What's New

What's New in Version 13.1

IDE

- Multicursor
- Ability to override the IDE shortcuts in the embedded local terminal

Code Editing

- Restoring default template settings
- Ability to show quick documentation for a live template from a suggestion list
- Postfix templates

Configuring Project and IDE

- Enabling national characters in properties files
- Option to override the IDE shortcuts
- Option to automatically collapse one-line methods is enabled by default
- Ability to configure postfix templates
- Ability to define location of proxy auto configuration file while auto-detect proxy settings
- Ability to restore default settings of a modified live template
- Ability to manage the list of trusted certificates

Code Analysis

- NotNull is defined as the default behavior

Languages, Frameworks and Technologies

- Servers certificates are checked
- Untrusted servers can be accepted of rejected

Running

- Ability to filter thread dump by a word in a stacktrace

Debugging

- Placing breakpoints on the folded methods

Version Control

- Option 'Rearrange entries' is available in the 'Reformat Files' dialog
- 'Rearrange code' option is available in the 'Commit Changes' dialog

Inspections

- Inspection 'Inconsistent line separators'

Groovy and Groovy-Based Frameworks

- Parameter info for type parameters

Java EE and Web Development

- OpenShift debug support

Miscellaneous Improvements

- Ability to pin scroll bar in the Differences Viewer
- Ability to open file from the Differences Viewer in the editor

What's New in Version 13.0.2

Code Editing

- The way of showing national characters in properties files depends on the 'Native-to-ascii conversion' setting

Miscellaneous Improvements

- Changing log settings

What's New in Version 13.0

IDE

- Quick access for tool windows
- Lens mode
- Presentation mode
- Reopening a project via Task bar Jumplist
- Embedded local terminal
- Abbreviation can be added to an action name, to ease Search Everywhere

Projects and Modules

- New directory types are available: resources and generated sources
- Excluded status supported for library items
- Reopening a project from the Task bar
- Ability to create package information file

Code Editing

- Zooming in the Editor
- Disable/enable reformatting of a part of source code
- Viewing quick documentation on mouse move

Configuring Project and IDE

- Copying Code Style Settings
- Possibility to disable code formatting using special markers
- Continuation indent in JSP code style
- Ability to create a default indentation for Velocity files
- The 'Force rearrange' option is added to the arrangement tab for XML files
- Ability to specify a rule that controls a namespace attribute position
- Wide screen support
- New default theme in community edition
- Configuring font size for the Presentation mode

- Ability to toggle the lens mode
- Ability to show quick documentation on mouse move
- Ability to change color using intention action, when gutter icons are not shown
- Emmet: Fuzzy search available for CSS
- Emmet is enabled for CSS and XML separately
- Turning on/off the preview of XML Emmet abbreviation
- Function 'clipboard()' added to the live template expressions
- Function 'escapeString()' added to the live template expressions

Refactoring

- Move Static Member refactoring for PHP static methods, variables, and constants is supported
- Ability to choose to see or not file after a copying

Code Analysis

- @Contract Annotations

Languages, Frameworks and Technologies

- Downloading stubs for TypeScript definition files and configuring them as IntelliJ IDEA JavaScript library
- Karma unit testing framework is supported
- NodeJS debugger backend for Chrome has been completely redesigned based on the V8 Debugging Protocol
- Installing and removing External software using Node Package Manager
- Drupal support:

  modules, themes, and core. Drupal-aware coding assistance, including support of hooks. Drupal native command line tool Drush.

- EJS templates are supported

Tool Windows

- Quick access for tool windows
- Wide screen support

Plugins

- Functionality of some plugins merged into IntelliJ IDEA

  GenerateToString, JarFinder, Inspection Gadgets, Intention Power Pack are no longer bundled plugins. Their functionality merged into IntelliJ IDEA.

Navigation and Search

- Go to directory
- Ability confine search in file to comments and strings only
- Search everywhere
- Ability to find comments and strings in path

Debugging

- Configurable port for built-in web server

- JavaScript debugger backend for Chrome has been completely redesigned based on the WebKit Remote Debugging Protocol
- Viewing the dynamic HTML source code and the DOM structure of the page actually opened in the editor. The HTML source is updated dynamically upon every action performed on the page
- CORS Control in Chrome Extension
- Specifying the actual IDE port in the Chrome Extension connection settings when the default one us busy and IDE has to capture another one
- NodeJS debugger backend for Chrome has been completely redesigned based on the V8 Debugging Protocol
- Built-in server port is available for the JavaScript debugger
- Built-in server port can be available externally

Markup languages and style sheets

- Ability to change color using intention action, when gutter icons are not shown
- Enabling Emmet separately for XML and CSS
- History is added to the Surround with Live Template (Emmet)
- Viewing the dynamic HTML source code and the DOM structure of the page actually opened in the editor. The HTML source is updated dynamically upon every action performed on the page
- CORS Control in Chrome Extension
- Specifying the actual IDE port in the Chrome Extension connection settings when the default one us busy and IDE has to capture another one

Version Control

- Mercurial: light-weight branches (bookmarks) are supported
- Basic support for Mercurial tags: showing tags in the Log tab of the Changes tool window, tagging tips of repositories
- Integration with SVN 1.8 is supported
- Mercurial: tags for tips of repositories shown

Android

- Completion for the minSdkVersion and the targetSdkVersion attributes in the AndroidManifest.xml file: the known API levels and the corresponding version names are displayed in a suggestion list
- Quick documentation look-up (Ctrl+Q) for Android XML tags
- A new Add Method intention action for method name attributes, such as Button.onCLick and similar
- In layout definition files, layout_ attributes are displayed on top in suggestion lists for completion
- On-the-fly validation of ID types for fields that are validated by the compiler. Highlighting invalid ID's
- Adding the android:layout_width and android:layout_height in single views and view groups automatically
- Adding closing tags </ViewGroup> and </> automatically
- Android inspection: Lint API check
- Generating a related layout definition file (content view) during the creation of an activity or a fragment

- Navigating between an activity or a fragment and its related layout definition file
- Navigating from a component to its declaration in the AndroidManifest.xml file
- Navigation between the source code of an activity or a fragment and the related layout definition file
- Running and debugging a custom .apk file built from a user-defined artifact
- The Logcat tab of the Android tool window is activated automatically every time an application is deployed and launched successfully. The default behaviour can be changed by clearing the Show logcat automatically check box in the Run/Debug configuration: Android Application dialog box
- Filtering out logcat message by Process ID (PID), Application (Log Tag), and by Java package (class path)
- Launching a debugging session for a running application through a run/debug configuration
- Merging the manifests of library modules into the manifest of the entire application
- Turning pre-dexing of library module dependencies on and off
- Renaming the Android application package (application ID) during build
- Signing packages in the debug mode with user-defined certificates
- Suppression compression of resources during packaging
- Accessing Android SQLite databases on internal and external storages from IntelliJ IDEA
- Ability to create Gradle-based Android projects
- The Insert line break after last attribute option added to the code style settings for Android XML files

Inspections

- Inspection 'Constant conditions and exceptions': new UI, new description

  Button 'Configure Assert/Check Methods' removed, since now contracts are used; check boxes 'Ignore assert statements' and 'Warn when reading a value guaranteed to be constant' are added.

- Ability to check that FreeMarker references are resolved correctly
- The Missing PHPDoc Comment inspection can be skipped if a function or method does not contain any parameters and/or return values

Intention Actions and Quick Fixes

- Groovy: Ability to convert the if-else statement to conditional expression

  For example, 'if (abc) return 2 else return 3' such if-statement will be converted to 'return abc?2:3' conditional expression.

- Groovy: Ability to introduce String variable from the selected part of String literal

PHP Support

- Move Static Member refactoring for PHP static methods, variables, and constants is supported
- Validating the local and remote configuration of the Xdebug or Zend Debugger tool with the possibility to specify the folder to create a validation script in
- Composer Dependency Manager: previously downloaded packages are marked with a tick in the Add Composer Dependency dialog box
- Drupal support:

  modules, themes, and core. Drupal-aware coding assistance, including support of hooks. Drupal native command line tool Drush.

- Enabling and disabling automatic upload of PHPUnit tests to the server before run

Groovy and Groovy-Based Frameworks

- Call hierarchy for Groovy is now supported
- Automatic language injection into predefined methods is now supported for Groovy

  Place your caret on the string, press 'Alt + Enter' and select 'Inject Language/Reference' to open a list of languages and references.
- Language injection is available for GString in Groovy
- Ability to convert the 'if-else' statement to conditional expression for Groovy

  For example, 'if (abc) return 2 else return 3' such if-statement will be converted to 'return abc?2:3' conditional expression.
- Ability to introduce String variable from the selected part of String literal
- Ability to change the order of 'else-if' in an 'if' statement
- Ability to compile Gradle-based Android projects

Data Access Support

- DDL data sources can be created by means of drag and drop
- DB data sources for H2 and SQLite can be created by means of drag and drop
- Ability to pin the Result tab provided
- The structure view for tables improved
- Read-only status for tables in the Table Editor supported
- Transposed Row view is available for tables
- Settings for the database, hibernate and JPA consoles, and the table editor are now available in one place
- You can close several database connections at once
- Different data sources can now have different colors
- Ability to show or hide table constraints provided
- Read-only status for DB data sources is supported
- Database tab added to Event log

JavaScript Support

- Downloading stubs for TypeScript definition files and configuring them as IntelliJ IDEA JavaScript library
- Configurable port for built-in web server
- JavaScript debugger backend for Chrome has been completely redesigned based on the WebKit Remote Debugging Protocol
- Karma unit testing framework is supported
- CORS Control in Chrome Extension
- Specifying the actual IDE port in the Chrome Extension connection settings when the default one us busy and IDE has to capture another one
- NodeJS debugger backend for Chrome has been completely redesigned based on the V8 Debugging Protocol
- Installing and removing External software using Node Package Manager
- EJS templates are supported

Testing Support

- Karma unit testing framework is supported
- Using cookies in testing RESTful Web services

Remote Hosts

- The HTTPS connection to Web servers is supported

Command Line Tools

- Zend Framework 2, Laravel, and Doctrine command line tools are supported
- Support of Drush, a Drupal-specific command line tool
- The character set to show the output of a command line tool in the Command Line Tool console can be chosen from the Console Encoding drop-down list

Build Tools

- Ability to create a Gradle project through the project wizard

Java EE and Web Development

- OpenShift support

Miscellaneous Improvements

- Submit Feedback command creates a YouTrack issue
- Ability to rename of move velocity_implicit.vm

What's New in Version 12.1

IDE

- Line endings control in the Status Bar
- Consolidating bookmarks, breakpoints, and favorites in a single tool window

Code Editing

- Ability to view and change line separators for the existing files
- To declare language injections you can now use comments
- Resource bundle editor respects escaped unicode characters

Configuring Project and IDE

- Possibility to define fall-back fonts
- Tab Other: HEX Colors
- Configuring Closure Linter quality checker
- Fall-back fonts
- Ability to configure Emmet for HTML, XML, CSS
- Simplified installation of JetBrains-provided plugins

Refactoring

- Groovy: Arbitrary method names support

Languages, Frameworks and Technologies

- Dart support including transpilation to JavaScript, running, and debugging
- Typescript support including transpilation to JavaScript, running, and debugging
- Time tracking support
- Emmet support

Tool Windows

- Possibility to add external files to Favorites by drag'n'drop
- Consolidating bookmarks, breakpoints, and favorites in a single tool window
- JetGradle tool window: new context menu options
- JetGradle tool window: new filter options
- HTML 5 outline mode is added
- Time tracking tool window

Plugins

- Separate button for installing JetBrains-provided plugins

Navigation and Search

- Possibility to navigate through the bookmarks using the Favorites tool window
- Possibility to navigate through the project favorites, bookmarks, and breakpoints using the Favorites tool window
- New actions to select by camel humps when use camelhumps feature is disabled

Debugging

- List of breakpoints is shown in the Favorites tool window
- Breakpoints are configurable from the Favorites tool window

Markup languages and style sheets

- File Watchers for transpiling SASS, LESS, and SCSS into CSS and for compressing CSS

PHP Support

- Running PHPUnit without installation from phpunit.phar archive
- Running PHPUnit using autoload.php from the Composer Dependency Manager

Groovy and Groovy-Based Frameworks

- Arbitrary method names support for Groovy refactoring

  Method names with non conventional syntax will be wrapped in quotation marks when they are renamed. For example,the method name f*oo is returned as "f*oo" when the method is renamed.

Data Access Support

- Auto sync is now available for databases
- Database access via SSH is supported

ActionScript and Flex

- Integration with the Adobe AIR SDK provided

JavaScript Support

- File Watchers for transpiling CoffeeScript, TypeScript, and Dart into JavaScript and for compressing JavaScript
- Closure Linter quality checker

Build Tools

- IntellijIDEA automatically reconfigures library folders for each new version of Gradle

Miscellaneous Improvements

- Ability to show progress indicators for modern operating systems

What's New in Version 12.0

IDE

- The Welcome screen has been redesigned
- Editor tabs can be sorted in alphabetical order
- Multi-selection works in the lists of recently opened and recently edited files
- Ability to cancel a compilation or build process with a shortcut

Projects and Modules

- New way of creating projects from scratch
- New way of importing sources into IntelliJ IDEA
- Possibility to delete project templates
- 'Remove' command is added to the context menu of the module dependencies

Code Editing

- Action to restore default font size
- Additional step to select vararg arguments in method calls
- Possibility to copy reference to a line of source code
- Table view provided for CSV and TSV files
- Navigating between custom regions
- Scopes for TODO items
- File template 'Implemented method body' takes predefined variables
- File template 'Overridden method body' takes predefined variables
- It is possible to expand the basic code completion suggestion list by applying same action once more
- Possibility to complete literals
- Basic code completion is performed after typing the middle of a word
- Since code completion works after typing any number of characters in the middle of a word, using asterisk wildcard is not required
- Second smart type code completion allows completing static method calls and constant references
- Action 'Fix Doc Comment'
- Hyperlinks to the symbols in the Quick Definition tooltip
- Pinning the Quick Definition tooltip results in showing Documentation tool window

- Pinning the Quick Documentation Lookup results in showing Documentation tool window
- Action 'Move Caret to Matched Brace' has been added
- Hungry Backspace action has been added to the platform

Compiler

- New approach to compilation

Configuring Project and IDE

- Possibility to turn on/off external compiler process
- Annotation profiles
- capability to rearrange Java code
- Possibility to rearrange ActionScript code
- Possibility to turn on/off parallel compilation
- Possibility to define bytecode versions on the project and module levels
- Darcula theme is available
- Option to change font size on the editor tabs
- Enabling notification in console on stderror/stdout for external tools
- Three possible behaviors on project opening
- Possibility to enable parsing PAC file
- Streamlined UI of Keymaps
- Possibility for groovyScript macro to take multiple arguments
- Customizable date format
- Language level: Java 8 now includes lambda support, type annotations etc

Refactoring

- Groovy: new options in extract method refactoring

Languages, Frameworks and Technologies

- Java 8 with lambdas, type annotations etc
- Code assistance for Drools Expert, a rule-based language from JBoss
- Basic TypeScript support
- Basic Dart support
- Basic Vaadin support
- Cucumber for Java and Groovy support
- CloudBees support
- Cloud Foundry support
- TomEE server support
- Support for FogBugz issue tracking system
- Support for Mantis issue tracking system
- Support for Assembla issue tracking system

Tool Windows

- New tool window provided for working with app server run configurations and associated artifacts

- New design of the Database tool window
- Documentation tool window
- Icon 'Export Inspection Results' is added to the toolbar of the Inspection tool window
- New design of the Structure tool window
- Scope-based TODO items

Navigation and Search

- Speed search in Live Templates allows searching for any text in the template abbreviation, body, or description
- Possibility to specify characters located anywhere inside name in question
- Navigating between custom regions
- Ability to switch between the 'Find in Path' and 'Replace in Path' dialogs
- Search for usages starts without a modal dialog box
- Possibility to find all files matching a certain mask regardless of the contents

Running

- Notification on stderror and stdout output in console
- New location of the check box 'Single instance only'
- Capability to group run/debug configurations in folders
- CloudBees run/debug configuration
- Cloud Foundry run/debug configuration
- Run/debug configuration for Cucumber Java
- Run/debug configuration for MXUnit
- Run/Debug configuration for TomEE server

Version Control

- Creating gists from console output is supported
- Subversion: editing messages for previous commits is available from the Repository tab of the Changes tool window, from the results of the Browse Changes action, and from the Show History for ZFIle/Folder
- Mercurial integration allows you to see all the files updated within a particular commit from the history of one of these files
- Possibility to remove untracked files that prevent check-out
- Advanced handling of platform-specific line endings (LF-CRLF).

  The Difference Viewer points at discrepancies in line endings (LF-CRLF). For Git repositories, %product% displays a warning when you are about to commit CRLFs and offers to set the core.autocrlf setting for you.
- TFS 11 (aka TFS 2012) is supported
- Possibility to color-mark directories with the changed descendants

Android

- When a drawable resource is deleted, IntelliJ IDEA suggests to delete the alternative resources as well
- For string resource extraction, the suggested values are selected by default
- Android 4.2 SDK (Jelly Bean) is supported

- Creating and Editing Layout using UI Designer
- Refactoring Android XML Layout Files
- Configuring Android application packages (APK) as artifacts
- Extracting Android application packages (APK) in the debug mode
- Logcat has become a tab of the Android tool window

Application Servers

- TomEE application server
- Cloud Foundry server
- CloudFBees server

Inspections

- JDK 8: Lambda-related inspections and intention actions, method and constructor references support
- Static import inspection: option to ignore static imports in test code
- JUnit inspections: old style JUnit test method in JUnit 4 class
- TestNG inspections:

  assertEquals() between objects of inconvertible types; Expected exception never thrown in test method body; Message missing on TestNG assertion; Misordered 'assertEquals()' arguments.
- Probable bugs: Modified inspection 'Constant conditions & expressions'
- Internationalization issues: Modified inspection 'String concatenation' now has the option to ignore string concatenation inside toString methods
- Modified inspection 'Unnecessary 'this' qualifier' now has the option to ignore assignments to fields
- Modified Java - Control Flow issues inspection 'Unnecessary 'continue' statement' now has the option 'Ignore in 'then' branch of 'if' statement with 'else' branch
- JQuery selector inspection for detecting JQuery selectors anti patterns added

Intention Actions and Quick Fixes

- Groovy: New Intention actions

  'Remove unnecessary return', 'Alias import statically imported member', 'Copy String concatenation text to the clipboard', 'Change access modifiers (private/protected/public)' and more.
- Groovy: ability to see where the methods came from in the code completion pop-up

  Now, in the method suggestion list you can see which mechanism was used to invoke a certain method.
- Groovy: A mixin support is now available

PHP Support

- Syntax highlighting in .ini files
- PHAR format is supported: browsing archives, code completion, search, reference resolution, etc
- Using custom coding standards with the root directory outside the default PHP Code Sniffer's Standards directory
- Running PHP Code Sniffer inspection in batch mode

- Model View Controller for Symfony2 and Yii frameworks
- Associating a keyboard shortcut with a Phing build target
- Configuring Phing build properties externally
- Using IntelliJ IDEA macros in externally configured Phing build properties

Seam Support

- Refactoring Android XML Layout Files

Spring Support

- Spring Batch support added
- Spring Integration support extended to version 2.2
- Spring Web Flow support is now available
- Spring Security support has been enhanced

Groovy and Groovy-Based Frameworks

- Possibility to assign colors for labels in Groovy code

  New option 'Label' is added to the list of Colors & Fonts for Groovy in Editor Settings.

- New Intention action

  'Remove unnecessary return', 'Alias import statically imported member', 'Copy String concatenation text to the clipboard', and more.

- Option to add spaces within GString injection braces

  New option 'GString injection braces' is added to Spaces tab in Code Style Settings.

- Extract Parameter refactoring for Groovy
- Griffon: ability to create additional Griffon structures
- Improve formatting for closures
- New options in extract method refactoring
- Possibility for groovyScript macro to take multiple arguments

Data Access Support

- Structure view is available for DDL SQL files
- You can run SQL scripts for a number of data sources at once
- You can edit definitions of views, functions, procedures and packages

ActionScript and Flex

- Integration with the Apache Flex SDK provided
- Improved structure view for ActionScript classes and MXML components
- The SWF metadata tag can now be used to control HTML wrapper properties
- You can now run and debug your FlexUnit tests using a built-in mobile device emulator
- Application debugging is supported for iOS mobile devices
- iOS Simulator support provided for Mac computers
- You can now use one SDK to build your app and then use a debugger from a different SDK
- ActionScript Compiler 2.0 (ASC 2.0) is supported
- New easier way to specify dependencies on runtime-loaded modules

JavaScript Support

- New CoffeeScript refactorings
- Advanced configuration settings for JSHint and JSLint code quality tools
- JQuery selector inspection for detecting JQuery selectors anti patterns added
- JavaScript libraries can be configured at the project level (project and global libraries)

Testing Support

- JUnit support: Inspection 'Old style JUnit test method in JUnit 4 class' added
- TestNG support: JUnit inspections are ported to TestNG
- Possibility to view byte code right from the statistics pop-up
- Cucumber for Java and Groovy support
- Highlighting of undefined steps in Cucumber feature files
- Code completion for feature descriptions
- Intention action that allows creating Cucumber step definition of an undefined step
- Inspection that highlights duplicated step definitions
- Possibility to navigate from a .feature file to a step definition
- Scenario outline support, with syntax highlighting, code inspections and quick fixes

  Quick fix to create missing Examples section; Quick fix to add missing colon after the keyword Examples.
- Running Cucumber tests: all feature in a directory, a feature or a single scenario within a feature
- MXUnit support

Remote Hosts

- Configuring automatic upload of changed files to the default server so only manually saved files are deployed automatically
- Ftps synchronization

Command Line Tools

- Integration with command line tools is provided at IntelliJ IDEA level, so once configured, a tool is available in all IntelliJ IDEA projects
- Executing command line scripts automatically after committing changes to the repository
- Reloading Symfony commands from the executable file without re-configuring the tool
- Support of Symfony2
- Integration with the Yii command line tool is supported

Java EE and Web Development

- Cloud Foundry support
- CloudBees support

Miscellaneous Improvements

- PROJECT_NAME variable is available in file templates
- Managing tasks and context: integration with PivotalTracker, Redmine, Trac, FogBugz, Mantis, Assembla, and Generic Server issue tracking systems is supported
- Multiselection is available in the list of tasks. Several tasks can be deleted at once

- New way to store product memory heapsize setting on MAC
- Ability to find RGB, HSB and hex values of any color component
- Possibility to switch between the panes of the Differences viewer
- Use soft wraps option
- Intention actions to check and edit regular expressions
- New way to manage cases of unicode literals on OS X

What's New in Version 11.1

IDE

- ALT+Click editor tab to close other tabs
- Pinch-to-Zoom in the editor tabs
- Possibility to view all editor tabs and choose the active tab

Code Editing

- Sticky selection
- Custom code folding regions
- Live template soutp added for Java and Groovy
- Possibility to surround with region folding comments
- Chained expression completion
- Emacs actions can be invoked via Go to Action
- More Emacs actions have been added to the platform (kill ring, sticky selection, adjust line); keyboard shortcuts have been added in the Emacs scheme

Configuring Project and IDE

- New way to access IDE Settings via Go to Action
- New way to access project Settings
- Improved code style settings
- Code Style Schemes dialog provides a better way to manage code styles
- Possibility to hide navigation pop-up frames on focus loss
- Disable mnemonic in controls allows the Mac users to search for @ sign in the find in path dialog box
- Preselect old name
- Option to automatically collapse closures is enabled by default
- Possibility to recognize line comments if it starts at the very first position on a line
- Command line options for Chrome

Languages, Frameworks and Technologies

- Improved Gradle support with numerous new features
- Possibility to preview compiled CoffeeScript files

Tool Windows

- Coverage tool window
- JetGradle tool window
- New design of the Project tool window

Navigation and Search

- Possibility to include non-menu commands by pressing shortcut once more

Running

- Improved coverage with a tool window to display results
- Possibility to launch CoffeeScript files directly, without conversion to JavaScript
- Ability to run interactive Groovy console (Groovy shell)

Debugging

- Hot-swap settings is now available for Groovy classes

Version Control

- Streamlined UI for Git branches
- Creating new Git branches
- Checking out Git branches
- New merging, deleting, and comparing Git branches
- Ability to switch to Perforce online mode in the Changes tool window
- For Perforce integration, two refresh modes are supported
- Integration with Subversion 1.7 is supported
- Local working copies can be created with Subversion 1.7
- Possibility to notify about VCS root errors
- Check box for memorizing password

Intention Actions and Quick Fixes

- New intention actions

  Intention action to create @flt variable comments right inside macros and functions is available on parameter declaration.

PHP Support

- PHP built-in Web server supported in projects with PHP interpreter 5.4
- Checking PHP code quality on the fly using PHP Code Sniffer
- Creating unit tests using PHPUnit Skeleton Generator
- Code coverage for PHPUnit testing
- Setting predefined Drupal coding standards as default

Groovy and Groovy-Based Frameworks

- Possibility to link a Gradle project to IntelliJ IDEA project
- Groovy Shell is available in Grails applications
- Live template soutp added
- Chained expression completion
- Access to a Groovy console in Java projects and Grails applications
- Possibility to add a special agent for hot-swapping Groovy classes
- Extended color scheme for Groovy (local variable, reassigned local variable, parameter, reassigned parameter, method declaration)

- Dedicated tool window for Gradle

ActionScript and Flex

- One SDK type (Flex SDK) for the various target platforms
- Build configurations are now used to control compilation and packaging
- New run/debug configuration types (Flash App and Flash Remote Debug)
- Convenient UI for working with build configurations
- Improved UI for AIR application packaging

JavaScript Support

- Code coverage for JavaScript Unit testing using the istanbul coverage tool
- Switching between JavaScript language versions to choose the one that fits the targeted browser
- JavaScript Strict mode supported

Testing Support

- Possibility to configure coverage colors right from the statistics pop-up

Miscellaneous Improvements

- Ability to print out help pages in pdf format from the web site
- MONTH_NAME_SHORT and MONTH_NAME_FULL variables are available in file templates
- ability to manage case of unicode literals
- Web Module: a new module type for developing web applications using programming languages other than Java

What's New in Version 11.0

IDE

- Possibility to preserve temporary files
- Possibility to import project from Gradle
- Possibility to reopen a project from the Welcome screen
- Possibility to drag and drop a project onto the Welcome screen
- Possibility to create modules around existing source

Code Editing

- Column selection mode
- Smart Enter now applies to JavaDoc
- Expanding suggestion list on second pressing action shortcut
- Highlighting level of the current file can be configured in the Analyze menu
- New PHPDoc formatting options in compliance with Zend, PEAR and other standards

Configuring Project and IDE

- Ability to copy code style from another language
- Possibility to attach sources on-the-fly
- Possibility to create libraries using JAR files selected in the Project tool window

- Possibility to move a module library to the project or global level
- Possibility to create a copy of a global library at the project level
- Possibility to move a project library to the global level
- Live templates can be created and edited right in the Settings | Live Templates dialog box
- Predefined code style for Symfony2 and in accordance with the PSR1/PSR2 PHP coding standards
- Language-specific code style settings
- Smart end in JavaDoc comments
- Smart indent in JavaDoc
- Safe write mode
- More convenient user interface for managing library contents
- New UI for configuring facet auto-detection

Languages, Frameworks and Technologies

- Velocity version 1.7 is supported
- Improved Gradle support
- CoffeeScript support
- Play! support

Tool Windows

- Tool Windows are now available on the View | Tool Windows menu
- Event Log tool window shows all IDE notifications and allows you to take action where needed
- Dedicated tool window for managing your project favorites
- Dedicated JSTestDriver tool widow for starting the JsTestDriver server for running JavaScript unit tests in the browser
- Tool windows are accessible from the view menu
- Structure tool window shows anonymous classes

Navigation and Search

- Navigation commands are available on the Navigate menu
- Possibility to navigate to anonymous classes
- Search functionality is in the Edit menu
- Multiline search and replace in the current file

Running

- Possibility to navigate from backtrace in the Watches tab to source

Debugging

- Ability to automatically detect a stack trace in the clipboard
- Adjust Range command is now available for lists

Markup languages and style sheets

- Less support
- Updated HTML 5 schema

- Zen coding 0.7

Version Control

- Reviewing changes to be checked in (Digest view) right in the Commit dialog box
- Viewing local changes from base revision in a dedicated pane on the Local tab of the Changes tool window
- Git Fetch is performed silently, without showing the Fetch Settings dialog box
- Initial support of git gists for sharing code snippets on GitHub
- Possibility to view change details for a file
- Revision graphs for Git
- Details section allows you to preview results before commit

Android

- Android 4 SDK (Ice Cream Sandwich) is supported
- IntelliJ IDEA detects the resource type and qualifier of a new Android resource and saves it accordingly in the relevant folder
- Previewing layout from the editor without launching a physical device or emulator. All the changes are reflected immediately in a dedicated Android Preview tool window
- Built-in obfuscation for Android applications
- Running Android applications on USB devices supported

Intention Actions and Quick Fixes

- 'Check RegExp' checks regular expressions on-the-fly
- 'Inject Language - RegExp' converts any string into a regular expression

PHP Support

- New PHPDoc formatting options in compliance with Zend, PEAR and other standards
- Debugging single php http requests supported
- Generating PHP unit tests improved

Web Resources

- Synchronizing folders in the difference viewer

Spring Support

- Improvements in Spring support

  Improved performance for large models, new features in bean dependency diagram, "Generate @Autowired dependency" action, etc.; Spring 3.1 @Configuration contexts, @ComponentScan, @ImportResource, @Profile, @PropertySource; Spring Testing: @ContextConfiguration configured with @Configuration array, @ActiveProfile.

- Spring Data JPA is supported

Groovy and Groovy-Based Frameworks

- New Groovy intention actions (Replace qualified reference with import; Add single-member static import; Add on-demand static import)
- Unwrapping statements is now available for Groovy

- Grails 2.0.0 features (Controller actions as Methods, new JUnit Testing API, access to Gorm API from Java classes, Detached Criteria, etc.)
- Web Flow support
- Grails Resources plugin
- Standalone GORM
- Spock framework
- Groovy 1.9 support
- Introduce Parameter refactoring for Groovy can be used to introduce closure parameters
- Code Coverage in Grails run/debug configuration

UML

- Changing module dependencies with UML diagram

Data Access Support

- Possibility to change an SQL dialect for an SQL or DDL file open in the editor

JavaScript Support

- The mark object action is available in the JavaScript debugger
- JavaScript unit testing
- Built-in JSLint code quality checker
- Node.js core module sources can be configured both as a global or as a project library
- Node.js: coding assistance, running, debugging, and unit testing

Testing Support

- Code Coverage colors can be changed
- Single action to run with coverage
- Unit Testing for JavaScript. JSTestDriver Assertion, QUnit, and Jasmine frameworks are supported
- Running JavaScript unit tests in browser
- Generating PHP unit tests improved

Remote Hosts

- Synchronizing local and remote folders in the difference viewer

Build Tools

- Importing project from external Gradle model
- Import from Gradle

Java EE and Web Development

- Jboss 7 support
- WebSphere 8 support

Miscellaneous Improvements

- Quick hide/show tool window buttons
- Macros functionality is in the Edit menu

- Possibility to compare binary files
- Hierarchies are built in the Navigate menu
- Find Action is in Help menu
- New way of creating and deleting tasks via tasks combo
- Possibility to view task description
- Possibility to show tasks combo in the main toolbar

What's New in Version 10.5

Code Editing

- Moving lines up and down
- CamelHumps in code completion are detected without using the Shift key
- Horizontal arrow keys in code completion
- Sorting of a suggestion list
- In JavaDoc, the caret is placed to an expected position on pressing Enter
- Changing font size in the Quick Documentation window
- You can drag-and-drop schema, table and field names from the Data Sources tool window into the editor

Configuring Project and IDE

- Horizontal scroll bar is not shown when soft wraps are used
- Console history depth is configured for the consoles of all types
- Adding closing tag in JavaDoc
- Option to automatically collapse multiline //-style comments

Refactoring

- Inline Method refactoring allows viewing the number of invocations of a method
- In-place ExtractConstant refactoring for Java
- In-place Introduce Field refactoring is available for Java
- In-place Introduce Parameter refactoring for Java
- Improved Introduce Variable/Field/Constant refactorings for ActionScript

Languages, Frameworks and Technologies

- Java 7 is supported
- Google app engine: arguments can be passed to the server

Tool Windows

- Project tool window: Copy reference works for packages in the Package view
- Data Sources tool window: you can drag-and-drop schema, table and field names into the editor
- Data Sources tool window: schema objects can now be grouped by their types
- A new tool window for working with Spring Roo console is available

Navigation and Search

- The switcher can navigate between the split editor tabs and floating tool windows

- Navigation between JavaScript tags in HTML files
- Scroll to top / scroll to bottom
- Swipe gestures for back and forward navigation are supported for os x
- Improved search and replace in the current file
- Regexp shows replacement preview
- Easy initiating new search
- Horizontal scrolling with the mouse wheel

Running

- Run/debug configurations for Jetty are now available

Markup languages and style sheets

- Xslt 2.0 and XPath 2.0 support

Version Control

- ability to review TODO items before commit
- Setting up connection to Perforce server has become highly fail-proof
- Viewing the GitHub version of a file from IntelliJ IDEA

Android

- The Honeycomb platform is supported
- Debugging an already running process is available
- Android DX tool supported

Inspections

- Simplify annotations

  Reports spaces between @ and name, unnecessary parentheses and value="foo" constructs which can be shortened, and offers a quickfix. (Since 10.0.2).

- Double literal cast to float

  Reports double literal expressions which are immediately cast to float, and offers to replace with an equivalent float literal. (Since 10.0.2).

PHP Support

- Viewing parameter info for methods defined through the @method phpDocumentor tag is available
- Dedicated Command Line Tools tool window
- IntelliJ IDEA recognizes .htaccess files and provides syntax highlighting, formatting, code completion and documentation lookup for common directives
- Multiuser debugging via XDebug proxy servers supported
- Command Line Tools functionality is provided through a dedicated tool window with the capability to navigate through the list of executed commands and their output and save the output in a text file
- Phing build tool is supported
- Workaround for the XDebug on FreeBSD crash can be enabled
- Zero-configuration debugging

Spring Support

- Spring Roo console is now available
- The Spring 3.1 c-namespace support is available
- The Spring 3.1 bean definition profiles are supported
- The Spring Integration framework is supported
- Spring profiles panel can be shown in the editor

Groovy and Groovy-Based Frameworks

- Intention action: if the caret is placed on a field in a domain class, an intention action suggests to make this field nullable, or unique
- Intention action: create action from usage
- Introduce Field refactoring is available for Groovy
- Creating actions and views from usage
- Groovy 1.8 support
- Intention action to create Groovy tests
- Navigation to Groovy tests
- Introduce Parameter refactoring for Groovy is available
- Grails tool window shows the Scripts directory

Google Web Toolkit (GWT) Support

- Coding assistance, including code completion, error highlighting, and finding usages, for declarations of external resources through the <ui:with> element
- Creating GWT event and event handler classes supported
- The possibility to choose the starting page of the application even if showing the application in the browser is suppressed
- Flexible configuration of the GWT compiler with the possibility to specify for each GWT module whether you need it compiled on make or not

UML

- Possibility to view changes on UML diagrams for ActionScript classes

Data Access Support

- You can create data sources by importing Tomcat configuration files (context.xml)
- A dedicated dialog is now available for working with the history of executed SQL statements
- Updated differences viewer for folders and database objects
- Table and column usages include database schema usages
- Possibility to compare schemas of two data sources
- You can drag-and-drop schema, table and field names from the Data Sources tool window into the editor
- JDBC drivers are downloaded right from Data Sources tool window

ActionScript and Flex

- Possibility to view changes in ActionScript source files in a structured form on UML diagrams
- When moving an ActionScript class or interface to a package, a package chooser is now available

- The Move Inner to Upper Level refactoring is available for moving out-of-package entities into a package
- Development of AIR applications for mobile devices is supported
- Import of FXP files is supported
- Type chooser in the Introduce Variable/Field/Constant dialogs for ActionScript
- When importing Flexmojos projects, you can turn auto-generation of Flex compiler configuration files on and off

JavaScript Support

- JavaScript debugging is now available for Chrome

Testing Support

- Ability to choose language for a test class
- capability to turn off display of the Code Coverage dialog

Build Tools

- Keep source and test folders on reimport

Java EE and Web Development

- Integration with Jetty is now supported

Miscellaneous Improvements

- Jetty integration plugin comes bundled with IntelliJ IDEA
- Updated ruby on rails, php, Python/Django and Scala plugins

What's New in Version 10.0

Projects and Modules

- You can now control which library classes are included in the compilation result and which are not using the Export and the Scope options

Code Editing

- You can now open and modify language injections in the editor, as if you were working with the source code in the corresponding language
- When performing the Introduce Variable refactoring, you can specify the name of the new variable right in the editor

Refactoring

- When performing the Introduce Variable refactoring, you can now use smart expression selection in JavaScript and ActionScript
- The in-place code editing is now available for the Introduce Variable refactoring
- When working with Flex and ActionScript, you can now use the Change Method Signature, Delegate Methods, Extract Interface, Extract Superclass and Introduce Parameter refactorings
- Search for references option is now available for Rename
- Search for references option is available for moving files and directories
- Search for references option is available for renaming files

Code Analysis

- Analysing source code for nullable and non-nullable elements

Languages, Frameworks and Technologies

- Google App Engine. Support of JPA and JDO persistence
- Google App Engine. Run enhancer is available for JDO and JPA

Running

- Referencing labeled variables, watches, and objects by labels is available
- Default run/debug configurations are presented under the Defaults node
- Possibility to automatically bring the editor forward on hitting a breakpoint

Markup languages and style sheets

- SASS3 support, with syntax highlighting, indentation-based code folding, and color editor
- LESS support, with compilation and coding assistance

Version Control

- On-the-fly spellchecking in commit messages is available
- Integration with GitHub is available. Now you can register GitHub accounts, create and clone repositories from IntelliJ IDEA

Android

- Navigation between resource definition files and R.java by clicking the Go to icon in the left gutter area
- Unit testing for Android applications is supported
- Sharing Android source code and resource using library projects is available
- Extracting a signed Android package is available

Inspections

- Running inspections by name

PHP Support

- Integration with Zend Debugger is supported
- Integration with the Zend Framework is available

Groovy and Groovy-Based Frameworks

- Search for usages is available for *.gsp files
- Rename refactoring is available for *.gsp files
- Grails filters can be created
- Means of navigation between Grails controllers and views
- History is now available in the Run Grails Target dialog box
- Debugging *.gsp files
- Find Usages is now available for the variables defined via g:set tag, and for the fields of the domain classes

Google Web Toolkit (GWT) Support

- GWT UiBinder is supported
- The possibility to choose whether to open a GWT application in one of the configured browsers or not

UML

- UML Class diagram is available from different places, for example, from the Navigation bar
- creating notes in UML class diagram

Data Access Support

- Input pane of the Database console opens as an editor tab. Editing tables is available from the Results pane
- Operations with database binary large objects (BLOBs) are supported
- Tables can be edited using a graphical Table Editor. The range of records to be displayed can be controlled through filters
- Possibility to view parameter information for prepared SQL statements

ActionScript and Flex

- The Change Method Signature, Delegate Methods, Extract Interface, Extract Superclass and Introduce Parameter refactorings are now available for Flex and ActionScript
- Smart code completion is now available
- You can generate ActionScript classes and class members using UML diagrams
- The Create Subclass and Implement Interface intention actions are now available for ActionScript classes and interfaces
- A multithreaded built-in Flex compiler shell is now available

Testing Support

- Unit testing for Android applications is supported

Build Tools

- Introduce Property refactoring
- Viewing Maven dependencies in a UML class diagram, excluding dependencies, navigating to source code
- Ability to download external libraries via Maven public repositories
- Possibility to use corporate repositories, if they are managed by Nexus

Java EE and Web Development

- Code completion, error highlighting, and basic refactorings are now available for the most popular JSF component libraries
- WebLogic server run/debug configurations have become more flexible
- ColdFusion is supported

Miscellaneous Improvements

- If the main toolbar is hidden, the docked Navigation bar shows run/debug configuration controls
- Power save mode
- Detaching editor tabs
- Task and Context Management is now available in the Community edition

- Support for JIRA, YouTrack, Lighthouse, Pivotal Tracker, RedMine and GitHub issue tracking systems
- XML breadcrumbs

What's New in Version 9.0

Projects and Modules

- Dependency Scopes: compile, test, runtime, provided

Code Editing

- Autofolding now includes anonymous classes, generic parameters, i18n strings, and more
- Built-in spellchecker
- Several live templates can have same abbreviations
- Create Java class action
- Asterisk * wildcard can be used in code completion popup
- Easy class exclusion from auto-import and code completion

Configuring Project and IDE

- New masks are added to the list of resource patterns in compiler options

Refactoring

- Improved refactoring feedback informs you about the potential conflicts

Code Analysis

- Dataflow from here

Languages, Frameworks and Technologies

- Google app engine support
- Tapestry integration

  Tapestry integration includes syntax and error highlighting, navigation between classes and templates, intention actions for creating components, pages and mixins, and code analysis.
- OSGi support

  OSGi application development includes facet, run/debug configuration, syntax and error highlighting, code completion, and automatic project configuration.
- Android module
- Scala support available via Scala plugin, with the dedicated facet and coding assistance
- La Clojure support available via La Clojure plugin, with the dedicated facet and coding assistance
- PostgreSQL and Derby SQL dialects are now supported
- Running and debugging Android applications
- OSGi facet

Tool Windows

- Tool window tabbed content mode

Navigation and Search

- Using switcher to navigate between open files and tool windows
- Declaration pop-up window for annotation types now shows @ to tell annotation types from interfaces
- Improved UI of the Show Usages pop-up window
- Possibility to specify multiple comma-separated file masks in the Find/Replace in Path dialogs

Running

- Run application without using toolbar: Alt+Shift+F10
- Dedicated tab for dump analysis

Version Control

- Subsets of changes to apply can be flexibly configured
- Integration with Subversion 1.6 is supported
- Configuring protection of inactive changelists
- VCS quick list
- Viewing differences between the local copies of all the project files one after another and their updates from the server

Android

- Developing Android applications using the emulator and a real device

Inspections

- Convert to atomic intention action

  "Convert to atomic" intention action replaces variable type with the corresponding atomic type.

- New code inspections

  New code inspections powered by InspectionsGadget, for example, related to 'assert' statement.

- Global unused declaration inspection

PHP Support

- Php development and debugging
- Creating PHPDoc blocks
- Run/debug configuration: php web application
- Run/debug configuration: PHPUnit
- Run/debug configuration: PHPUnit on server

Spring Support

- Spring Security 3.0 Support

Groovy and Groovy-Based Frameworks

- Smart type code completion after new in Groovy
- Gradle is now supported
- Simplified way of creating Grails or Griffon Application modules
- Groovy classes, interfaces, enumerations and annotations are created in a single action

- Run/Debug configuration for Griffon applications
- Per-project Gant home setting
- Per-project Gradle home setting
- Grails tool window
- Griffon tool window

UML

- Configuring default settings for UML class diagram
- Adding node elements to UML class diagram
- Viewing changed classes in UML class diagram
- Viewing subtypes, super classes, and classes used in signatures
- UML presentation of changed classes is now available from the Changes tool window

Data Access Support

- Possibility to view parameter information in JDBC console

ActionScript and Flex

- Ctrl+Q for AsDoc
- FlexUnit
- Easy import from Adobe Flash Builder
- New Flex refactorings: Extract Method, Introduce Variable, Introduce Constant, Introduce Field, Inline, Pull Members Up, Push Members Down
- Type, Method, Call Hierarchy for Flex sources
- Automatic generation of Bindable getters/setters, event handlers

Build Tools

- Fully integrated Maven 2.2 support
- Find Usages
- Improved navigation between modules and pom dependencies with Ctrl+B
- Rename refactoring for properties defined in custom filters files
- Classpath for Maven-based projects is built following Maven dependency mechanism
- Add Maven Dependency quick-fix for unresolved classes in Java code
- Parent and dependencies generation in pom files with Alt+Insert

Java EE and Web Development

- Surround with () for HQL/JPAQL
- Updating a running Java ee application
- Jsf 2.0 support

Miscellaneous Improvements

- Improved file indexing

  IntelliJ IDEA performance is way better due to backgroundable file indexing, with limited navigation / editing / VCS functionality being available during the indexing process.

- On-the-fly module reloading

  IntelliJ IDEA features automatic module reloading, which means that changes in .iml files (in particular, after updating from version control) no longer require reopening the entire project.

- Platform-specific ide help shows keyboard shortcuts for your particular operating system
- Possibility to open files in associated applications
- Possibility to perform clipboard operations between IntelliJ IDEA and Explorer/Finder
- Psi structure viewer
- Preview of small icons in Java classes