Introduction to Programming
with Java, for Beginners

Arrays of Objects

Array of Primitives

```
int[] data;

data = new int[3];
data[0] = 5;
data[1] = 10;
```

Array of Objects

```
Counter[] counters;

collectors = new Counter[3];
  it has 3 references to Counters in it
  all of these references are initially null

collectors[0]=new Counter();
collectors[0].inc();
collectors[1]=new Counter();
collectors[0].getCount();
```

Declaring & Initializing Object Arrays

```
Person[] people = {new Person("jo"),new Person("flo")};
Person[] people = new Person[] {new Person("jo"),
  new Person("flo")};
```

• Point p1 = new Point(0,0);
  Point[] points1 = {p1, new Point(0, 10)};
  Point[] points2 = new Point[] {p1, new Point(0, 10)};

Note: The advantage of using the "new type[]" syntax is that it can be used in an assignment statement that is not a variable declaration statement.
Person Array

```java
Person[] people;

people = new Person[3];

people[0]=new Person("jo");
people[1]=new Person("flo");
```

E.g. Person Database (contd..)

```java
public class PersonDB{
    private Person[] people;
    public PersonDB(){
        people = new Person [] {new Person("jo",25),
                               new Person("flo",18),
                               new Person("mo", 19)};
    }

    /** Calculates and returns the average age. */
    public double getAverageAge(){
        double sum = 0;
        for(int i = 0; i < people.length ; i++){
            sum = sum + people[i].getAge();
        }
        return (sum/people.length);
    }
}
```

"Traversing" Arrays of Objects

- We’ve used loops to traverse arrays of primitives
- We can do the same with arrays of objects