Animal Class Hierarchy

```
           Animal
 - String name
 + Animal()
 + Animal(String name)
 + void move()
```

```
        Dog

 + Dog()
 + Dog(String name)
 + void move()
 + void bark()
```

```
        Bird

 + Bird()
 + Bird(String name)
 + void move()
```
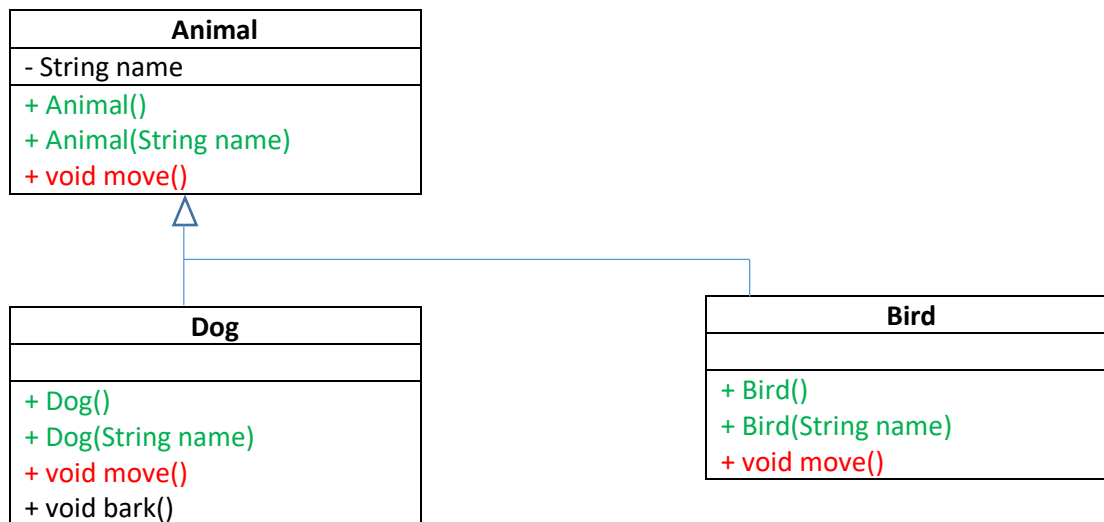
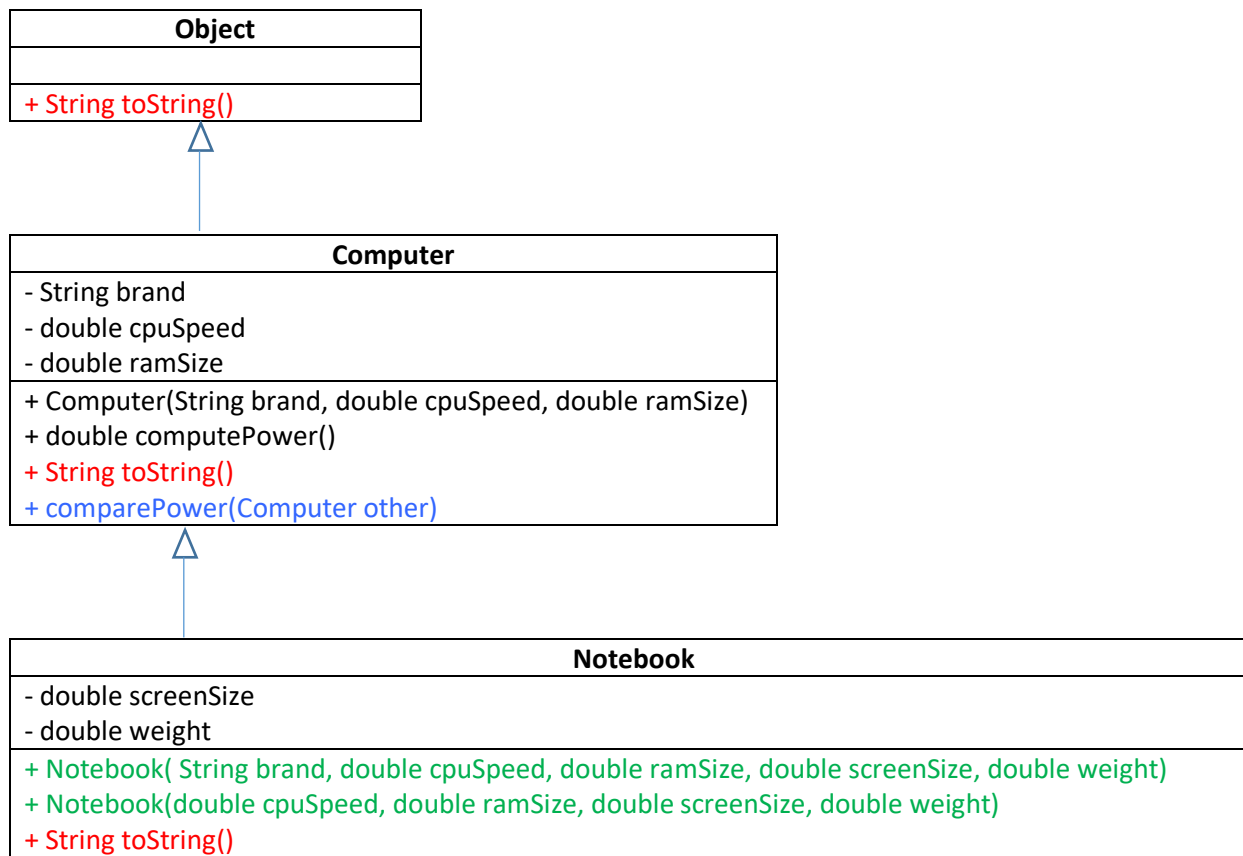The move() method shows method overriding in Java.
Multiple versions of constructors show method overloading in Java.
+ means public; **-** means private

Computer Class Hierarchy

```
          Object

 + String toString()
```

```
                     Computer
 - String brand
 - double cpuSpeed
 - double ramSize
 + Computer(String brand, double cpuSpeed, double ramSize)
 + double computePower()
 + String toString()
 + comparePower(Computer other)
```

```
                                  Notebook
 - double screenSize
 - double weight
 + Notebook( String brand, double cpuSpeed, double ramSize, double screenSize, double weight)
 + Notebook(double cpuSpeed, double ramSize, double screenSize, double weight)
 + String toString()
```

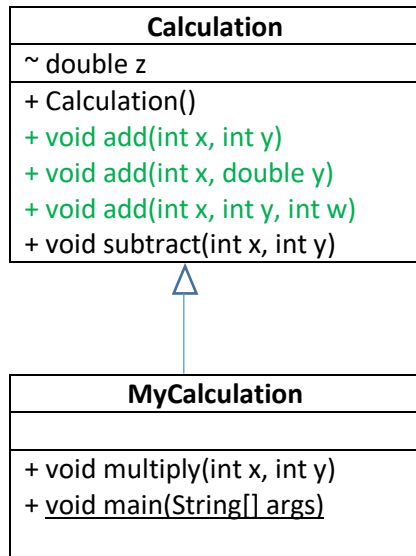The toString() method shows method overriding in Java. Class Object is shown because of the toString() method.
Multiple versions of constructors show method overloading in Java.
The comparePower(..) method shows polymorphism in Java.

With polymorphism, we don't have to write overloaded comparePower(...) method for different types of computers.

Calculation Class Hierarchy

Calculation
~ double z
+ Calculation() + void add(int x, int y) + void add(int x, double y) + void add(int x, int y, int w) + void subtract(int x, int y)

△
|

MyCalculation
+ void multiply(int x, int y) + void main(String[] args)

Multiple versions of add(...) methods show method overloading in Java.

+ means public;   **-** means private;

~ means default access level (which is accessible in the same package)

Underlined means static method or static data field