Week 4

Strings, `if`/`else`, `return`, user input

Strings

index	0	1	2	3	4	5	6	7
or	-8	-7	-6	-5	-4	-3	-2	-1
character	P	.		D	i	d	d	y

- Accessing character(s):
  variable [**index**]
  variable [**index1** : **index2**]

  - index2 exclusive
  - index1 or index2 can be omitted (end of string)

```
>>> name = "P. Diddy"
>>> name[0]
'P'
>>> name[7]
'y'
>>> name[-1]
'y'
>>> name[3:6]
'Did'
>>> name[3:]
'Diddy'
>>> name[:-2]
'P. Did'
```

python™

String Methods

Java	Python
length	len(**str**)
startsWith, endsWith	startswith, endswith
toLowerCase, toUpperCase	upper, lower, isupper, islower, capitalize, swapcase
indexOf	find
trim	strip

```
>>> name = "Martin Douglas Stepp"
>>> name.upper()
'MARTIN DOUGLAS STEPP'
>>> name.lower().startswith("martin")
True
>>> len(name)
20
```

python™

for Loops and Strings

- A `for` loop can examine each character in a string in order.

  for **name** in **string**:
      statements

```
>>> for c in "booyah":
...     print c
...
b
o
o
y
a
h
```

raw_input

`raw_input` : Reads a string from the user's keyboard.

 – reads and returns an entire line of input

```
>>> name = raw_input("Howdy. What's yer name?")
Howdy. What's yer name? Paris Hilton

>>> name
'Paris Hilton'
```

python™

raw_input

- `raw_input` can be used to read a number from the user's keyboard by casting the result as an int
  - Only numbers can be cast as ints!
  - Example:

    ```
    age = int(raw_input("How old are you? "))
    print "Your age is", age
    print "You have", 65 - age, "years until retirement"
    ```

    Output:

    ```
    How old are you? 53
    Your age is 53
    You have 12 years until retirement
    ```

python™

Exercise

- Write a program that reads two employees' hours and displays each's total and the overall total.
  - Cap each day at 8 hours.

```
Employee 1: How many days? 3
Hours? 6
Hours? 12
Hours? 5
Employee 1's total hours = 19 (6.33 / day)

Employee 2: How many days? 2
Hours? 11
Hours? 6
Employee 2's total hours = 14 (7.00 / day)

Total hours for both = 33
```

Formatting Text

"format string" % (**parameter**, **parameter**, **...**)

- *Placeholders* insert <u>formatted values</u> into a string:
  - `%d`   an integer
  - `%f`   a real number
  - `%s`   a string

  - `%8d`   an integer, 8 characters wide, right-aligned
  - `%08d`   an integer, 8 characters wide, padding with 0s
  - `%-8d`   an integer, 8 characters wide, left-aligned
  - `%12f`   a real number, 12 characters wide
  - `%.4f`   a real number, 4 characters after decimal
  - `%6.2f`   a real number, 6 total characters wide, 2 after decimal

```
>>> x = 3; y = 3.14159; z = "hello"
>>> text = "%-8s, %04d is close to %.3f" % (z, x, y)
>>> print text
hello   , 0003 is close to 3.142
```

python™

if

`if` **condition**:
    statements

  – Example:

```
gpa = int(raw_input("What is your GPA? "))
if gpa > 2.0:
    print "Your application is accepted."
```

if/else

```
if condition:
    statements
elif condition:
    statements
else:
    statements
```

- Example:
```
gpa = int(raw_input("What is your GPA? "))
if gpa > 3.5:
    print "You have qualified for the honor roll."
elif gpa > 2.0:
    print "Welcome to Mars University!"
else:
    print "Your application is denied."
```

python™

Logical Operators

Operator	Meaning	Example	Result
==	equals	`1 + 1 == 2`	True
!=	does not equal	`3.2 != 2.5`	True
<	less than	`10 < 5`	False
>	greater than	`10 > 5`	True
<=	less than or equal to	`126 <= 100`	False
>=	greater than or equal to	`5.0 >= 5.0`	True

Operator	Example	Result
and	`(2 == 3) and (-1 < 5)`	False
or	`(2 == 3) or  (-1 < 5)`	True
not	`not (2 == 3)`	True

python™

Exercise

- Write a program that judges a couplet by giving it one point if it
  - is composed of two verses with lengths within 4 chars of each other,
  - *"rhymes"* (the two verses end with the same last two letters),
  - *alliterates* (the two verses begin with the same letter).
- A couplet which gets 2 or more points is "good"

```
Example logs of execution:
(run #1)
First verse: I joined the CS party
Second verse: Like "LN" and Marty
2 points: Keep it up, lyrical genius!

(run #2)
First verse: And it's still about the Benjamins
Second verse: Big faced hundreds and whatever other
synonyms
0 points: Aw, come on.  You can do better...
```