Oracle® JavaScript Extension Toolkit (Oracle JET)

What's New in Oracle JET

19.0.0

G33838-02

August 2025

New Features Described in Oracle JET's Documentation for Release 19.0.0

Here are updates to the documentation for Oracle JET to describe new features and enhancements added to Release 19.0.0.

For other changes in this release, see the product Release Notes.

Oracle JET

Documentation updates for this release of Oracle JET.

- Oracle JET CLI `add translation` command and support for ICU translation bundles

  A new command-line argument (`add translation`) facilitates the creation and management of ICU translation bundles in Oracle JET apps. See Work with ICU Translation Bundles in an Oracle JET Virtual DOM App.

- Configuration property to generate source map files

  A new property, `generateSourceMaps`, in the `oraclejetconfig.json` file controls the generation of source map files. When `true`, the Oracle JET CLI configures Terser and RequireJS packages to generate source map files when you build your Oracle JET app. Its default value is `false`. See Properties in the oraclejetconfig.json File and Debug Web Apps.

- Configuration property to enable use of NPM's `--legacy-peer-deps` flag

  The `oraclejetconfig.json` file supports a new configuration property, `enableLegacyPeerDeps`, which, when set to `true`, implements support for the optional inclusion of the `--legacy-peer-deps` flag during NPM package installation. When `true`, the Oracle JET CLI modifies any invoked existing NPM install commands, such as the Oracle JET CLI's `add docgen` command, to include the `--legacy-peer-deps` flag. You need to add the `enableLegacyPeerDeps` property to the `oraclejetconfig.json` file if you want to use this optional capability. The default value of `false` means that NPM installations proceed without the `--legacy-peer-deps` flag. See Properties in the oraclejetconfig.json File.

- TypeScript

  This release of JET uses version `5.8.3` of TypeScript. For more detail about TypeScript in Oracle JET apps, see Configure Oracle JET Apps for TypeScript Development.

- Migration log file

  The Oracle JET CLI's `migrate` command now writes a file named `ojet.migrate.log` to the root directory of the Oracle JET app that you attempt to migrate. This log file contains details about the updates (for example, properties added to the `oraclejetconfig.json` file) that the `migrate` command makes. For more information about migrating your Oracle JET app to this release, see Oracle JET App Migration to Current Release.

Oracle JET Audit Framework

Using and Extending the Oracle JET Audit Framework has been updated in the following ways.

New and changed JAF audit features:

- Oracle JAF now supports WAI-ARIA 1.2, including properties like `aria-modal` introduced in WAI-ARIA 1.1. Previously, it supported WAI-ARIA 1.0.
- Oracle JAF in a non-Oracle JET project

  If an Oracle JAF configuration file (for example, `jafconfig.json`) sets the `typescript.compile` property to `true` for VComponent compilation, but no `oraclejetconfig.json` file is found in the root directory (indicating a non-Oracle JET project), Oracle JAF now continues with audits where previously it terminated due to an inability to set up VComponent compilation.

- JAF includes updates to existing rules:
  - The `csp-html-unsafe-expr` rule now detects use of `async await` in a JET expression.
  - The `jetwc-use-public-apis` rule in the `builtinJetWcRules` rule set is now enabled by default.
  - `oj-acc-select-aria-label`

    This rule no longer checks no longer checks the `oj-c-select-multiple` and `oj-c-select-single` Core Pack components to avoid false positives.
  - `oj-html-alta-deprecated`

    This rule's message has been revised to request you to remove all Alta CCS or SCSS as soon as possible.
  - `oj-html-form-bindif, oj-tsx-form-bindif, and oj-html-binding-as-slot-child`

    The `oj-html-form-bindif` and `oj-tsx-form-bindif` rules now handle the `oj-c-form-layout` Core Pack component in addition to the legacy `oj-html-form-layout` component, while the `oj-html-binding-as-slot-child` rule now exempts both `oj-c-form-layout` and `oj-form-layout`.

  - The severity of the `oj-html-msgs-cat-text` rule is elevated to `blocker` in the `redwood-strict` profile.
  - `oj-html-wctag-for` and `oj-html-wctag-id`

    These rules now check that the related `component.json` metadata has a *`type`* property of *`composite`* or is omitted.
  - `oj-tsx-ojattr`

    The severity of this rule is now critical for the case where a kebab-case attribute is assigned a string of the form `[[…]]` or `{{…}}` and, additionally, warns in a message that the result is coerced to a string.

- JAF includes new rules:
  - `oj-acc-navigable-link and oj-tsx-acc-navigable-link`

    ```
    Rule: oj-acc-navigable-link  Severity: blocker  Message ID:
    jet-2190
    Rule: oj-tsx-acc-navigable-link  Severity: blocker  Message ID:
    jet-7278
    In order to be keyboard accessible, hyperlinks must be defined
    with either an 'href' or a 'tabindex'
    property (or both).
    ```
  - `oj-css-bp-zindex`

    ```
    Rule: oj-css-bp-zindex  Severity: major[1]  Message ID's:
    jet-6075, 6076
    For best practices, CSS z-index property values should not be
    used. Use of JET standard secondary
    window components (popup, dialog or messages) is recommended for
    proper UI element stacking.

    [1] Possibility of more than 1 severity since there are 2 message
    ID's
    ```
  - `oj-cca-compjson-prop-conflict`

    ```
    Rule: oj-cca-compjson-prop-conflict  Severity: major[1]  Message
    ID's: jet-8045, 8046, 8047
    JET custom components should not define custom property names
    that conflict with HTML global
    attributes. Overriding standard HTML APIs in a custom component
    implementation can cause unexpected
    behavior and bugs that are hard to diagnose.

    As an alternative, JET custom VComponents have the ability to
    observe a subset of the available
    global attributes of their HTML custom element. Refer to the
    ObservedGlobalProps helper type in the
    JET VComponent documentation for further details.
    ```

[1] Possibility of more than 1 severity since there are 3 message ID's

- oj-html-badge-maintenance

  Rule: oj-html-badge-maintenance Severity: info Message ID: jet-0017
  class 'oj-badge' is moved to maintenance status starting in JET 19.0, and is superseded by
  oj-c-badge.

- oj-html-corequired and oj-tsx-corequired

  Rule: oj-html-corequired Severity: blocker Message ID: jet-2260
  Rule: oj-tsx-corequired Severity: blocker Message ID: jet-7280
  Some components need to be configured such that at least one property or slot must be populated out
  of a given set for the component to function correctly (the 'anyOf' requirement), or alternatively
  the component should not be configured with properties and slots from a given set (the 'not'
  requirement). This requirements are often used for accessibility.

- oj-html-oj-ux-ico-slot

  Rule: oj-html-oj-ux-ico-slot Severity: blocker Message ID: jet-8040
  Menus and other action components that can be configured with an icon starting with
  'oj-ux-ico-', should always have the icon applied via a span tag in the assigned
  slot. Applying the icon class anywhere else will result in the component not
  functioning as expected.

- oj-html-redwood-ojoption-disabled

  Rule: oj-html-redwood-ojoption-disabled Severity: blocker Message ID: jet-8055
  By extending the profile 'redwood-strict', Redwood compliance is required by the Audit. Redwood
  compliant designs do not allow individual child <oj-option> components of <oj-radioset> and
  <oj-checkboxset> to be disabled, and should be removed. If an option is not available for
  interaction by the user, it should not be included in the set of options.

  Extend the `redwood-strict` profile to enable this rule. It is disabled by default.

- oj-html-nav-tab-nointeractive

  Rule: oj-html-nav-tab-nointeractive Severity: blocker Message ID: jet-8050
  The <template> 'itemTemplate' slot in <oj-tab-bar> and <oj-navigation-list> must not contain
  interactive tags. That is, content other than <li>, <a>, , , <oj-badge>, text,
  <oj-bind-text>, <oj-bind-if>, <oj-bind-for-each>, or <oj-if>.

- oj-tsx-tabbar-progressive-popup

  Rule: oj-tsx-tabbar-progressive-popup Severity: minor Message ID: jet-7285
  For JET 18.0.0 and later, <oj-tab-bar> truncation="progressive" should always
  be used with overflow="popup" for handling overflow properly. If truncation="progressive" is used alone, for long labels it is possible that
  all items may not fit within oj-tab-bar despite being truncated.

- oj-tsx-prop-ko-binding

  Rule: oj-tsx-prop-ko-binding Severity: critical Message ID: jet-7275
  When components are used in VDOM, properties should be using the appropriate VDOM
  style of binding syntax which uses single curly braces. Knockout style binding
  syntax using string values with pairs of square brackets or curly braces will be
  simply treated as a literal strings and this is likely to have unexpected
  effects on the operation of the component.

- jetwc-use-public-apis

  Rule: jetwc-use-public-apis Severity: blocker[1] Message ID's: jetwc-0250, 0251,
  0252, 0253
  This rule checks that any components or APIs used by a component are
  either from within the same pack or are marked as public and from a declared
  dependency

  [1] Possibility of more than 1 severity since there are 4 message ID's

Changes for rule writers:

- A new API, `isFolderWritableSync()`, in the file system utilities library allows you to check if a folder has write access permission. You can access this API through `<rule_context>.utils.fsUtils`. See FsUtils: File System Functions.
- New listener types:
  - `jetcorecomp` listens for Oracle JET Core Pack components.
  - `globtag` has been extended to optionally accept the name of an HTML global element as an argument. For example, `"globtag(a)"` or `"globtag(span)"`.
  - `attr-aria` for HTML and `TsxAria` for TSX allows a rule to listen directly for all `aria-*` properties instead of having to examine all element properties.
  - `requirements` for web components that have `requirements` metadata.
  - See Listener Types for HTML and JSON Rules.
  - `TsxRequirements` is called for web components that have `requirements` metadata. See Listener Types for TSX Rules.
- During the JAF pre-audit phases, the `jafLib` interface, available through `ruleCtx`, has been expanded to include the `isCLI()`, `isAMD()`, and `isAPI()` methods, which were previously available only in the main audit phase. See JafLib: JAF Core Access Methods.

File path separators for the Windows platform

A future release of Oracle JAF will remove support for glob paths with escaped backslashes on Windows. To prepare for this change, review `oraclejafconfig.json` properties like `"files"`, `"exclude"`, and `"components"` and replace values that use escaped backslashes to use forward slashes.

For example, if you encounter a value similar to the following for the `"files"` property:

```
"files": [
      ".\\src\\**\\*.html",
```

replace it as follows:

```
"files": [
      "./src/**/*.html",
```

Failure to make this change will result in referenced files not being processed by JAF on Windows in a future release.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Oracle JavaScript Extension Toolkit (Oracle JET) What's New in Oracle JET, 19.0.0
G33838-02