OOP Exam report

31.june 2022 - Candidate number: 1246

The principles of OOB and how I have used them in my project

Inheritance

Since this is a quiz project I have used inheritance in many ways to streamline the process of making questions and how they relate to the quizzes. A question is already inherited from the parent class of quiz which has its fields. However instead of inheriting from a question class they just inherit from the quiz class and are split up into two. Multiple Choice and binary choice questions. Both of these inherit the field of the question having a string that is the question and an int which is the value linked to what part of the answer is correct. I have not added the choices here as they are different enough that they have to be in their own specific class in my opinion.

Abstraction

Abstraction links into the quiz interface that is in the project. The interface is the template for every question and is therefore simple to let the other subclasses work off it.

Encapsulation

Almost every method in the project has been created to be used multiple times. Great example of this is the JDBC class which mostly consists of methods to either write to or read from the databases. There are a few small methods as well which just make things easier for me when making methods. I have a small method called GenerateRandomNumber which does just as you think. It generates a random number in the range I have pre specified in parameters of the method. Very nice for the randomization of the questions.

Polymorphism

Since we are working with a quiz I had to make sure different values could work with different methods and classes. Since the two question types we have, Binary and multiple choice are so different I had to find similarities I could work with. Since they both have topics they share from the enum I had created I made sure they both used the same one. Sadly i found no use of either method overloading or overriding as is normal when talking about polymorphism.