18/1/2019

09.30 - 11.30am

Basement 1, Kevin Street

DUBLIN INSTITUTE OF TECHNOLOGY

DT228 BSc. (Honours) Degree in Computer Science

Year 3

WINTER EXAMINATIONS 2018/2019

SOFTWARE ENGINEERING III [CMPU3038]

MR. CIARAN CAWLEY
DR. DEIRDRE LILLIS
MR. PATRICK CLARKE

FRIDAY 18TH JANUARY 9.30 A.M. – 11.30 A.M.

2 HOURS

ANSWER **THREE** QUESTIONS OUT OF **FOUR**.

ALL QUESTIONS CARRY 33 MARKS EACH.

ONE COMPLIMENTARY MARK SHALL BE AWARDED.

Note: *If asked in any question to provide an example of code, you may use any appropriate language of your choice or pseudo code in your answer.*

Q.1 **(a)** Given the structural model below. Answer the following questions.

(i) Explain, in detail, the meaning of the association between the two classes.

[5 Marks]

(ii) Using the example, explain what is meant by *referential integrity* in the context of Object Oriented Programming.

[5 Marks]

(iii) Show, using code examples, how referential integrity would be implemented for this example.

[12 Marks]

(b) The *Principle of Least Knowledge* promotes decoupling so that communication only occurs between appropriate objects. Using a suitable class diagram and sequence diagram example, explain how this principle can be breached while still allowing correct business logic to be performed.

[11 Marks]

Q.2 **(a)** Briefly describe what is meant by a *structural software design pattern.*

[4 marks]

(b) The *Bridge* is one of a number of structural design patterns that are often collectively referred to as 'wrapper' patterns. Give the intent of **three other** 'wrapper' patterns that you are familiar with, clearly explaining the differences between the three intents.

[9 marks]

(c) The intent of the Bridge pattern is stated below.

"Decouple abstraction from implementation so that the two can vary independently"

Using an example of your choice, explain what is meant by this intent. In your answer, provide a structural view of the example classes.

[10 Marks]

(d) Using your example from **part (c)** of this question. Explain how and why this pattern favours *association* over *inheritance*.

[10 Marks]

Q.3 For each of the software design patterns listed below, provide the following:

(i) The *Intent* of the pattern.
(ii) A description of a context in which the pattern could be applied.
(iii) An example of the code that would implement it.

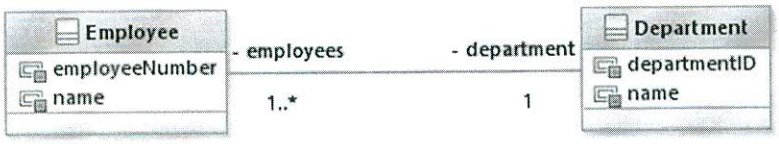
- Front Controller [11 Marks]
- Singleton [11 Marks]
- Factory [11 Marks]

Q.4 There are many instances within business software systems where the applications are forced to use *Relational Database Management Systems (RDBMS)* to store data. Designing and implementing an *Object Oriented* system can lead to substantial challenges when it comes to persisting objects. Answer the following questions in relation to these challenges.

(i) Outline the goals of what is termed *Object Relational Mapping (ORM)*.
[4 Marks]

(ii) Briefly explain what is meant by a *top down* approach when dealing with *ORM* and, using an example, discuss three different approaches in which a class inheritance hierarchy could be mapped to a relational model.
[14 Marks]

(iii) Given the following class structure, describe the database tables and their columns that could be used to store instances of both classes. In your answer clearly identify how the association is implemented.



[6 Marks]

(iv) Describe, using a clearly labelled diagram, the architecture you would use if you planned to use a *Data Access Object* and an *ORM* tool such as Hibernate as part of your persistence solution.
[5 Marks]

(v) Outline four features that you would expect to find in any commercial *ORM* tool.
[4 Marks]