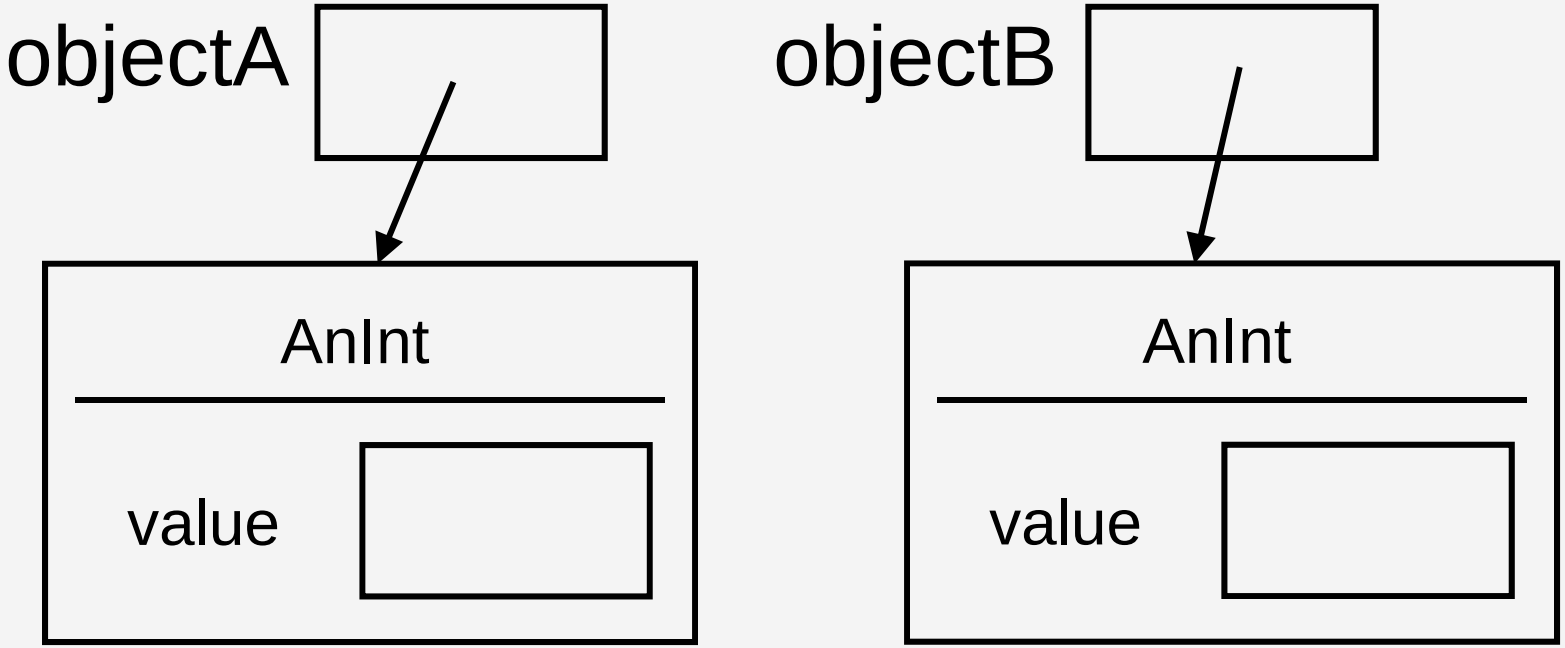
ITEC1620
Object-Based Programming

Lecture 13

References

A Simple Class

```
AnInt objectA = new AnInt();
AnInt objectB = new AnInt();
```



Value Assignments

```
objectA.value = 5;
objectB.value = 7;

objectA.value =objectB.value;
objectB.value = 10;
System.out.println(objectA.value);
System.out.println(objectB.value);
```

Value Assignments II

```
objectA =objectB;
objectB.value = 12;
System.out.println(objectA.value);
System.out.println(objectB.value);
```

- How is objectA.value affected?

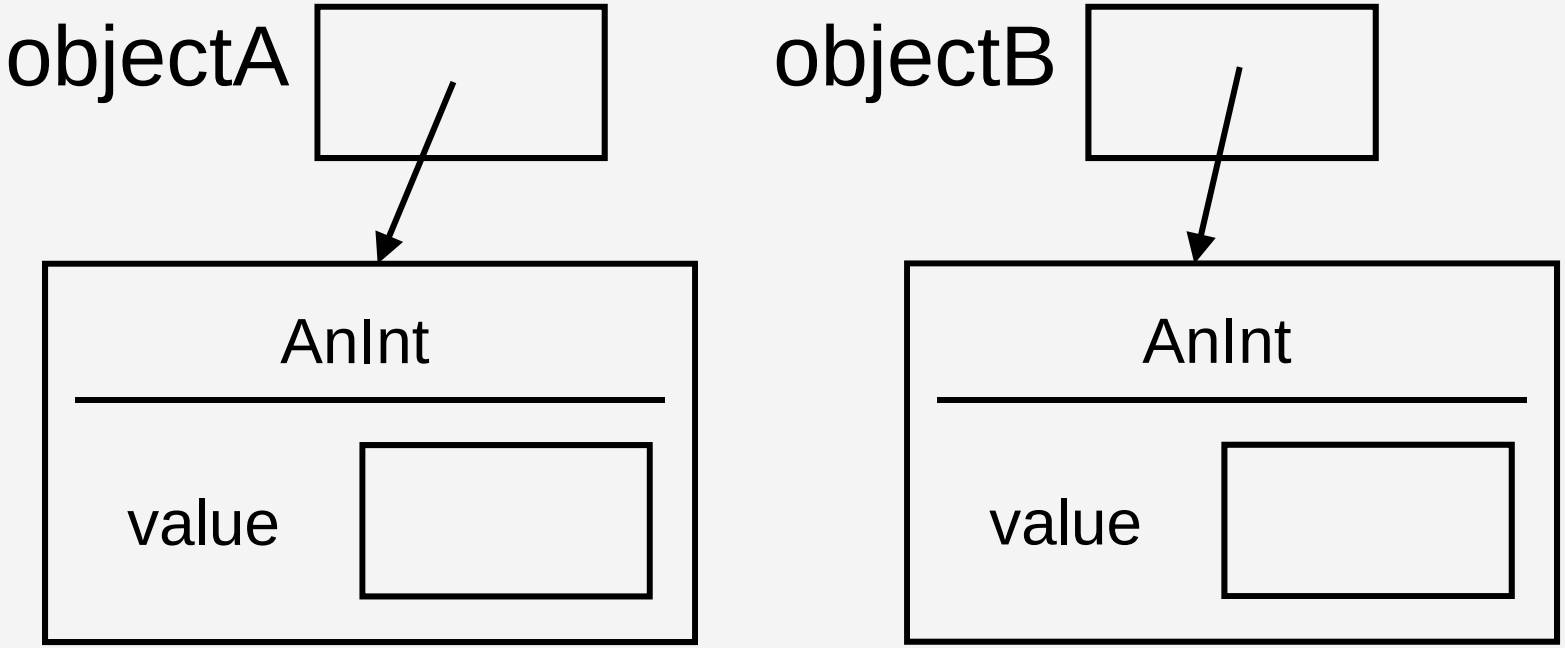
Value Assignments III

- Assignment copies a value from one memory location to another
- What are the memory locations, and what do they hold?

- Note the difference between copying values and copying references

Value Assignments IV

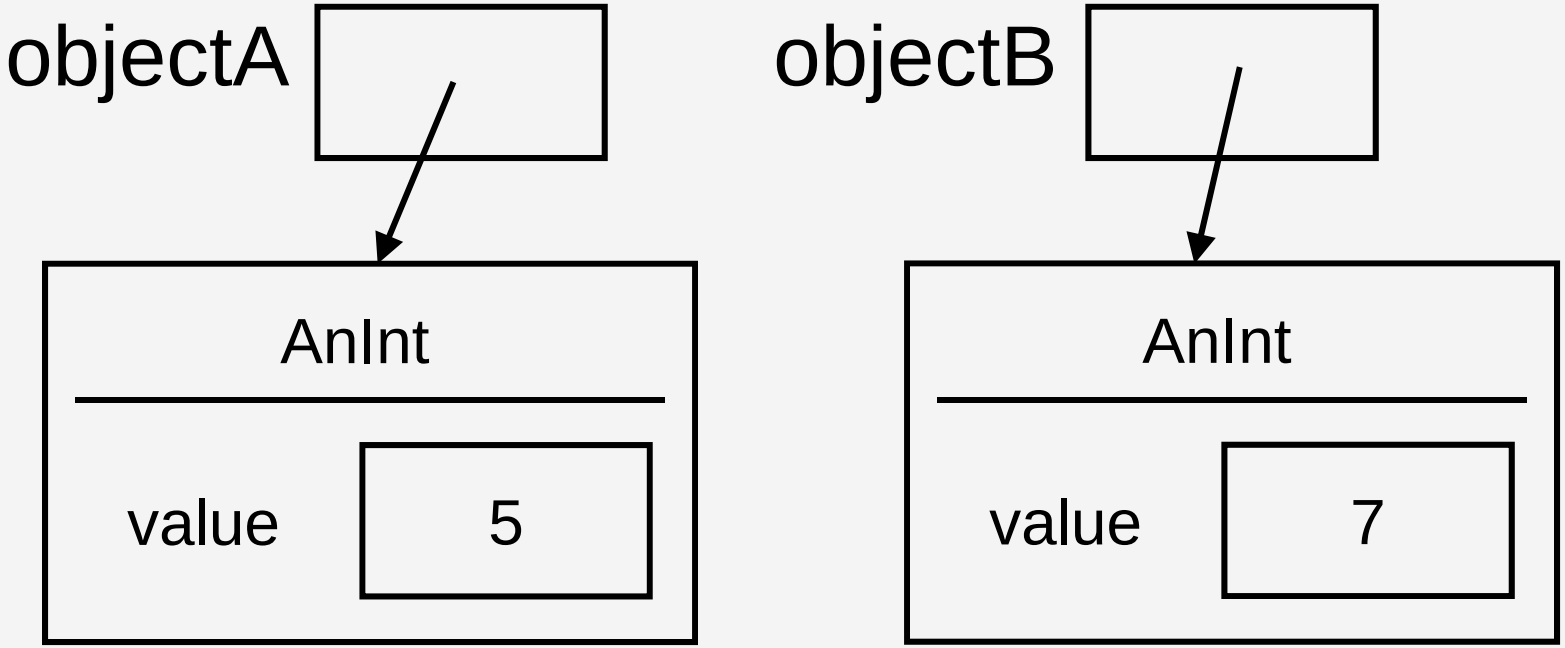
```
AnInt objectA = new AnInt();
AnInt objectB = new AnInt();
```



Value Assignments V

objectA.value = 5;

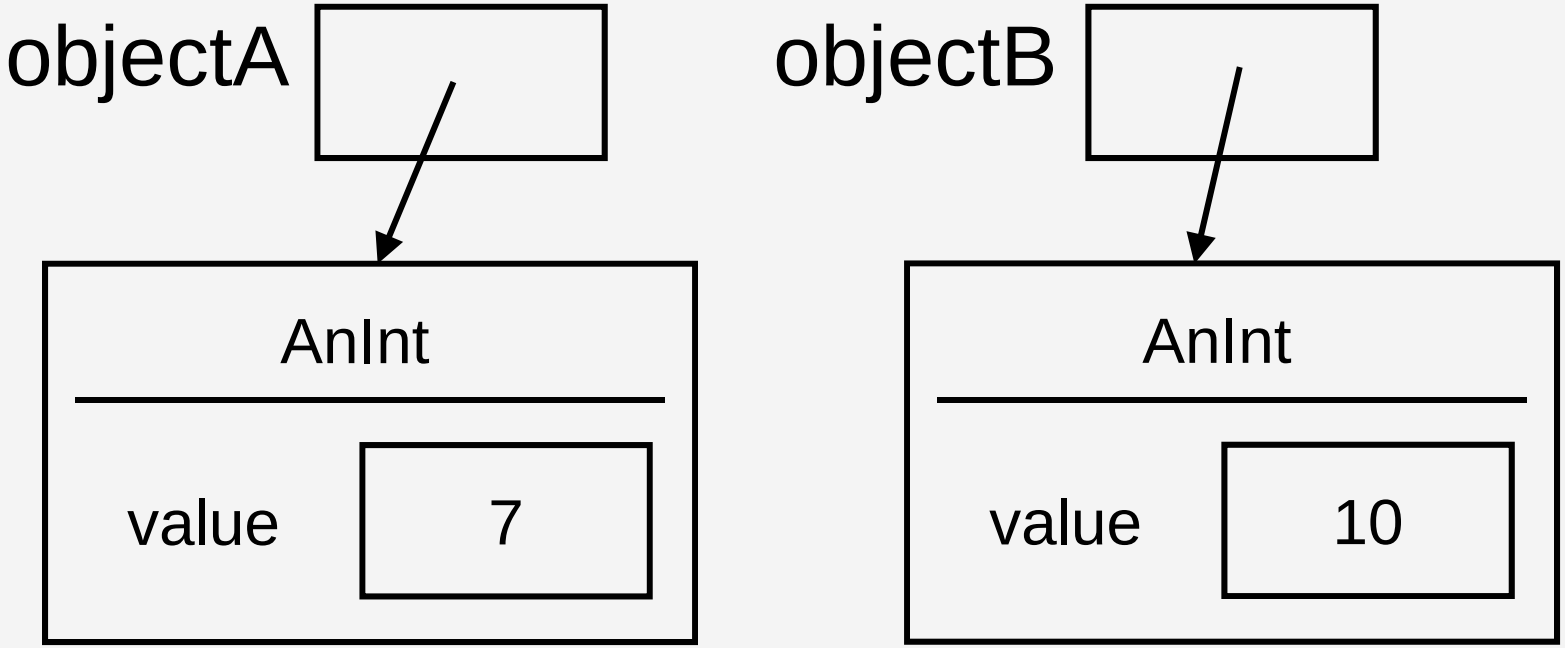
objectB.value = 7;



Value Assignments VI

objectA.value =objectB.value;

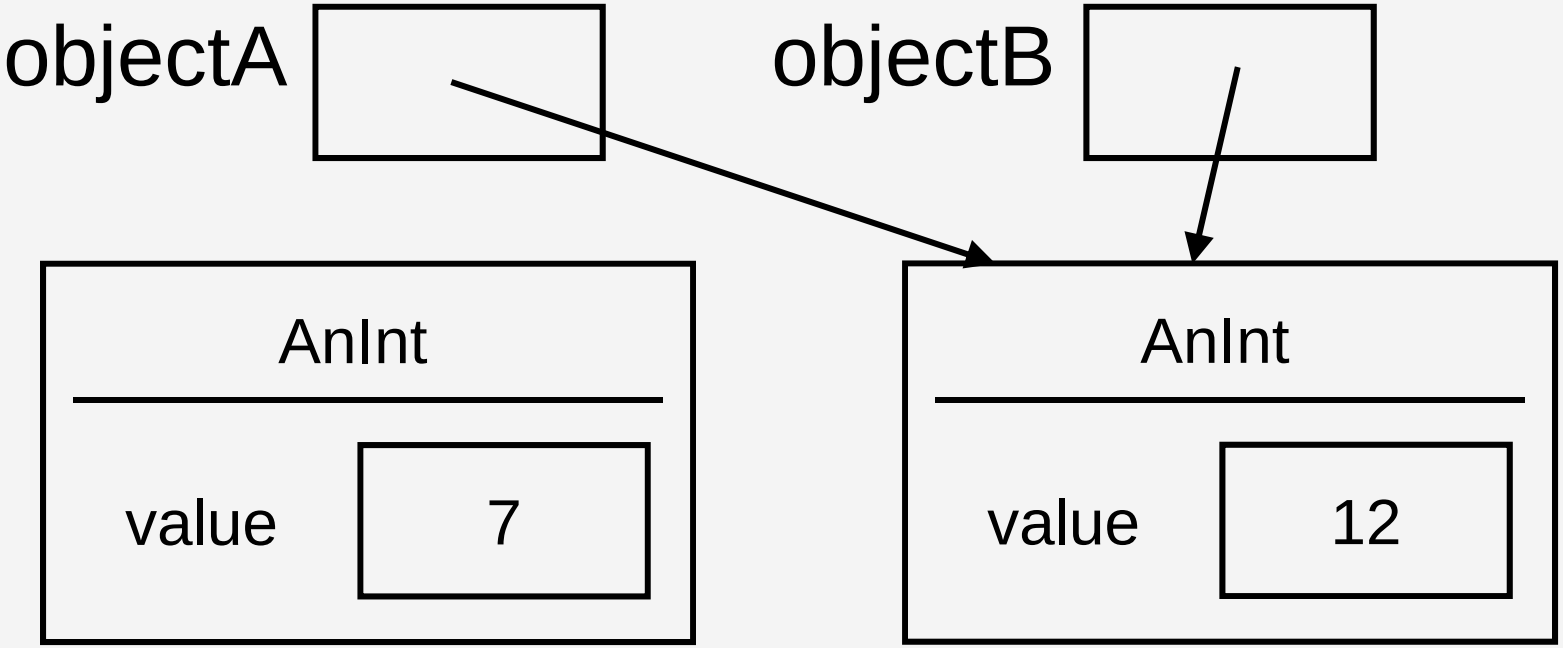
objectB.value = 10;



Value Assignments VII

objectA =objectB;

objectB.value = 12;



Value Assignments VIII

- What does objectA = objectB do?
  - Copies contents in objectB's memory location into objectA's memory location
- What is objectB.value?
  - objectB follow the arrow…
- What is objectA.value?
  - objectA follow the arrow…

Value Assignments IX

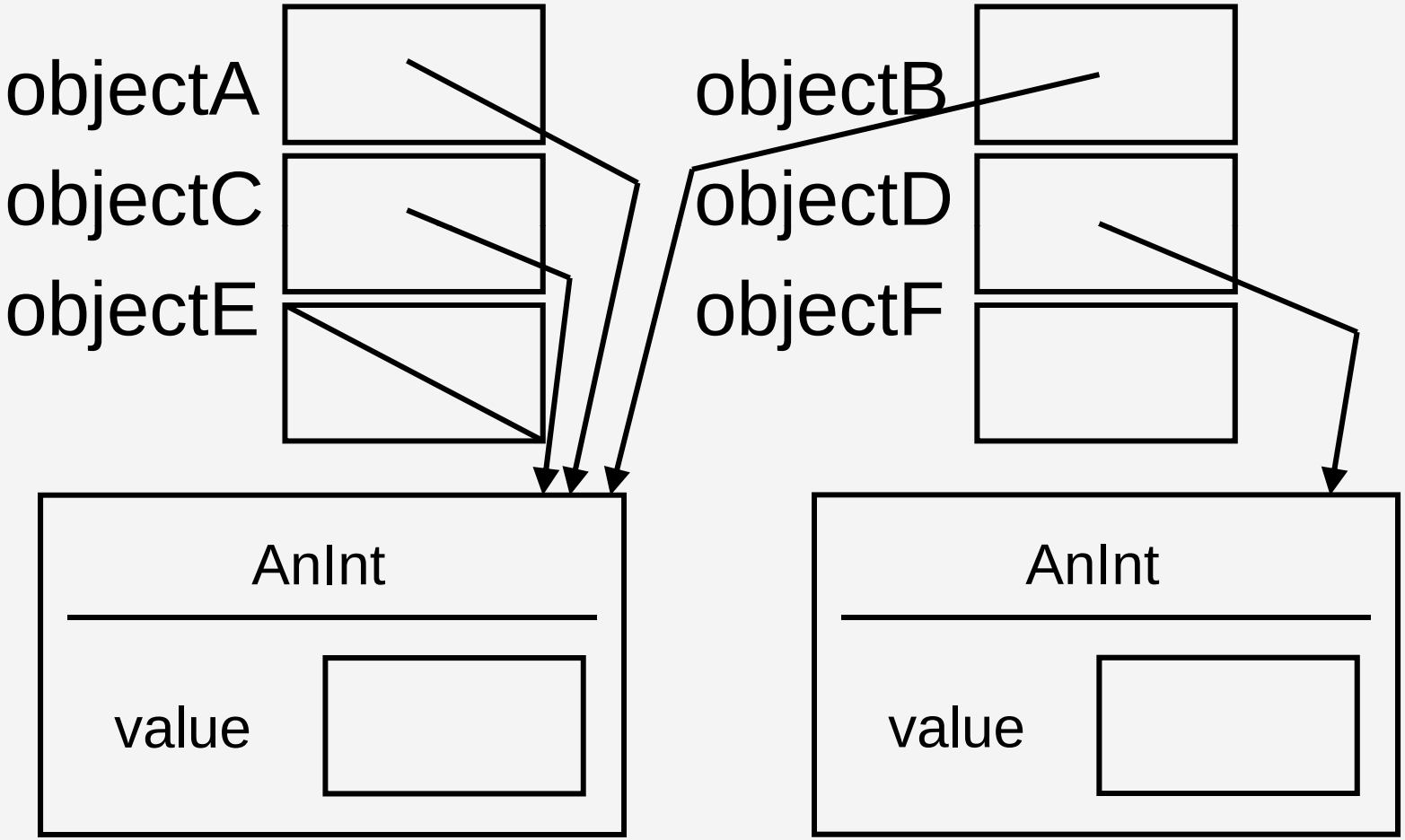
- After objectA = objectB, both identifiers have references to the same object
  - objectA and objectB “are” the same object
  - Mathematical equating
- Any change to objectA is also a change to objectB
- Any change to objectB is also a change to objectA

Questions?

Creating and Copying

```
AnInt objectA = new AnInt();
AnInt objectB = objectA;
AnInt objectC = objectB;
AnInt objectD = new AnInt();
AnInt objectE = null;
AnInt objectF;
```

Creating and Copying II



Creating and Copying III

- How many objects created?
- How many identifiers declared?

- What is the difference between objectE and objectF?

Null vs. Nothing

```
objectF.value;
```

- Syntax error – compiler sees nothing

```
objectE.value;
```

- Runtime error – OS tries to follow null
  - Dot means “follow the arrow” – no arrow

```
if (objectE != null && objectE.value > 0)
   sum += objectE.value;
```

Equals and Equality

identifier1 == identifier2

- What does the == relational operator do?
  - Determines if the two identifiers have the same (or compatible) data types and the same contents in memory

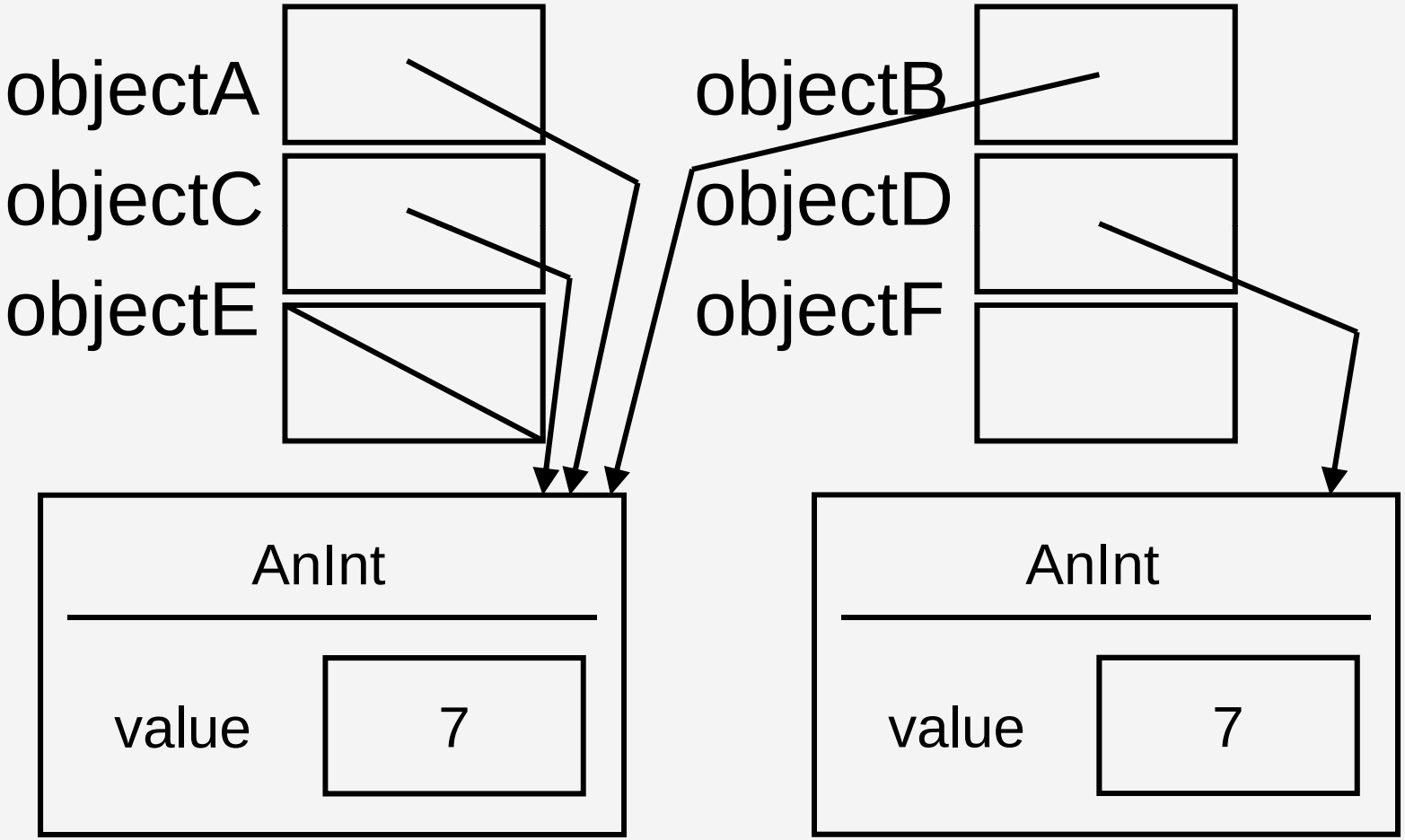
Equals and Equality II

- Use previous objects

```
objectA.value = 5;
objectB.value = 7;
objectD.value = 7;

objectA == objectB          // true or false
objectB == objectD          // true or false
```

Equals and Equality III



Equals and Equality IV

- objectA and objectB refer to the same object
  - Same contents in their memory locations
  - == (equality) returns true – same object
- objectB and objectD refer to different objects
  - Different arrows in their memory locations
  - == returns false – different objects

Equals

- Equality asks if two objects are really one object
- Equals asks if two objects are clones
- Java doesn't know what is in our objects – user defined datatype
  - Must provide a method for each class

```
public boolean equals () {…}
objectB.equals(objectD)          → true
```

Questions?

Review

- Whenever objA == objB is true, then objA.field == objB.field must also be true
  - True or false?
- Whenever objA == objB is false, then objA.field == objB.field must also be false
  - True or false?

Review II

```
AnInt x = new AnInt();
AnInt y = new AnInt();
AnInt z = new AnInt();
x.value = 5;
y.value = 7;
z.value = x.value;
x = y;
y.value = 10;
```

Review III

- What are the final contents of x.value, y.value, and z.value?

Sample Final

```
public class AnInt
{
   public int data;

   public AnInt (int data)
}
```

Sample Final II

```
public class IndInt
{
   public int data;
   public AnInt obj;

   public IndInt (int data, AnInt obj)
}
```

Sample Final III

```
public class MainClass
{
    public static void main(String[] args)
    {
        AnInt x = new AnInt(3);
        AnInt y = new AnInt(5);
        AnInt z = new AnInt(7);
        IndInt a = new IndInt(2, x);

        // Part 1 – draw the object diagrams at this time
```

// Part 1 – draw the object diagrams at this time

Sample Final IV

```
        z.data = a.data;
        x = new AnInt(1);
        y = a.obj;
        a.obj.data = 8;

        // Part 2 – draw the object diagrams at this time
    }
}
```

// Part 2 – draw the object diagrams at this time

Readings and Assignments

- Tutorial – References