

# Package ‘JBrowseR’

July 20, 2022

**Title** An R Interface to the JBrowse 2 Genome Browser

**Version** 0.9.1

**Description** Provides an R interface to the JBrowse 2 genome browser.

Enables embedding a JB2 genome browser in a Shiny app or R Markdown document. The browser can also be launched from an interactive R console. The browser can be loaded with a variety of common genomics data types, and can be used with a custom theme.

**License** Apache License (>= 2)

**URL** <https://gmod.github.io/JBrowseR/> <https://github.com/GMOD/JBrowseR>

**Encoding** UTF-8

**RoxygenNote** 7.1.2

**Imports** htmltools, htmlwidgets, reactR, stringr, magrittr, readr, jsonlite, httpuv, mime, cli, ids, dplyr

**Suggests** testthat (>= 3.0.0), knitr, rmarkdown

**Config/testthat/edition** 3

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Elliot Hershberg [aut, cre] (<<https://orcid.org/0000-0003-2068-3366>>), the JBrowse 2 Team [aut]

**Maintainer** Elliot Hershberg <eahershberg@gmail.com>

**Repository** CRAN

**Date/Publication** 2022-07-19 22:30:02 UTC

## R topics documented:

assembly	2
default_session	3
JBrowseR	4
JBrowseR-shiny	4
json_config	5

serve_data . . . . .	6
text_index . . . . .	7
theme . . . . .	8
tracks . . . . .	8
track_alignments . . . . .	9
track_data_frame . . . . .	10
track_feature . . . . .	11
track_variant . . . . .	12
track_wiggle . . . . .	13

<b>Index</b>	<b>14</b>
--------------	-----------

---

assembly	<i>Create an assembly for a custom JBrowse view</i>
----------	---

---

## Description

Creates the necessary configuration string for an indexed fasta or bgzip fasta so that it can be used as the assembly in a JBrowse custom linear genome view.

## Usage

```
assembly(assembly_data, bgzip = FALSE, aliases = NULL, refname_aliases = NULL)
```

## Arguments

assembly_data	the URL to your fasta file
bgzip	whether or not your fasta is bgzip compressed
aliases	a vector of strings of the aliases for the assembly
refname_aliases	the URL to a file containing reference name aliases. For more info see <a href="https://jbrowse.org/jb2/docs/config_guide#configuring-reference-name-aliasing">https://jbrowse.org/jb2/docs/config_guide#configuring-reference-name-aliasing</a>

## Details

The string returned by `assembly` is stringified JSON. JBrowseR is an interface to JBrowse 2, which receives its configuration in JSON format. The stringified JSON returned by `assembly` is parsed into a JavaScript object in the browser, and is used to configure the genome browser.

It is important to note that while only the fasta file is passed as an argument, `assembly` assumes that a fasta index of the same name is located with the fasta file (as well as a gzi file in the case of a bgzip fasta).

For example:

```
assembly("data/hg38.fa")
```

Assumes that `data/hg38.fa.fai` also exists.

```
assembly("data/hg38.fa", bgzip = TRUE)
```

Assumes that data/hg38.fa.fai and data/hg38.fa.gzi both exist.

This is a JBrowse 2 convention, and the default naming output of samtools and bgzip.

For more information on creating these files, visit [https://jbrowse.org/jb2/docs/quickstart\\_cli#adding-a-genome-assembly](https://jbrowse.org/jb2/docs/quickstart_cli#adding-a-genome-assembly)

### Value

a character vector of JBrowseR assembly configuration

### Examples

```
assembly("https://jbrowse.org/genomes/hg19/fasta/hg19.fa.gz", bgzip = TRUE)
```

---

default_session	<i>Create a default session for a custom JBrowse view</i>
-----------------	---

---

### Description

Creates the necessary configuration string for a default session for your browser. A default session is the set of tracks that are displayed when your browser is first displayed.

### Usage

```
default_session(assembly, displayed_tracks, display_assembly = TRUE)
```

### Arguments

assembly            the config string generated by assembly

displayed\_tracks    a vector of tracks generated by a track\_\* command.

display\_assembly    a boolean determining whether the reference sequence is visible or not. TRUE by default.

### Value

a character vector of stringified JSON configuration for the defaultSession to be used by the browser when first loaded

### Examples

```
# create the assembly configuration
assembly <- assembly("https://jbrowse.org/genomes/hg19/fasta/hg19.fa.gz", bgzip = TRUE)

# create variant and wiggle tracks
variant <- track_variant(
  "clinvar.vcf.gz",
  assembly
```

```

)
wiggle <- track_wiggle(
  "read-cov.bw",
  assembly
)

# create a default session with those tracks open by default
default_session <- default_session(
  assembly,
  c(variant, wiggle)
)

```

---

JBrowseR

*R interface to JBrowse 2 genome browser*


---

### Description

Embed a JBrowse 2 linear genome view in your Shiny app, Rmd document, or interactive R console.

### Usage

```
JBrowseR(view, ..., width = NULL, height = NULL, elementId = NULL)
```

### Arguments

view	Which JBrowse 2 view to use. View, JsonView, ViewHg19, ViewHg38
...	The parameters passed on to the view
width	The width of the htmlwidget
height	The height of the htmlwidget
elementId	The elementId of the htmlwidget

### Value

an htmlwidget of the JBrowse 2 linear genome view.

---

JBrowseR-shiny

*Shiny bindings for JBrowseR*


---

### Description

Output and render functions for using JBrowseR within Shiny applications and interactive Rmd documents.

**Usage**

```

JBrowseROutput(outputId, width = "100%", height = "400px")

renderJBrowseR(expr, env = parent.frame(), quoted = FALSE)

JBrowseR_html(id, style, class, ...)

```

**Arguments**

outputId	output variable to read from
width	Must be a valid CSS unit or a number, which will be coerced to a string and have 'px' appended.
height	Must be a valid CSS unit or a number, which will be coerced to a string and have 'px' appended.
expr	An expression that generates a JBrowseR
env	The environment in which to evaluate expr.
quoted	Is expr a quoted expression (with quote())? This is useful if you want to save an expression in a variable.
id	htmltools id
style	htmltools style
class	htmltools class
...	Additional arguments passed on

**Value**

the Shiny UI bindings for a JBrowseR htmlwidget  
the Shiny server bindings for a JBrowseR htmlwidget  
the root HTML element to render the React component in

---

```
json_config
```

*Read in a JBrowse 2 JSON configuration file*

---

**Description**

Reads in a JSON file with values for configuring your browser. Looks for assembly, tracks, default-Session, and theme. Only assembly is explicitly required for a working browser.

**Usage**

```
json_config(file)
```

**Arguments**

file	the file path or URL to a JBrowse 2 configuration
------	---

**Details**

Note: this is the most advanced API. It offers full control to do anything possible in JavaScript with an embedded JBrowse 2 React component, but comes with a steeper learning curve. For more details on JBrowse 2 configuration, visit: [https://jbrowse.org/jb2/docs/config\\_guide](https://jbrowse.org/jb2/docs/config_guide)

An example JSON config is provided with this package

**Value**

a character vector of JSON configuration from a JBrowse 2 configuration file

**Examples**

```
## Not run: json_config("./config.json")
```

---

serve_data	<i>Serve a local data directory for use with a browser</i>
------------	--

---

**Description**

This is a utility function that can be used to server a local directory with data so that it can be used in the genome browser.

**Usage**

```
serve_data(path, port = 5000)
```

**Arguments**

path	The path to the directory with data to serve
port	The port to serve the directory on

**Details**

Note: This is intended for local development and use. For a production deployment, refer to the vignette on creating URLs for more robust options.

**Value**

a list containing information about the newly created HTTP server including the host, port, interval, and URL. The list also contains the stop\_server() function which can be used to stop the server

**Examples**

```
## Not run:
server <- serve_data("~/path/to/my-data")
# use server$stop_server() to stop
```

```
## End(Not run)
```

---

text_index	<i>Create configuration for a JBrowse 2 text index</i>
------------	--

---

### Description

Creates the necessary configuration string for an adapter to a text index for gene name search in the browser.

### Usage

```
text_index(ix_uri, ix_uri, meta_uri, assembly)
```

### Arguments

ix_uri	the URI for the ix file
ixx_uri	the URI for the ixx file
meta_uri	the URI for the JSON metadata file
assembly	the assembly associated with the text index

### Details

Note: this function currently only supports aggregate indexes.

For more information on JBrowse 2 text indexes, visit: [https://jbrowse.org/jb2/docs/config\\_guide/#text-searching](https://jbrowse.org/jb2/docs/config_guide/#text-searching)

### Value

a character vector with the JSON text index adapter.

### Examples

```
text_index(  
  "https://jbrowse.org/genomes/hg19/trix/hg19.ix",  
  "https://jbrowse.org/genomes/hg19/trix/hg19.ixx",  
  "https://jbrowse.org/genomes/hg19/trix/meta.json",  
  "hg19"  
)
```

---

theme

*Create a theme for a custom JBrowse 2 view*

---

### Description

Creates the necessary configuration string for a custom theme palette for your browser. Accepts up to four hexadecimal colors. For more information on how JBrowse 2 custom themes work, visit [https://jbrowse.org/jb2/docs/config\\_guide#configuring-the-theme](https://jbrowse.org/jb2/docs/config_guide#configuring-the-theme)

### Usage

```
theme(primary, secondary = NULL, tertiary = NULL, quaternary = NULL)
```

### Arguments

primary	the primary color of your custom palette
secondary	the secondary color of your custom palette
tertiary	the tertiary color of your custom palette
quaternary	the quaternary color of your custom palette

### Value

a character vector of stringified theme JSON configuration to configure a custom color palette for the browser

### Examples

```
theme("#311b92")
theme("#311b92", "#0097a7")
theme("#311b92", "#0097a7", "#f57c00")
theme("#311b92", "#0097a7", "#f57c00", "#d50000")
```

---

tracks

*Create a set of tracks for a custom JBrowse 2 view*

---

### Description

Accepts any number of tracks, returns the configuration string necessary to load these tracks into your JBrowse view.

### Usage

```
tracks(...)
```

**Arguments**

... The tracks to be added to the JBrowse 2 view

**Value**

a character vector of stringified JSON configuration for all tracks to add to the browser

**Examples**

```
# create an assembly configuration and alignments track
assembly <- assembly("https://jbrowse.org/genomes/hg19/fasta/hg19.fa.gz", bgzip = TRUE)
alignments <- track_alignments("alignments.bam", assembly)

# create a tracks configuration with the alignments track
tracks(alignments)
```

---

track_alignments	<i>Create an AlignmentTrack for a custom JBrowse 2 view</i>
------------------	---

---

**Description**

Creates the necessary configuration string for an indexed BAM or CRAM alignment so that it can be used in a JBrowse custom linear genome view.

**Usage**

```
track_alignments(track_data, assembly)
```

**Arguments**

track\_data the URL to the BAM/CRAM alignments  
assembly the config string generated by assembly

**Details**

It is important to note that while only the BAM/CRAM file is passed as an argument, tracks\_alignment assumes that a BAM/CRAM index of the same name is located with the file

For example:

```
track_alignments("data/alignments.bam")
```

Assumes that data/alignments.bam.bai also exists.

This is a JBrowse 2 convention, and the default naming output of samtools

For more information on creating an index with samtools, visit <https://www.htslib.org/>

**Value**

a character vector of stringified AlignmentTrack JSON configuration

## Examples

```
assembly <- assembly("https://jbrowse.org/genomes/hg19/fasta/hg19.fa.gz", bgzip = TRUE)

track_alignments("alignments.bam", assembly)
track_alignments("alignments.cram", assembly)
```

---

track\_data\_frame      *Create a track from an R data frame for a custom JBrowse 2 view*

---

## Description

Creates the necessary configuration string for an R data frame so that it can be viewed as a track in a JBrowse custom linear genome view.

## Usage

```
track_data_frame(track_data, track_name, assembly)
```

## Arguments

track_data	the data frame with track data. Must have cols: chrom, start, end, name. The column additional can optionally be include with more feature information. If a score column is present, it will be used and the track will be rendered to display quantitative features.
track_name	the name to use for the track
assembly	the config string generated by assembly

## Value

a character vector of stringified track JSON configuration

## Examples

```
assembly <- assembly("https://jbrowse.org/genomes/hg19/fasta/hg19.fa.gz", bgzip = TRUE)

df <- data.frame(
  chrom = c(1, 2),
  start = c(123, 456),
  end = c(789, 101112),
  name = c('feature1', 'feature2')
)

track_data_frame(df, "my_features", assembly)
```

---

track_feature	<i>Create a FeatureTrack for a custom JBrowse 2 view</i>
---------------	--

---

### Description

Creates the necessary configuration string for an indexed GFF3 file so that it can be used in a JBrowse custom linear genome view.

### Usage

```
track_feature(track_data, assembly)
```

### Arguments

track_data	the URL to the GFF3 file
assembly	the config string generated by assembly

### Details

It is important to note that while only the GFF3 file is passed as an argument, `tracks_variant` assumes that a GFF3 index of the same name is located with the file

For example:

```
track_feature("data/features.gff")
```

Assumes that `data/features.gff.tbi` also exists.

This is a JBrowse 2 convention, and the default naming output of `tabix`

For more information on creating an index with `tabix`, visit <https://www.htslib.org/>

### Value

a character vector of stringified FeatureTrack JSON configuration

### Examples

```
assembly <- assembly("https://jbrowse.org/genomes/hg19/fasta/hg19.fa.gz", bgzip = TRUE)
track_feature("features.gff", assembly)
```

---

track_variant	<i>Create a VariantTrack for a custom JBrowse 2 view</i>
---------------	--

---

### Description

Creates the necessary configuration string for an indexed VCF file so that it can be used in a JBrowse custom linear genome view.

### Usage

```
track_variant(track_data, assembly)
```

### Arguments

track_data	the URL to the VCF file
assembly	the config string generated by assembly

### Details

It is important to note that while only the VCF file is passed as an argument, `tracks_variant` assumes that a VCF index of the same name is located with the file

For example:

```
track_alignments("data/variants.vcf")
```

Assumes that `data/variants.vcf.tbi` also exists.

This is a JBrowse 2 convention, and the default naming output of `tabix`

For more information on creating an index with `tabix`, visit <https://www.htslib.org/>

### Value

a character vector of stringified VariantTrack JSON configuration

### Examples

```
assembly <- assembly("https://jbrowse.org/genomes/hg19/fasta/hg19.fa.gz", bgzip = TRUE)
track_variant("variants.vcf", assembly)
```

---

`track_wiggle`*Create a WiggleTrack for a custom JBrowse 2 view*

---

**Description**

Creates the necessary configuration string for a bigWig file so that it can be used in a JBrowse custom linear genome view.

**Usage**

```
track_wiggle(track_data, assembly)
```

**Arguments**

<code>track_data</code>	the URL to the bigWig file
<code>assembly</code>	the config string generated by assembly

**Value**

a character vector of stringified WiggleTrack JSON configuration

**Examples**

```
track_wiggle(  
  "https://jbrowse.org/genomes/hg19/COL0829/colo_normal.bw",  
  assembly("https://jbrowse.org/genomes/hg19/fasta/hg19.fa.gz", bgzip = TRUE)  
)
```

# Index

`assembly`, [2](#)

`default_session`, [3](#)

`JBrowseR`, [4](#)

`JBrowseR-shiny`, [4](#)

`JBrowseR_html` (`JBrowseR-shiny`), [4](#)

`JBrowseROutput` (`JBrowseR-shiny`), [4](#)

`json_config`, [5](#)

`renderJBrowseR` (`JBrowseR-shiny`), [4](#)

`serve_data`, [6](#)

`text_index`, [7](#)

`theme`, [8](#)

`track_alignments`, [9](#)

`track_data_frame`, [10](#)

`track_feature`, [11](#)

`track_variant`, [12](#)

`track_wiggle`, [13](#)

`tracks`, [8](#)