

■ 배열의 최대값 구하기 (재귀호출 사용)

```
int[] array = {3, 8, 1, 6, 2, 10, 5};
```

```
int max = findMax(arr, 0, arr.length - 1);  
System.out.println("배열의 최대값: " + max);
```

```
private static int findMax(int[] arr, int start, int end) {  
    if (start == end) {  
        // 배열의 크기가 1인 경우  
        return arr[start];  
    } else {  
        // 배열을 두 부분으로 나누어 각 부분에서 최대값을 찾음  
        int mid = (start + end) / 2;  
        int max1 = findMax(arr, start, mid);  
        int max2 = findMax(arr, mid + 1, end);  
  
        // 두 부분에서 찾은 최대값 중 큰 값을 반환  
        return max1 > max2 ? max1 : max2;  
    }  
}
```

```
findMax( arr, 0,6)  
max1= findMAX( arr, 0, 3) 8  
max2= findMAX( arr, 4, 6) 10  
return max1 > max2 ? max1 : max2;
```

```
findMax( arr, 0,3)  
max1= findMAX( arr, 0, 1)  
max2= findMAX( arr, 2, 3)  
return max1 > max2 ? max1 : max2;
```

```
findMax( arr, 0,1)  
max1= findMAX( arr, 0, 0)  
max2= findMAX( arr, 1, 1)  
return max1 > max2 ? max1 : max2;
```

```
findMax( arr, 2,3)  
max1= findMAX( arr, 2, 2)  
max2= findMAX( arr, 3, 3)  
return max1 > max2 ? max1 : max2;
```

```
findMax( arr, 0,0)  
return arr[0]
```

```
findMax( arr, 1,1)  
return arr[1]
```

```
findMax( arr, 2,2)  
return arr[2]
```

```
findMax( arr, 3,3)  
return arr[3]
```

findMax(arr, 4,6)

```
max1= findMAX( arr, 4, 5)
max2= findMAX( arr, 5, 6)
return max1>max2?max1: max2;
```

```
findMax( arr, 4,5)
max1= findMAX( arr, 4, 4)
max2= findMAX( arr, 5, 5)
return max1>max2?max1: max2;
```

```
findMax( arr, 5,6)
max1= findMAX( arr, 5, 5)
max2= findMAX( arr, 6, 6)
return max1>max2?max1: max2;
```

```
findMax( arr, 4,4)
return arr[4]
```

```
findMax( arr, 5,5)
return arr[5]
```

```
findMax( arr, 5,5)
return arr[5]
```

```
findMax( arr, 6,6)
return arr[6]
```