**Polymorphism**

1. What does the ***word*** 'polymorphism' mean?

Polymorphism is made of two word: ‘Poly’ meaning many, and ‘morph’ meaning forms.

1. What does it mean when we apply polymorphism to OO design? Give a simple Java example.

In OO design, something is polymorphic if it can be regarded as another type of object which is not the class it was initialised as. For instance, a dog could be considered a mammal (inheritance) and also something which sheds fur (interface).

1. What can we use to implement polymorphism in Java?

Inheritance, or parent classes, allow an object to be considered an instance of its parent class.

Interfaces also allow this, allowing something to be categorised by a particular behaviour it has.

1. How many 'forms' can an object take when using polymorphism?

Unlimited?

1. Give an example of when you could use polymorphism.

One might have an array list of card objects, containing different sub-classes of the abstract card class: birthday card, Christmas card etc.

**Composition**

1. What do we mean by 'composition' in reference to object-oriented programming?

Composition refers to the instance variables contained within an object and the types of those variables.

1. When would you use composition? Provide a simple example in Java.

See question 5.

1. What is/are the advantage(s) of using composition?

Allows collection of different sub-classes under one array, and means a variable doesn’t need to receive exactly one type of variable during construction.

1. What happens to the behaviours when the object composed of them is destroyed?

If there are no other references to those behaviours/variables in memory, they are cleaned up by Java several times per second.