

# Singleton

Limits number of instances that can be created, and provides easy access to it.

Similar to global variables, but number of instances is limited, and scope is limited to the scope of the class.

Lazy instantiation

Easy to change your mind and allow multiple instances

Examples: Logging, Configuration, ThreadScheduler, LoadBalancer

```
public Configuration {  
  
    private int maxConnections;  
    private int backupInterval;  
    private File logFile;  
  
    public Configuration(File configFile) {  
        // open and read config file  
    }  
  
    public int getMaxConnections() {  
        return maxConnections;  
    }  
  
    public int getBackupInterval() {  
        return backupInterval;  
    }  
  
    public File getLogFile() {  
        return logFile;  
    }  
}
```

```
public Configuration {

    private static Configuration instance;

    public static Configuration getInstance() {
        if (instance == null)
            instance = new Configuration("DB.config");
        return instance;
    }

    private int maxConnections;
    private int backupInterval;
    private File logFile;

    private Configuration(File configFile) {
        // open and read config file
    }

    public int getMaxConnections() {
        return maxConnections;
    }

    public int getBackupInterval() {
        return backukpInterval;
    }

    public File getLogFile() {
        return logFile;
    }
}
```