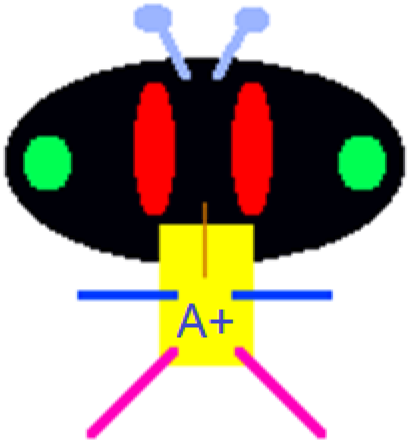
A+ Computer Science

Return Methods

Objects

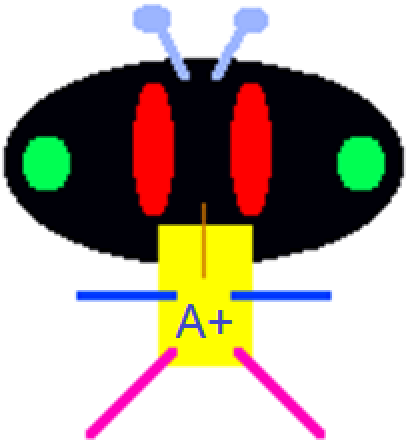
Object Instantiation

```
new Scanner(System.in);

new AplusBug();
```



References

```
Scanner keyboard =
                new Scanner(System.in);

AplusBug dude;
dude = new AplusBug();
```



Constructors

Constructors

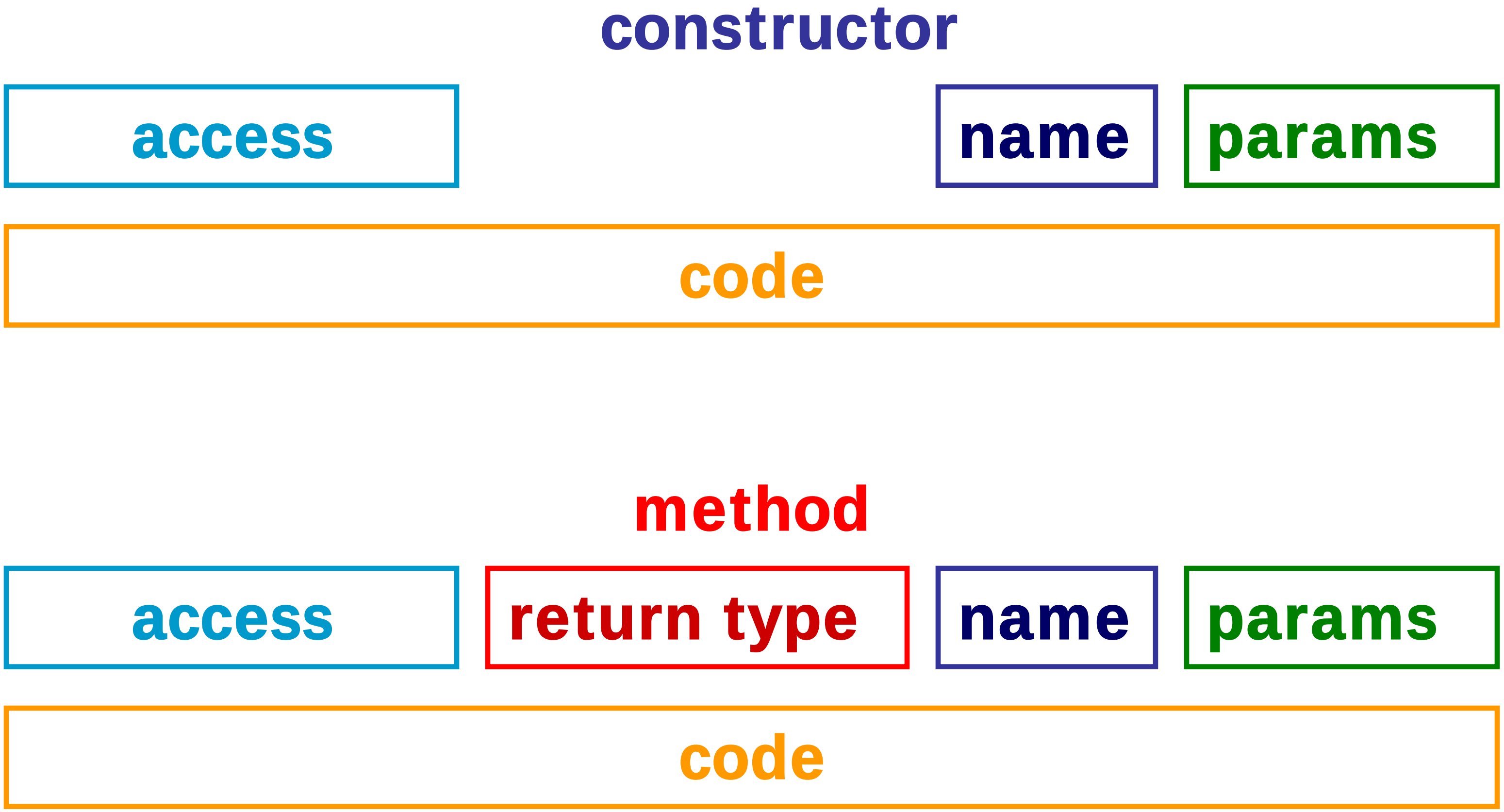
very similar to methods

have the same name as the class

have no return type – no void,int,etc.

initialize all instance variables

Constructors vs. Methods



constructor

access | name | params

code

method

access | return type | name | params

code

Default Constructor

```
class Triangle
{
  private int sideA, sideB, sideC;

  public Triangle()
  {
    sideA=0;
    sideB=0;
    sideC=0;
  }
}
```

```
Triangle triangle = new Triangle();
```

constructorone.java

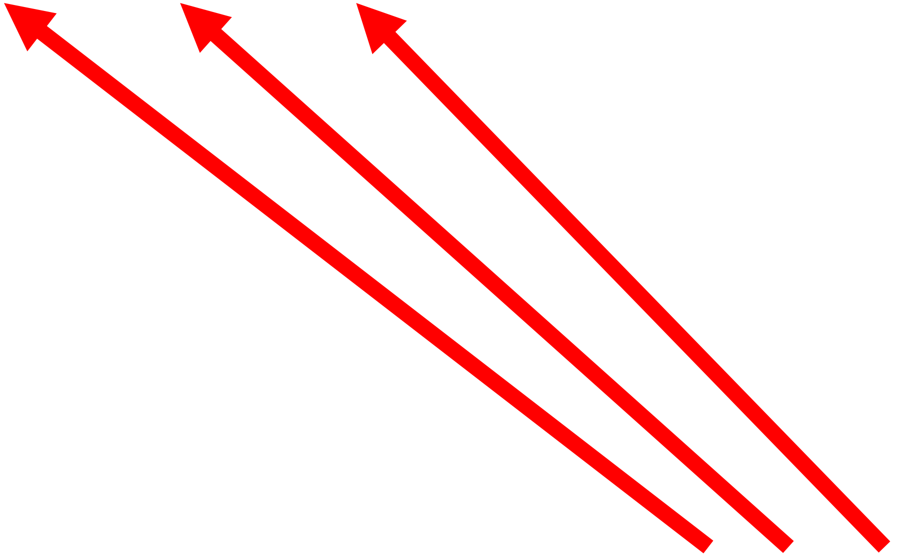
Initialization Constructor

```
public Triangle(int a, int b, int c)
{
   sideA=a;
   sideB=b;
   sideC=c;
}
```

Constructors often have parameters. The parameters allow data to be passed into the class so that it can be assigned to the instance variables / data fields.

Initialization Constructor

```
class Triangle
{
  private int sideA, sideB, sideC;

  public Triangle(int a, int b, int c)
  {
    sideA=a;
    sideB=b;
    sideC=c;
  }
}
```



```
Triangle triangle = new Triangle(3,4,5);
```

constructortwo.java

Variable Scope

Scope

```
{
   int fun = 99;
}
```

Any variable defined inside of braces, only exists within those braces.

That variable has a scope limited to those braces.

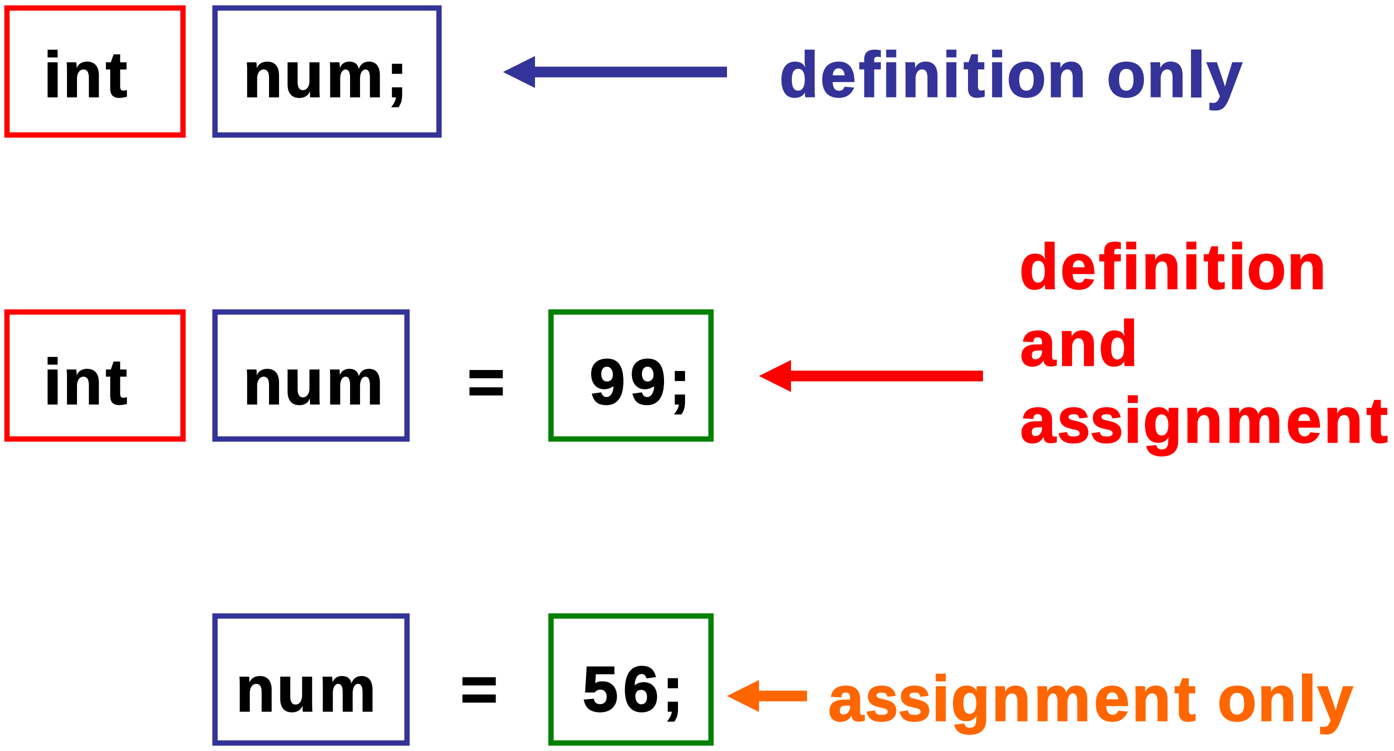
Instance Variables

When you need many methods to have access to the same variable, you make that variable an instance variable.

The scope of an instance variable is the entire class where that variable is defined.

calc.java
calcrunner.java

Defining vs. Assigning



int num; ← definition only

int num = 99; ← definition and assignment

num = 56; ← assignment only

Local Variables

When you need only one method to have access to a variable, you should make that variable a local variable.

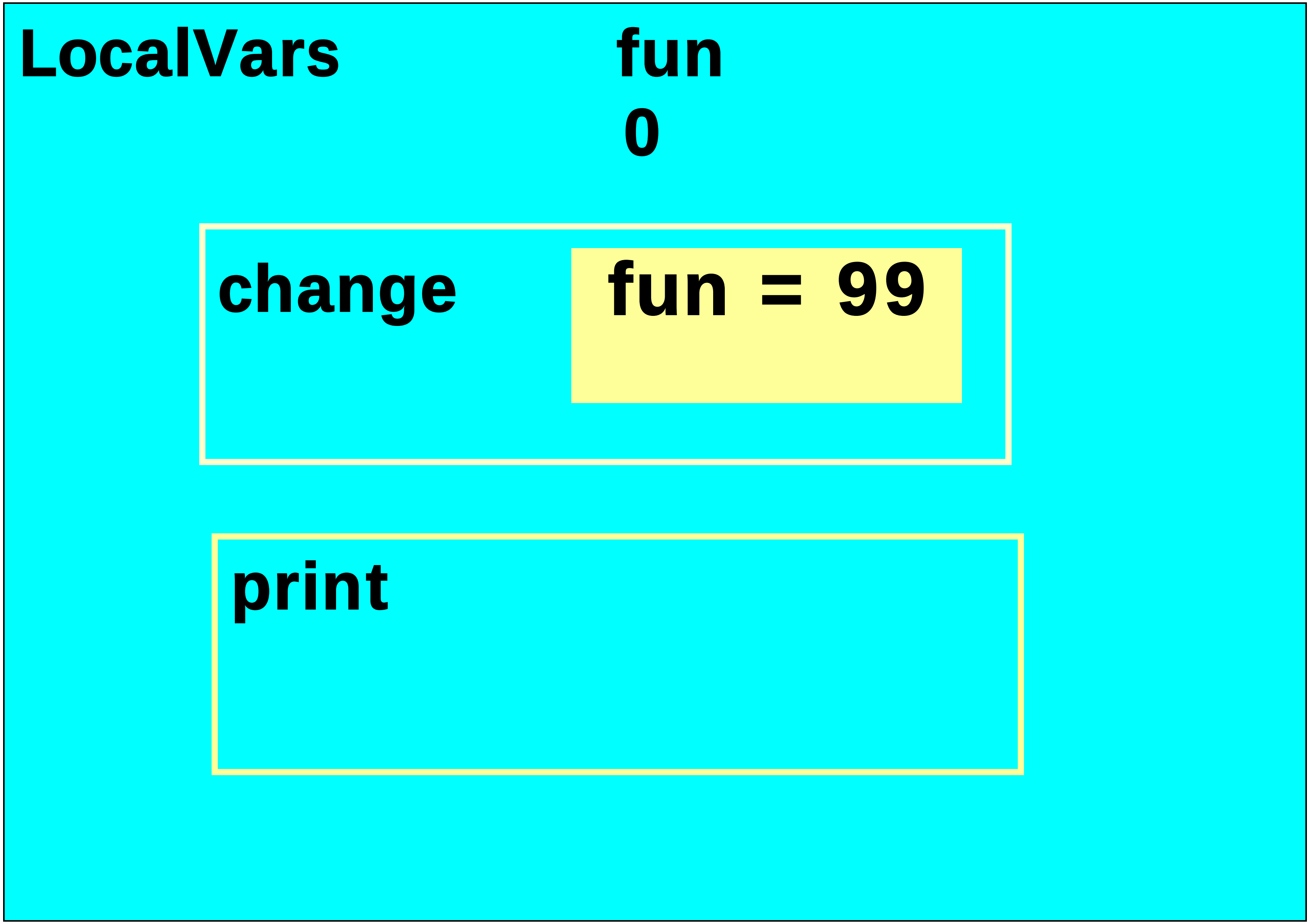
The scope of a local variable is limited to the method where it is defined.

Local Variables

```
public class LocalVars
{
   private int fun;          //instance variable

   public void change()  {
     int fun = 99;           //local variable
   }

   public void print()  {
     System.out.println(fun);
   }

   public static void main(String args[])
   {
     LocalVars test = new LocalVars();
     test.change();
     test.print();
   }
}
```

OUTPUT

```
0
```



LocalVars

fun

0

change

fun = 99

print

localvars.java

Return Methods

Return Methods

Return methods perform some action and return a result back to the calling location.

```
int num = keyboard.nextInt();
```

nextInt() returns an int back to the calling location.

The value returned is assigned to num.

Return Methods

```
Scanner keyboard =
                  new Scanner(System.in);


int num = keyboard.nextInt();
out.println(num);
```

INPUT

1

OUTPUT

1

return method

num

1

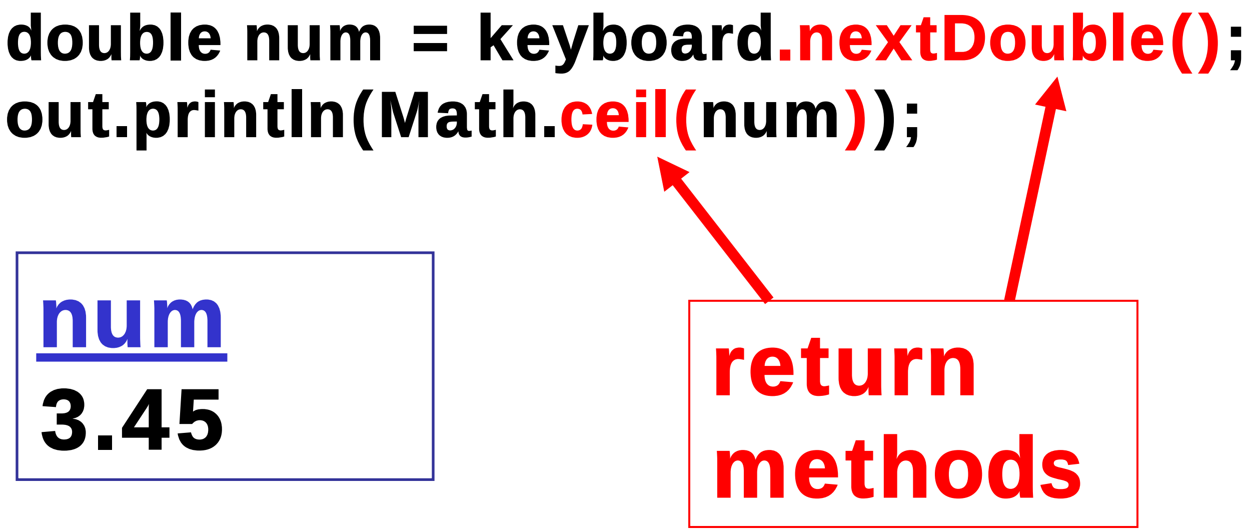
Math Return Methods

Math
frequently used methods

Name	Use
floor(x)	rounds x down
ceil(x)	rounds x up
pow(x,y)	returns x to the power of y
abs(x)	returns the absolute value of x
sqrt(x)	returns the square root of x
round(x)	rounds x to the nearest whole number
min(x,y)	returns smallest of x and y
max(x,y)	returns biggest of x and y
random()	returns a double >=0.0 and < 1.0

Math Methods

```
Scanner keyboard =
                    new Scanner(System.in);
```



```
double num = keyboard.nextDouble();
out.println(Math.ceil(num));
```

return methods

num
3.45

INPUT
3.45

OUTPUT
4.0

Math Methods

```
out.println(Math.floor(3.254));
out.println(Math.ceil(2.45));
out.println(Math.pow(2,7));
out.println(Math.abs(-9));
out.println(Math.sqrt(256));
out.println(Math.sqrt(144));
out.println(Math.round(3.6));
out.println(Math.max(5,7));
out.println(Math.max(5,-7));
out.println(Math.min(5,7));
out.println(Math.min(5,-7));
```

OUTPUT

```
3.0
3.0
128.0
9
16.0
12.0
4
7
5
5
-7
```

Math Methods

```
out.println(Math.random()*10);
int num = (int)(Math.random()*10);
out.println(num);
```

OUTPUT

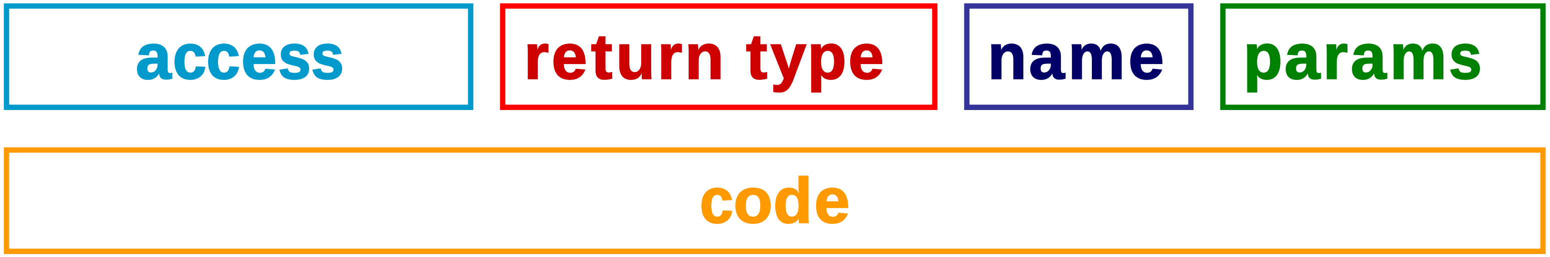
```
7.564
4
```

random() returns a double in the range 0.0 to 1.0, not including 1.0.

mathmethods.java
randomone.java

User-Defined Return Methods

Return Methods



access return type name params

code

Return Methods

```java
public int twice(int x)
{
   return 2*x;
}
```

return method

The call twice(3) would return 6.

The call twice(11) would return 22.

returnone.java
calc.java
calcrunner.java

OOP Pieces

Constructors

```
public Triangle()
{
   sideA=0;
   sideB=0;
   sideC=0;
}
```

Constructors are similar to methods. Constructors set the properties of an object to an initial state.

Constructors

```
public Triangle(int a, int b, int c)
{
   sideA=a;
   sideB=b;
   sideC=c;
}
```

Constructors are similar to methods. Constructors set the properties of an object to an initial state.

Modifier Methods

```
public void setSides(int a, int b, int c)
{
    sideA=a;
    sideB=b;
    sideC=c;
}
```

Modifier methods are methods that change the properties of an object.

Accessor Methods

```
public void print()
{
  out.println(sideA + " " + sideB + " " + sideC);
}
```

Accessor methods are methods that retrieve or grant access to the properties of an object, but do not make any changes.

triangle.java
trianglerunner.java

Work on Programs!

Crank

Some Code!

A+ Computer Science

Return Methods