

Overview

The week 9 practical was to model the the concept of a museum, to do this we would an online UML diagram creator. The first step was to identify all the classes required, following this I needed to choose the appropriate attributes and methods for the classes. It was then required to determine the all the associations between the created classes. In order to complete the practical, I used methods such as encapsulation as well as using constructors.

Design

The first class, which I would also consider as the main class, I decided to make was the museum class. I decided to this because many of the other classes are related to it moreover this class would contain the most methods and attributes. The museum attributes consisted of: a name – which is a string, a type – which is a Boolean variable as it is either virtual or physical, location which is a string, entry fee, which is float type variable, number of item, which is an integer.

The next class I created was the item class, this would represent all the items in the museum. The class had the following attributes: a unique catalogue number- which is integer, title- which is a string, type- which is a string and value – which is a float type variable.

Next I created a class for the exhibitions, this class contained the following attributed: title- which is a string, and finally start and end dates – which are both date type variables.

I then created a class for the Facilities offered by the museum, this class contained the following attributes: restrooms, cafes, parking and shops which are all strings.

Finally, I created the employee class, which had the following attributes: name – which was a string, contact number – which was also a string because a contact number will never need to have any mathematical operations preformed on it, role- which was a string and salary – which was a float type variable.

All the attributes for each of the classes where private to ensure each class was encapsulated. Because the attributes where all private they can not be directly seen from any of the other objects. To access them I used getter and setter methods.

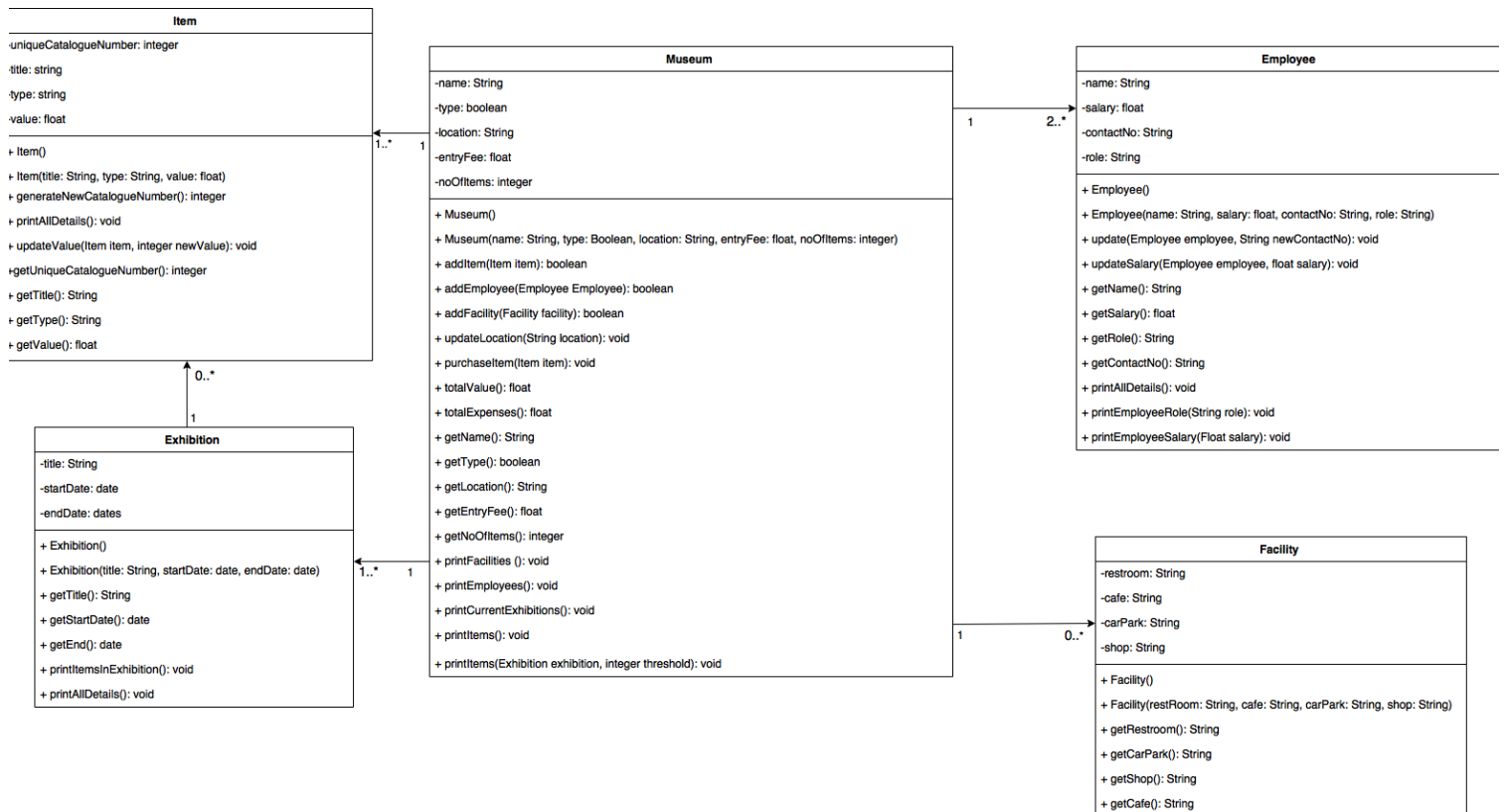
The next task was to introduce associations between classes with the correct multiplicity and navigability details for associations. Following this, I specified how the number of associations between classes. Because the museum class had to contain a number of items, one or more exhibitions, zero or more facilities and at least two employees, all the other classes where associated with this class and so I created to corresponding associations with the correct multiplicities. I then made an association between exhibitions and items so that an exhibition could consists of a subset of items. Moreover, I have ensured each class contains at least two constructors. The final step was to add all the other methods which would allow the classes to preform operations.

Research

A composite structure diagram may be used when developing and designing object-oriented software, this is because a composite structure diagram is used to show the internal structure of a class as well as all the collaborations that this structure makes possible. A composite structure diagram typically includes the internal parts, a representation of the way in which the parts interact with each other or in some cases the way in which objects interact with the class and also with the outside world. A composite structure defines the set of interconnected elements the work together when a program is executed to complete certain tasks.

A communication diagram is used to model the interactions between objects. Communication diagrams give a coherent view of the way in which related objects interact with one another this in turn means the communication diagram can be used to see how messages are passed between objects in a system.

UML Diagram



Evaluation

I have been able to successfully create a UML class diagram to model a museum. In my diagram I have made sure that all the classes are encapsulated by making the attributes private. I have therefore also added the necessary getter and setter methods to allow the attributes to be used and manipulated sensibly. In the diagram I have also include multiplicity and navigability details for associations. Moreover, I have ensured each class contains at least two constructors. I have therefore fulfilled the requirements as well as adding in my own extensions.

Conclusion

In conclusion I happy with my UML diagram and the program which I made using the diagram. The classes I created allowed me to successfully model the the concept of a museum and the attributes in each class meant that all the instances of each class created, are able to have sensible sufficient information stored about them. Each class is also able to make use of a verity of methods and constructors, which in turn makes the program more efficient and maintainable.