Collections in Java

- Arrays are collections
  - Arrays are objects
  - Arrays are sequences of elements in base type
  - These elements are collected together in one object: the array
  - Java includes many other collection mechanisms
  - Arrays good for some applications (fixed-length sequences), not others (varying-length sequences)
  - Other collections tuned for different purposes
  - General observation holds, however:
    - Collections are objects...
    - ...that contain other objects in a given type
- We'll study two (more in CMSC132): Stack, ArrayList

Stacks in Java

- Recall: a stack is a data structure ("device" for holding values) — FILO (First In, Last Out)
- Typical operations on a stack
  - push: add a new value into the stack
  - pop: remove the most recently added value still in stack
  - top: return the most recently added value in stack
  - isEmpty: returns true if the stack is currently empty or false otherwise
Example of stack concept (not Java specific)

- Stack `s`
  - `s.isEmpty() == ??` true
  - `s.push(3);` `s.isEmpty() == ??` false
  - `s.push(4);` `s.peek == ??` 4
  - `s.pop();` `s.push(5);` `s.peek == ??` 5

Stacks in Java (cont.)

- Java includes a generic class for stack objects
  - Stack objects contain other objects
  - All objects in stack must have same type
  - Only objects may be stored in stacks (no primitive-type values)
- Syntax: `Stack<T>`
  - `Stack<T>` is a generic class
  - `T` is a class variable representing the base type
  - Replace `T` by a specific type to get a stack of that type of elements
  - Class is in `java.util` package
- Documentation: [http://java.sun.com/j2se/1.5.0/docs/api/java/util/Stack.html](http://java.sun.com/j2se/1.5.0/docs/api/java/util/Stack.html)
- See example: `StackExample.java`
  
  ```java
  Stack<String> stack = new Stack<String>();
  ```
  Creates a stack of strings
  extend this to be stack of cats
  extend this to be stack of integer values