**Factory Method**

Name: Nguyễn Đình Thọ

Class: SE1612-NET

Student code: HE150527

**I – Research**

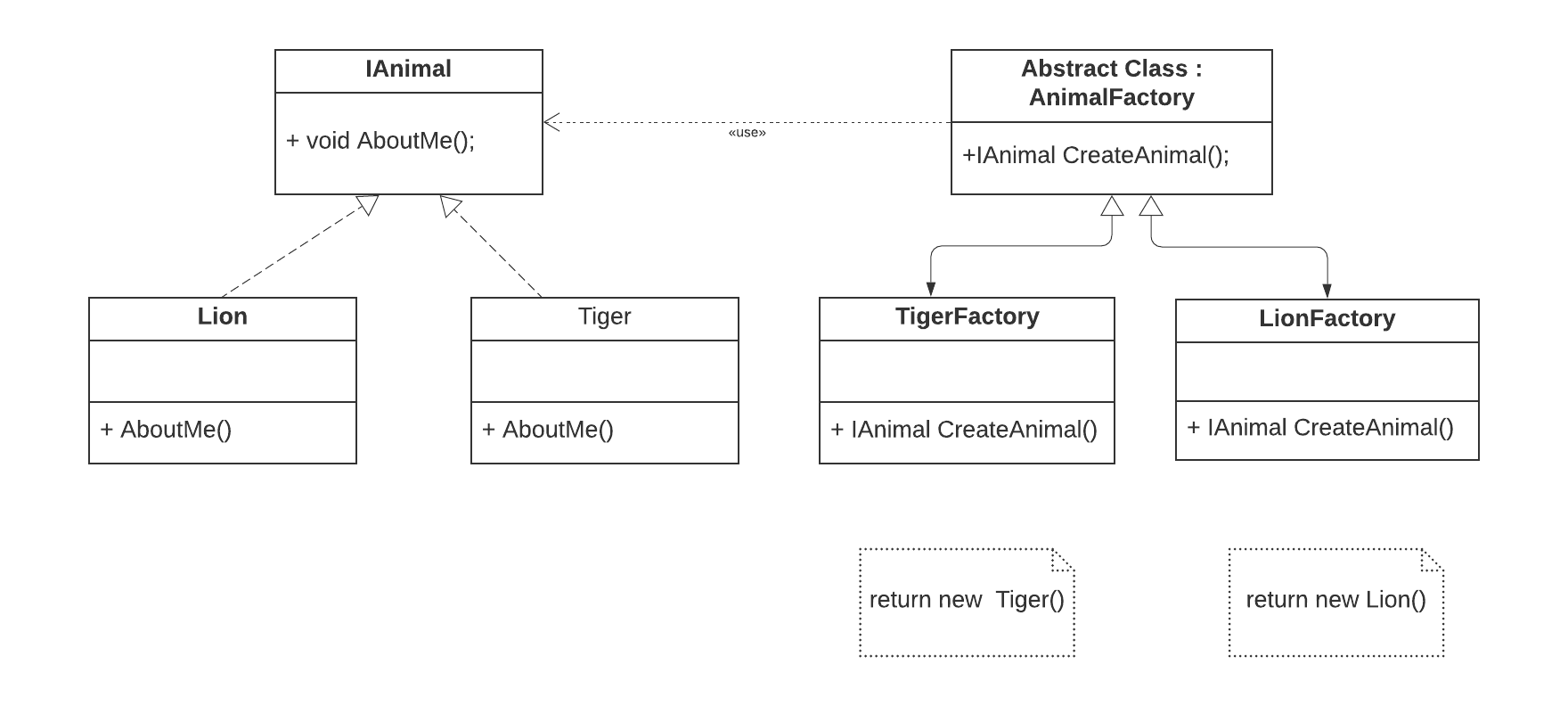
1. **Purpose of Factory Method**

* The Factory Method design pattern defines an interface for creating an object, but let subclasses decide which class to instantiate. This pattern lets a class defer instantiation to subclasses.
* The new operator considered harmful.
* Defining a "virtual" constructor.

**2.Frequency of use**:

 (high)

**3. Factory Method Structural**



* Explained class diagram:

+IAnimal : implement the IAnimal interface method

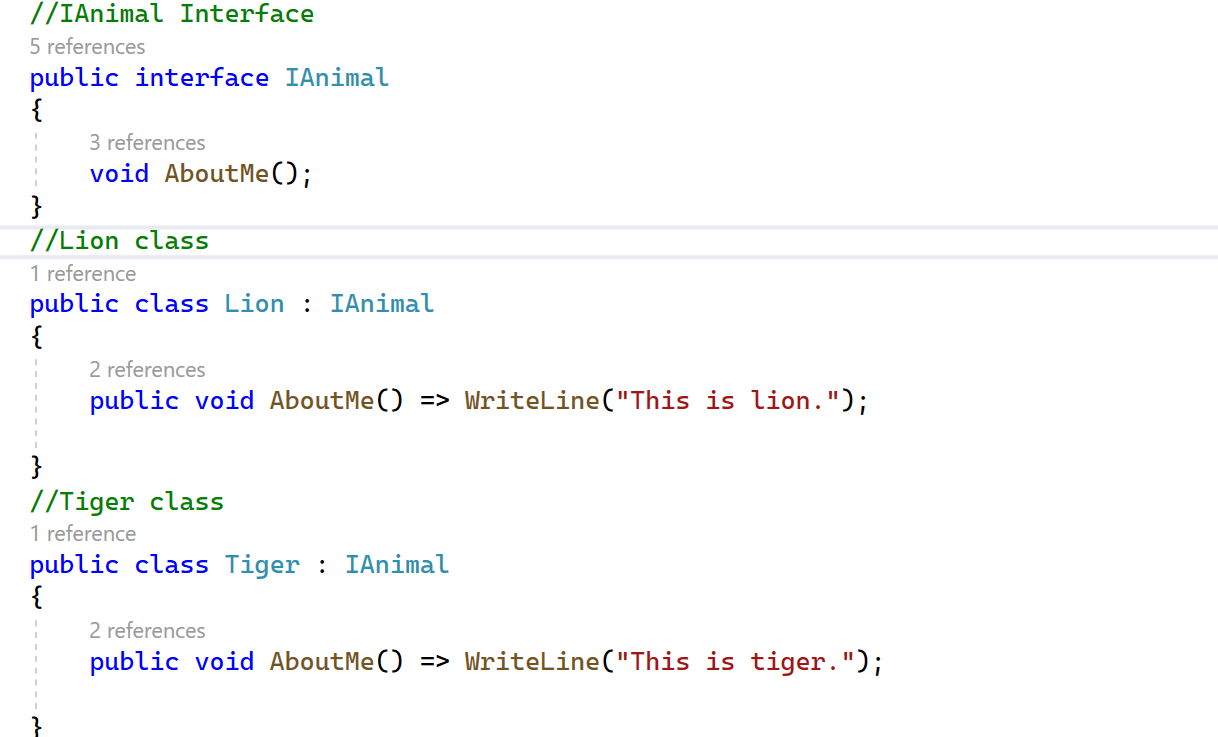
+ Lion : Class Lion implement IAnimal

+Tiger : Class Tiger implement IAnimal

+AnimalFactory : AnimalFactory method lets a class defer instantiation to subclassed. the following method will create a Tiger or a Lion, but at this point it does not know whether it will get a lion or a tiger. it will be decided by the subclassed i.e.lionfactory or TigerFactory , So, the following method is acting like a factory (of creation)

+TigerFactory : TigerFactory is used to create Tiger

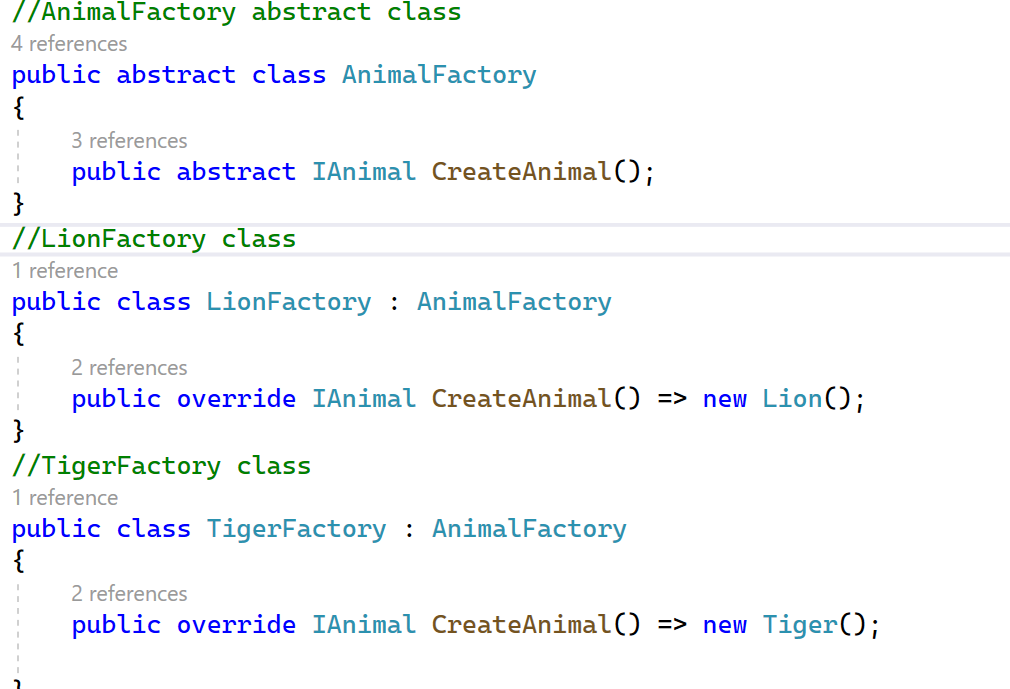
+LionFactory : LionFactory is used to create Lions

-Explained Code 

Create an interface named IAnimal and this have method AboutMe();

Class Lion implement IAnimal and Method AboutMe() print “This is lion.”

Class Tiger implement IAnimal and Method AboutMe() print “This is tiger.”



AnimalFactory abstract class have Abstract Method named AnimalFactory this return IAnimal

LionFacetory extend AnimalFactory initialization new Lion() and returns it.

TigerFactory extend AnimalFactory initialization new Tiger()and returns it.



Class Program have Main method, Main method create List have value type is AnimalFactory

In List<AnimalFactory> create new TigerFactory() and new LionFactory()

And used foreach to call CreateAnimal().AbountMe();

The CreateAnimal() method will be called to the class based on the inner object

If object inside is TigerFactory() method CreateAnimal in TigerFactory class will called

If object inside is LionFactory() method CreateAnimal in LionFactory class will called

Similar, AboutMe() method is called to the class based on the inner object.

Code will print “This is tiger” when object is TigerFactory()

Code will print “This is lion” when object is LionFactory()

**II.Conclusion:**

Factory method helps you to separate the mechanism of object creation and reuse it. Also by taking advantage of polymorphism for Factory, we can depending on the conditions to use the appropriate factory.