** COMSATS University Islamabad, Lahore Campus**

***Lab Task - DP***

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Course Title: | Object Oriented Software Engineering | | | | Course Code: | CSC 304 | | Credit Hours: | 3(2,1) |
| Course Instructor IndhsdjsIInstructorInstructor/s: | Yella Mehroze | | | | Theory/Lab Name: | Lab | | | |
| Semester: | 4th | Batch: | Fa19-BSE | Section: | A-B | Depart | CS | | |

Student Name: AREEB AHMED

Registration ID : FA19-BSE-022

**Outline**

* **Identification of patterns from scenarios**
* **Drawing Class Diagram**

# **Task 1**

**Find the most appropriate design pattern for the following problems:**

# **Question: 1**

You are building an inheritance hierarchy of products that your company sells; however, you want to reuse several classes from one of your suppliers. You cannot modify your suppliers' classes. How do you ensure that the facilities of the suppliers' classes can still be used polymorphically?

## **Name of pattern selected**

Adapter Pattern

## **Reason of selection**

Because we have a limit that we can't change the class of suppliers. We use the adapter class to ensure that provider class resources can still be used polymorphically.

# **Question: 2**

You want to represent genealogical information in a family tree. You need to represent that some bloodlines end without any descendants. You don't want your class diagram to have to change every time a new family is added as a node to the family tree

## Name of pattern selected

The General Hierarchy Pattern

## Reason of selection

To represent the category of objects, where other objects may not be subordinate we use this pattern.

# **Question: 3**

**You have created a subsystem with 25 classes. You know that most other subsystems will only access about 5 methods in this subsystem; how can you simplify the view that the other subsystems have of your subsystem?**

## Name of pattern selected

Façade Pattern

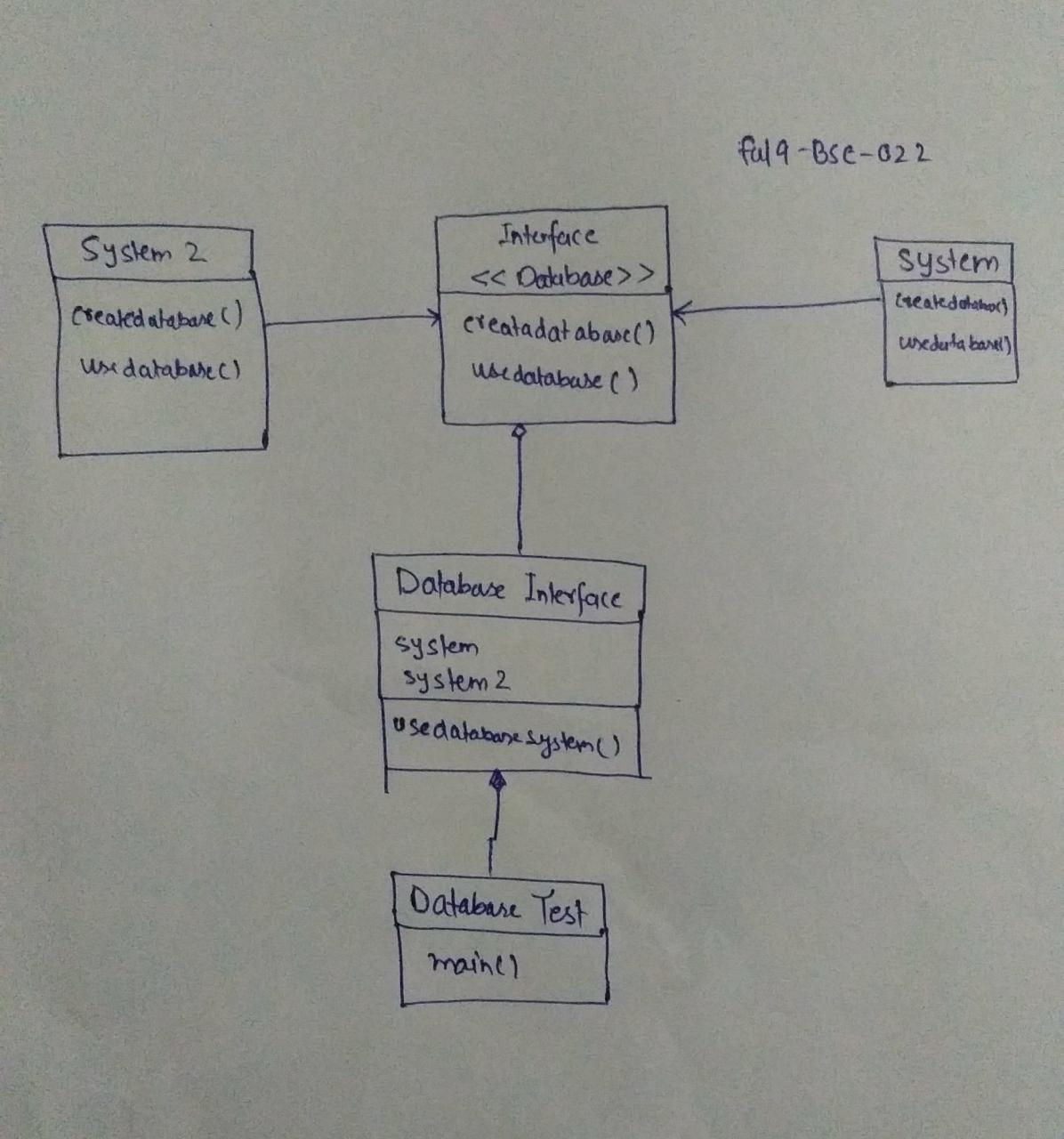
## Reason of selection

To simplify client view we must use the Façade pattern. The façade section will deal with all other connected sections.

# **Task 2**

Suppose that you want to be able to use different database systems in different versions of an application. To facilitate interchanging databases, you create a Facade interface plus Facade classes associated with each specific database system. Draw the class diagram that corresponds to this.

**Solution (Class Diagram)**

****

***For any queries***

[***yellamehroze@cuilahore.edu.pk***](mailto:yellamehroze@cuilahore.edu.pk)