Arrays

### Exercise10.1

1. An Array consist of an ordered collection of similar items. An array, as a whole, has a single name, and the items in an array are referred to in terms of their position within the array.
2. Using the index of an element in a array.

### Exercise 10.2

1. a. 23 b. 12 c.156
2. throws an ArrayIndexOutOfBoundsException if they are out of bounds.

### Exercise 10.3

1. int[] abc = new int[500];

for (int i = 0; I < abc.length; i++)

system.out.println(abc[i]);

1. for (int I = abc.length-1; I >= 0; i--)

system.out.println(abc[i]);

1. int index= a.length;

for(int i= a.length-1; I>= 0; i--){

if(a[i]<0)

index=I;

}

1. a. Gets the absolute each number in the array

b. adds all the value in the array

### Exercise 10.4

1. a. double[] ddd = new double[15];

b. String[] yes= new String[20];

1. The list of numbers between the braces {1,2,3,4}
2. A. int[] testscore = { 100,90,75,60,88};

b. double[] rate = {.12,.05,.15}

c. String[] name ={Josh, Martin};

### Exercise 10.5

1. The computer sends back garbage.

### Exercise 10.6

1. Parallel arrays are several arrays with the same number of elements that work in tandem to organize data.
2. Come back too
3. For (int x=0; x < numbers.length; x++)

System.out.printf(“Name: %-%20s Numbers: %20d%n” name[x], number[x]);

### Exercise 10.7

1. A. For (int element : abc)

System.out.println(abc[element]);

b. What does this do?

1. I have no Idea?

### Exercise 10.8

1. The assignment operator makes the two variables refer to the same array object.
2. As with other objects, arrays are not copied when the assignment operator is used. Therefore, one must instantiate a new array of the same type and length as the original, transfer the items to the new arrays, and return the new array.
3. double average(double[]a){  
   double sum = 0;  
   for (int i : a)  
   sum += i;  
   return sum / a.length;  
   }
4. int[] subArray(int[] a, int first, int last){  
   int[] result = new int[last - first + 1];  
   for (int i = first; i <= last; i++)  
   result[i - first] = a[i];  
   return result;  
   }

### Exercise 10.9

### Student getHighStudent(Student[] students){ Student highest = null; for (int i = 0; i < studentCount; i++) if (highest == null ||  students[i].getHighScore() > highest.getHighScore()) highest = students[i]; return highest; }

1. I have not idea..

### Review pg. 377

1. A. vaild

b. vaild

c. unvaild

d. unvaild

E. unvaild

f. vaild

G. vaild

H. vaild

I. unvaild

J. vaild

1. A. 6

b. 3

c. 3

D. error, greater than the size of the array

E. 10

F. error, can’t square a array

1. A. Array type is int can’t make 10 new doubles

b. can’t make 1.5 objects in an array

c. array can’t have a negative

4. What?

// project 10-1

**import** java.util.Scanner;

**import** java.util.Arrays;

**public** **class** project101 {

**public** **static** **void** main(String args[])

{

**int**[] evenList = **new** **int**[10];

**int** [] oddList = **new** **int**[10];

**int**[] negativeList = **new** **int**[10];

**int** num, even = 0,odd = 0,neg = 0;

String evenstr = **null** , oddstr = **null**, negivtive = **null**;

Scanner reader = **new** Scanner(System.in);

**for** (**int** i =0; i < 10; i++)

{

System.out.println(" enter in a number(#" + (i+1) + ") :: " );

num = reader.nextInt();

**if** (num % 2 ==0)

{

evenList[even] = num;

even++;

}

**else** **if** (num % 2 ==1)

{

oddList[odd] = num;

odd++;

}

**else** **if** ( num <0)

{

negativeList[neg] = num;

neg++;

}

}// end of for loop

**for** (**int** y= 0; y < evenList.length;y++)

{

**if** (evenList[y] > 0)

evenstr = evenList[y] +" ";

**if** (oddList[y] > 0)

oddstr = oddList[y] +" ";

**if** (negativeList[y] > 0)

negivtive = negativeList[y] +" ";

}// end of loop number 2

System.out.print(" Even numbers are:: " + evenstr + " \n Odd numbers are:: " + oddstr + "\n The negitves are:: " + negivtive);

}// end of main

}// end of project 10-1 class

// project 10-2

**import** java.util.Scanner;

**public** **class** Project102 {

**public** **static** **double** arraydoubles(**double**[] a)

{ **double** sum = 0, aver = 0;

**int** count = 0;

**for** (**int** i =0; i < a.length;i++ )

{

sum += a[i];

count++;

}

aver = sum /count;

**return** aver;

}// end of double

**public** **static** **void** main(String args[])

{

**double**[] numbers = **new** **double**[10];

**double** num = 0;

**double** aver = 0;

Scanner reader = **new** Scanner(System.in);

**for** (**int** i =0; i < 10; i++)

{

System.out.println(" enter in a number(#" + (i+1) + ") :: " );

numbers[i] = reader.nextDouble();

}// end of for loop

aver = *arraydoubles*(numbers);

System.out.print(" the avarage is:: " + aver + "\n ");

**for** ( **int** y = 0; y < numbers.length; y++)

{

**if** (numbers[y] > aver)

System.out.print(" "+ numbers[y]);

}

}// end of main

}// end of class project 10-2