

# Download biodiversity data with galah: : CHEAT SHEET



galah is an interface to biodiversity data hosted by the [Atlas of Living Australia \(ALA\)](https://ala.org.au). It enables users to locate and download species occurrence records (observations, specimens, eDNA records, etc.), taxonomic information, or associated media such as images or sounds, and to restrict their queries to particular taxa or locations.

## Build a query

No matter what kind of data you want to return in [R-Studio](#), **every download query** consists of the **same building blocks**.

1. Start a query with **galah\_call()**
2. **Modify a query** with **galah\_** functions
3. **Return data** with **atlas\_** functions

### AN EXAMPLE QUERY:

```
galah_call() |>
  galah_identify("reptilia") |>
  galah_filter(year > 2010,
               cl22 == "Victoria") |>
  atlas_occurrences()
```

Start a query

Modify a query

Return data

## dplyr syntax



galah supports many common dplyr functions to be used in queries. These can directly replace the equivalent galah functions, highlighted in **aqua**.

## Download data

### COUNTS

**atlas\_counts()** | **count()** Return the number of records that match a query

Return number of observations

```
count  galah_call() |>
       galah_identify("reptilia") |>
       galah_filter(year == 2020) |>
       atlas_counts()
       40284
```

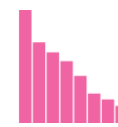
Return number of species

```
count  galah_call(type = "species") |>
       galah_identify("reptilia") |>
       galah_filter(year == 2020) |>
       count()
       537
```

Return grouped counts

```
galah_call() |>
  galah_identify("reptilia") |>
  galah_filter(year == 2020) |>
  galah_group_by(order) |>
  atlas_counts()
```

order	count
Squamata	40284
Crocodylia	6388
Testudines	4301
Rhynchocephalia	1



### OCCURRENCES

**atlas\_occurrences()** | **collect()** Return species occurrence records that match a query

```
galah_config(email = "your-email-here")
```

```
galah_call() |>
  galah_identify("perameles") |>
  galah_filter(year > 2021) |>
  atlas_occurrences()
```

Must use an ALA-registered email

Register at [ala.org.au/](https://ala.org.au/)



decimalLongitude	decimalLatitude	eventDate	scientificName	taxonConceptID
-43.2	148.	2023-01-06 12:46:00	<i>Perameles gunnii</i>	<a href="https://biodiversity.org.au/afd/taxa/03b412...">https://biodiversity.org.au/afd/taxa/03b412...</a>
-43.1	147.	2022-10-07 10:38:26	<i>Perameles gunnii</i>	<a href="https://biodiversity.org.au/afd/taxa/03b412...">https://biodiversity.org.au/afd/taxa/03b412...</a>
-43.1	148.	2022-09-18 10:13:00	<i>Perameles gunnii</i>	<a href="https://biodiversity.org.au/afd/taxa/03b412...">https://biodiversity.org.au/afd/taxa/03b412...</a>
-43.1	148.	2022-01-15 13:39:00	<i>Perameles gunnii</i>	<a href="https://biodiversity.org.au/afd/taxa/03b412...">https://biodiversity.org.au/afd/taxa/03b412...</a>

i = 1,658 more rows

### SPECIES LISTS

**atlas\_species()** | **collect()** Return species information for each species that matches a query

```
galah_config(email = "your-email-here")
```

```
galah_call() |>
  galah_identify("perameles") |>
  galah_filter(year > 2021) |>
  atlas_species()
```

Must use an ALA-registered email

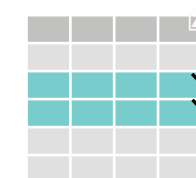
Register at [ala.org.au/](https://ala.org.au/)

kingdom	phylum	class	order	family	genus	species
Animalia	Chordata	Mammalia	Peramelemorphia	Peramelidae	Perameles	<i>Perameles nasuta</i>
Animalia	Chordata	Mammalia	Peramelemorphia	Peramelidae	Perameles	<i>Perameles gunnii</i>
Animalia	Chordata	Mammalia	Peramelemorphia	Peramelidae	Perameles	<i>Perameles notina</i>
Animalia	Chordata	Mammalia	Peramelemorphia	Peramelidae	Perameles	<i>Perameles fasciata</i>

i = 1 more row

### MEDIA & IMAGES

**atlas\_media()** | **collect()** + **download\_media()** Return species occurrence records with associated images, sounds or media that match a query. Return matching records with **atlas\_media()**, then use **collect\_media()** to download locally



Returns output like **atlas\_occurrences()** with media metadata



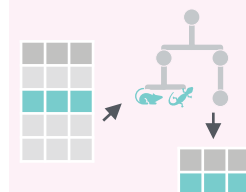
```
galah_config(email = "your-email-here")
```

```
galah_call() |>
  galah_identify("perameles") |>
  galah_filter(year > 2021) |>
  atlas_media()
```

```
galah_call() |>
  galah_identify("perameles") |>
  galah_filter(year > 2021) |>
  atlas_media() |>
  download_media(path = "path-to-folder",
                filesize = "thumbnail")
```

Download "full" or "thumbnail" size images

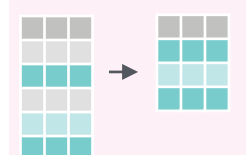
### MODIFY QUERY ON THE SERVER BEFORE DOWNLOADING:



#### galah\_identify() | identify()

Filter query to specific identified taxa

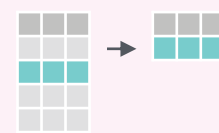
```
galah_call() |>
  galah_identify("mammalia", "reptilia") |>
  atlas_counts()
```



#### galah\_apply\_profile()

Apply a set of data quality filters to narrow a query

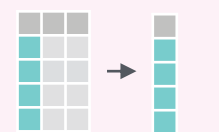
```
galah_call() |>
  galah_apply_profile(ALA) |>
  atlas_counts()
```



#### galah\_filter() | filter()

Filter query to rows that meet a logical criteria

```
galah_call() |>
  galah_filter(year == 2020) |>
  atlas_counts()
```



#### galah\_select() | select()

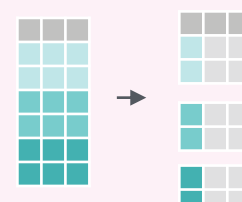
Filter query to return specified columns

```
galah_call() |>
  galah_select(scientificName, eventDate) |>
  atlas_occurrences()
```

#### galah\_group\_by() | group\_by()

Filter query to rows that meet a logical criteria

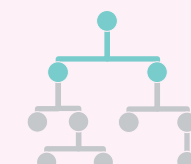
```
galah_call() |>
  galah_group_by(year, species) |>
  atlas_counts()
```



#### galah\_geolocate() | st\_crop()

Specify the location of data returned with a polygon or bounding box

```
bbox <- tibble(xmin = xx, xmax = dd,
               ymin = xx, ymax = dd)
galah_call() |>
  galah_geolocate(bbox,
                  type = "bbox") |>
  atlas_occurrences()
```



#### galah\_down\_to()

Specify the lowest taxonomic level in a query for a taxonomic tree

```
galah_call() |>
  galah_identify("fungi") |>
  galah_down_to(phylum) |>
  atlas_taxonomy()
```

## Choose an atlas



The Global Biodiversity Information Facility (GBIF) network consists of a series of ‘nodes’ — **Living Atlases**—that collate biodiversity data from their own countries. GBIF acts as an umbrella organisation to store data from all nodes.

**galah supports data downloads for more than 10 Living Atlases and GBIF.**

See full list:

[https://galah.ala.org.au/R/articles/choosing\\_an\\_atlas.html](https://galah.ala.org.au/R/articles/choosing_an_atlas.html)

```
galah_config(atlas = "Austria")
```

```
galah_config(atlas = "GBIF")
```

## Advanced downloads

lifecycle experimental

Data queries with many rows and columns (ie fields) can become very large. Large queries can take a long time to download, and sometimes timeout in R if a session is closed for any reason.

galah supports **dplyr collapse()**, **compute()** & **collect()** functions to **break up building, sending and downloading a query into separate steps**. This allows for larger or more complex queries to download without timing out in R.



**collapse(.data)** Create the query to be sent to the atlas

```
galah_call(type = "occurrences") |>
  galah_identify("coleoptera") |>
  galah_filter(
    year == 2022,
    occurrenceStatus == "PRESENT") |>
  collapse()
```



**compute(.data)** Send the query to the specified atlas. Sent queries can download without timing out in R

```
galah_call(type = "occurrences") |>
  galah_identify("coleoptera") |>
  galah_filter(
    year == 2022,
    occurrenceStatus == "PRESENT") |>
  compute()
```



**collect(.data)** Once a query is complete, download data locally

```
galah_call(type = "occurrences") |>
  galah_identify("coleoptera") |>
  galah_filter(
    year == 2022,
    occurrenceStatus == "PRESENT") |>
  collect()
```

**collect() runs collapse() & compute() under the hood to create and send a query**

## Lookup information — galah provides look-up functions to help users find ways to modify their queries



The living atlases store a huge amount of information, above and beyond only occurrence records. This information can be useful for modifying queries.

### Taxonomic information

Look up taxonomic names before downloading data from the ALA using **atlas\_** functions.



**search\_taxa(...)** Search for valid taxonomic names, look up taxonomic information, disambiguate homonyms

```
search_taxa("reptilia", "mammalia")
```

Specify taxonomic levels in a tibble using “specificEpiphet”

```
search_taxa(tibble::tibble(
  class = "aves",
  family = "pardalotidae",
  genus = "pardalotus",
  specificEpiphet = "punctatus"))
```

Specify taxonomic levels in a tibble using “scientificName”

```
search_taxa(tibble::tibble(
  family = c("pardalotidae", "maluridae"),
  scientificName = c("Pardalotus striatus striatus",
    "malurus cyaneus")))
```

**search\_identifiers(...)** Search for unique identifiers of a taxon. Identifiers are assigned by atlases to identify all taxonomic clades.

```
search_identifiers(
  query =
    https://id.biodiversity.org.au/node/apni/2914510
)
```

### Values

Users may wish to see what values are within a chosen field, profile or list to modify a query or understand more about the information of interest. The **values functions can be piped from supported search\_all() functions** to see this information.



Specified field

Field values

**show\_all(type, limit = NULL)** Show all available options or categories for a specified type of information

### Configuration



**atlases** Show what atlases are available

```
show_all(atlases)
search_all(atlases, "Brazil")
```



**apis** Show what APIs & functions are available

```
show_all/apis)
search_all/apis, "counts")
```



**reasons** Show what values are acceptable as “download reasons” for a specified atlas

```
show_all(reasons)
search_all(reasons, "research")
```

### Filters



**fields** Show fields that are stored in an atlas

```
show_all(fields)
search_all(fields,
  "australian states")
```



**assertions** Show data quality checks run by each atlas

```
show_all(assertions)
search_all(assertions, "longitude")
```



**licenses** Show what copyright licenses are applied to media

```
show_all(licenses)
search_all(licenses, "CC BY")
```

**show\_values(df)** Search for valid taxonomic names, look up taxonomic information, disambiguate homonyms

```
search_all(fields, "cl22") |>
  show_values()
```

field	count
cl22	New South Wales
cl22	Victoria
cl22	Queensland
cl22	South Australia
cl22	Western Australia
cl22	Northern Territory
cl22	Australian Capital Territory
cl22	Tasmania
cl22	Ashmore and Cartier Islands
Cl22	Coral Sea Islands

**search\_all(type, query)** Search for a specific option or category for a specified type of information

### Taxonomy



**ranks** Show valid taxonomic ranks (eg Kingdom, Class)

```
show_all(ranks)
search_all(ranks, "suborder")
```

### Group filters



**profiles** Show what data profiles (sets of data quality filters) are available

```
show_all(profiles)
search_all(profiles, "ALA")
```



**lists** Show what species lists are available

```
show_all(lists)
search_all(lists, "EPBC")
```

### Data providers



**providers** Show which institutions has provided data

```
show_all(providers)
search_all(providers, "botanic")
```



**collections** Show the specific collections within those institutions

```
show_all(providers)
search_all(collections,
  "antarctic")
```



**datasets** Show all the data groupings within those collections

```
show_all(datasets)
search_all(datasets, "river")
```

**search\_values(df, query)** Search for valid taxonomic names, look up taxonomic information, disambiguate homonyms

```
search_all(fields, "cl22") |>
  search_values("territory")
```

field	count
cl22	Northern Territory
cl22	Australian Capital Territory

### Want ideas on how to visualise your data?

For more information, including tutorials & examples, check out **ALA LABS**.

