# C++ VS. JAVA

C++ arrays are a primitive data type

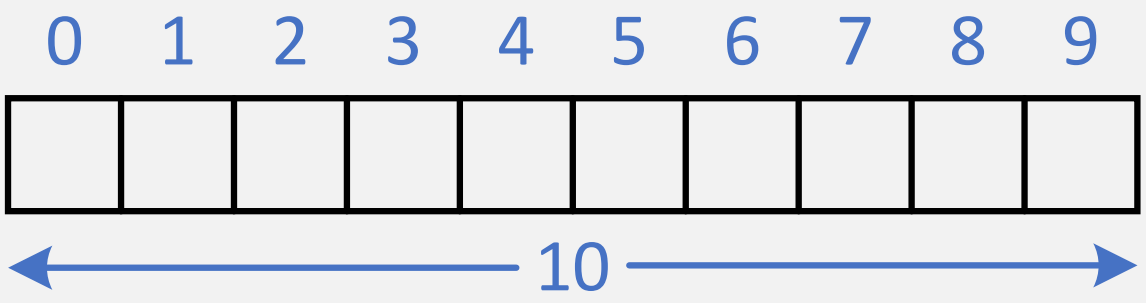
Java arrays are objects

Delroy A. Brinkerhoff

# TRACKING THE SIZE

- Java arrays have a length attribute (instance field)
  - `int[] scores = new int[10];`
  - `scores.length`
- C++ arrays are really pointers – they do not have attributes or fields
  - Track the size with a (named) constant
  - `const int size = 10;`
  - `int scores[size];`

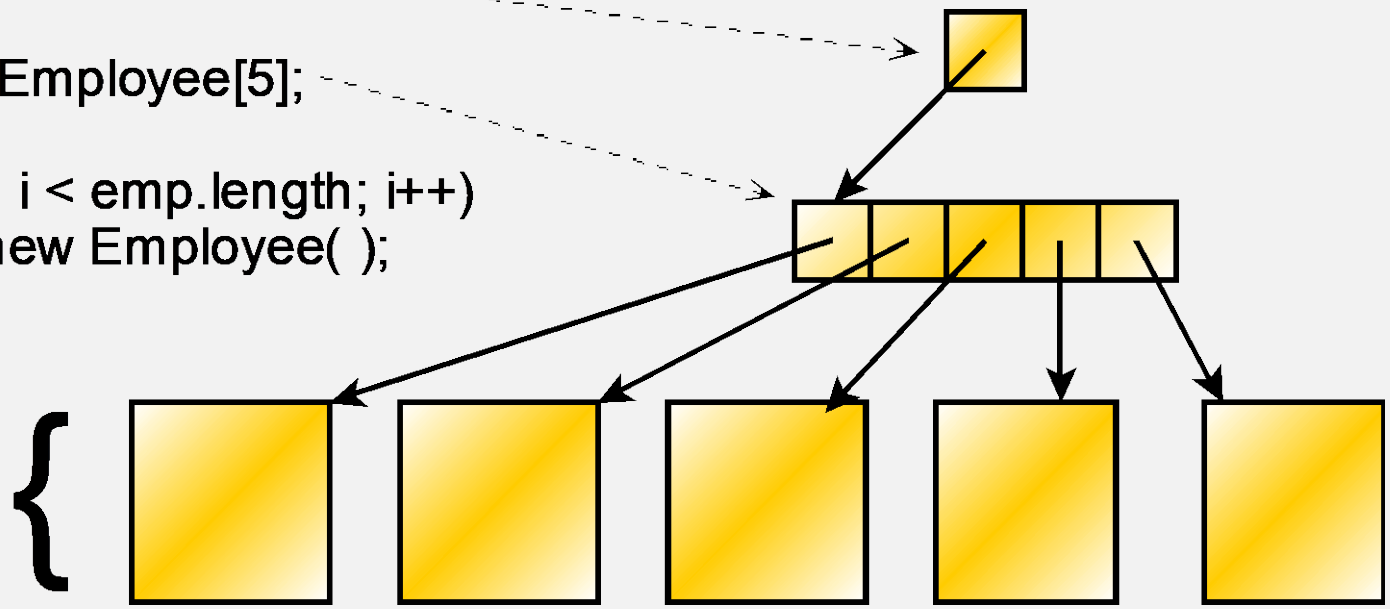
# BOUNDS CHECKING




- Java checks each array index, throws an exception if the index is out of bounds
  - `scores[i]`, throws an exception if i < 0 or if i ≥ 10
- C++ does not check any array index
  - `scores[-1]` or `scores[10]` will crash the program or corrupt adjacent data

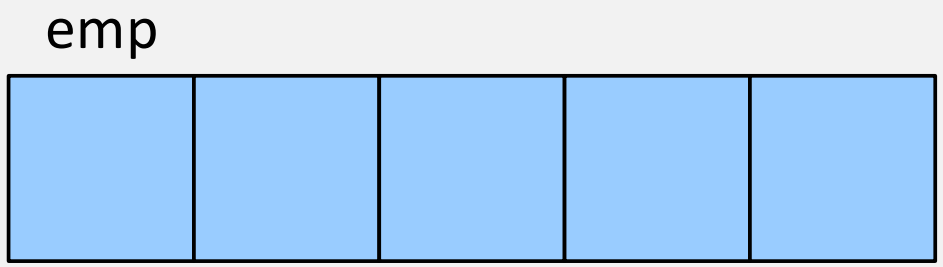
# JAVA: ARRAYS OF OBJECTS

```
Employee[ ] emp;

emp = new Employee[5];

for (int i = 0; i < emp.length; i++)
    emp[i] = new Employee( );
```



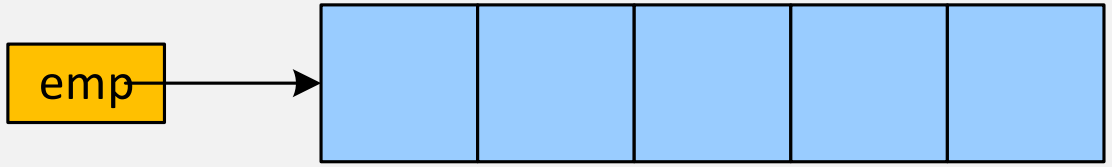
# C++: AUTOMATIC ARRAYS OF OBJECTS

```
Employee emp[5];
```



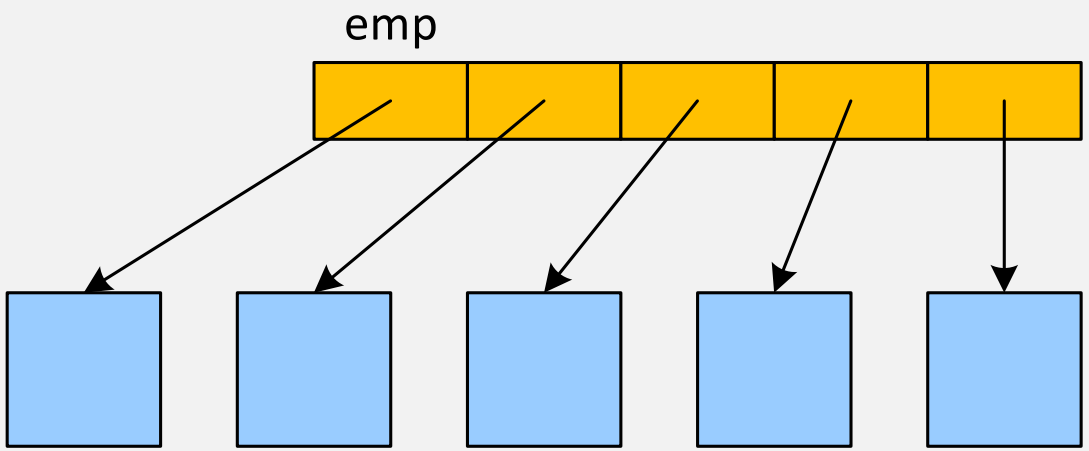
# C++: DYNAMIC ARRAYS OF OBJECTS

```
Employee* emp;
emp = new Employee[5];
```



# C++: ARRAY OF POINTERS TO OBJECTS

```
Employee* emp[5];
for (int i = 0; i < 5; i++)
    emp[i] = new Employee;
```



# C++: POINTER TO AN ARRAY OF POINTERS

```
Employee** emp;
emp = new Employee*[5];
for (int i = 0; i < 5; i++)
    emp[i] = new Employee;
```