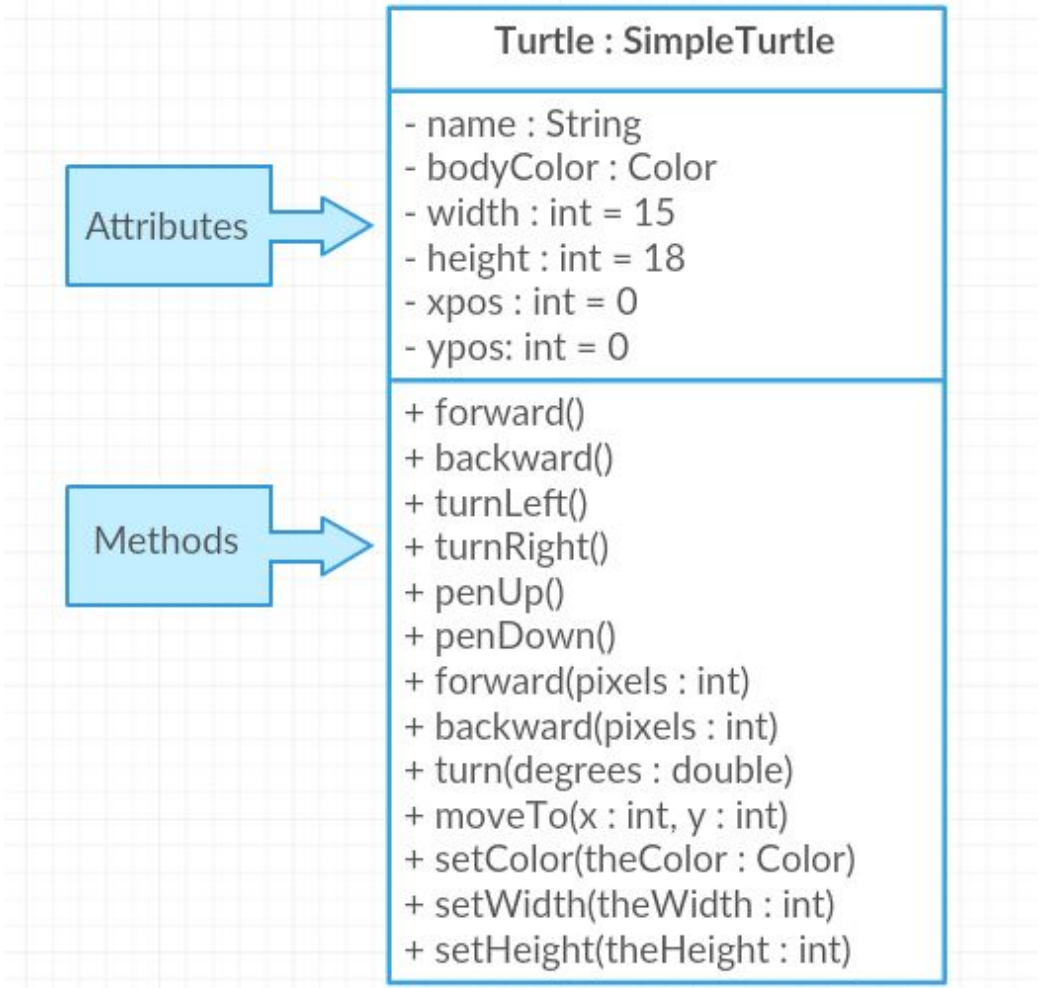
2.2 Creating and Initializing Objects: Constructors


Turtle : SimpleTurtle
- name : String
- bodyColor : Color
- width : int = 15
- height : int = 18
- xpos : int = 0
- ypos: int = 0
+ forward()
+ backward()
+ turnLeft()
+ turnRight()
+ penUp()
+ penDown()
+ forward(pixels : int)
+ backward(pixels : int)
+ turn(degrees : double)
+ moveTo(x : int, y : int)
+ setColor(theColor : Color)
+ setWidth(theWidth : int)
+ setHeight(theHeight : int)
Attributes
Methods

Among Us Characters

AmongUsCharacter : Blue
//Attributes
//Methods

Constructor

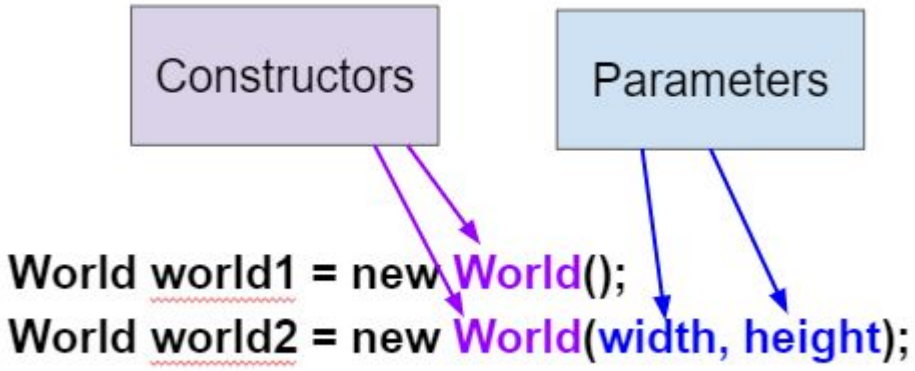
Method used to create an instance of a class.

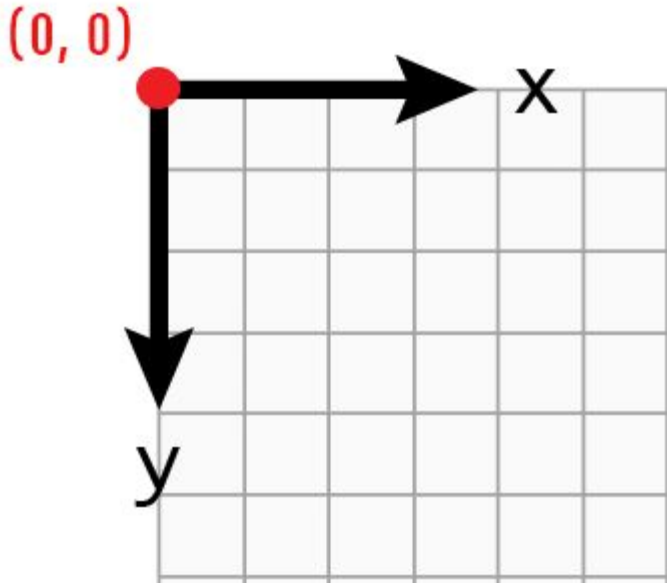
```
// To create a new object and call a constructor write:
// ClassName variableName = new ClassName(parameters);
World habitat = new World();    // create a new World object
Turtle t = new Turtle(habitat); // create a new Turtle object
```

A new object is created with the new keyword followed by the class name (new Class())

Overloading Constructors

There can be more than one constructor defined in a class. This is called **overloading** the constructor.





Dog

```
public class Dog {
    public String breed;
    public int age;
    public String color;

    public Dog() {
        breed = “pug”;
        age = 3;
        color = “brown”;
    }

    public Dog(String a, int b, String c) {
        breed = a;
        age = b;
        color = c;
    }
}
```

Default constructors still need a value to be assigned to the instance variables

Parameter constructors still need a value to be assigned to the instance variables

```
World world1 = new World(); // creates a 640x480 world

World world2 = new World(300,400); // creates a 300x400 world


Turtle t1 = new Turtle(world1);

Turtle t2 = new Turtle(50, 100, world1);
```

Notice here that the order of parameters matters

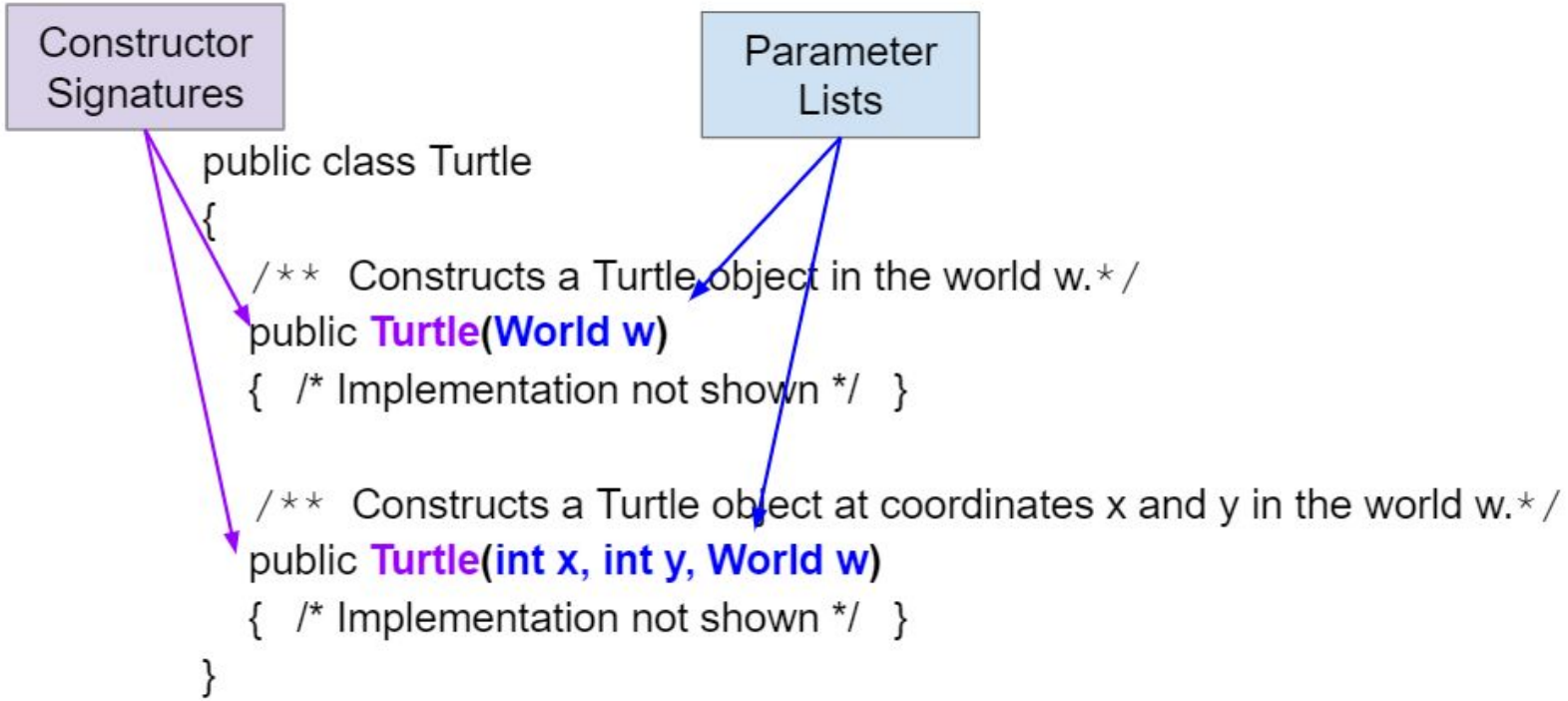
Character constructor

```
public class AmongUsCharacter {
    public String name;
    public String color;
    public boolean imposter;
    public int size;

    public AmongUsCharacter(String a, String b, boolean c, int d){
        name = a;
        Color = b;
        Imposter =c;
        Size = 3;
```

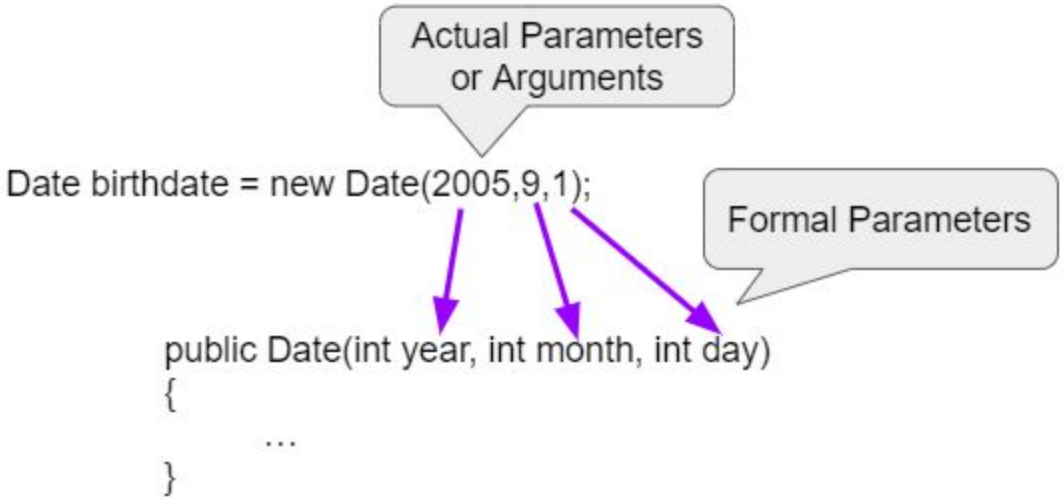
You can declare an object variable and initialize it to null

```
World world1 = new World();

Turtle t1 = null;

t1 = new Turtle(world1);

// declare and initialize t2

Turtle t2 = new Turtle(world1);
```

Constructor Signatures



Formal and Actual Parameters



Date(2005,9,1) - This is **call by value** which means that copies of the actual parameter values are passed to the constructor. These values are used to initialize the object's attributes.

```
public class Date
{
  /** instance variables for attributes */
  private int year;
  private int month;
  private int day;

  /** a constructor to initialize the attributes
      to today's date  */
  public Date()
  { /* Implementation not shown*/ }

  /** a constructor to initialize the attributes
     for a Date with the given parameters */
  public Date(int year, int month, int day)
  { /* Implementation not shown*/ }
}
```

Attributes

Constructors

Student

```
public class Student {
    public String name;
    public int gradeLevel;
    public double gpa;

    public Student(String n, int gl, double g) {
        name = n;
        gradelevel = gl;
        gpa = g;
    }
}
```

```
// It's a good idea to use a
single
// letter from the fields you are
// initializing in the
constructor.
// It keeps things simple.
```

What would you add to this constructor?
Can you think of some situations where you would want to leave out some of the fields?

AP Example 1

```
public class Cat
{
    private String color;
    private String breed;
    private boolean isHungry;
    public Cat()
    {
        color = "unknown";
        breed = "unknown";
        isHungry = false;
    }
    public Cat(String c, String b, boolean h)
    {
        color = c;
        breed = b;
        isHungry = h;
    }
}
```

I. `Cat a = new Cat();`
II. `Cat b = new Cat("Shorthair", true);`
III.
```
String color = "orange";
boolean hungry = false;
Cat c = new Cat(color, "Tabby", hungry);
```

AP Example 2

```
public class Movie
{
    private String title;
    private String director;
    private double rating;
    private boolean inTheaters;
    public Movie(String t, String d, double r)
    {
        title = t;
        director = d;
        rating = r;
        inTheaters = false;}
    public Movie(String t)
    {
        title = t;
        director = "unknown";
        rating = 0.0;
        inTheaters = false;
    }
}
```

Which of the following code segments will construct a Movie object m with a title of “Lion King” and rating of 8.0?

A. Movie m = new Movie(8.0, "Lion King");

B. Movie m = Movie("Lion King", 8.0);

C. Movie m = new Movie();

D. Movie m = new Movie("Lion King", "Disney", 8.0);

E. Movie m = new Movie("Lion King");