

# CS150 - File I/O

Note Title

3/5/2012

## Announcements

- HW due Thursday
- Next HW due Wed. up Wednesday after break

## Ch.8 : Input + Output

So far, we've seen standard I/O.

raw\_input(). function:

- Takes input up to first 'enter' from keyboard (not including newline)

- Returns a string

- Optional input - string for prompting

## Print Statement

- Accepts 0 or more inputs, separated by commas.
- Not really a function - no ( ) around inputs
- Automatically casts every input to str( ).
- Each comma is a space, & end is a newline

Can avoid the newline : print 'I like',  
print 'cookies'

→ I like cookies

## Formatted strings

Want to print

'Cardinals: ranked  $\frac{1}{\text{rank}}$  of  $\frac{30}{\text{total}}$ '

where we have variables.

How?

[team='Cardinals'  
rank=1  
total=30]

print team + ': ranked ' + str(rank) + ' of ' + str(total)

print team + ': ranked', rank, 'of', total

Easier way: %os , %od → place holders  
for later variables

Ex:

print '%os: ranked %od of %od teams' % (team,  
rank, total)

The diagram illustrates the mapping of printf format specifiers to their corresponding arguments. A red bracket labeled "string" points to the "%os" specifier, which is part of the string "ranked %od of %od teams". Another red bracket labeled "numbers" points to the "%od" specifier, which is part of the same string. A third red bracket labeled "(team, rank, total)" points to the final "%od" specifier, which is part of the tuple "(team, rank, total)".

Fills in variables from end tuple to  
given spots, assuming they match  
the correct type.)

## More formatting

Can even use this to specify precision:

```
print 'You owe $%.2f, cash.' % 350
```

```
print '%4d' % 12
```

```
print '%04d' % 12
```

```
print '%-4d' % 12
```

## More examples

\%7.3f\n + \%7.3f' \% (80.0/3.0, 3.0/4.0)

Note:

Not dependent on print, also:

message = ' %s: ranked %d of %d teams' %  
(team, rank, total)

## Practice

Set variables named `person`, `day`, +  
`month` to be your information.

Use string formatting to produce the  
format:

Erin's birthday is March 27.  
"person"                                    "month"                                    "day (an int)

Pt 2: Redo the previous to work on 3  
lists storing names, days, + months,  
+ produce output for each triple.