

IST 261 Guided Study Group Session #2

Interfaces, CRUD
operations, and Introduction
to Persistent Data

Join the GSG GroupMe!



Table of Contents



1. Session Breakdown
2. Interfaces and Use Case
3. CRUD Operations
4. Persistent Data
5. Example Problem
6. Group Breakout work session
7. Solution walk-through
8. Conclusion

Session Breakdown

15 Minutes: Basic Lecture going over concepts

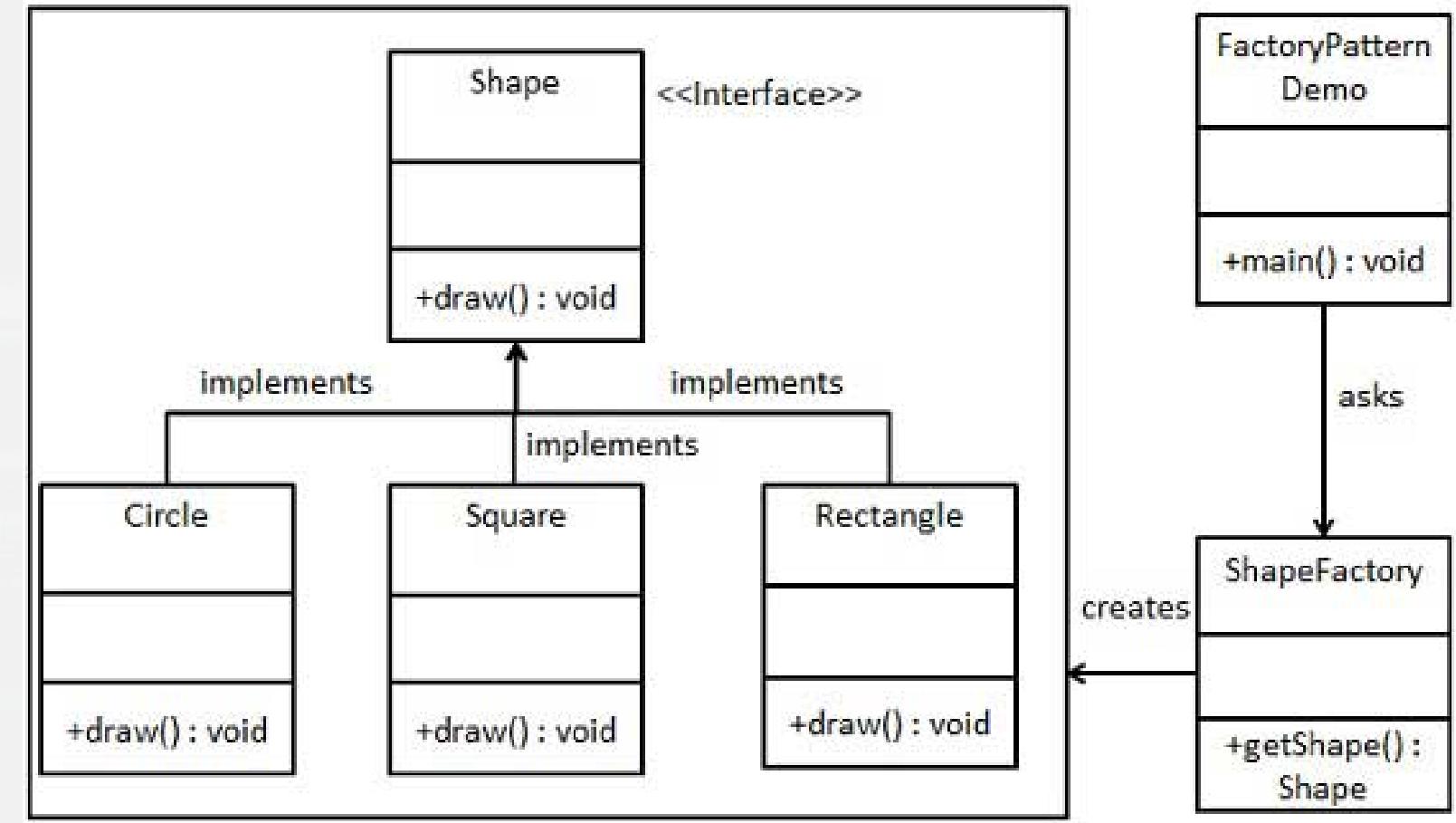
20 Minutes: Work through example together

15 Minutes: Extend example in groups

10 Minutes: Go through solution

Interfaces and Use Case

- Interfaces are another way to achieve abstraction in OOP.
 - Instead of Abstract Class, use an Interface. is-a v.s. has-a
- Great for creating a consistent API for interacting with classes that have the same has-a relationship.
- Do not have to spend as much time being bogged-down figuring out what type of class you are using.



Source: https://www.tutorialspoint.com/design_pattern/factory_pattern.htm

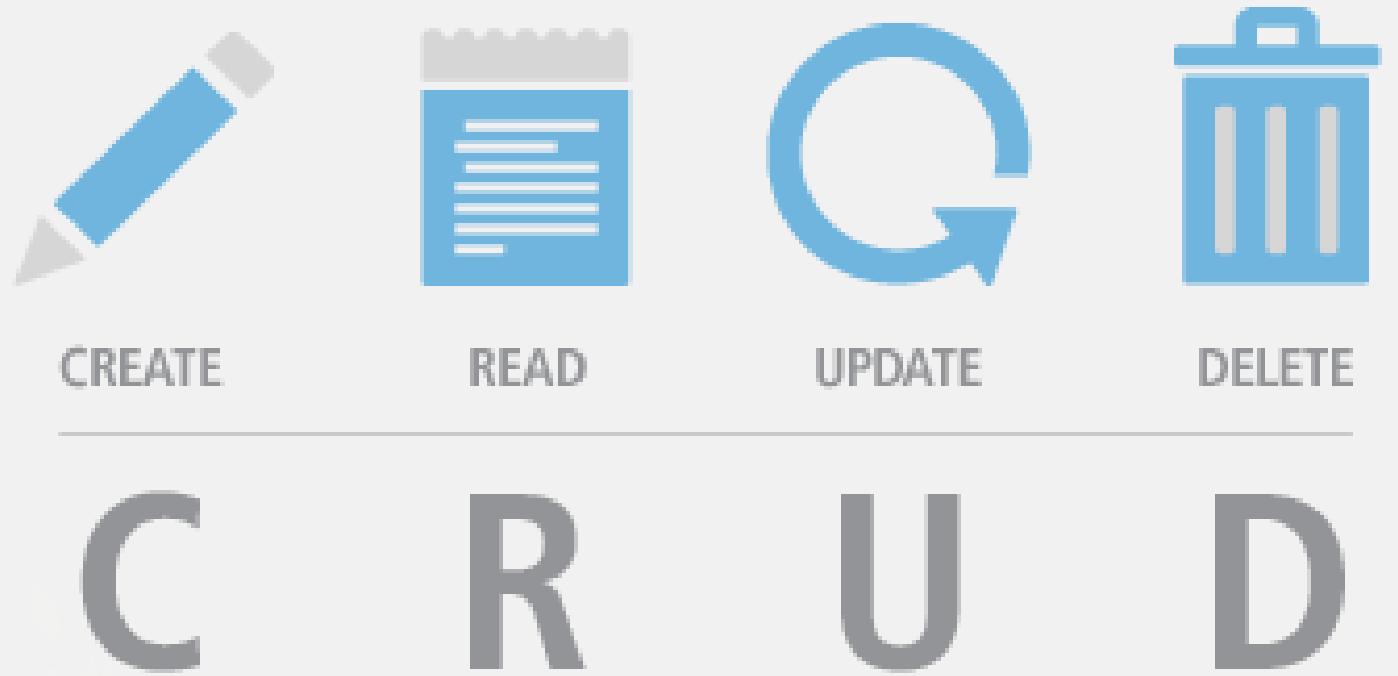


The Factory Design Pattern is great example of how you can use interfaces in your project.

Think of how you could use a Factory when creating user accounts in your project

CRUD Operations

- The four basic functions of persistent data storage.
- Each letter refers to an operation that can be performed on persistent data.
 - Ex. SQL databases, HTTP web apps, a variable in a static Java class, etc.
- Great for compartmentalizing operations on data and breaking it down into class methods.
- Operations can be bound to ActionListeners in Java Swing.



Source: <https://www.c-sharpcorner.com/UploadFile/francissvk/crud-operations-in-Asp-Net-mvc-using-ado-net/>



Persistent Data

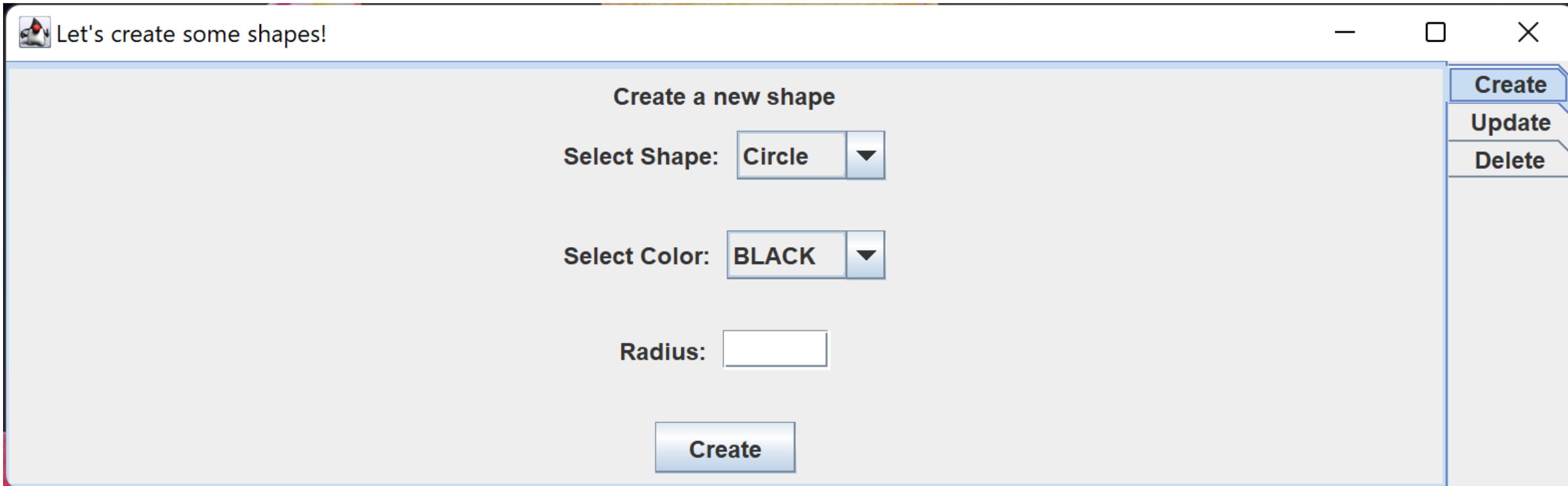
- Interact with persistent data through CRUD operations in your application.
- Data continues to be accessible by other classes and methods after being used by another method or class.
 - Data persists after program completion. Can be loaded back in by your program.
- Many ways to make data persistent!
 - MySQL (Relational), MongoDB (Document-Oriented), Redis (Key-Value), etc.
 - In IST 261, we will use an ArrayList in the Controller class.



```
new ArrayList<>();
```



CRUD-enabled Shape Drawer



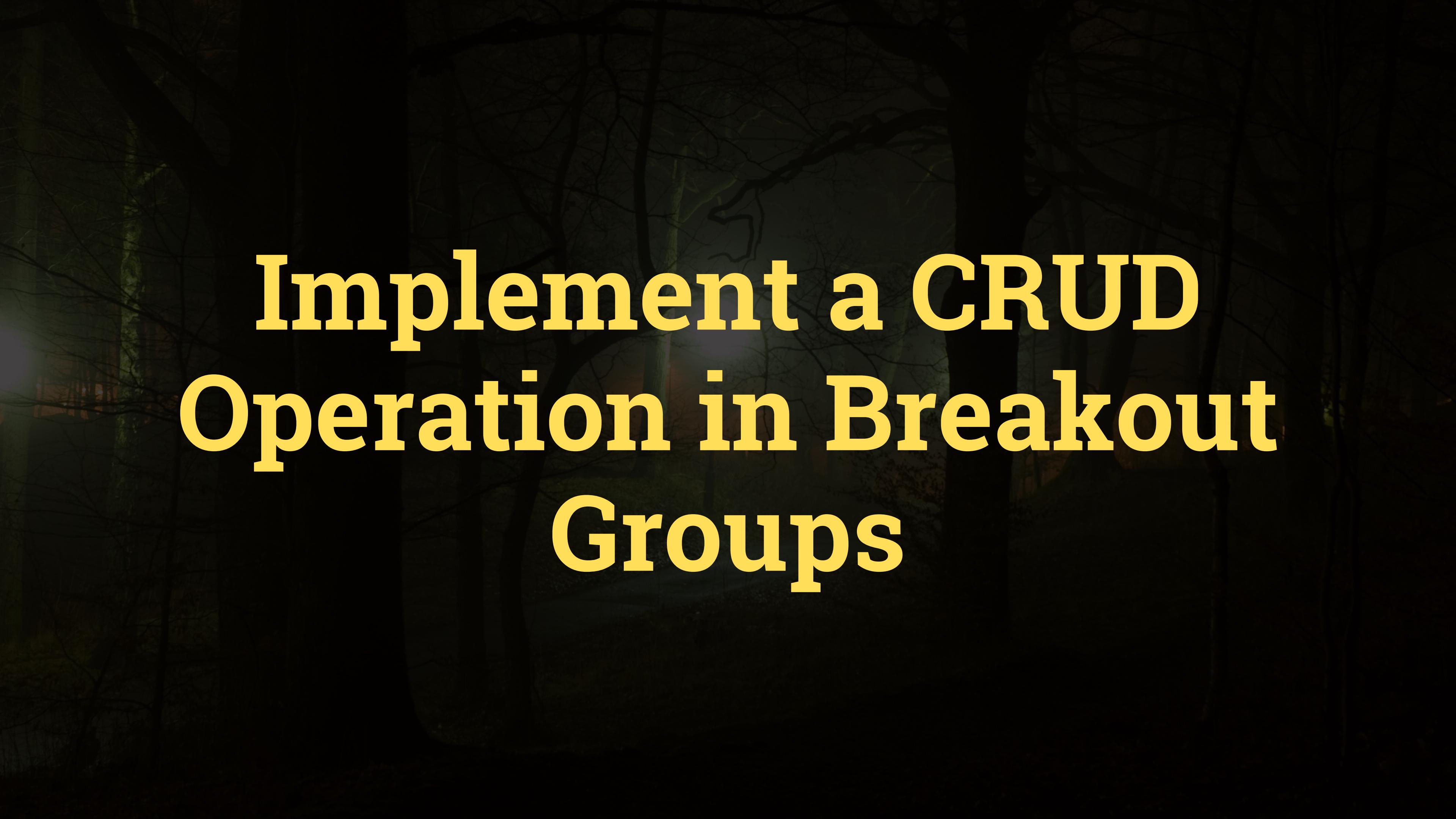
Clone the starter code here:

```
git clone https://github.com/NucciTheBoss/ist261_gsg_tutoring.git
```

After cloning the repository, open *shape_drawer* as a project in IntelliJ IDEA



**Working on the
Example together**



Implement a CRUD Operation in Breakout Groups

Solution Walkthrough



Join the GSG GroupMe!

