Synchronized vs Non-Synchronized in Java

Feature
Synchronized Method
Non-Synchronized Method

Definition
A method that is synchronized, meaning it can only be accessed by one thread at a time.
A method that is not synchronized, allowing multiple threads to access it simultaneously.

Keyword
Uses the `synchronized` keyword.
Does not use the `synchronized` keyword.

Thread Safety
Provides thread safety by preventing concurrent access by multiple threads.
Does not provide thread safety; concurrent access can lead to race conditions.

Performance
Can be slower due to the overhead of acquiring and releasing locks.
Generally faster as there is no overhead of synchronization.

Synchronized vs Non-Synchronized in Java

Usage
Used when shared resources need to be accessed or modified by multiple threads.
Used when there is no shared resource, or thread safety is not a concern.

Example

public synchronized void synchronizedMethod() {
        // method code
    }

public void nonSynchronizedMethod() {
        // method code
    }

Block Synchronization
Allows synchronizing a block of code within a method using `synchronized` block.
No equivalent; the entire method is non-synchronized.

Locks
Acquires a lock on the object (or class, if the method is static) before executing.
Does not acquire any locks.

Deadlock Potential
Can lead to deadlock if not used

carefully, especially when multiple locks are involved.
No risk of deadlock as no locks are used.

Reentrancy
Allows the same thread to re-enter the synchronized method/block if it already holds the lock.
Not applicable.

Scope of Synchronization
Applies to the entire method or specified block of code.
No synchronization, so no scope limitation.