

1. (a) A parameter is a variable declared in the header of a method (within the parentheses). An argument is an expression that occurs in a call of a method.

1. It's alright if you left out the (*but not ...*) parts.

(b) 5. Line 3. Scope 4..6.

8. Line 2. Scope 2..18 but not 3..6, 9..12, 13..16.

10. Line 9. Scope 9..12..

15. Line 13. Scope 13..16.

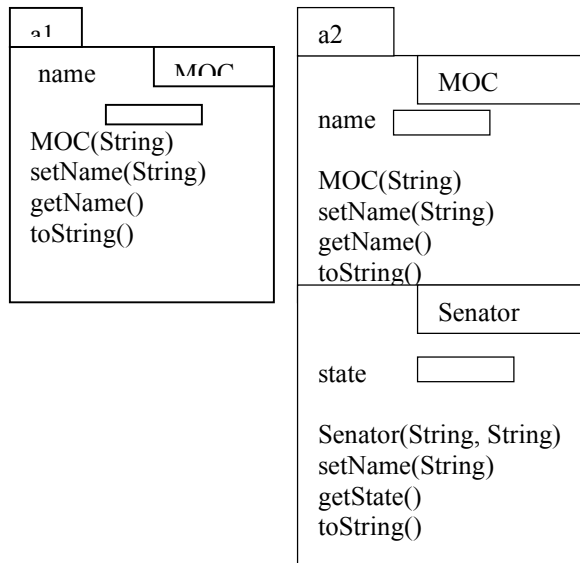
1. (c) 1. ILLEGAL. Cast **double** to **int** not automatic.

2. d contains 3.0 3. b contains **true**

4. e contains 36.0 5. s contains "usa"

6. ILLEGAL. Can't divide a String by an **int**.

2. We abbreviate "MemberOfCongress" by "MOC". We put the types of pars within parentheses in showing a method; you didn't have to do this.



3. (a) Senator is the subclass.

(b) toString() and getName are overridden.

(c) 1: 2.

2: "Member of Congress John McCain"

3: "Senator Hillary Clinton of NY"

4: ERROR. Field name is private.

5: "Senator Sen. Hillary R. Clinton"

(d) 1: create a new instance of class <class name>. 2: execute the constructor call. 3: yield as the value of the new expression the name of the new instance.

6: 0

4a. /** An instance represents a congressional rep

* -- a member of the House. */

public class Representative extends

MemberOfCongress {

/* District of the representative. Has form "nnxx"
or "nxx" where n is a digit and xx is the state.*/

private String district;

/* Affiliation: "D" or "R" or "I" */

private String party;

/** Constructor: a rep named n in district d and
with party p. Precondition: d and p must conform
to the rules on fields district and party. */

public Representative(String n, String d, String p) {
 super(n);
 district= d;
 party= p;
}

/** = the district */

public String getDistrict() { return district; }

/** = the party of the representative */

public String getParty() {
 if (party.equals("D"))
 return "Democrat";
 if (party.equals("R"))
 return "Republican";
 return "independent";
}

/** = a representation of this representative */

public String toString() {
 return "Representative " + getName() +
 " (" + party + ") of " + district;
}

4 (b).

assertEquals("Representative Pelosi (Democrat) of
8CA", a.toString());

assertEquals("Representative Who-2 (Republican) of
10NY", b.toString());

assertEquals("Representative Who-3 (independent) of
6CA", c.toString());

5. /** = the district, as specified on Prelim 1 */

public String formatDistrict() {
 String digits; // The digit(s) of the district
 String state; // The state
 if (district.length() == 3) {
 digits= district.substring(0, 1);
 state= district.substring(1, 3);
 } else {
 digits= district.substring(0, 2);
 state= district.substring(2, 4);
 }

if (digits.endsWith("1") && !digits.equals("11"))
 return digits + "st district of " + state;

if (digits.endsWith("2") && !digits.equals("12"))
 return digits + "nd district of " + state;

if (digits.endsWith("3") && !digits.equals("13"))
 return digits + "rd district of " + state;

return digits + "th district of " + state;

}