Create new Object:constructor

Dog myDog=new Dog()

Dog yourDog=new Dog("Anny")

Dog herDog=new Dog("Mim","male")

Dog hisDog=new Dog("Lili","female",2)

Object Basics
The Life Cycle of an Object

- Create a object
- Destroy a object

object Basics
The Life Cycle of an Object

- A typical Java program creates **many objects**, which interact with one another by sending each other **messages**. Through these **object interactions**, a Java program can implement a GUI, run an animation, or send and receive information over a network. Once an object has completed the work for which it was created, it is **garbage-collected** and its resources are recycled for use by other objects.

Create new Object:constructor

- The constructors are special methods in a java class. We use constructors to create a object

  1)A concrete class can have more than one constructor. If you don't define a constructor in a concrete class, It will get a default constructor automatically.The default constructor can not take parameter.

  2) The constructor name is same as class name.The constructor returns a object. When you write constructor, you do not need define return type

Create new Object:constructor

- public class Dogx extends Pet{
- public Dogx() {
- }
- public Dogx(String myName) {
- name=myName;
- }
- public Dogx(String myName,String myGender){
- name=myName;
- gender=myGender;
- }
- public Dogx(String myName,String myGender,int myAge){
- name=myName;
- gender=myGender;
- age=myAge;
- }

Create new Object:constructor

- Dogx myDog=new Dogx()
- Dogx yourDog=new Dogx("Anny")
- Dogx herDog=new Dogx("Mim","male")
- Dogx hisDog=new Dogx("Lili","female",2)

Create new Object:constructor

- **Using Objects**
- Once you've created an object, you probably want to use it for something. You may need information from it, want to change its state, or have it perform some action. Objects give you two ways to do these things:

1. Manipulate or inspect its variables.
2. Call its methods.

Create new Object:constructor

- **Referencing an Object's Variables**
- objectReference.variableName
- **Calling an Object's Methods**
- objectReference.methodName(argumentList);
- or objectReference.methodName();

Create new Object:constructor

- **Cleaning Up Unused Objects**
- *garbage collection*.
- An object is eligible for garbage collection when there are no more references to that object.
- *The Garbage Collector:*
- The garbage collector does garbage collection automatically .

Create new Object:constructor

- The garbage collector automatically cleans up unused objects. An object is unused if the program holds no more references to it. You can explicitly drop a reference by setting the variable holding the reference to null.