**Companion objects:**

Companion objects in Kotlin provide a way to associate methods and properties with a class rather than with instances of that class. They are equivalent to static blocks in Java, allowing for common functionalities to be accessed without creating an instance of the class. Companion objects are defined using the `companion object` keyword within the class and can be accessed using the class name directly. They are often used to define factory methods, constants, or utility functions related to a class.

Instantiation In Kotlin:

val obj = MyClass("example")

**Why Kotlin?**

Kotlin, created by JetBrains, is favored over Java due to its concise syntax, null safety feature, and reduced boilerplate code. Google promotes Kotlin because as it Operates on the JVM, Kotlin seamlessly interoperates with existing Java codebases and offers features like higher-order functions, extension functions, data classes, and sealed classes, enhancing developer productivity and code maintainability.

**How Kotlin compiled?**

