

# Abstraction

## Classes and Methods

---

Produced      Dr. Siobhán Drohan  
by:            Mairead Meagher



Waterford Institute *of* Technology  
INSTITIÚID TEICNEOLAÍOCHTA PHORT LÁIRGE

Department of Computing and Mathematics  
<http://www.wit.ie/>

# Topic List

---

- Abstract vs Concrete
- Abstract Methods and Classes
- Network-V4 (recap)
- Network-V5 (with abstraction)

# Abstract vs Concrete

---

- Abstract
  - Implementation delayed
    - abstract method has no code
    - cannot instantiate an abstract class (it has, by definition “unfinished” methods)
- Concrete
  - Ready to go.
  - Everything up to now has been concrete.

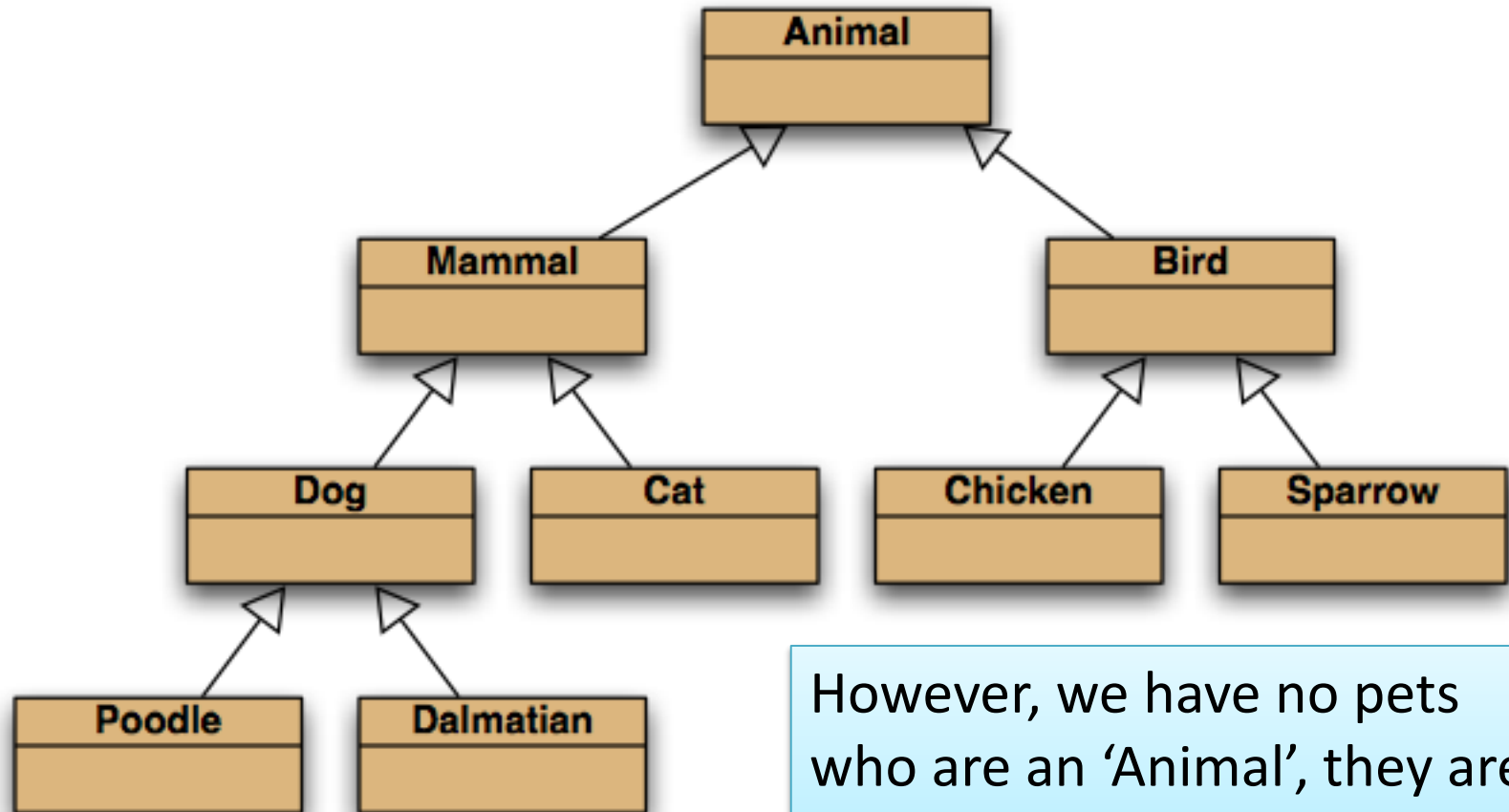
# Topic List

---

- Abstract vs Concrete
- Abstract Methods and Classes
- Network-V4 (recap)
- Network-V5 (with abstraction)

# Recap: Inheritance hierarchies

---



However, we have no pets who are an 'Animal', they are either a Dog, a Cat, a Chicken, etc.

# Abstract Methods

---

- Abstract methods have `abstract` in the signature.
- Abstract methods have no body.
  - ‘We promise to write this later. Every (concrete) subclass of this class will have this implemented in the subclass.’
- Abstract methods make the class abstract.
  - Think about why this is?

# Abstract Classes

---

- An abstract class is a class that contains zero or more abstract methods.
- An class that has an abstract method must be declared abstract.
- Abstract classes cannot be instantiated.
- Abstract classes function as a “base” for subclasses.
  - abstract classes can be subclassed.
- Concrete subclasses complete the implementation.

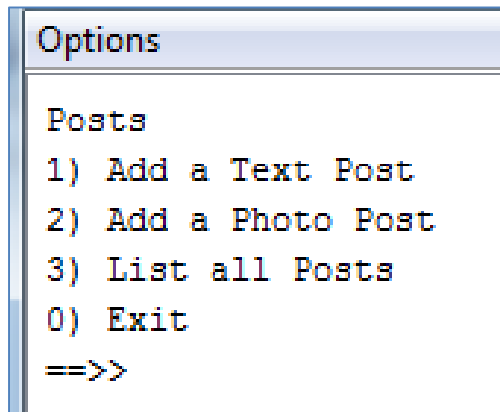
# Topic List

---

- Abstract vs Concrete
- Abstract Methods and Classes
- Network-V4 (recap)
- Network-V5 (with abstraction)

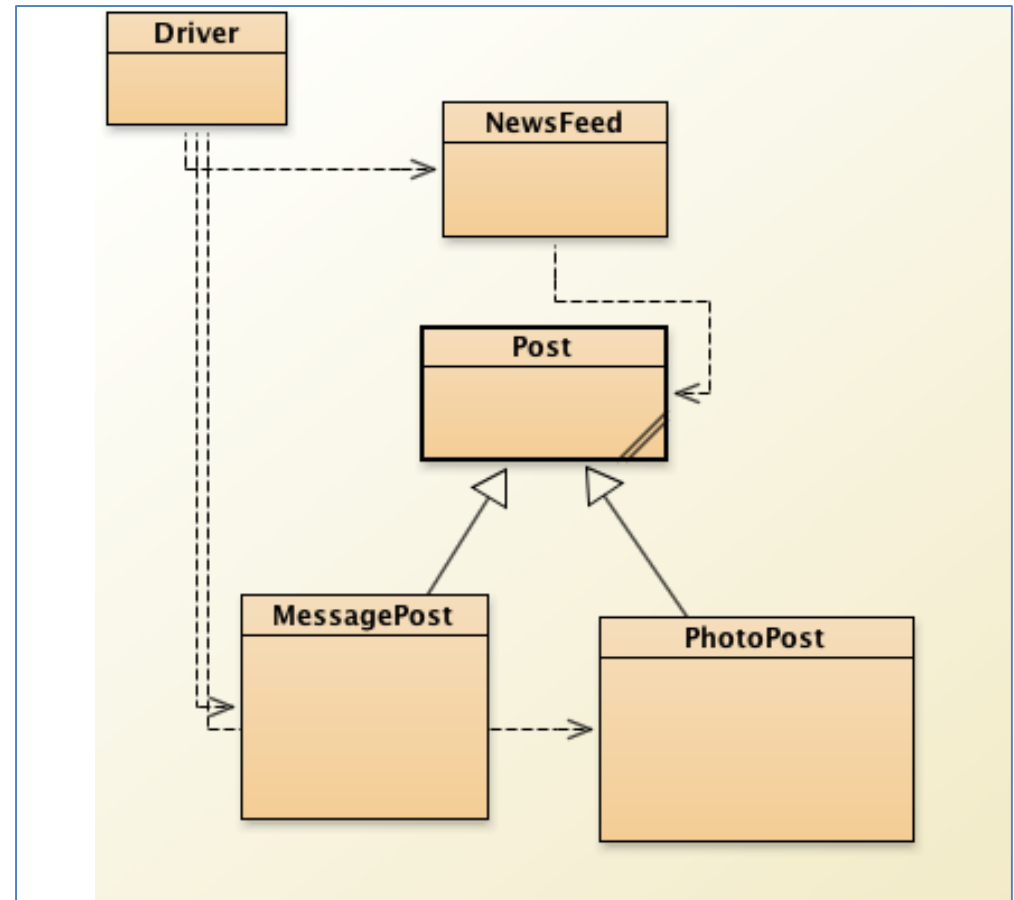


# Recap: Network-V4



Our news feed displays either MessagePost or PhotoPost objects.

We never create a Post object but our ArrayList is of Post.



# Topic List

---

- Abstract vs Concrete
- Abstract Methods and Classes
- Network-V4 (recap)
- Network-V5 (with abstraction)

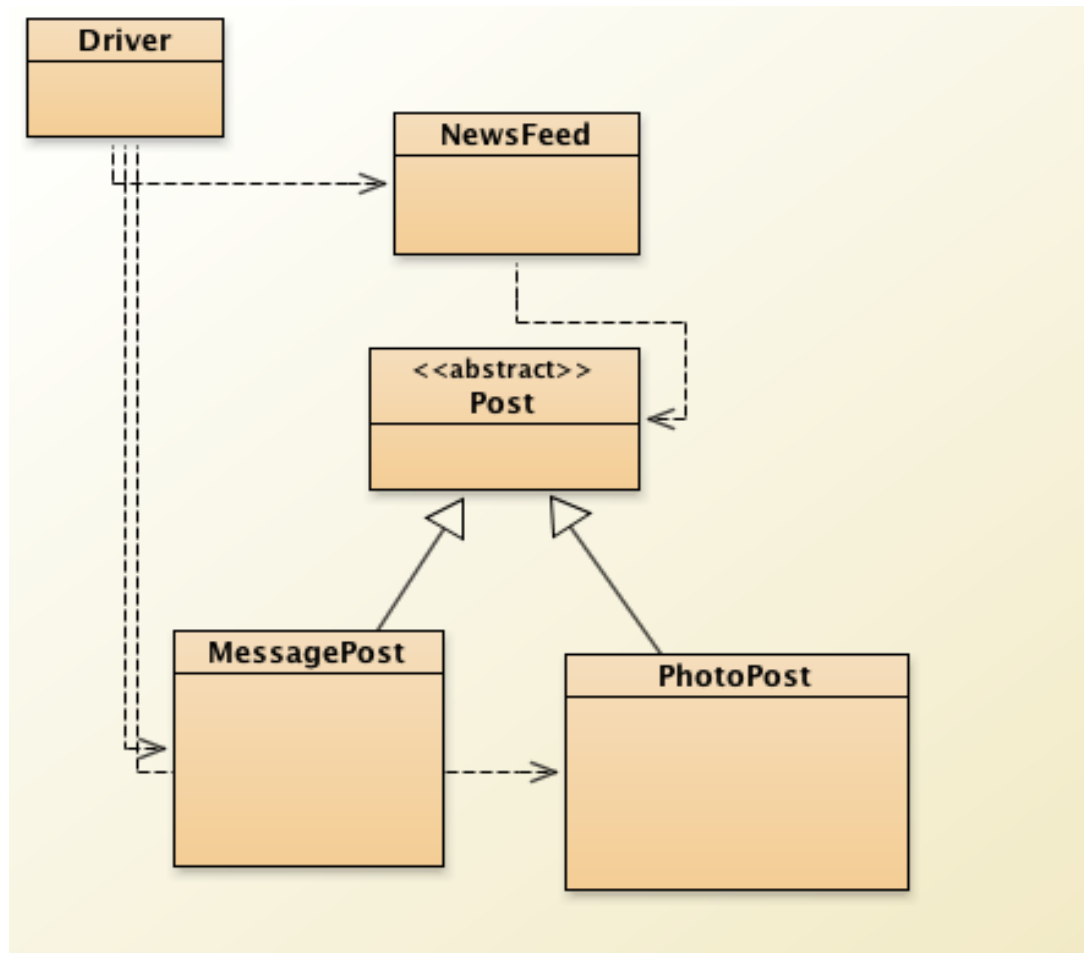
# Network-V5 (Post as an abstract class)

---

- We can never create a 'post' object
  - We cannot instantiate one.
- In Post, we define fields, methods that can be used later for all subclasses (using super)
  - e.g. display(), constructor..

# Network-V5 (Post as an abstract class)

---



# Syntax for abstract classes

---



```
public abstract class Post
{
    private String username; // username of the post's author
    private long timestamp;
    private int likes;
    private ArrayList<String> comments;
```

# displayExtract() as an abstract method

---

- If you wish all subclasses of a class to implement a particular method as part of its code, simply write an abstract method heading in superclass.
- Each subclass must have this method fully coded.

# displayExtract() as an abstract method

---

```
abstract String displayExtract();
```

**Post**

```
String displayExtract()  
{  
    return "Message extract "+ message.substring(0,10) + "....";  
}
```

**MessagePost**

```
String displayExtract()  
{  
    return "Photo caption: " + caption.substring(0,10) + "....";  
}
```

**PhotoPost**

# Interfaces Topic

---

- Next, we will look at interfaces which are used when you can see a 'multiple inheritance' in your class design.
- Multiple inheritance is not allowed in Java so we use interfaces instead.



**Any  
Questions?**

