DigiPen INSTITUTE OF TECHNOLOGY

CS 175
Advanced Scripting

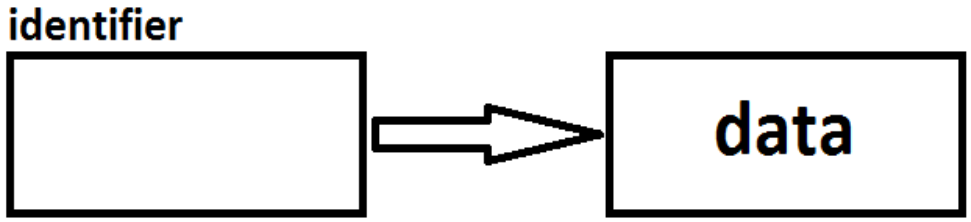
References

What are references?

What are refernces?

• A reference data type (a.k.a object) is only a means to access the data.

ReferenceType identifier = data;



Note: In AS3, all complex types are references.

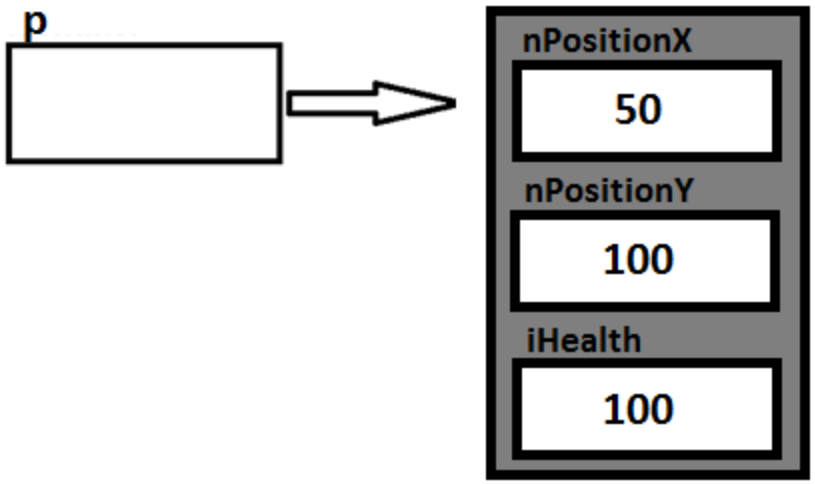
The Player Class

```
1  package
2  {
3      public class Player
4      {
5          public var nPositionX:Number;
6          public var nPositionY:Number;
7          public var iHealth:int;
8
9          public function Player(nPositionX_:Number, nPositionY_:Number)
10         {
11             nPositionX = nPositionX_;
12             nPositionY = nPositionY_;
13             iHealth = 100;
14         }
15
16         public function TraceInfo():void
17         {
18             trace("Position X: "+ nPositionX);
19             trace("Position Y: "+ nPositionY);
20             trace("Health: "+ iHealth);
21         }
22     }
23 }
```

Using the Player class

```
1  var p:Player = new Player(50.0, 100.0);
2  p.TraceInfo();
```

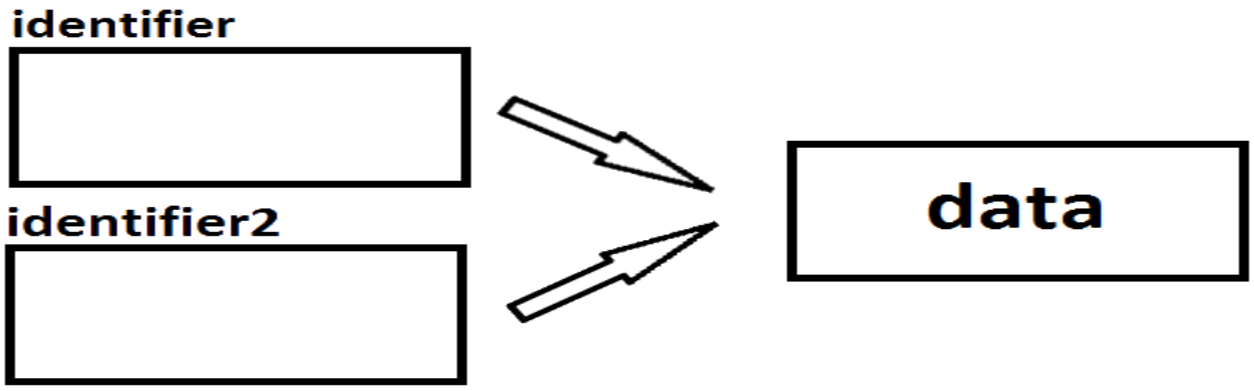
p is a reference to a Player object which means that through p we can access the Player memory/data that was allocated by the new operator and initialized by the Player class constructor.



Referencing the same data

- Suppose the following declaration is made:

ReferenceType identifier = data;
ReferenceType identifier2 = identifier;



Both variables are referencing the same data, so if any of them change the data then it is changed for both.

Referencing the same data

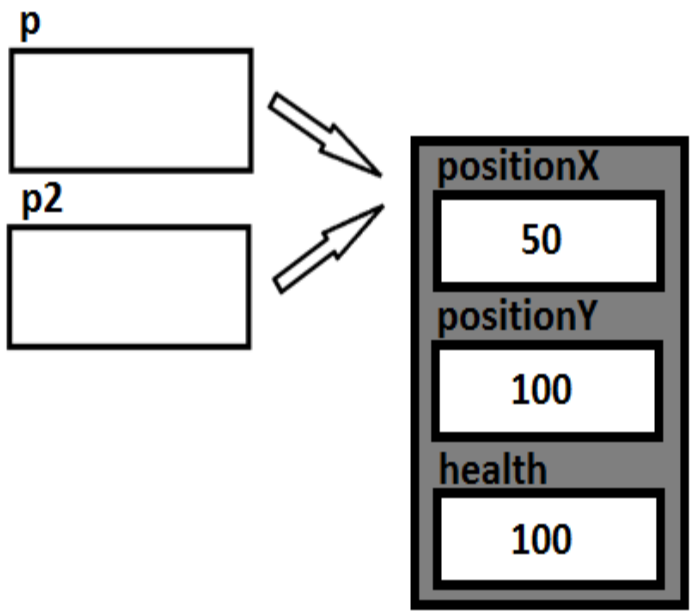
```
1  var p:Player = new Player(50.0, 100.0);
2  trace("p content:");
3  p.TraceInfo();
4  trace();
5
6  var p2:Player = p;
7  trace("p2 content:");
8  p2.TraceInfo();
9  trace();
10
11 p.nPositionX = 50;
12 p.nPositionY = 100;
13
14 trace("p content:");
15 p.TraceInfo();
16 trace();
17
18 trace("p2 content:");
19 p2.TraceInfo();
20 trace();
```

p content:
Position X: 50
Position Y: 100
Health: 100

p2 content:
Position X: 50
Position Y: 100
Health: 100

p content:
Position X: 50
Position Y: 100
Health: 100

p2 content:
Position X: 50
Position Y: 100
Health: 100

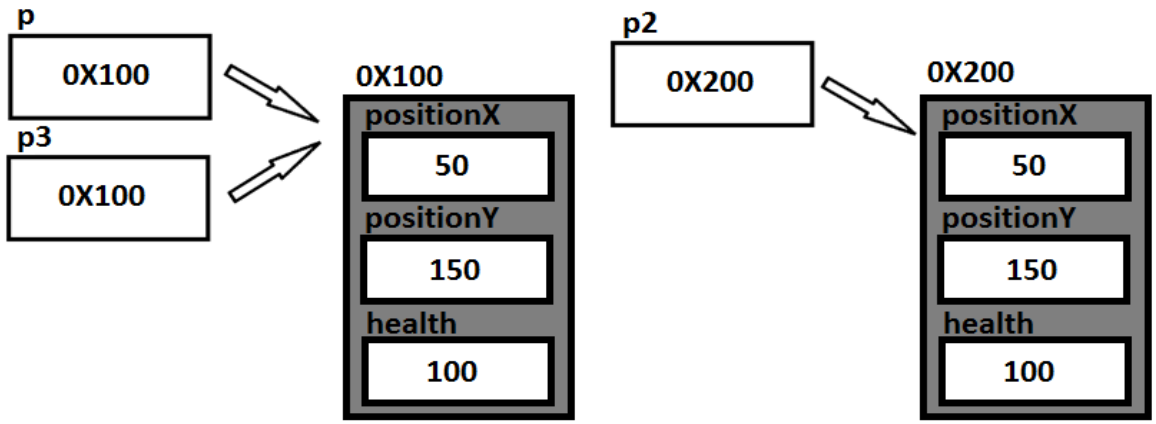


Comparing References

```
1  var p:Player = new Player(50.0, 100.0);
2  var p2:Player = new Player(50.0, 100.0);
3  var p3:Player = p;
4
5  trace("p == p2: " + (p == p2));
6  trace("p == p3: " + (p == p3));
```

Output:
p == p2: false
p == p3: true

Even though all three Player references have the same values in their properties, when comparing two references of the same type we are actually comparing the addresses they are pointing to.



p == p2 is actually doing 0X100 == 0X200 which results to a false.
p == p3 is actually doing 0X100 == 0X100 which results to a true.

The End ☺