This keyword in Java

Why and how we use it

Produced by:

Dr. Siobhán Drohan
Ms. Mairead Meagher
Ms. Siobhan Roche

Ollscoil Teicneolaíochta an Oirdheiscirt
South East Technological University

Department of Computing and Mathematics
http://www.setu.ie/

this keyword

- Examine a class called Spot – it contains many fields e.g.:
  - xCoord, yCoord, diameter

```
class Spot{
  float xCoord, yCoord;
  float diameter;
  int red, green, blue;

  Spot(float xPos, float yPos, float diamtr)
  {
     xCoord = xPos;
     yCoord = yPos;
     diameter = diamtr;
  }
```

this keyword

- One of the Spot constructors takes three parameters:
  - xPos, yPos, diamtr

```
class Spot{
  float xCoord, yCoord;
  float diameter;
  int red, green, blue;

  Spot (float xPos, float yPos, float diamtr)
  {
     xCoord = xPos;
     yCoord = yPos;
     diameter = diamtr;
  }
```

this keyword

- It would be nice to name the parameters passed into the Spot constructor the **same names as the instance fields**.

- This is called **name overloading**.

- But how will Java know which variable we are referring to?

```
class Spot{
  float xCoord, yCoord;
  float diameter;
  int red, green, blue;

  Spot(float xPos, float yPos, float diamtr)
  {
     xCoord = xPos;
     yCoord = yPos;
     diameter = diamtr;
  }
```

this keyword

We can use the this keyword to distinguish between them

```
class Spot{
  float xCoord, yCoord;
  float diameter;
  int red, green, blue;

 Spot(float xCoord, float yCoord, float diameter)
 {
   this.xCoord    = xCoord;
   this.yCoord    = yCoord;
   this.diameter  = diameter;
 }
```

this keyword

this refers to the current object fields.

```
class Spot{
 float xCoord, yCoord;
 float diameter;
 int red, green, blue;

Spot(float xCoord, float yCoord, float diameter)
{
  this.xCoord = xCoord;
  this.yCoord = yCoord;
  this.diameter = diameter;
 }
```

this keyword

These are local fields that are destroyed as soon as the Spot constructor finishes executing.

```
class Spot{
  float xCoord, yCoord;
  float diameter;
  int red, green, blue;

  Spot(float xCoord, float yCoord, float diameter)
  {
    this.xCoord = xCoord;
    this.yCoord = yCoord;
    this.diameter = diameter;
  }
```

this keyword – other examples

```
void colour (int red, int green, int blue)
 {
   this.red = red;
   this.green = green;
   this.blue = blue;
   fill (red, green, blue);
 }

 void colour (int gray){
   this.gray = gray;
   fill (this.gray);
 }
```

To clairfy, in the statement:

this.x = **x;**

Where **this.x** refers to the object's property / field

and **x** on it's own
is the parameter passed in to the method

substitute x for any property/field

This describes **name overloading**

Questions?

References

- Reas, C. & Fry, B. (2014) Processing – A Programming Handbook for Visual Designers and Artists, 2nd Edition, MIT Press, London.