Data Access Objects (DAO) Patterns

Tessema Mengistu (Ph.D.)

Department of Computer Science

Virginia Tech

Mengistu@vt.edu

Outline

- Design Pattern
  - Singleton
  - Data Access Object

Design Patterns

- Design Patterns
  - Names, abstracts, and identifies the key aspects of a common design structure that make it useful for creating a reusable object-oriented design
  - Represent the best practices used by experienced object-oriented software developers
  - Provides
    - Common platform for developers
    - Best Practices

Singleton Design Pattern

- Singleton
  - Ensure a class only has one instance, and provide a global point of access to it
  - The class can ensure that no other instance can be created
  - It can provide a way to access the instance
  - Implementation:
    - Private default constructor
    - Static creation member that acts as a constructor

Data Access Object (DAO) Pattern

- Separates low level data accessing API or operations from high level business services
- It involves:
  - Data Access Object Interface
    - Defines the standard operations to be performed on a model object(s).
  - Data Access Object concrete class
    - Implements above interface
    - This class is responsible to get data from a data source
  - Model Object or Value Object
    - This object is class containing get/set methods to store data retrieved using DAO class

Data Access Object (DAO) Pattern

- Advantages
  - Transparency
  - Easier Migration
  - Easier to maintain and manage
- Disadvantage
  - Additional extra layer

References

- **Design Patterns: Elements of Reusable Object-Oriented Software.** E. Gamma, R. Helm, R. Johnson, J. Vlissides. Addison-Wesley Professional.
- https://www.oracle.com/java/technologies/dataaccessobject.html