**Java Key Words**

**final**

* Zmienna: nie można zmienić jej referencji (nie wartości). Może być inicjalizowana tylko raz. Przykładowo tworząc listę i oznaczając ją jako final, możemy ją jedynie raz zainicjalizować new ArrayList<>(), ale możemy do niej dodać elementy.
* Metoda: nie pozwala override’ować tej metody
* Klasa: nie może być dziedziczona

**Abstract**

* Klasa: Nie może być inicjalizowana. Jest tworzona po to, aby być klasą nadrzędną do dziedziczenia.

Klasa abstrakcyjna może:

- definiować metody, które mogą być używane przez klasy dziedziczące.

- definiować metody abstrakcyjne, które klasa dziedzicząca będzie musiała zaimplementować,

- zapewnia wspólny interfejs,

* Metoda: Metoda abstrakcyjna nie posiada implementacji i musi zostać zaimplementowana przez klasę dziedziczącą.

**synchronized**

It is used to prevent concurrency. Synchronized keyword can be applied to static/non-static methods or a block of code. Only one thread at a time can access synchronized methods

**volatile**

It is used to mark a Java variable as “being stored in main memory”. So declaring a variable as *volatile* thus guarantees the visibility for other threads of writes to that variable.

JVM can not reorder read and write operations of volatile variable.

**transient**

Not required to be serialized.

**static**

Everything defined as static is related to class and not to object.

A static method can be accessed without creating the objects. It can only access static variables.

A class cannot be declared static except inner class. But a class can be said to be static if all the variables and methods of the class are static, and the constructor is private.