# Package 'ipanema'

May 9, 2025

**Type** Package

**Title** Read Data from 'LimeSurvey'

**Version** 1.2.0

**Description** Read data from 'LimeSurvey'
(<https://www.limesurvey.org/>)
in a comfortable way.
Heavily inspired by 'limer'
(<https://github.com/cloudyr/limer/>),
which lacked a few comfort features for me.

**License** MIT + file LICENSE

**URL** https://gitlab.com/REDS1736/ipanema

**Encoding** UTF-8

**Imports** base64enc, DBI, dplyr, httr, jsonlite, magrittr, RMySQL

**RoxygenNote** 7.3.2

**Depends** R (>= 2.10)

**NeedsCompilation** no

**Author** Maximilian Hagspiel [aut, cre, cph]

**Maintainer** Maximilian Hagspiel <maxhag@mailbox.org>

**Repository** CRAN

**Date/Publication** 2025-05-09 16:20:02 UTC

## Contents

---

`base64_to_df` *base64_to_df*

---

## Description

Convert a base64 representation of a CSV table into a ‘data.frame‘ object.

## Usage

```
base64_to_df(x)
```

## Arguments

| | |
|---|---|
| `x` | The base64-encoded CSV string |

## Value

A ‘data.frame‘ object containing the data from ‘x‘.

---

`connect_to_limesurvey` *connect_to_limesurvey*

---

## Description

Connect to ‘LimeSurvey‘ instance via the RPC and a direct MySQL connection. Store the RPC session key in ‘options('limesurvey_session_key')‘. Store the MySQL connection object in ‘options('limesurvey_mysql_connection')‘. Store the RPC URL in ‘options('limesurvey_api_url')‘.

## Usage

```
connect_to_limesurvey(
  api_url,
  limesurvey_username,
  limesurvey_password,
  mysql_host,
  mysql_port,
  mysql_dbname,
  mysql_table_prefix,
  mysql_username,
  mysql_password
)
```

### Arguments

| | |
|---|---|
| `api_url` | URL to the ‘LimeSurvey‘ RPC, e.g. ’http://localhost/index.php/admin/remotecontrol’ |
| `limesurvey_username` | Username for the ‘LimeSurvey‘ API |
| `limesurvey_password` | Password for the ‘LimeSurvey‘ API |
| `mysql_host` | Hostname of the MySQL server used by ‘LimeSurvey‘ |
| `mysql_port` | Port on which the MySQL server listens for connections |
| `mysql_dbname` | Name of the database on the MySQL server which is used by ‘LimeSurvey‘ |
| `mysql_table_prefix` | Prefix for all table names, e.g. "lime_" |
| `mysql_username` | Username for the MySQL server |
| `mysql_password` | Password for the MySQL server |

### Value

No return value, called for side effects

### Examples

```
# This example assumes a locally hosted `LimeSurvey` instance using a locally
# hosted MySQL server
## Not run:
connect_to_limesurvey(
  api_url = 'https://localhost/index.php/admin/remotecontrol',
  limesurvey_username = 'admin',
  limesurvey_password = '1234admin',
  mysql_host = '127.0.0.1',
  mysql_port = 3306,
  mysql_dbname = 'limesurvey',
  mysql_table_prefix = '',
  mysql_username = 'lime',
  mysql_password = '1234lime'
)

## End(Not run)
```

---

## `fix_column_data_types` *fix_column_data_types*

---

### Description

Freshly exported data has all item-data columns as type "character". This function converts these columns to ideal types (e.g. integer). Currently simply converts all multiple-choice columns to integer. Future task: Add conversion to other data types as needed.

## Usage

```
fix_column_data_types(df_in)
```

## Arguments

| | |
|---|---|
| df_in | The ‘data.frame‘ object to fix. |

## Value

A ‘data.frame‘ object containing the data from ‘df_in‘ but with fixed column data types.

---

`get_answer_options` *get_answer_options*

---

## Description

Get the answer options to a question with pre-defined answer options (e.g. a multiple choice question).

## Usage

```
get_answer_options(question_code)
```

## Arguments

| | |
|---|---|
| question_code | Code by which to identify the question. Follows a dot-based naming scheme: <group title>.<subquestion title>. |

## Value

‘data.frame‘ object with the columns ‘code‘ and ‘answer‘ in which each row represents one answer option where ‘code‘ is the encoded value (as found in datasets exported by ‘get_survey_data()‘ and ‘answer‘ is the answer option text as seen by survey users).

## Examples

```
# This example assumes a locally hosted `LimeSurvey` instance using a locally
# hosted MySQL server.
# On this `LimeSurvey` instance, there is a survey with the ID 123456.
# In this survey, a multiple-choice question identified by the code "bdi.01"
# is used.
# For this question, this example retrieves the possible answer options.
## Not run:
connect_to_limesurvey(
  api_url = 'https://localhost/index.php/admin/remotecontrol',
  limesurvey_username = 'admin',
  limesurvey_password = '1234admin',
  mysql_host = '127.0.0.1',
  mysql_port = 3306,
```

```
  mysql_dbname = 'limesurvey',
  mysql_table_prefix = '',
  mysql_username = 'lime',
  mysql_password = '1234lime'
)

answer_options <- get_answer_options("bdi.01")

## End(Not run)
```

---

`get_question_text` *get_question_text*

---

## Description

Get the question text (e.g. "How have you been feeling?") to a question in the dataset.

## Usage

```
get_question_text(question_code)
```

## Arguments

| | |
|---|---|
| `question_code` | Code by which to identify the question. Follows a dot-based naming scheme: <group title>.<subquestion title>. |

## Value

'character' object containing the question text

## Examples

```
# This example assumes a locally hosted `LimeSurvey` instance using a locally
# hosted MySQL server.
# On this `LimeSurvey` instance, there is a survey with the ID 123456.
# In this survey, a multiple-choice question identified by the code "bdi.01"
# is used.
# For this question, this example retrieves the question text which was shown
# to the user when answering the questionnaire.
## Not run:
connect_to_limesurvey(
  api_url = 'https://localhost/index.php/admin/remotecontrol',
  limesurvey_username = 'admin',
  limesurvey_password = '1234admin',
  mysql_host = '127.0.0.1',
  mysql_port = 3306,
  mysql_dbname = 'limesurvey',
  mysql_table_prefix = '',
  mysql_username = 'lime',
```

```
  mysql_password = '1234lime'
)

q_text <- get_question_text("bdi.01")

## End(Not run)
```

---

`get_sql_varname` *get_sql_varname*

---

## Description

Get the internal SQL field name (e.g. "697929X4X21") to a question from a specific survey in the dataset.

## Usage

```
get_sql_varname(question_code, survey_id)
```

## Arguments

| | |
|---|---|
| `question_code` | Code by which to identify the question. Follows a dot-based naming scheme: <group title>.<subquestion title>. |
| `survey_id` | Survey-ID of the survey from which to select the question. |

## Value

'character' object containing the field name

## Examples

```
# This example assumes a locally hosted `LimeSurvey` instance using a locally
# hosted MySQL server.
# On this `LimeSurvey` instance, there is a survey with the ID 123456.
# In this survey, a multiple-choice question identified by the code "bdi.01"
# is used.
# For this question, this example retrieves name of the SQL table field in
# which `LimeSurvey` internally stores the responses to this question.
## Not run:
connect_to_limesurvey(
  api_url = 'https://localhost/index.php/admin/remotecontrol',
  limesurvey_username = 'admin',
  limesurvey_password = '1234admin',
  mysql_host = '127.0.0.1',
  mysql_port = 3306,
  mysql_dbname = 'limesurvey',
  mysql_table_prefix = '',
  mysql_username = 'lime',
```

```
  mysql_password = '1234lime'
)

q_varname <- get_sql_varname("bdi.01", 123456)

## End(Not run)
```

---

`get_survey_data` *get_survey_data*

---

## Description

Get collected data from a specific survey on the connected 'LimeSurvey' instance. Includes complete and incomplete cases! Returns 'NULL' if no data has been collected in this survey.

## Usage

```
get_survey_data(survey_id, completion_status = "all")
```

## Arguments

| | |
|---|---|
| `survey_id` | ID of the survey from which the collected data shall be extracted. 6-digit integer. |
| `completion_status` | 'complete' = Return only complete cases; 'incomplete' = Return only incomplete cases; 'all' = Return both. |

## Value

A 'data.frame' object containing the survey data. Column names follow a dot-based naming scheme: <group title>.<subquestion title>. 'NULL' if no data has been collected.

## Examples

```
# This example assumes a locally hosted `LimeSurvey` instance using a locally
# hosted MySQL server.
# On this `LimeSurvey` instance, there is a survey with the ID 123456.
## Not run:
connect_to_limesurvey(
  api_url = 'https://localhost/index.php/admin/remotecontrol',
  limesurvey_username = 'admin',
  limesurvey_password = '1234admin',
  mysql_host = '127.0.0.1',
  mysql_port = 3306,
  mysql_dbname = 'limesurvey',
  mysql_table_prefix = '',
  mysql_username = 'lime',
  mysql_password = '1234lime'
)
```

```
df_data <- get_survey_data(123456)

## End(Not run)
```

---

`get_survey_id` *get_survey_id*

---

## Description

Get numerical LimeSurvey ID of the survey with the given title.

## Usage

```
get_survey_id(survey_title)
```

## Arguments

| | |
|---|---|
| `survey_title` | TItle of the survey. String. |

## Value

An integer Survey ID which can be used as a parameter in 'get_survey_data()'

## Examples

```
# This example assumes a locally hosted `LimeSurvey` instance using a locally
# hosted MySQL server.
# On this `LimeSurvey` instance, there is a survey with the title 'mysurvey'.
## Not run:
connect_to_limesurvey(
  api_url = 'https://localhost/index.php/admin/remotecontrol',
  limesurvey_username = 'admin',
  limesurvey_password = '1234admin',
  mysql_host = '127.0.0.1',
  mysql_port = 3306,
  mysql_dbname = 'limesurvey',
  mysql_table_prefix = '',
  mysql_username = 'lime',
  mysql_password = '1234lime'
)

survey_id <- get_survey_id('mysurvey')
df_data <- get_survey_data(survey_id)

## End(Not run)
```

## limesurvey_api_call *limesurvey_api_call*

### Description

Perform a call to the 'LimeSurvey' RPC API.

### Usage

```
limesurvey_api_call(method, params = list(), ...)
```

### Arguments

| | |
|---|---|
| method | Name of the API method to call. A complete list of methods can be found here: https://api.limesurvey.org/classes/remotecontrol_handle.html |
| params | Parameters to pass to the API |
| ... | Additional parameters passed from above |

### Value

A list containing the de-serialized response.

## wipe_survey_data *wipe_survey_data*

### Description

Delete all data collected by this survey.

### Usage

```
wipe_survey_data(survey_id)
```

### Arguments

| | |
|---|---|
| survey_id | ID of the survey from which the collected data shall be deleted. 6-digit integer. |

### Value

Nothing. Function is called for side effects on SQL table.

## Examples

```
# This example assumes a locally hosted `LimeSurvey` instance using a locally
# hosted MySQL server.
# On this `LimeSurvey` instance, there is a survey with the ID 123456.
## Not run:
connect_to_limesurvey(
  api_url = 'https://localhost/index.php/admin/remotecontrol',
  limesurvey_username = 'admin',
  limesurvey_password = '1234admin',
  mysql_host = '127.0.0.1',
  mysql_port = 3306,
  mysql_dbname = 'limesurvey',
  mysql_table_prefix = '',
  mysql_username = 'lime',
  mysql_password = '1234lime'
)

wipe_survey_data(123456)

## End(Not run)
```

# Index