

# Package: rpix (via r-universe)

October 31, 2024

**Type** Package

**Title** R package to manage dependencies with pixi

**Version** 0.1.0

**Author** Mikkel Roald-Arbøl

**Maintainer** Mikkel Roald-Arbøl <yourself@somewhere.net>

**Description** More about what it does (maybe more than one line) Use four spaces when indenting paragraphs within the Description.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** true

**Config/testthat/edition** 3

**Remotes** r-lib/usethis, etiennebacher/altdoc

**URL** <https://www.roald-arboel.com/rpix/>

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.1

**Imports** blogdown, renv

**Suggests** altdoc, testthat (>= 3.0.0)

**Repository** <https://roaldarbol.r-universe.dev>

**RemoteUrl** <https://github.com/roaldarbol/rpix>

**RemoteRef** HEAD

**RemoteSha** 181ac5a020c4e7f473e85f943adfaf023e565498

## Contents

add . . . . .	2
init . . . . .	2
pixi_installed . . . . .	3
remove . . . . .	3
search . . . . .	4

<b>Index</b>	<b>5</b>
--------------	----------

---

add *Add dependencies*

---

### Description

add() will add dependencies to the pixi.toml. It will only add if the package with its version constraint is able to work with rest of the dependencies in the project. More info on [multi-platform configuration](#).

### Usage

```
add(package_names)
```

### Arguments

package\_names Package name(s) to be added.

### Value

Doesn't return any objects.

### Examples

```
## Not run:  
add("tibble")  
  
## End(Not run)
```

---

init *Create a new pixi project*

---

### Description

init() initializes a pixi.toml file and also prepares a .gitignore to prevent the environment from being added to git.

### Usage

```
init()
```

### Value

Doesn't return any objects.

---

pixi_installed	<i>Check pixi installation</i>
----------------	--------------------------------

---

**Description**

Tests whether pixi is installed on the system

**Usage**

```
pixi_installed()
```

**Value**

Doesn't return any objects.

---

remove	<i>Remove dependencies</i>
--------	----------------------------

---

**Description**

remove() will add dependencies to the pixi.toml. It will only remove if the package with its version constraint is able to work with rest of the dependencies in the project. More info on [multi-platform configuration](#).

**Usage**

```
remove(package_names)
```

**Arguments**

package\_names Package name(s) to be removed.

**Value**

Doesn't return any objects.

**Examples**

```
## Not run:  
remove("tibble")  
  
## End(Not run)
```

---

search

*Search for dependency*

---

**Description**

Search a package, output will list the latest version of the package.

**Usage**

```
search(package_name = NULL)
```

**Arguments**

package\_name    Package name to be added.

# Index

add, [2](#)

init, [2](#)

pixi\_installed, [3](#)

remove, [3](#)

search, [4](#)