

## Chapter 9 CRT

Surkhab Mundi

1. The value of the index of the third element in an array is 2. This is because we count zero when counting elements in an array, rather than starting at one. As such, we can use a simple formula where index value is equal to 1 - the element position
2. The declaration for an array named quantities that stores 20 integers is as follows: **int[] quantities = new int[19];**
3. The declaration for an array named heights storing the numbers 1.65, 2.15 and 4.95 is as follows: **Double[] heights= {1.65, 2.15, 4.95}**
4. for (int grades: grades)

```
{  
  
    System.out.println(grades);  
  
}
```

5. a) 

```
int Data = 1;  
int[] array = new int[10];  
for (int i = 0; i < 10; i++)  
{  
    array[i] = array[i] + 1  
}  
  
array[0] = Data;
```

  
b) 

```
int Data = 10;  
int[] array = new int[10];  
for (int i = 0; i < 10; i++)  
{  
    array[i] = array[i + 1]  
}  
array[10] = Data;
```

6. When passing a single element in an array you are only taking a single value from the array, whereas when passing an entire array you are taking all the information contained in the array. As well, when passing an entire array you just write the type of array followed by square brackets and the array's name, whereas with a single element you are going to have to write down which element via the index and the square brackets.

7. An offset array index is used in situations where a standard array index would be inefficient, for example if there would be a large amount of elements unused. An offset array index allows you to start at whichever number is required for the specific situation which makes the entire process more efficient and gets rid of unnecessary numbers. Also, for example if you needed to count the number of individuals in a school, which is a very specific number of elements, a standard array index would not be adequate, as you cannot have 0.
8. The character would be "i".