

SCOPE

CS302 – Introduction to Programming
University of Wisconsin – Madison
Lecture 10

By Matthew Bernstein – matthewb@cs.wisc.edu

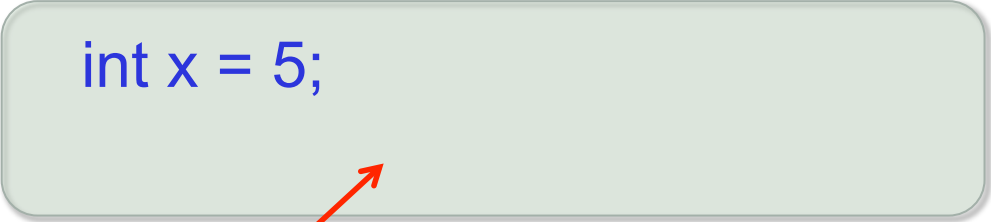
Scope

- Every variable has a **scope**
- The scope of a variable consists of all parts in the program where the variable can be used
- Two variables with the same name cannot have overlapping scope

Example

- In the following program the scope of the variable “x” consists of anywhere in the “main” method below its declaration:

```
public class ScopeExample
{
    public static void main(String args[])
    {
        int x = 5;
    }
}
```



The scope of “x”

What defines a variable's scope?

- A variable's scope is defined by the inner-most curly braces that contain it
- For example, in the following code segment, the variable "y" can only be used within the if-statement

```
if (true)
```

```
{
```

```
    int y = 4;
```

```
    .....
```

```
}
```

The scope of "y"

Example

```
public class ScopeExample
{
    public static void main(String args[])
    {
        int x = 5;

        if (true)
        {
            int y = 5;
            ...
        }
    }
}
```

The scope of "x"

The scope of "y"

Scope and for-loops

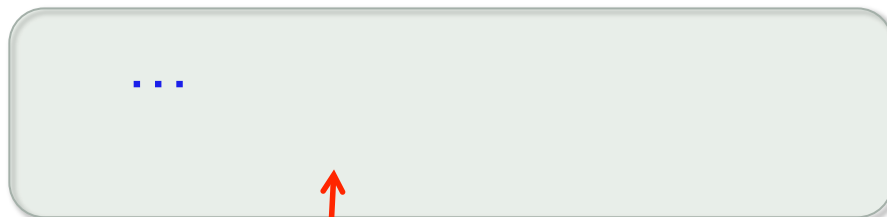
- If you declare your counter variable in the for-loop's declaration, the counter variable's for loop consists of the loop itself
- Example:

```
for (int i = 0; i < 10; i++)
```

```
{
```

```
    ...
```

```
}
```



The scope of "i"

Variables with same name

- We can declare variables with the same name as long as their scopes do not overlap
- The following is legal:

```
for (int i = 0; i < 10; i++)
```

```
{
```

```
    ...
```

```
}
```

```
for (int i = 0; i < 20; i++)
```

```
{
```

```
    ...
```

```
}
```



The scope of first "i"



The scope of second "i"

Variables with same name

- The following is **illegal** because the two scopes of “i” overlap

```
for (int i = 0; i < 10; i++)
```

```
{
```

```
    for (int i = 0; i < 20; i++)
```

```
    {
```

```
        ...
```

```
    }
```

```
}
```

The scope of second “i”

The scope of first “i”