QA\_consulting Pre-Assessment Day Pack

/\*\*

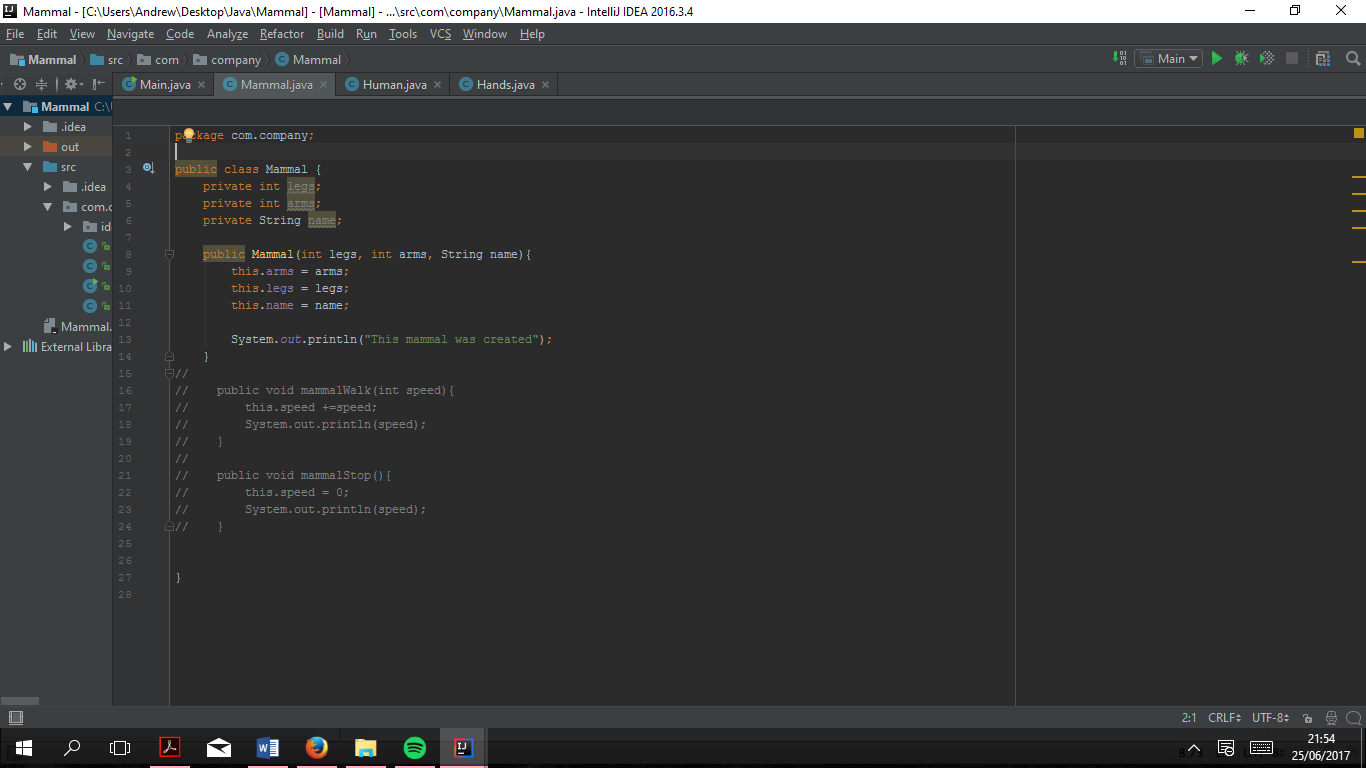
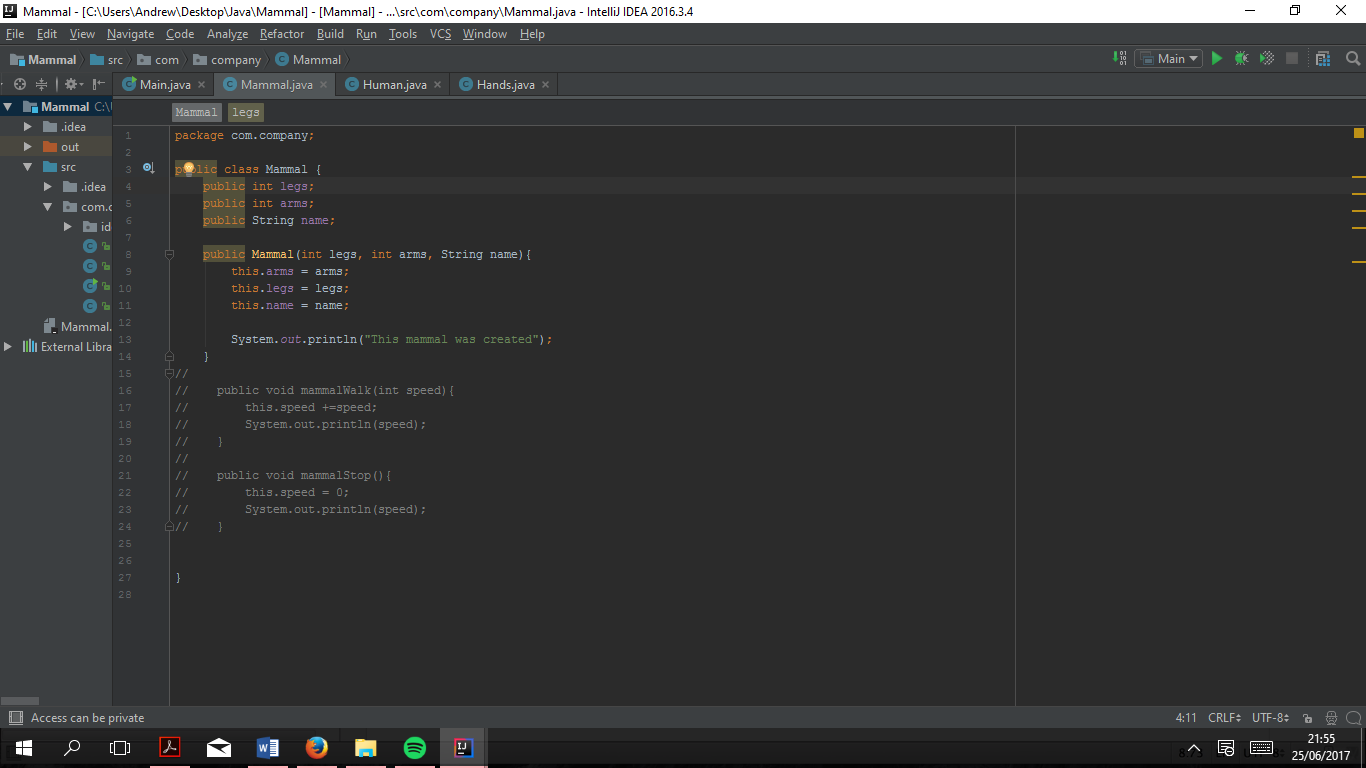
\* Created by Andrew Holmes on 23/06/2017

\*/

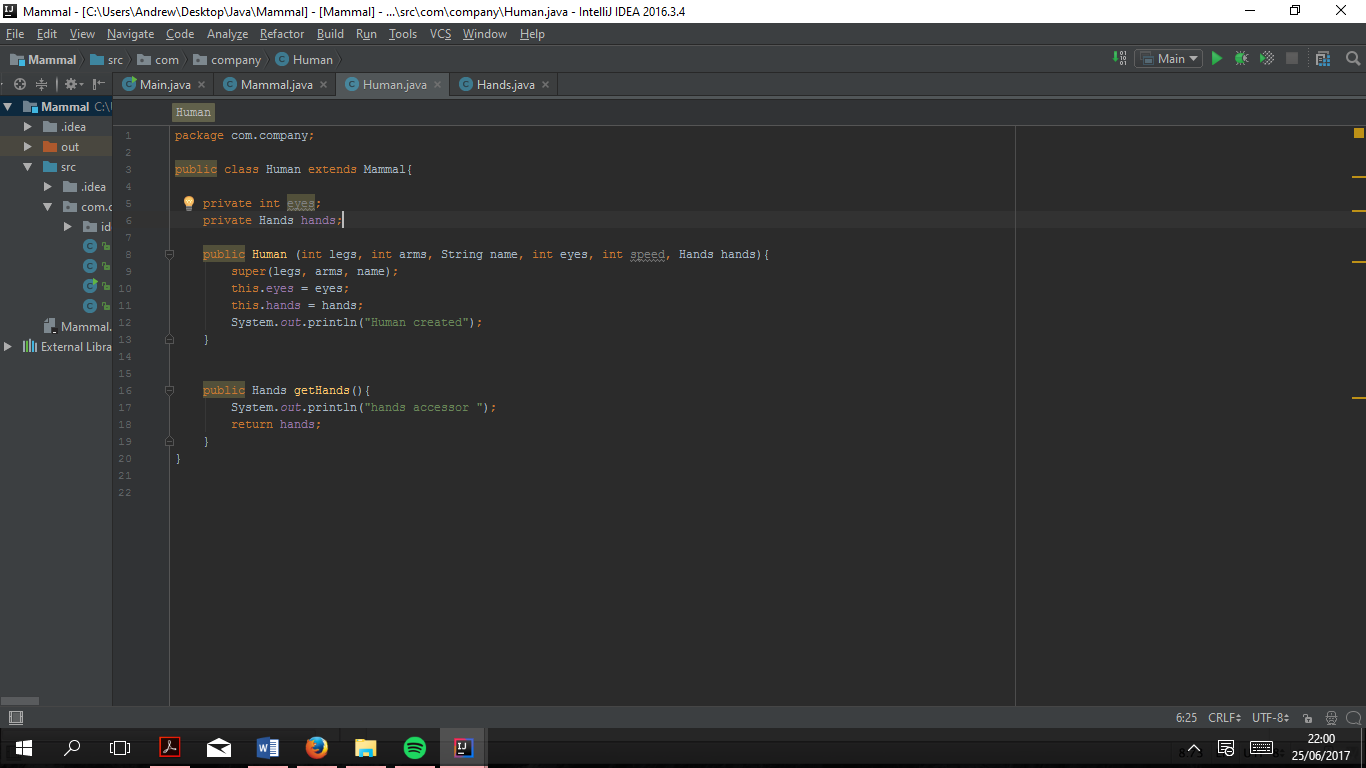
**Exercise One-**

**Encapsulation** - Restricts authorised access to classes that have have created, so outside classes cannot change the inner workings of the class. Encapsulation is implemented by making fields and methods private rather than public.

Without Encapsulation, using public keyword

With Encapsulation, using private keyword

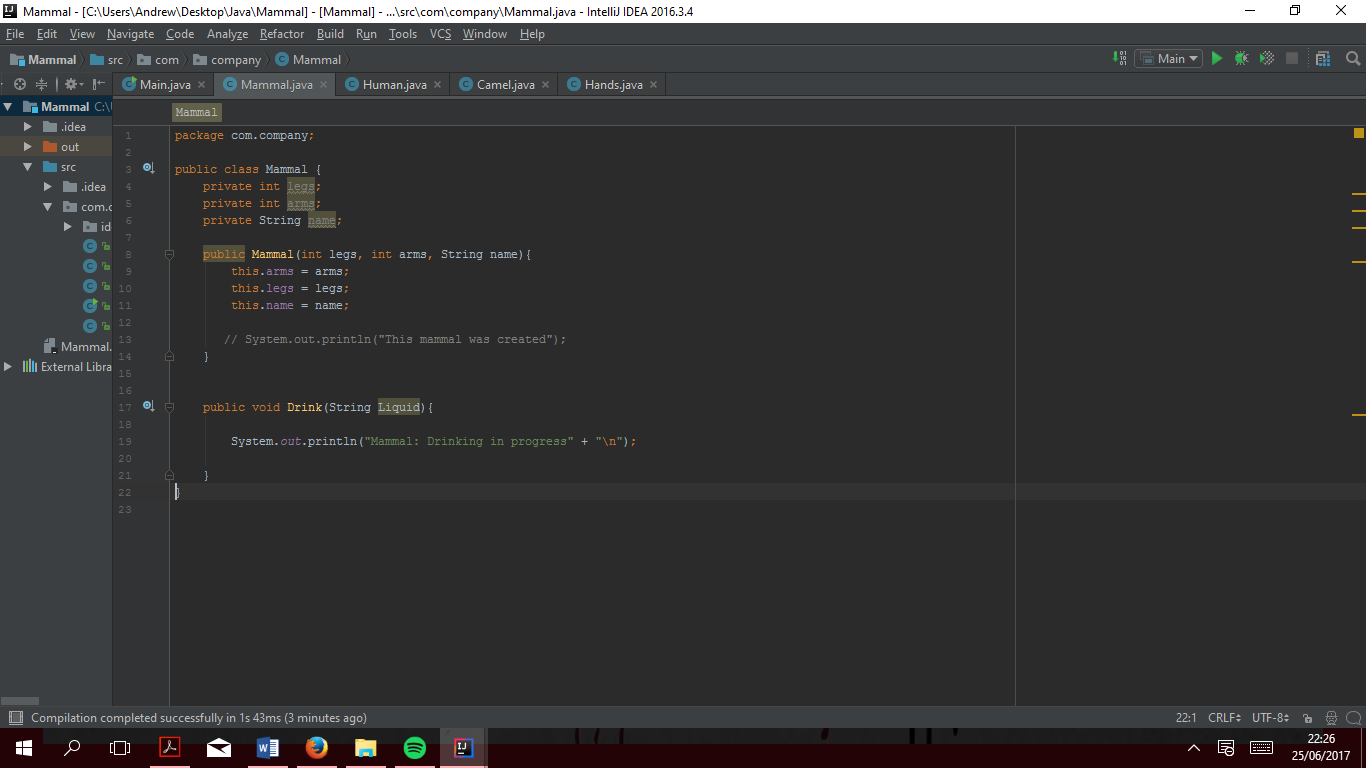
**Inheritance –** Classes inherit properties and behaviours from a super class. Implemented using the “Extends” keyword. This allows the sub-class to inherit from the super class. The keyword “Super” is used in the sub-class to call the constructor in the super class.



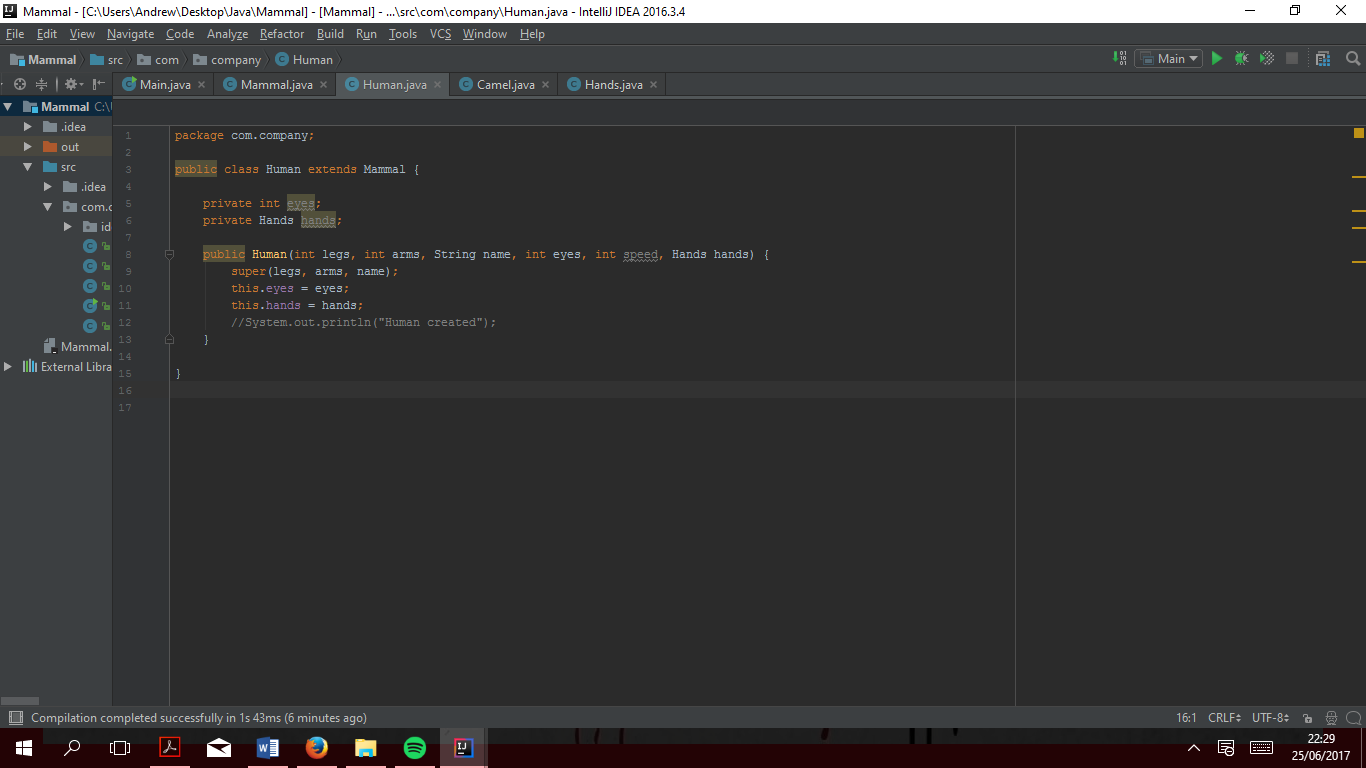
Use of keyword “extends” indicates that the Humans class is the sub-class of Mammal

Use of keyword “super” indicates which variables in the sub- class were inherited from the super class.

**Polymorphism –** Gives unique functionality to a class that has inherited from a base class. By using the keyword “@Override” sub-classes can take control away from the super class methods and make methods unique for itself.

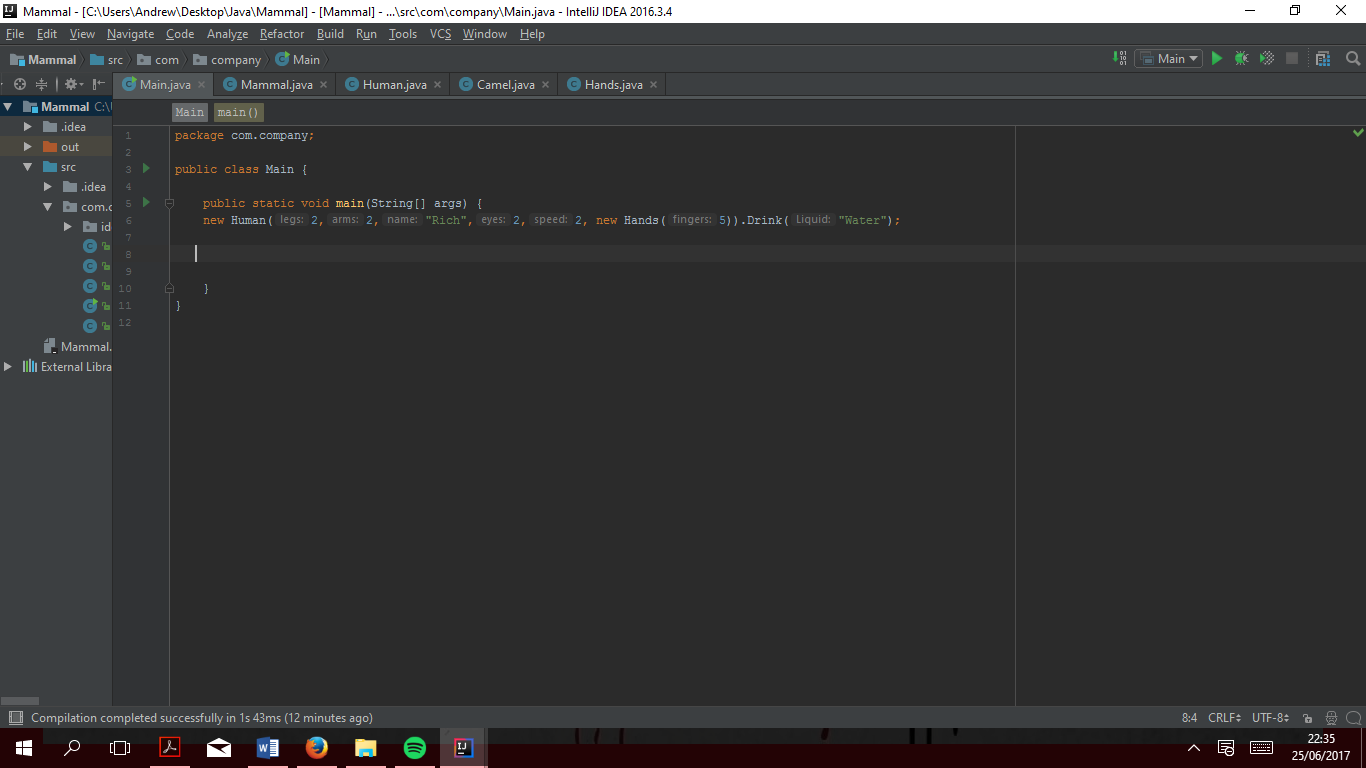


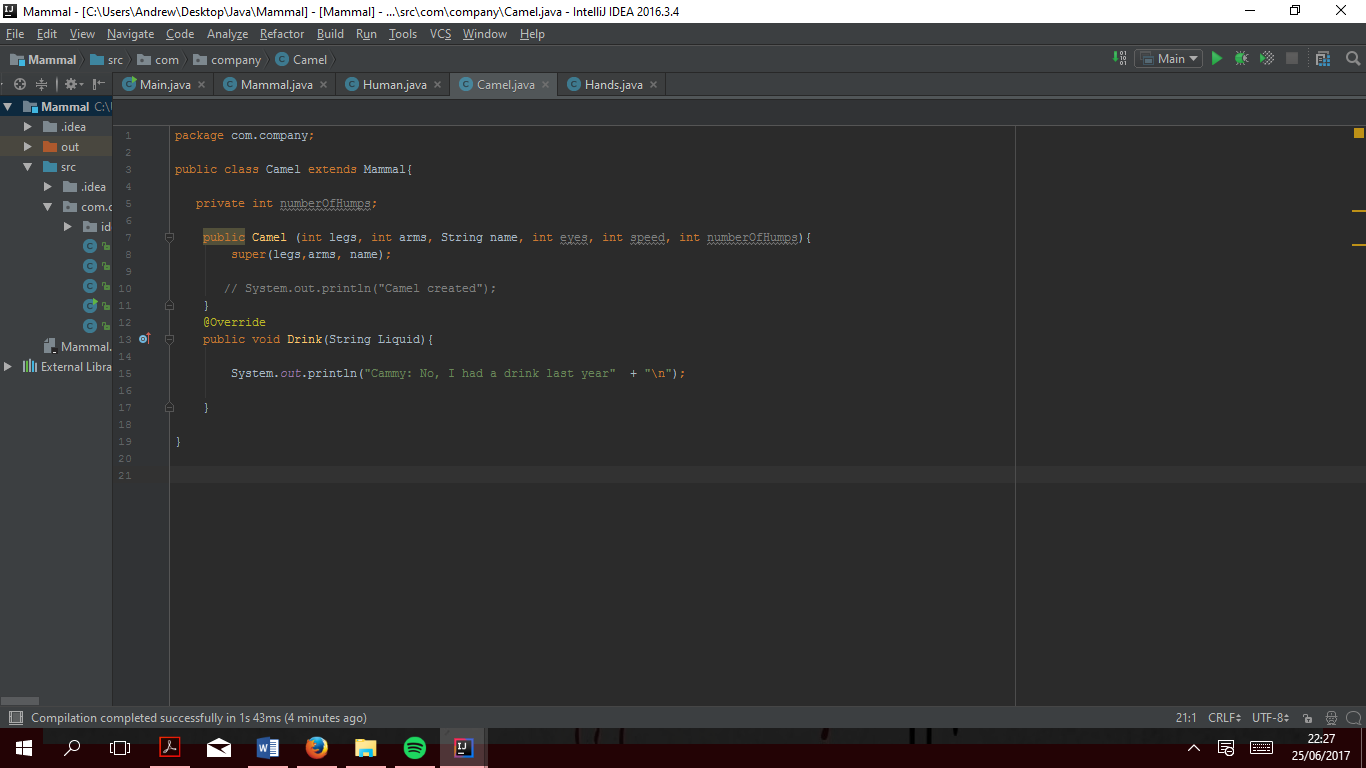
Super class method, which can be called by all inheriting sub-classes



As the method is inherited, it does not have to be in the sub-class

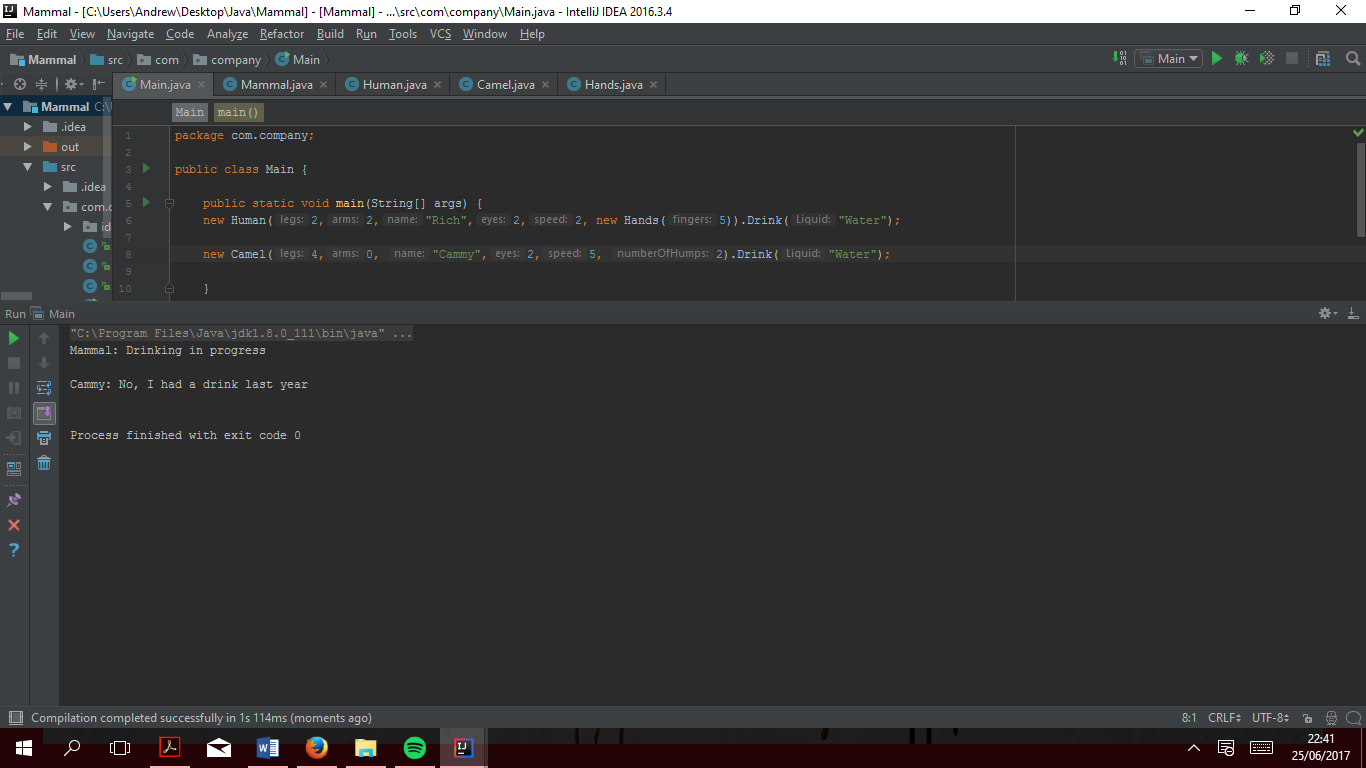
Inherited methods are called exactly the same way a normal methods, located in the sub-class, by using .methodname



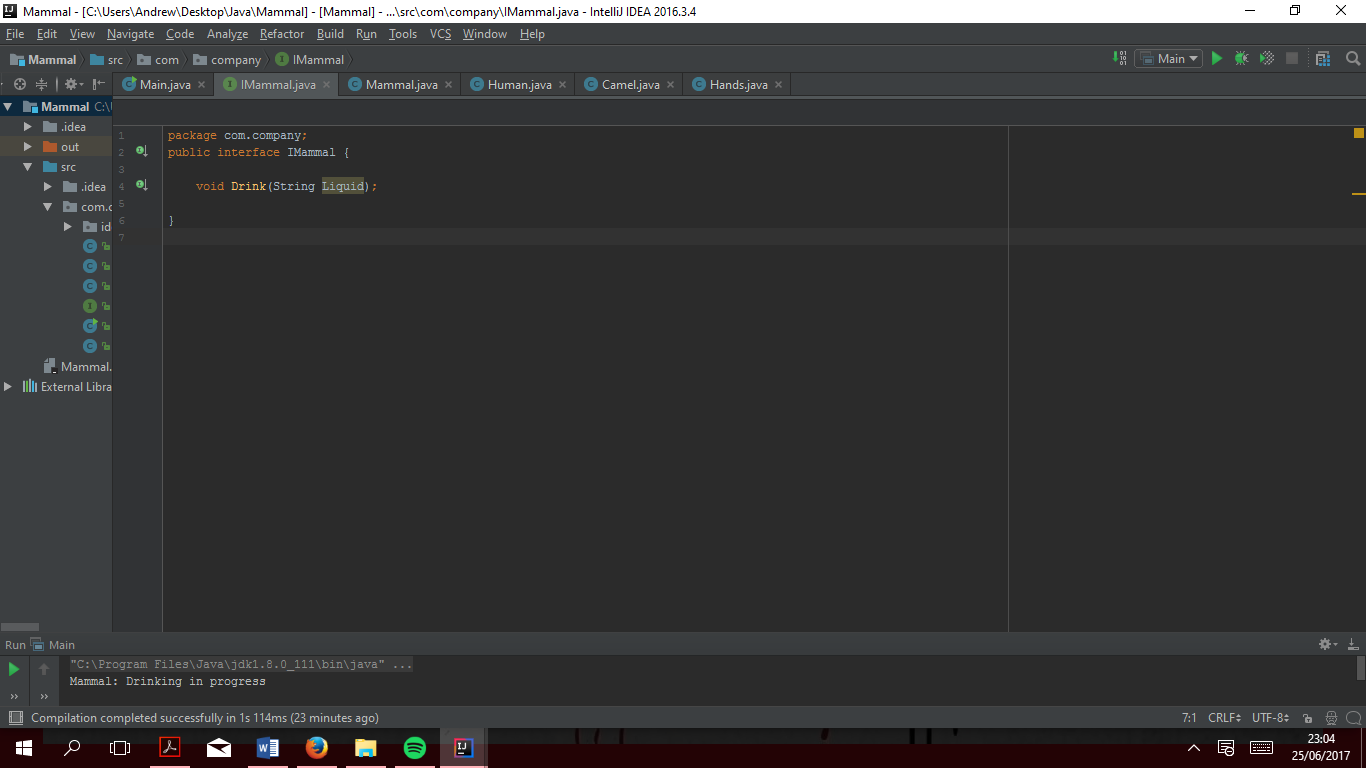
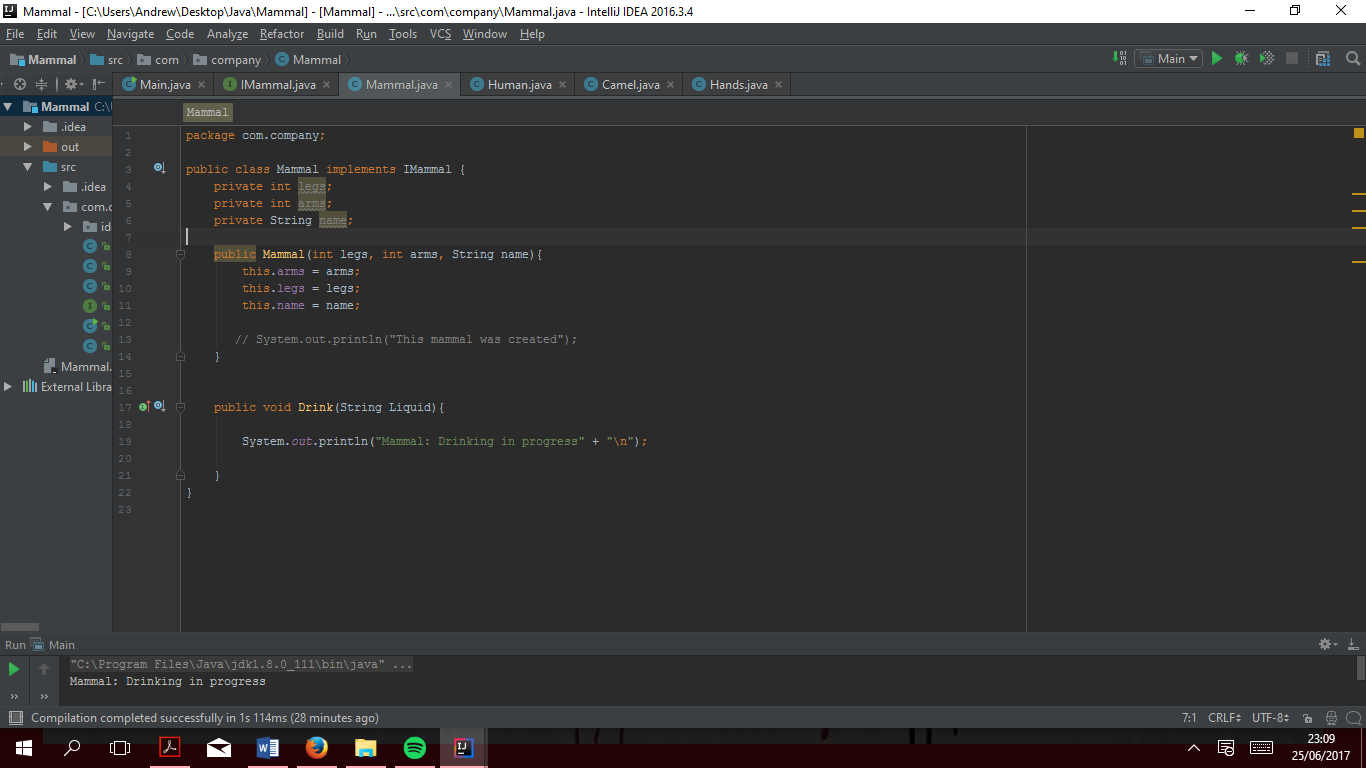


Inherited methods can be overridden to provide unique functionality for the sub-class. Override keyword indicates a method with unique functionally for sub-class

This is screenshot shows how both methods are called exactly the same way but functionality differs, this is polymorphism



**Abstraction –** Use an interface (API) to provide common behaviours used by several classes. Shields the inner workings from the classes i.e. classes can only see method names but not the integral code that is used to dictate the behaviour of the method. Similar to a template or contract in that each class created from the interface must use all methods for it to be a valid class.



Keyword “implements” is used to indicate that Mammal implements the interface IMammal

1

Shows a list of methods, without functionality, that must be used by any class that implements the interface

Shows that the file is an interface as opposed to a class. i.e. public class IMammal