

Annotating Source Code

In this section you will find the basics of IntelliJ IDEA's annotations, and the related procedures:

- [Enabling Annotations](#).
- [@Nullable](#) and [@NotNull](#) annotations, which are used, when IntelliJ IDEA supposes that a certain element can become Null.
- [@NonNls](#) annotation, which is used to ignore the hard-coded string literals.
- [@Contract](#) annotations, which is a powerful and flexible tool for making the APIs safer.
- [Annotating Source Code Directly](#), which describes how to add annotations to your source code.
- [Inferring Nullity](#), which describes how to analyze the source code for the possible [@Nullable](#) and [@NotNull](#) annotations.

Besides that, you can also find the following information:

- [External Annotations](#) that are stored outside the source code.
- [Using External Annotations](#)

Bundled **IntelliLang** plugin provides three new annotations you may find useful:

- [@Language](#): responsible for the Language injection feature.
- [@Pattern](#): is used to validate Strings against a certain regular expression pattern
- [@Subst](#): is used to substitute non-compile time constant expressions with a fixed value.

To learn more about these **IntelliLang** annotations, refer to the [plugin documentation](#).

See Also

Concepts:

- [Code Inspection](#)

Procedures:

- [Analyzing Applications](#)

Web Resources:

- [Developer Community](#) 