

Computer Sciences 302
Week 11 : Shadowed Fields in Java – Lecture Handout –
Instructor: Deb Deppeler

Fill in these fields (left to right) on the red scantron form (use #2 pencil):

1. Full Name: _____
2. NetID and Email: _____

Introduction

Fields become shadowed when a constructor or method contains a declaration of a parameter or local variable with the same name as the field.

Use the reserved word **this** to reference the field instead of the parameter or local variable.

1. Which choice completes this methods so that it will display the smaller of two values? Either value is displayed if they are same.

```
public void showSmaller( int a, int b ) {  
    if ( CODE )  
        System.out.println( a );  
    else  
        System.out.println( b );  
}
```

- A. CODE1: a < b
- B. CODE1: a > b
- C. CODE1: a == b
- D. CODE1: a != b
- E. a.equals(b)

2. What is output by the code fragment given the class definition of `Shadow`?

```
// Code Fragment
Shadow shadow = new Shadow();
shadow.Shadow();
System.out.println( shadow );
```

```
public class Shadow {

    private int shadow = 1;
    private final int N;
    private int m;

    public Shadow() {
        N = shadow;
    }

    public void Shadow() {
        int shadow = 2;
        shadow = shadow;
    }

    public String toString() {
        return N + ": shadow = " + shadow;
    }

}
```

3. Draw the Object (memory) diagram of the memory allocation during the program in question 2's execution.

4. What is output by the code fragment given the class definition of `Shadow`?

```
// Code Fragment
Shadow shadow = new Shadow(3);
System.out.println( shadow );
```

```
public class Shadow {

    private int shadow = 1;

    public Shadow(int shadow) {
        shadow = shadow;
    }

    public String toString() {
        return "shadow = " + shadow;
    }

}
```

5. What is output by the code fragment given the class definition of `Shadow`?

```
// Code Fragment
Shadow shadow = new Shadow(3);
shadow.Shadow();
System.out.println( shadow );
```

```
public class Shadow {
    private int shadow = 1;

    public Shadow(int shadow) {
        this.shadow = shadow;
    }
    public void Shadow() {
        int shadow = 2;
        this.shadow = shadow;
    }
    public String toString() {
        return "shadow = " + shadow;
    }
}
```

6. What is output by the code fragment given the class definition of `Shadow`?

```
// Code Fragment
Shadow shadow = new Shadow();
System.out.println( shadow );
```

```
public class Shadow {
    private int shadow = 1;
    private final int N;
    public Shadow( int shadow ) {
        N = this.shadow; this.shadow =
        shadow;
    }

    public Shadow() {
        this(5);
    }

    public String toString() {
        return N + ": shadow = " + shadow;
    }
}
```

7. What is output by the code fragment given the class definition of `Shadow`?

```
// Code Fragment
Object [][] wa; int n = 0;
wa = new Widget[10] [];
for ( int i=0; i < wa.length; i++ ) {
    wa[i] = new Widget[i];
    for ( int j=0; j < wa[i].length; j++ )
        wa[i][j]=new Widget("W_"+i+"_"+j,n);
}
System.out.println( wa.getPassCode() );
System.out.println( wa.toString() );
System.out.println( wa[4] );
System.out.println( wa[13].charAt(8) );
System.out.println( wa[5].getIndex() );
```

```
import java.util.Arrays;
public class Widget {
    public final String name;
    public final long id;
    private char [] passcode;
    public Widget( String s, long id ) {
        name = s; this.id = id; passcode = new char[4];
        for ( int i=0; i < passcode.length; i++ )
            passcode[i] = (char)('A'+(int)(Math.random()*26));
    }
    public String getName() { return name; }
    public char[] getPassCode() {
        return passcode;
    }
    public String getPassCode() {
        return Arrays.toString(passcode);
    }
}
```

8. Draw the Object (memory) diagram of the memory allocation during the program in question 2's execution.

9. Given the class Diamond:

```
public class Diamond {
    public final String name;
    private final double carat = 0.5;
    public static String cut;
    private double price; // price per carat weight 1.0
    public double getPrice() {
        return price / carat;
    }
    public void setPrice( double p ) {
        if ( p > price ) price = p;
    }
    public Diamond() {
        this("WHITE",1.0, "Princess", 1000);
    }
    public Diamond(String n, double carat, String cut, double p) {
        name = n;
        carat = carat;
        this.cut = cut;
        this.price = p;
    }
    public String toString() {
        return name + ": $" + getPrice() + " " + carat + " carats" +
            " cut: " + cut;
    }
}
```

What is the result of executing the GraderMain program?

```
public class GraderMain {
    public static void main( String [] args ) {
        Diamond diamond1 = new Diamond();
        diamond1.cut = "Princess";
        diamond1.setPrice( 500 );
        Diamond diamond2 = new Diamond("ORANGE",2.0,"Oval Brilliant",2000);
        System.out.println( diamond1 + "\n" + diamond2 );
    }
}
```

- A. WHITE: \$2000.0 1.0 carats cut: Princess
ORANGE: \$4000.0 0.5 carats cut: Oval Brilliant
- B. WHITE: \$1000.0 1.0 carats cut: Princess
ORANGE: \$4000.0 0.5 carats cut: Oval Brilliant
- C. WHITE: \$2000.0 0.5 carats cut: Oval Brilliant
ORANGE: \$4000.0 0.5 carats cut: Oval Brilliant
- D. WHITE: \$1000.0 1.0 carats cut: Princess
ORANGE: \$4000.0 2.0 carats cut: Oval Brilliant

10. Which **CODE1** and **CODE2** replacements will make this code fragment produce the output shown?

```
for ( int i = 20 ; i >= 0 ; CODE1 ) {
    System.out.print( i + " : " );
    for ( CODE2 ) {
        System.out.print( j + " " );
    }
    System.out.println();
}
```

Console output:

```
20: 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
10: 10 9 8 7 6 5 4 3 2 1
0:
```

- A. **CODE1**: `i = i - 10` **CODE2**: `int j = i ; j > 0 ; j--`
- B. **CODE1**: `i = i - 9` **CODE2**: `int j = i ; j < i ; j++`
- C. **CODE1**: `i = i - 10` **CODE2**: `int j = 0 ; j < i ; j--`
- D. **CODE1**: `i = i - 9` **CODE2**: `int j = 0 ; j <= i ; j++`
- E. **CODE1**: `i = i - 10` **CODE2**: `int j = i - 1 ; j > 0 ; j--`
11. Which statement in the main method of one class will call this method named `mystery()` if it is defined as shown and in a class named `Mystery`?

```
public static char mystery( double a , int b , String s ) { ... }
```

- A. `char[] c = this.mystery(3 , 3 , "333");`
- B. `char c = this.mystery(double 3.0 , int 3 , char[] "333");`
- C. `char c = Mystery.mystery(a = 3.0 , b = 3 , c = "333");`
- D. `char[] c = mystery.mystery(3 , (int) 3.5 , '3' + '3' + '3');`
- E. `char c = Mystery.mystery(3 , (int) 3.5 , null);`

12. Given this partial definition of the class `Radio`:

```
public class Radio {
    private double station;
    private int volume;
    public Radio( double s, int v ) {
        station = s;
        volume = f;
    }
    public boolean equals( Object obj ) {
        Radio r = (Radio) obj;
        return Math.abs( this.station - r.station ) < 0.1
            && this.volume <= r.volume;
    }
    ...
}
```

and the following local variables:

```
Radio r1 = new Radio( 93.1, 2 );
Radio r2 = new Radio( 93.2, 2 );
Radio r3 = new Radio( 93.3, 3 );
Radio r4 = new Radio( 93.3, 1 );
```

Which of these expressions returns **true** for the instances created above?

- A. `r1.equals(r2)`
- B. `r1.equals(r3)`
- C. `r2.equals(r3)`
- D. `r2.equals(r4)`
- E. `r3.equals(r4)`

13. What is displayed by this code fragment?

```
int [ ][ ] x = { {1,2} , {3} , {4,5,6} };
for ( int i = 0 ; i < x.length ; i++ ) {
    int [ ] y = new int[ x[i].length * 2 ];
    for ( int j = 0; j < x[i].length ; j++ ) {
        y[ j ] = x[ i ][ j ] ;
        y[ j + x[i].length ] = x[ i ][ j ]*2 ;
    }
    x[ i ] = y;
}
System.out.println( Arrays.toString( x[ 1 ] ) );
```

- A. `[1 , 2 , 2 , 4]`
- B. `[1 , 2 , 3 , 4 , 5 , 6]`
- C. `[3 , 4]`
- D. `[3 , 4 , 5 , 6]`
- E. `[3 , 6]`

14. What is the output when DirectorTester is run?

```
import java.util.ArrayList;
public class DirectorTester {
    public static void main(String [] args) {
        Director qt = new Director("Quentin Tarantino");
        Movie m1 = new Movie("PulpFiction", qt, 4.5 );
        Movie m2 = new Movie("Reservoir Dogs", qt, 5);
        Movie m3 = new Movie("Jackie Brown", qt, 5);
        System.out.println( qt.getBest() );
    }
}
```

```
public class Movie {
    private String name; Director director; double rating;
    public Movie( String n, Director d, double r ) {
        name = n;
        director = d;
        rating = r;
        d.add(this);
    }
    public String toString() { return name + " (" + rating + " stars)"; }
}
```

```
public class Director {
    private String name;
    private ArrayList<Movie> movieList = new ArrayList<Movie>();
    public Director( String name ) { this.name = name; }
    public void add( Movie m ) { movieList.add(m); }
    public Movie getBest() {
        return movieList.get(0);
    }
    public String toString() {
        String s = name + ": ";
        for ( int i = 0 ; i < movieList.size() ; i++ ) {
            s += movieList.get(i);
        }
        return s;
    }
}
```

- A. Jackie Brown (5.0 stars)
- B. Reservoir Dogs (5.0 stars)
- C. PulpFiction (4.5 stars)
- D. Quentin Tarantino: PulpFiction (4.5 stars)
- E. Quentin Tarantino: [PulpFiction (4.5 stars),Reservoir Dogs (5.0 stars),Jackie Brown (5.0 stars)]

.....

Check that you have answered all questions and bring your UW ID and turn in your exam question paper and scantron sheet.

.....