

1. What index value does the third element of an array have

The index value that the third element of an array is 2 because an array always starts from 0 making the third element of the array have an index value of 2

2. Write the declaration for an array named quantities that store 20 integers

```
int[] quantities = new int[20];
```

3. Write a declaration for an array named heights storing the numbers 1.65, 2.15 and 4.95

```
Double[] heights= double[] heights = {1.65, 2.15, 4.95};
```

4. Write a for-each statement that displays the integers values stored in an array named grades

```
for (int element : grades) {  
    System.out.println(element);  
}
```

6. How does passing an entire array to a method differ from passing a single element of the array?

Passing an entire array to a method gives the method access to the whole array because arrays are objects, so the method receives a reference to the original array and can change its elements. Passing a single element is different because only the value of that element is sent, not the actual array location, so the method cannot change the original array.

7. Why are offset array indexes required in some cases

Offset array indexes are used to start accessing or processing an array from a specific position instead of the first element.

8. What output is displayed by the statements below

```
String name = "Elaine"  
System.out.println(name.charAt(3));
```

It is displaying the i in Elaine because that's what name.charAt(3) is doing

10. Give an example of when a dynamic array might be a better structure voice over an array

When you code varies in size during run time and it is useful if it the code is needed to grow or shrink

11. How does the ArrayList indexOf() method determine equality between the object passed to the method and an element in the array?

The ArrayList indexOf() method checks equality using an element's equals() method. It searches the list from the start and returns the index of the first matching element, or -1 if none is found. If the argument is null, it looks for a null element instead.

12 .How can the values of wrapper class objects be compared

Values of wrapper class objects can be compared using the compareTo() method, which returns a negative, zero, or positive value depending on whether the first object is less than, equal to, or greater than the second. They can also be compared by converting the object to a primitive type and using standard operators.