

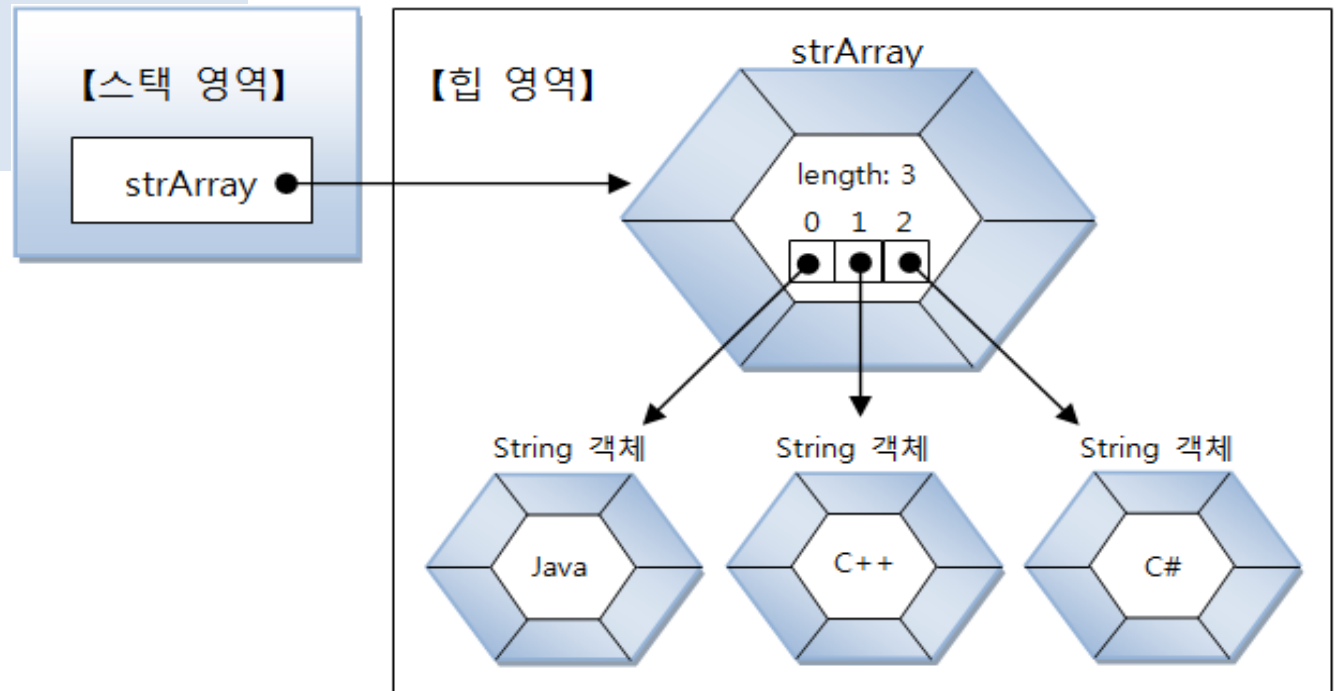
배열의 조작

배열의 조작

❖ 객체를 참조하는 배열

- 기본 타입(byte, char, short, int, long, float, double, boolean) 배열
 - 각 항목에 직접 값을 가지고 있음
- 참조 타입(클래스, 인터페이스) 배열 - 각 항목에 객체의 번지 가짐

```
String[] strArray = new String[3];  
strArray[0] = "Java";  
strArray[1] = "C++";  
strArray[2] = "C#";
```



배열의 조작

❖ 객체를 참조하는 배열 : ArrayReferenceObjectExample.java

```
public class ArrayReferenceObjectExample {  
  
    public static void main(String[] args) {  
  
        String[] strArray = new String[3];  
  
        strArray[0] = "Java";  
        strArray[1] = "Java";  
        strArray[2] = new String("Java");  
  
        System.out.println(strArray[0] == strArray[1]);  
        System.out.println(strArray[0] == strArray[2]);  
        System.out.println(strArray[0].equals(strArray[2]));  
    }  
}
```

배열의 조작

❖ 배열 복사

- 배열은 한 번 생성하면 크기 변경 불가
- 더 많은 저장 공간이 필요하다면 보다 큰 배열을 새로 만들고 이전 배열로부터 항목 값들을 복사

❖ 배열 복사 방법

- for문 이용
- `System.arraycopy()` 메소드 이용
- `Arrays` 클래스 이용

배열의 조작

❖ for 문으로 배열 복사: ArrayCopyByExample.java

```
public class ArrayCopyByExample {  
    public static void main(String[] args) {  
  
        int[] oldIntArray = { 1, 2, 3 };  
        int[] newIntArray = new int[5];  
  
        for (int i = 0; i < oldIntArray.length; i++) {  
            newIntArray[i] = oldIntArray[i];  
        }  
  
        for (int i = 0; i < newIntArray.length; i++) {  
            System.out.print(newIntArray[i] + ", ");  
        }  
    }  
}
```

배열의 조작

❖ System.arraycopy()로 배열 복사: ArrayCopyExample.java

```
public class ArrayCopyExample {  
    public static void main(String[] args) {  
        String[] oldStrArray = { "java", "array", "copy" };  
        String[] newStrArray = new String[5];
```

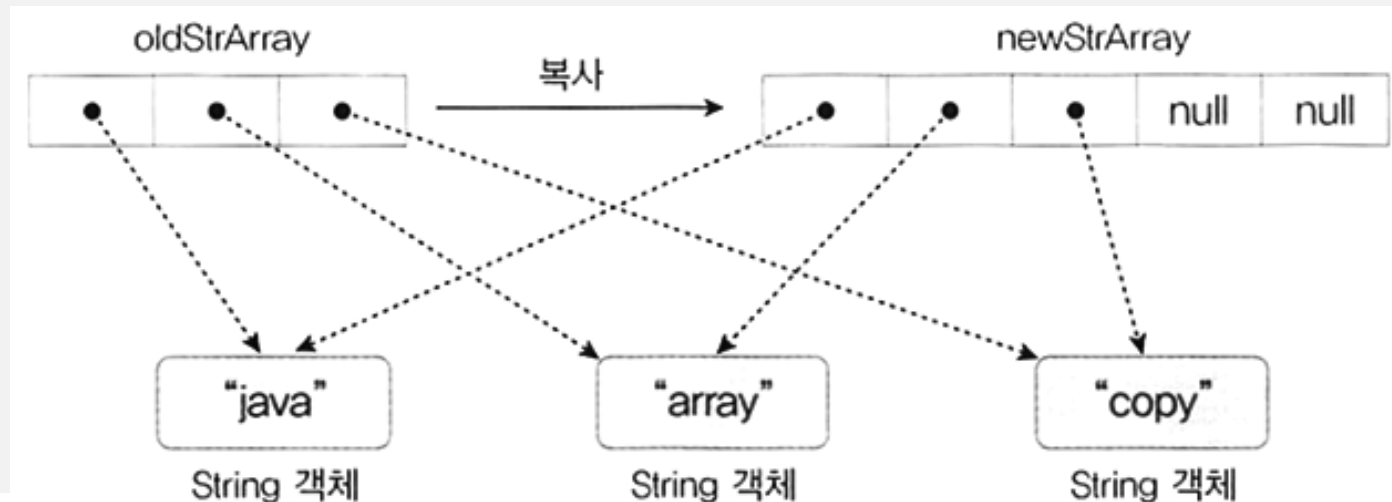
```
        System.arraycopy(oldStrArray, 0, newStrArray, 0,  
                           oldStrArray.length);
```

```
        for (int i = 0; i < newStrArray.length; i++) {  
            System.out.print(newStrArray[i] + ", ");
```

```
        }
```

```
    }
```

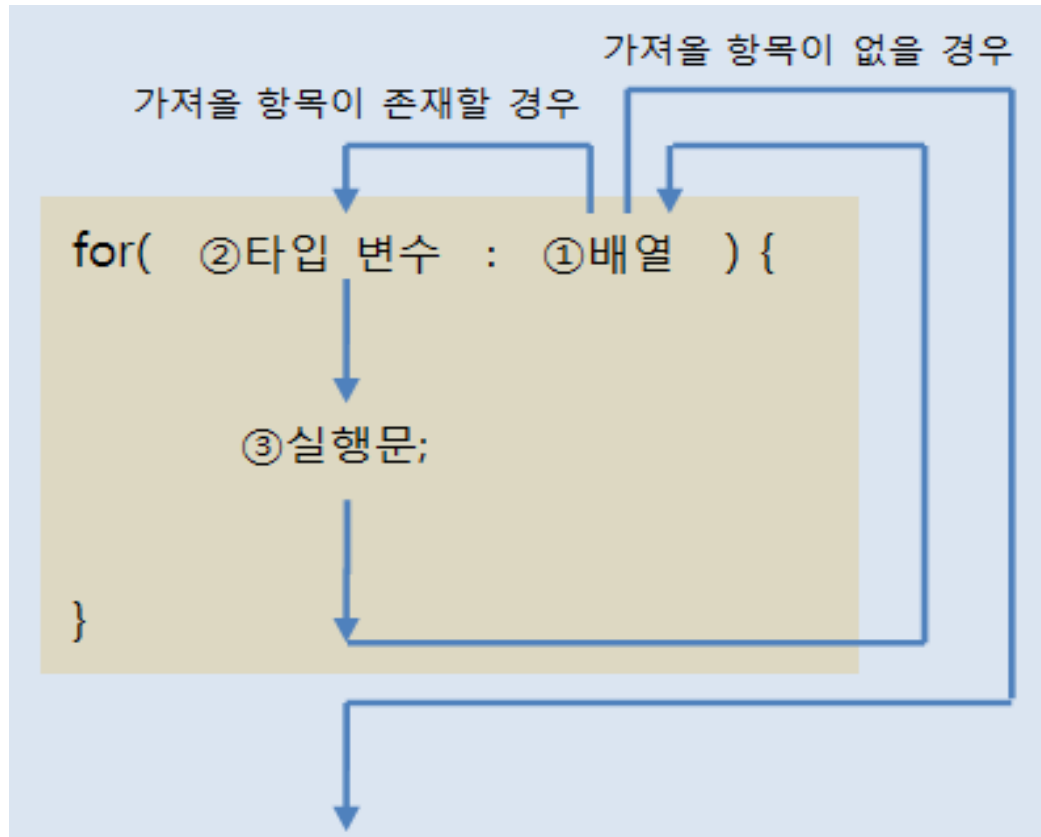
```
}
```



배열의 조작

❖ 향상된 for 문

- 배열 및 컬렉션(15장에서 다룸)의 항목 요소를 순차적으로 처리
- 인덱스 이용하지 않고 바로 항목 요소 반복



```
int[] scores = {  
    95, 71, 84, 93, 87  
};  
int sum = 0;
```

```
for (int score : scores) {  
    sum = sum + score;  
}
```

배열의 조작

❖ 향상된 for 문: AdvancedForExample.java

```
public class AdvancedForExample {  
    public static void main(String[] args) {  
        int[] scores = { 95, 71, 84, 93, 87 };  
  
        int sum = 0;  
        for (int score : scores) {  
            sum = sum + score;  
        }  
        System.out.println("점수 총합 = " + sum);  
  
        double avg = (double) sum / scores.length;  
        System.out.println("점수 평균 = " + avg);  
    }  
}
```