Features of Gradle

The following is the list of features that Gradle provides.

* **Declarative builds and build-by-convention** − Gradle is available with separate Domain Specific Language (DSL) based on Groovy language. Gradle provides a declarative language elements. Those elements also provides build-by-convention support for Java, Groovy, OSGI, Web and Scala.
* **Language for dependency based programming** − The declarative language lies on a top of a general purpose task graph, which you can fully leverage in your build.
* **Structure your build** − Gradle finally allows you to apply common design principles to your build. It will give you a perfect structure for build, SO that you can design well-structured and easily maintained, comprehensible build.
* **Deep API** − Using this API it allows you to monitor and customize its configuration and execution behavior to its core.
* **Gradle scales** − Gradle can easily increases their productivity, from simple and single project builds to huge enterprise multi-project builds.
* **Multi-project builds** − Gradle supports for multi-project builds and it supports partial builds. If you build a subproject Gradle takes care of building all the subprojects that it depends on.
* **Different ways to manage your builds** − Gradle supports different strategies to manage your dependencies.
* **Gradle is the first build integration tool** − Gradle fully supported for your ANT tasks, Maven and lvy repository infrastructure for publishing and retrieving dependencies. Gradle also provides a converter for turning a Maven pom.xml to Gradle script.
* **Ease of migration** − Gradle can easily adapt to any structure you have. Therefore you can always develop your Gradle build in the same branch where you can build live script.
* **Gradle Wrapper** − Gradle Wrapper allows you to execute Gradle builds on machines where Gradle is not installed. This is useful for continuous integration of servers.
* **Free open source** − Gradle is an open source project, and licensed under the Apache Software License (ASL).
* **Groovy** − Gradle's build script are written in Groovy. The whole design of Gradle is oriented towards being used as a language, not as a rigid framework. And Groovy allows you to write your own script with some abstractions. The whole Gradle API is fully designed in Groovy language.