Package 'g4sr'

September 17, 2021

Type Package

Title Go4Schools API Wrapper

Version 0.6.0

Maintainer Afsar Chowdhury <a.chowdhury@hydehighschool.uk>

Description An API wrapper for Go4Schools. Allows you to manage your school data.
Authentication is required for all requests. See your school's
administrator for an API key.

License GPLv3

URL https://github.com/afsarchowdhury/g4sr

BugReports https://github.com/afsarchowdhury/g4sr/issues

Encoding UTF-8

LazyData true

Imports crayon,
data.table,
dplyr,
ggplot2,
glue,
httr,
jsonlite,
lubridate,
magrittr,
stringr,
tidyr

RoxygenNote 7.1.1

Roxygen list(markdown = TRUE)

Suggests testthat,
knitr,
rmarkdown

VignetteBuilder knitr

R topics documented:

---

gfs_assessment_markbooks

Get markbooks.

Description

Returns the structure of markbooks in the chosen academic year, in terms of the marksheets they contain, and the markslots within each marksheet.

Usage

```
gfs_assessment_markbooks(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_assessment_markbooks(2020)
```

`gfs_assessment_marks`   *Get marks.*

Description

Returns details of the marks achieved by students for individual marks.

- Use `gfs_assessment_markbooks()` to resolve markbooks.
- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_assessment_marks(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_assessment_marks(2020)
```

`gfs_assessment_marksheets`
                      Get marksheets.

Description

Returns details of the grades achieved by students for individual marksheets.

- Use `gfs_assessment_markbooks()` to resolve markbooks.
- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_assessment_marksheets(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_assessment_marksheet_grades(2020)
```

```
gfs_attainment_exam_results
```
Get exam results.

Description

Returns details of the exam results achieved by students.

- Use `gfs_student_details()` to resolve student personal details.
- Use `gfs_teaching_subjects()` to resolve subject details.

Usage

```
gfs_attainment_exam_results(academicYear, yearGroup)
```

Arguments

academicYear   academic year as integer.

yearGroup   year group as string.

Examples

```
gfs_attainment_exam_results(2020, "11")
```

```
gfs_attainment_grades    Get grades.
```

Description

Returns details of the headline grades for students.

- Use `gfs_attainment_grade_types()` to resolve grade names.
- Use `gfs_student_details()` to resolve student personal details.
- Use `gfs_teaching_subjects()` to resolve subject details.

Usage

```
gfs_attainment_grades(academicYear, yearGroup)
```

Arguments

academicYear   academic year as integer.

yearGroup   year group as string.

Examples

```
gfs_attainment_grades(2020, "11")
```

gfs_attainment_grade_types

Get grade types.

Description

Returns details of the available types of headline grades.

Usage

```
gfs_attainment_grade_types(academicYear, yearGroup)
```

Arguments

academicYear   academic year as integer.

yearGroup   year group as string.

Examples

```
gfs_attainment_grade_types(2020, "11")
```

gfs_attainment_prior   *Get prior attainment.*

Description

Returns prior attainment for students.

- Use gfs_student_details() to resolve student personal details.

Usage

```
gfs_attainment_prior(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_attainment_prior(2020)
```

---

gfs_attendance_codes   *Get attendance codes.*

Description

Returns the Attendance codes and any aliases being used for the academic year.

Usage

```
gfs_attendance_codes(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_attendance_codes(2020)
```

---

gfs_attendance_student_lesson_marks
                          Get student lesson marks.

Description

Returns lesson attendance details for a particular date.

- Use `gfs_attendance_codes` to resolve attendance codes.
- Use `gfs_student_details()` to resolve student personal details.
- Use `gfs_classes()` to resolve timetabled classes.

Usage

```
gfs_attendance_student_lesson_marks(academicYear, goDate)
```

Arguments

academicYear   academic year as integer.

goDate   date as string in the form yyyy-mm-dd.

Examples

```
gfs_attendance_student_lesson_marks(2020, "2020-07-01")
```

```
gfs_attendance_student_session_marks
```
Get student session marks.

Description

Returns session attendance details for a particular date.

- Use `gfs_attendance_codes` to resolve attendance codes.
- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_attendance_student_session_marks(academicYear, goDate)
```

Arguments

academicYear	academic year as integer.
goDate	date as string in the form yyyy-mm-dd.

Examples

```
gfs_attendance_student_session_marks(2020, "2020-07-01")
```

```
gfs_attendance_student_summary
```
Get student attendance summary.

Description

Returns summary session data for all students for the academic year up to (but not including) today.

- Use `gfs_attendance_codes` to resolve attendance codes.
- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_attendance_student_summary(academicYear)
```

Arguments

academicYear	academic year as integer.

Examples

```
gfs_attendance_student_summary(2020)
```

```
gfs_behaviour_classification
```
Get classifications.

Description

Returns the current behaviour classification groups for the supplied academic year.

Usage

```
gfs_behaviour_classification(academicYear)
```

Arguments

```
academicYear     academic year as integer.
```

Examples

```
gfs_behaviour_classification(2020)
```

```
gfs_behaviour_events     Get events.
```

Description

Returns lists of events for a particular date within an academic year.

- Use `gfs_behaviour_event_types()` to resolve event characteristics.
- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_behaviour_events(academicYear, goDate)
```

Arguments

```
academicYear     academic year as integer.
goDate           date as string in the form yyyy-mm-dd.
```

Examples

```
gfs_behaviour_events(2020, "2020-03-19")
```

```
gfs_behaviour_events_range
```
Get events over a range of dates.

Description

Returns list of events over a range of dates within an academic year.

- Use gfs_behaviour_event_types() to resolve event characteristics.
- Use gfs_student_details() to resolve student personal details.

Usage

```
gfs_behaviour_events_range(academicYear, goDateStart, goDateEnd)
```

Arguments

academicYear	academic year as integer.
goDateStart	start date of range as string in the form yyyy-mm-dd.
goDateEnd	end date of range as string in the form yyyy-mm-dd.

Examples

```
gfs_behaviour_events_range(2020, "2020-03-16", "2020-03-19")
```

```
gfs_behaviour_event_types
```
Get event types.

Description

Returns the current definition of the behaviour types for the supplied academic year.

Usage

```
gfs_behaviour_event_types(academicYear)
```

Arguments

academicYear	academic year as integer.

Examples

```
gfs_behaviour_event_types(2020)
```

```
gfs_behaviour_followup
```
                              Get follow-up actions.

Description

Returns the current definitions for follow up behaviour actions for the supplied academic year.

Usage

```
gfs_behaviour_followup(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_behaviour_followup(2020)
```

```
gfs_behaviour_immediate
```
                              Get immediate actions.

Description

Returns the current definitions for immediate behaviour actions for the supplied academic year.

Usage

```
gfs_behaviour_immediate(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_behaviour_immediate(2020)
```

---

gfs_calendar   *Get calendar.*

Description

Returns details of when the school is open, closed or open for staff training only in the chosen academic year.

- Use `gfs_timetables()` to resolve date, week, and period details.

Usage

```
gfs_calendar(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_calendar(2020)
```

---

gfs_classes   *Get classes.*

Description

Returns details of all timetabled classes in the chosen academic year.

- Use `gfs_timetables()` to resolve date, week, and period details.
- Use `gfs_teaching_teachers()` to resolve teacher details.

Usage

```
gfs_classes(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_classes(2020)
```

`gfs_contacts_parental`   *Get parental contacts.*

Description

Returns details of parental contacts for students.

Usage

```
gfs_contacts_parental()
```

`gfs_reports`   *Get reports.*

Description

Returns details of available progress reports in the chosen academic year, including the grades and attributes associated with each set of reports.

Usage

```
gfs_reports(academicYear)
```

Arguments

`academicYear`   academic year as integer.

Examples

```
gfs_reports(2021)
```

`gfs_reports_attributes`
                              Get report attributes.

Description

Returns attributes for students in chosen academic year and report.

- Use `gfs_reports` to resolve name and points value for an attribute choice for the student.
- Use `gfs_teaching_subjects()` to resolve subject details.
- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_reports_attributes(academicYear, reportID)
```

Arguments

academicYear	academic year as integer.
reportID	report ID as integer.

Examples

```
gfs_reports_attributes(2021, 57102)
```

---

gfs_reports_comments   *Get report comments*

Description

Returns comments for students in chosen academic year and report.

- Use `gfs_reports` to resolve name and points value for an attribute choice for the student.
- Use `gfs_teaching_subjects()` to resolve subject details.
- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_reports_comments(academicYear, reportID)
```

Arguments

academicYear	academic year as integer.
reportID	report ID as integer.

Examples

```
gfs_reports_comments(2021, 57102)
```

---

gfs_reports_grades   *Get report grades*

Description

Returns grades for students in chosen academic year and report.

- Use `gfs_reports` to resolve name and points value for an attribute choice for the student.
- Use `gfs_teaching_subjects()` to resolve subject details.
- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_reports_grades(academicYear, reportID)
```

Arguments

academicYear   academic year as integer.

reportID   report ID as integer.

Examples

```
gfs_reports_grades(2021, 57102)
```

---

gfs_school   *Get school details.*

Description

Returns details of the school and the available academic years.

Usage

```
gfs_school()
```

---

gfs_setup   *Run set up for gfs functions.*

Description

Use this function to store your API key for use with gfs functions. Authentication is required for all requests. See your school's administrator for an API key.

Usage

```
gfs_setup(api_key)
```

Arguments

api_key   API key as string.

Examples

```
gfs_setup("a96650481f0245eea396726f85ac7049")
```

`gfs_student_classes`   *Get student classes.*

Description

Returns details of which students are linked to which timetabled classes in the chosen academic year.

- Use `gfs_classes()` to resolve class details.
- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_student_classes(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_student_classes(2020)
```

`gfs_student_demographics`

Get demographic attributes.

Description

Returns the following attributes for students for the chosen academic year: country of birth, home language, first language, and nationality.

- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_student_demographics(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_student_demographics(2020)
```

---

gfs_student_details   *Get students details.*

Description

Returns the current personal details (DOB, gender, names) of the students that were at school in the chosen academic year.

Usage

```
gfs_student_details(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_student_details(2020)
```

---

gfs_student_edu_details

                            Get education details.

Description

Returns education details for students.

- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_student_edu_details(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_student_edu_details(2020)
```

gfs_student_general   *Get general attributes.*

Description

Returns general student attributes for the chosen academic year. This will include all custom attributes and also system attributes not returned by other endpoints.

- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_student_general(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_student_general(2020)
```

gfs_student_medical   *Get student medical conditions.*

Description

Returns medical conditions for students.

- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_student_medical(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_student_medical(2020)
```

---

gfs_student_send   *Get SEND attributes.*

Description

Returns the following attributes for students for the chosen academic year: SEN code, SEN status, educational needs, gifted/talented, gifted/talented subject.

- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_student_send(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_student_send(2020)
```

---

gfs_student_sensitive *Get sensitive attributes.*

Description

Returns the following attributes for students for the chosen academic year: FSM, FSMEver6, EAL, disadvantaged, ethnicity, ethnicity code, ever in care, looked after, pupil premium indicator, service child, traveller type.

- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_student_sensitive(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_student_sensitive(2020)
```

```
gfs_teaching_departments
```
Get departments.

Description

Returns details of departments in the chosen academic year.

Usage

```
gfs_teaching_departments(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_teaching_departments(2020)
```

gfs_teaching_groups   *Get groups.*

Description

Returns details of teaching groups for each subject in the chosen academic year.

  • Use `gfs_teaching_subjects()` to resolve subject details.

Usage

```
gfs_teaching_groups(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_teaching_groups(2020)
```

---

gfs_teaching_groups_students

Get groups and students.

Description

Returns details of the students in each teaching group in the chosen academic year.

- Use `gfs_teaching_groups()` to resolve group details.
- Use `gfs_student_details()` to resolve student personal details.

Usage

```
gfs_teaching_groups_students(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_teaching_groups_students(2020)
```

---

gfs_teaching_groups_teachers

Get groups and teachers.

Description

Returns details of teachers assigned to each teaching group in the chosen academic year.

- Use `gfs_teaching_groups()` to resolve group details.
- Use `gfs_teaching_teachers()` to resolve teacher details.

Usage

```
gfs_teaching_groups_teachers(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_teaching_groups_teachers(2020)
```

gfs_teaching_subjects   *Get subjects.*

Description

Returns all subjects in the chosen academic year.

- Use `gfs_teaching_departments()` to resolve department details.

Usage

```
gfs_teaching_subjects(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_teaching_subjects(2020)
```

gfs_teaching_teachers   *Get teachers.*

Description

Returns details of the teachers in the chosen academic year.

Usage

```
gfs_teaching_teachers(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_teaching_teachers(2020)
```

gfs_timetables   *Get timetables.*

Description

Returns details of the timetables and their associated periods for the chosen academic year.

Usage

```
gfs_timetables(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
gfs_timetables(2020)
```

gfs_users_parent   *Get parents.*

Description

Returns details of parent user accounts.

Usage

```
gfs_users_parent()
```

gfs_users_staff   *Get staff.*

Description

Returns details of staff user accounts.

Usage

```
gfs_users_staff()
```

Index