Variables as Remote Control

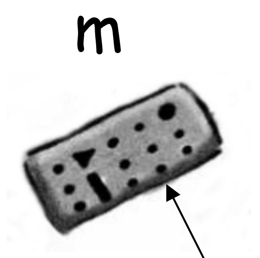
A useful memory aid
used in *Head First Java*

A Variable is a Reference

$\underbrace{\text{Person p}}_{}$ = $\underbrace{\text{new Person( )}}_{}$

a *reference* for sending commands to object

object



buttons on remote control are methods

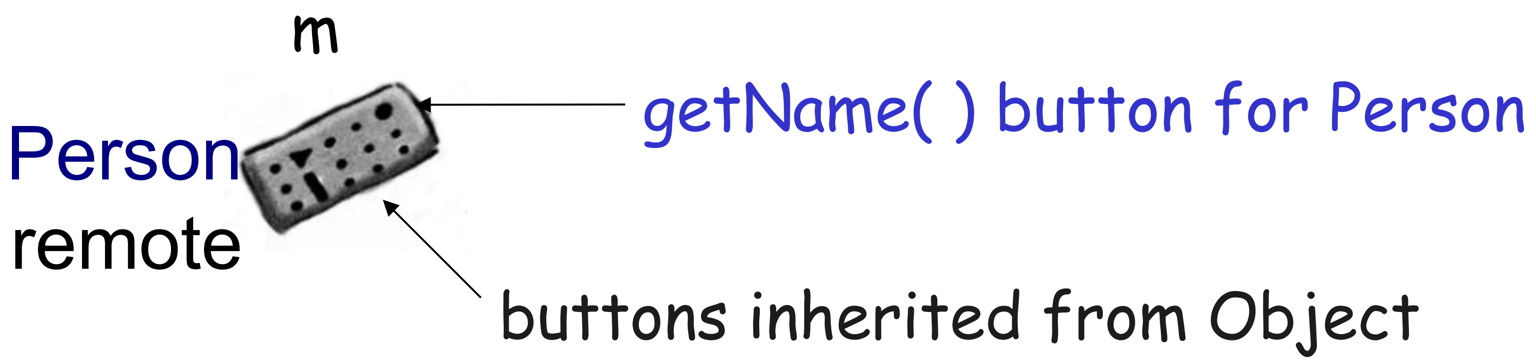
Person

```
#clone()
equals(Object)
finalize()
getClass()
hashCode()
toString()
wait()
getName(): Str
```

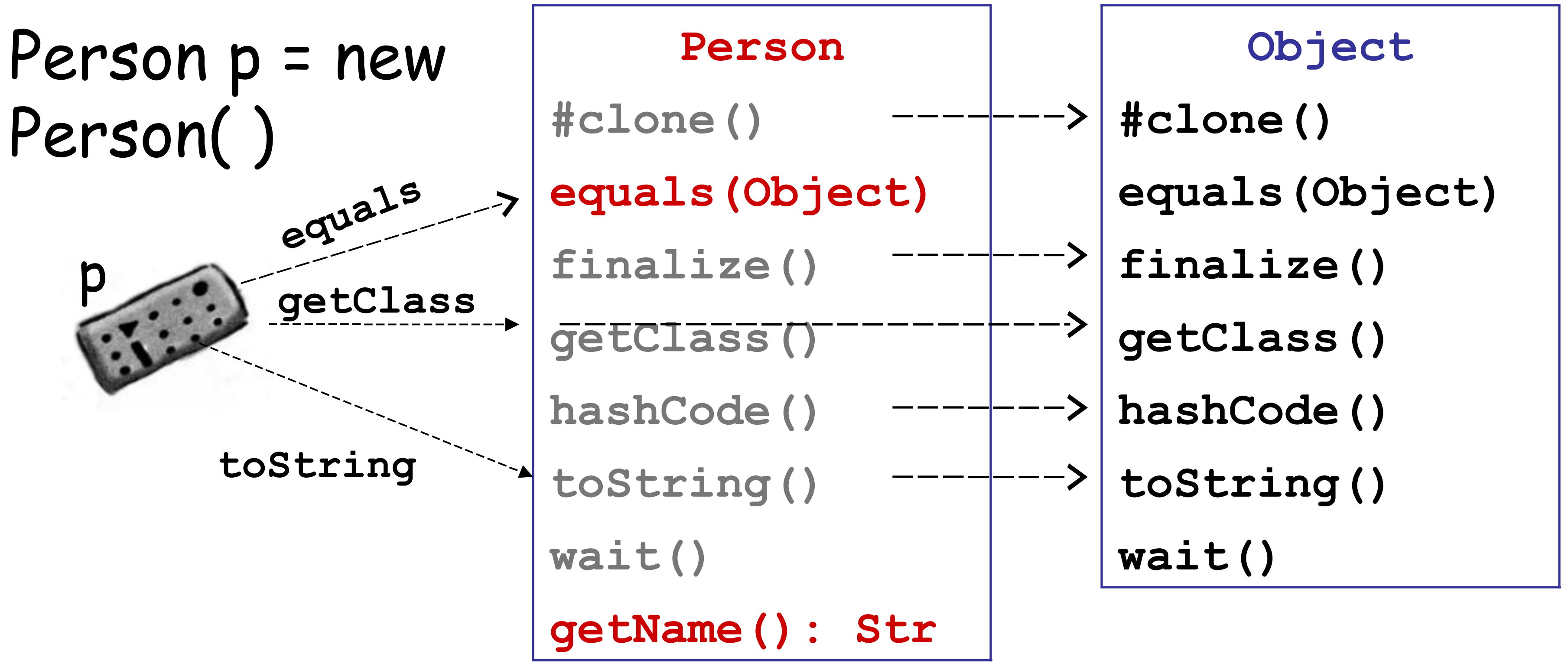
The Compiler decides what Buttons

$\underbrace{\text{Person p =}}$ xxxxxxxxxxxxxx

Compiler uses the *declared type* of variable to decide what buttons the remote control has.

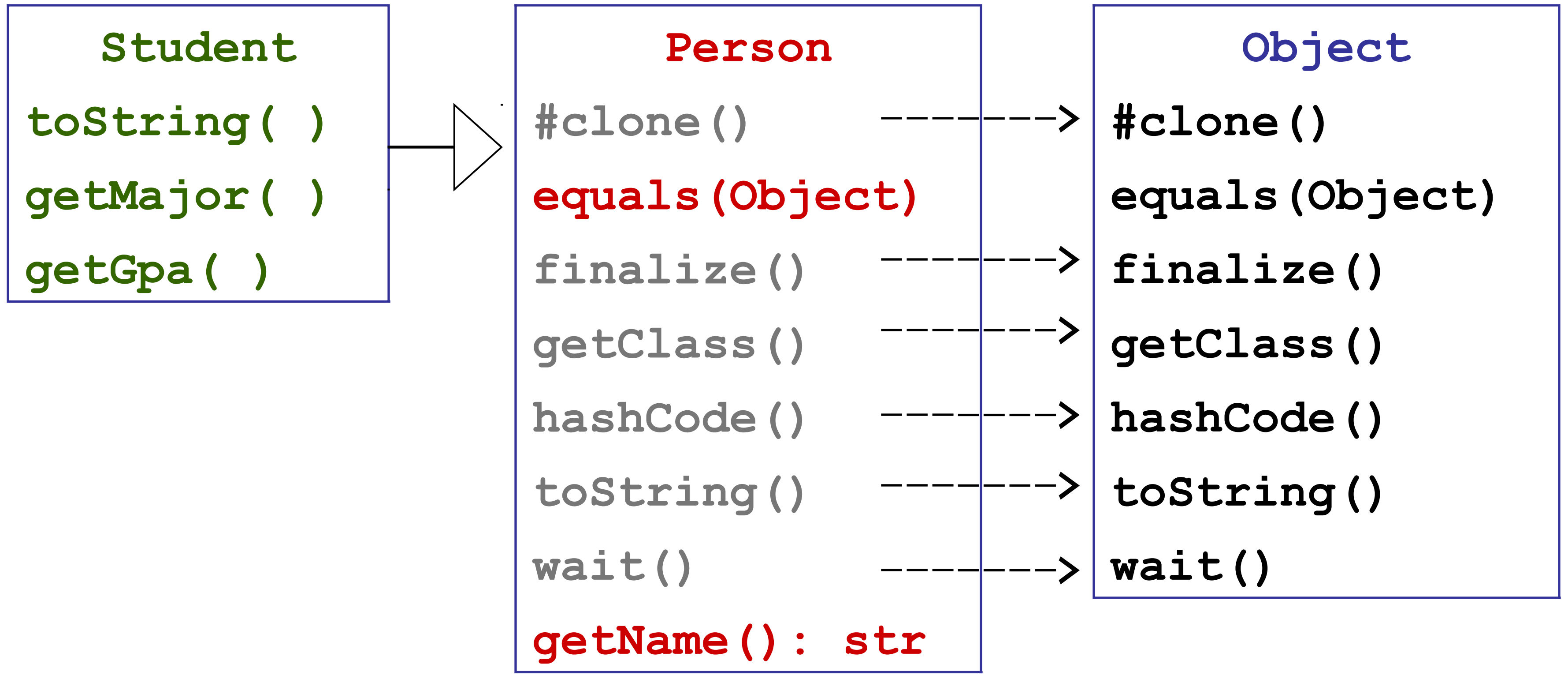


Invoking Methods



At runtime, JVM invokes method on actual object type. If a class *overrides* a method, the override is used.

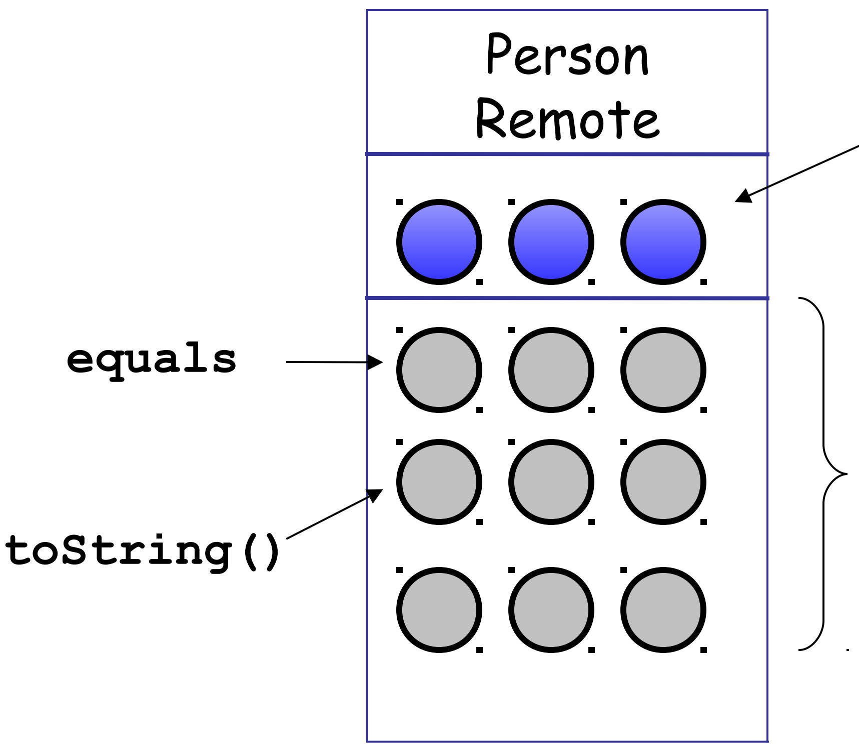
Student - Person - Object



```
class Student extends Person {
```

What Buttons Do You Have?

Person x = new Student( );

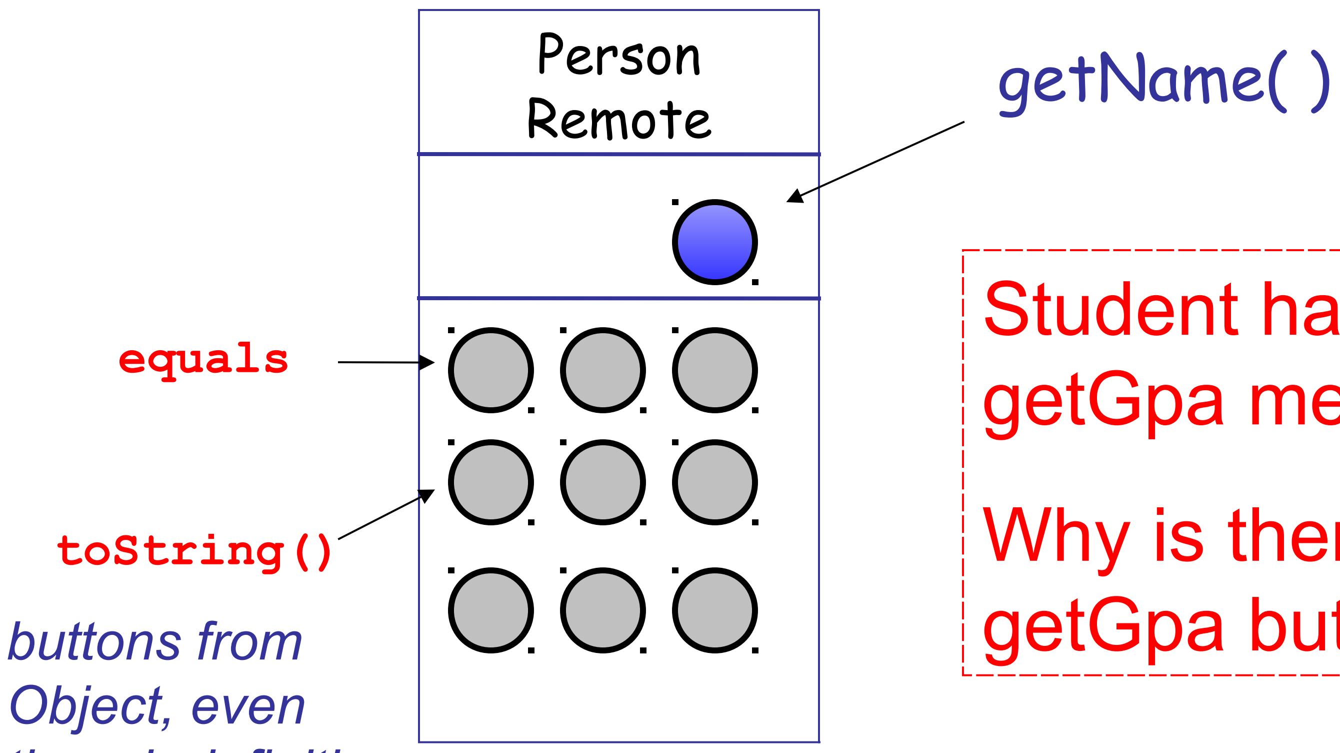


What buttons go here?

Buttons inherited from Object. Every remote control has these.

What Buttons Do You Have?

Person p = new Student( );

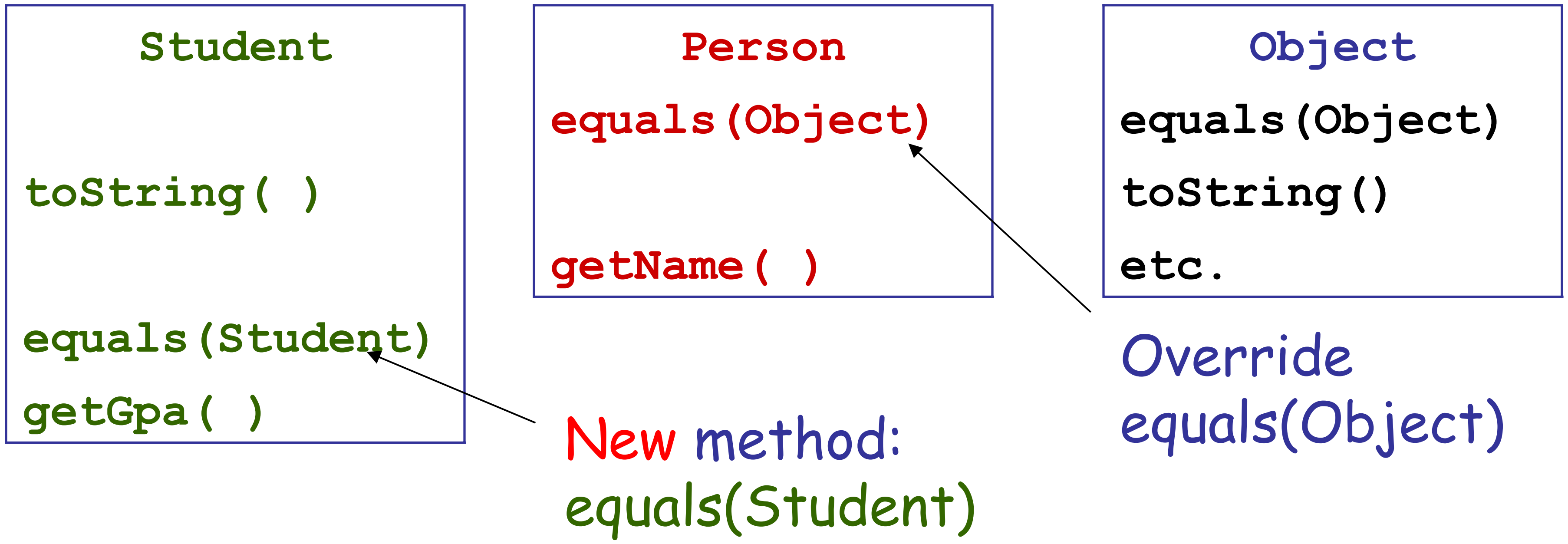


buttons from Object, even though definition is changed.

Student has a getGpa method.

Why is there no getGpa button?

Method Signature includes Parameter



```
class Student extends Person {
    public boolean equals( Student s ) // BAD IDEA
    public String toString( )
```

Which equals( ) is called?

```
Student

toString( )

equals(Student)
```

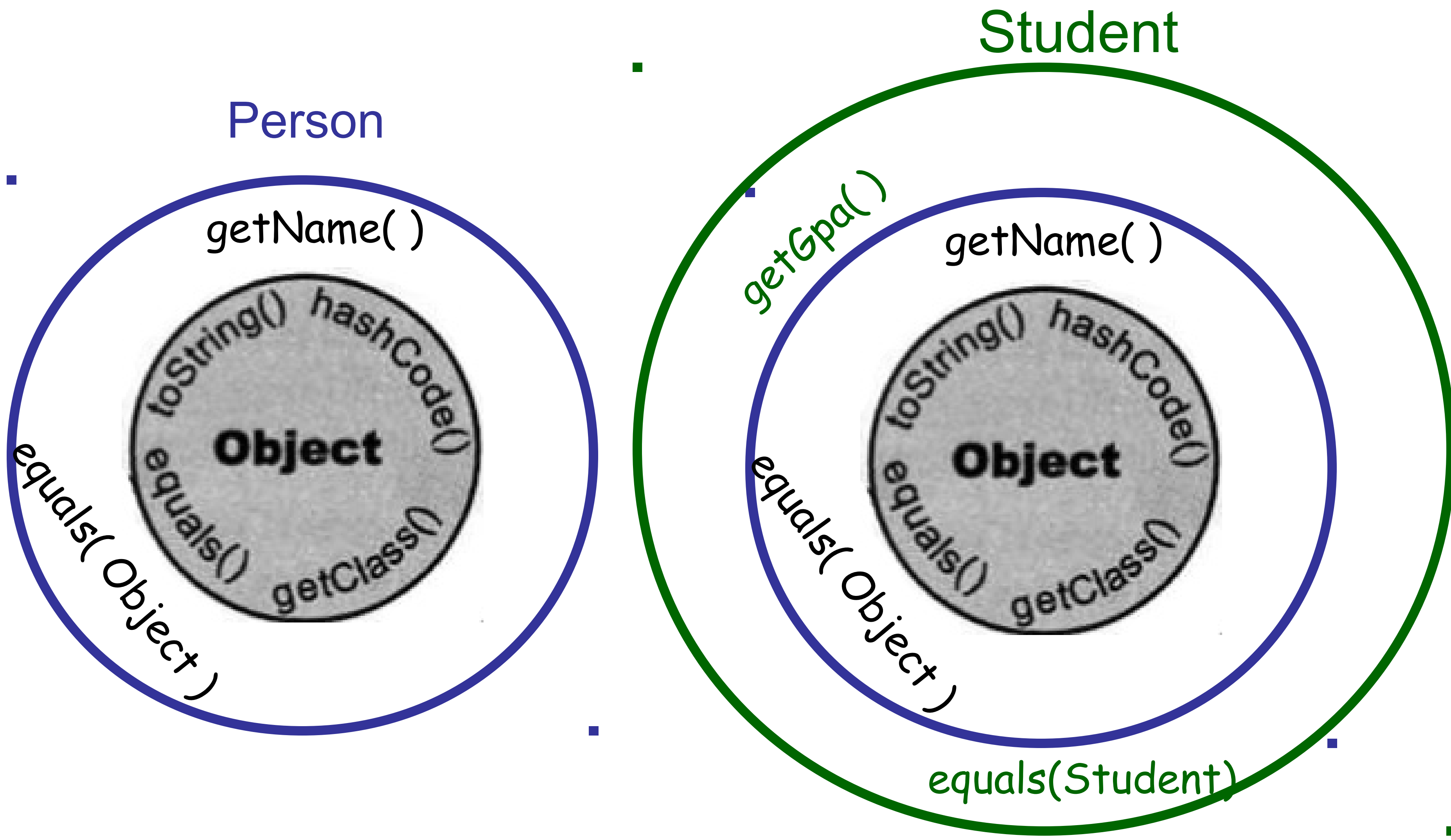
```
Person
equals(Object)

getValue( )
```

```
Object
equals(Object)
toString()
etc.
```

```
Student a = new Student();
Person b = new Student( );
//1.
b.equals( a )
//2.
a.equals( b )
```

Draw the remote control !

Another view of Inheritance

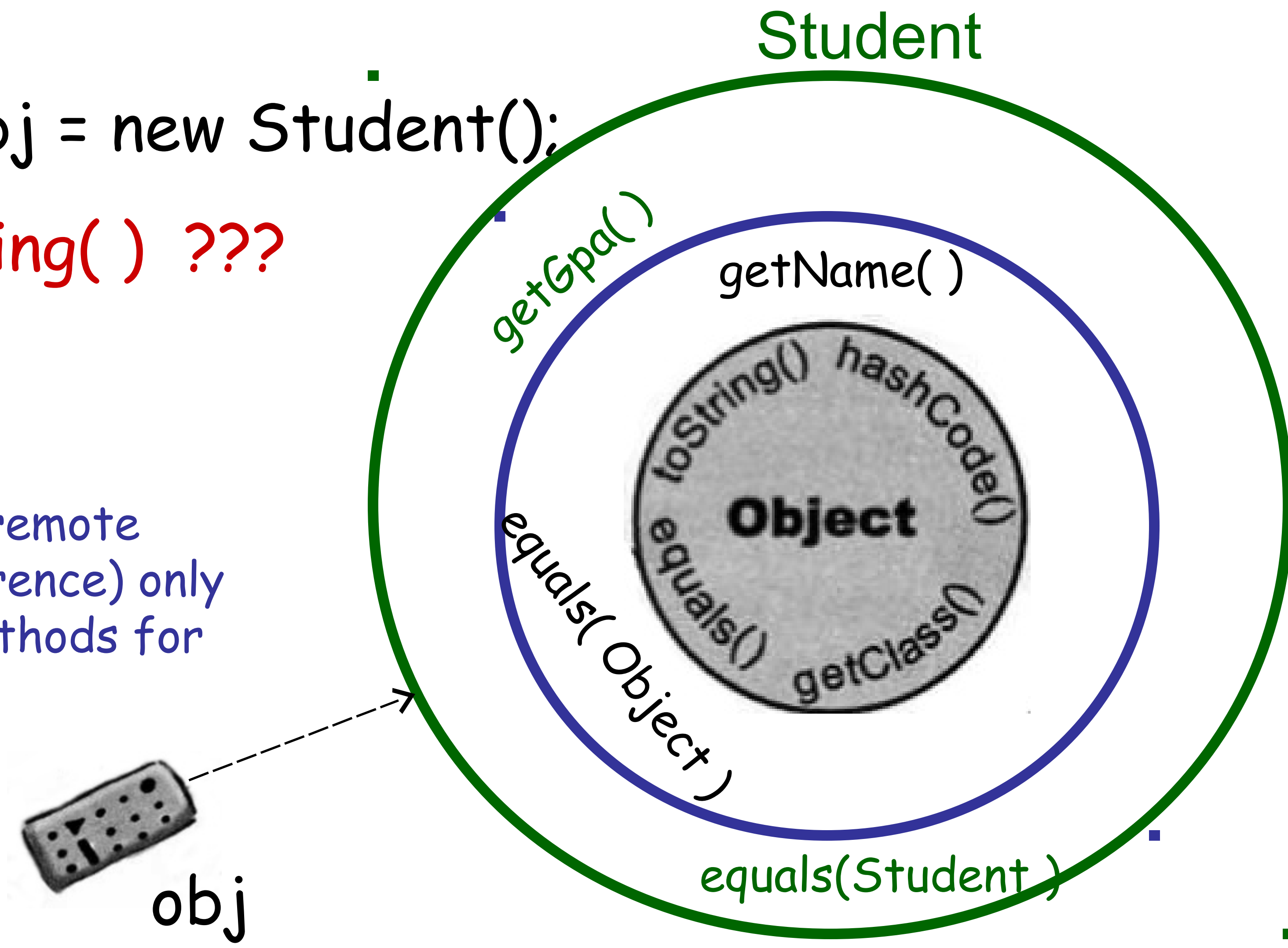


Object References

```
Object obj = new Student();
```

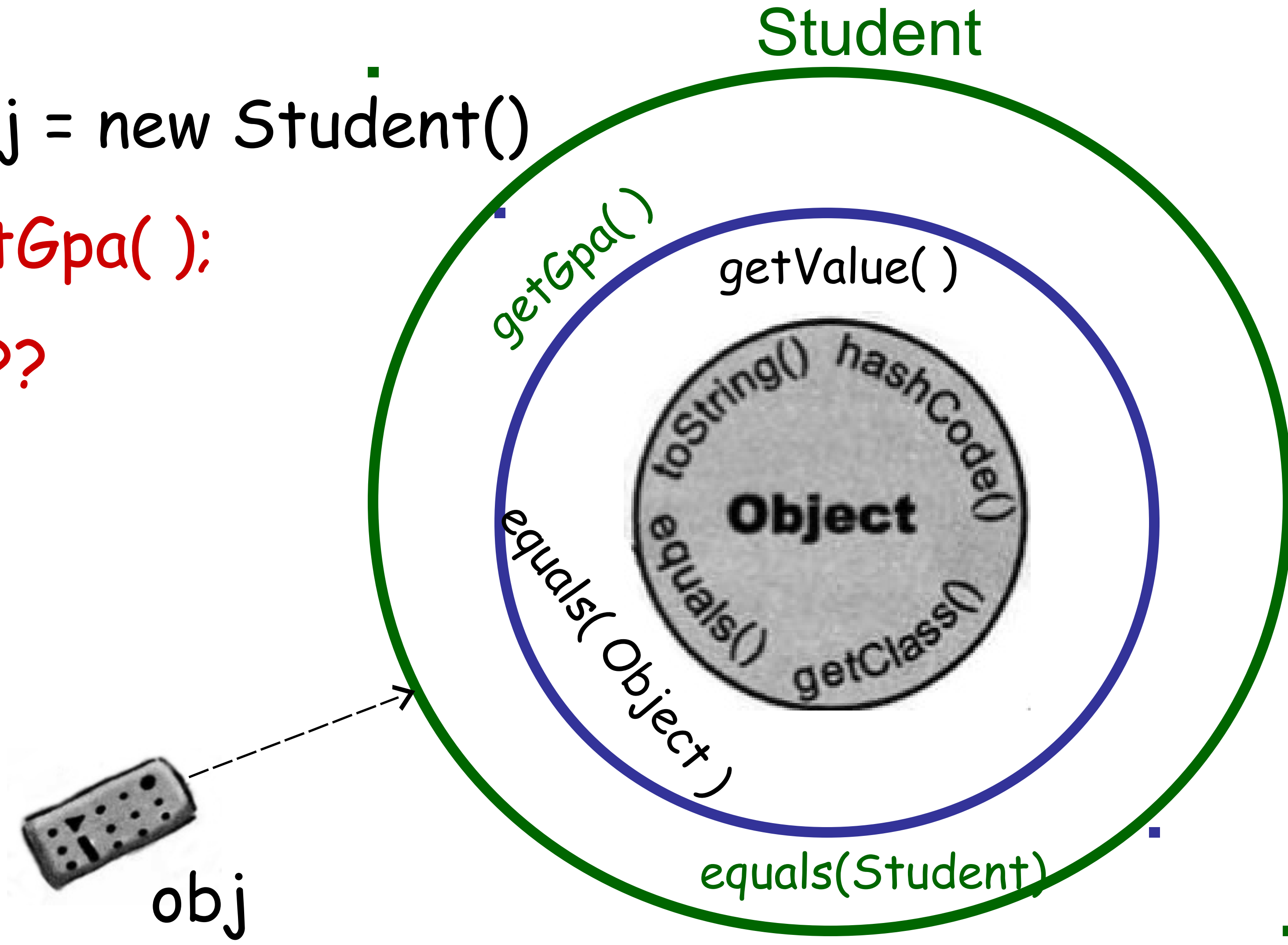
```
obj.toString( ) ???
```

An "Object" remote control (reference) only knows the methods for object.



How to Access the *Real* object

```
Object obj = new Student()
x = obj.getGpa( );
??? how ???
```


Student
getGpa( )
getValue( )
toString()
hashCode()
Object
equals()
getClass()
equals(Object)
equals(Student)
obj