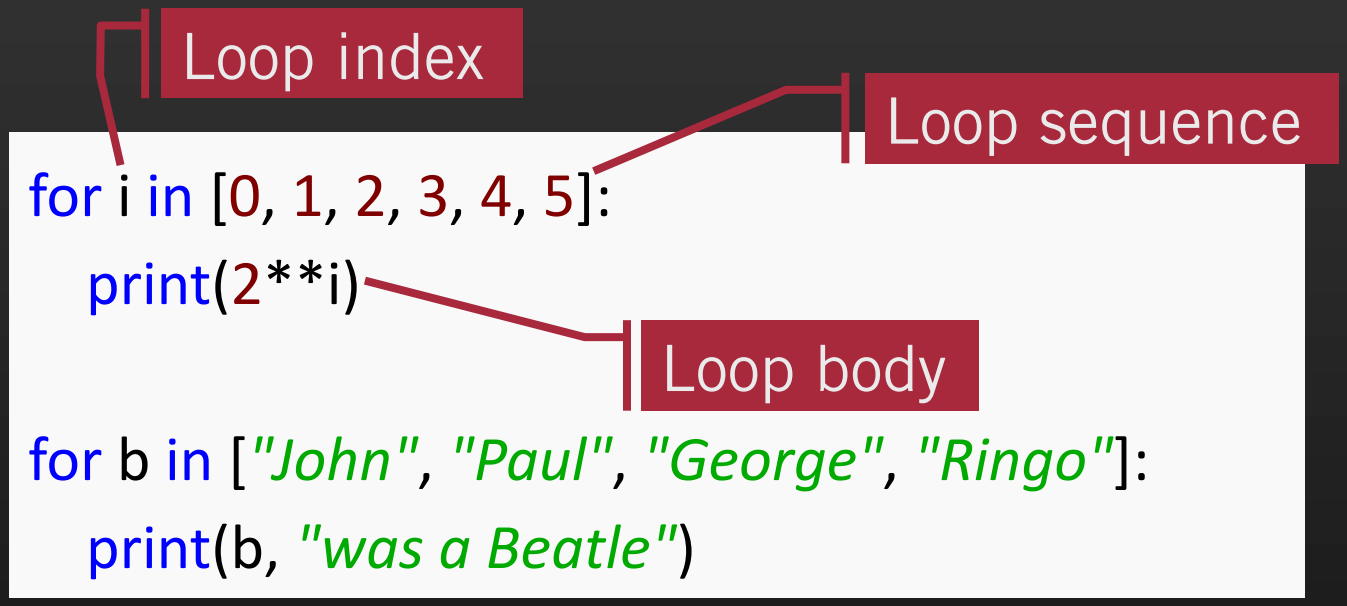
- *Loop*: a control structure for executing a portion of a program multiple times
- *Definite loop*: Python knows beforehand how many times to repeat the body of the loop
- Syntax:

  for <var> in <sequence> :
  <body>

- Semantics: Executes **<body>** once for every element of **<sequence>**, with **<var>** set to that element.

ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

Some Definite Loops



ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

The range Function

- Creates a list that is an *arithmetic sequence*
- General formats for range function:
  - **range(<expr>)**
  - **range(<expr>, <expr>)**
  - **range(<expr>, <expr>, <expr>)**

- Consider:

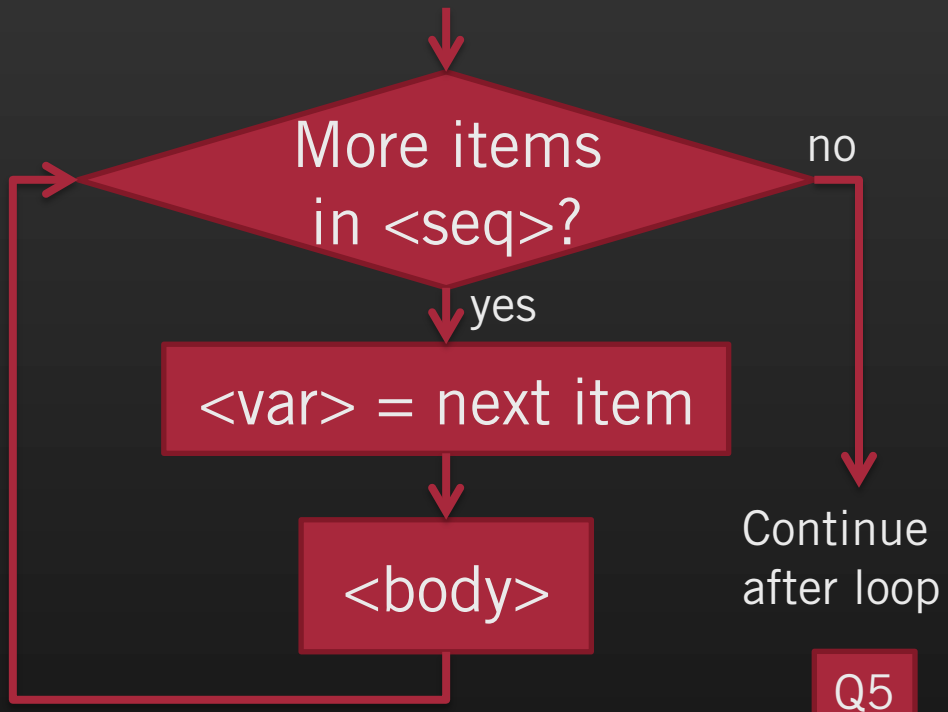
```
list(range(8))
list(range(1, 7))
list(range(3, 18, 2))
list(range(4,10,-1))
list(range(17, -5, -3))
```

Q4

Accumulator Loop

- Accumulator combines parts of a list
- Common technique!
- Consider:

```
a = 0
for j in [1, 2, 3, 4]:
    a = a + j
    print(a)
```



Q5

ROSE-HULMAN INSTITUTE OF TECHNOLOGY

Another loop with an accumulator

- Find the sum of the positive odd numbers that are ≤ 13
- Do it together as a class, in function **sumOddPositiveLessThan() in**

ROSE-HULMAN
INSTITUTE OF TECHNOLOGY

More math library components

Python	Mathematics	English
pi	π	Approximation of pi
e	e	Approximation of e
sin(x)	sin x	The sine of x
cos(x)	cos x	The cosine of x
tan(x)	tan x	The tangent of x
atan2(y, x)	$\tan^{-1}$ y/x	Arc tangent of angle of line from (0,0) to (x, y)
log(x)	ln x	The natural (base e) log of x
log10(x)	$\log_{10}$x	The base 10 log of x
exp(x)	e^x	The exponential of x

Math library functions

- Quadratic formula to find real roots for quadratic equations of the form $ax^2 + bx + c = 0$
- Solution:

$$x = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

$$x = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

- Write out the Python expression for the first formula.
- If time permits, test it in Eclipse

Q6

ROSE-HULMAN INSTITUTE OF TECHNOLOGY

Work Time

HW2 due Mon., HW3 due Tues.

ROSE-HULMAN
INSTITUTE OF TECHNOLOGY