Package 'hydeR'

March 9, 2024

Type Package

Title R functions for Hyde High School

Version 0.8.2

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

Description R funnctions for Hyde High School.
The package makes extensive use of g4sr, an API wrapper for Go4Schools.
Authentication is required for all g4sr requests. See your school's
administrator for an API key.

License GPLv3

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

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

Encoding UTF-8

LazyData true

Imports crayon,
data.table,
dplyr,
ggplot2,
glue,
httr,
jsonlite,
lubridate,
magrittr,
quanteda,
quanteda.textstats,
readtext,
scales,
stringr,
tidyr

Remotes afsarchowdhury/g4sr

RoxygenNote 7.2.3

Roxygen list(markdown = TRUE)

Suggests testthat,
knitr,
rmarkdown

VignetteBuilder knitr

R topics documented:

---

hhs_attainment   *Get clean attainment data.*

Description

Returns clean details of attainment in the chosen academic year for chosen year group.

Usage

```
hhs_attainment(academicYear, yearGroup)
```

Arguments

academicYear   academic year as integer.

yearGroup   year group as string.

Examples

```
hhs_attainment(2020, "10")
```

hhs_attainment_multiple

Get clean attainment data for multiple year groups.

Description

Returns clean details of attainment in the chosen academic year for multiple year groups.

Usage

```
hhs_attainment_multiple(academicYear, yearGroupFrom = "7", yearGroupTo = "11")
```

Arguments

academicYear	academic year as integer.
yearGroupFrom	starting year group as string. Defaults to "7".
yearGroupTo	ending year group as string. Defaults to "11".

Examples

```
hhs_attainment_multiple(2021, "7", "11")
```

hhs_attendance_student_lesson

Get clean student lesson marks.

Description

Returns clean details of student lesson marks for chosen academic year and date.

Usage

```
hhs_attendance_student_lesson(academicYear, goDate)
```

Arguments

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

Examples

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

hhs_attendance_student_lesson_range

Get clean student lesson marks over a range of dates.

Description

Returns clean student lesson marks over a range of dates for chosen academic year.

Usage

```
hhs_attendance_student_lesson_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

```
hhs_attendance_student_lesson_range(2021, "2021-07-12", "2021-07-16")
```

hhs_attendance_student_session

Get clean student session marks.

Description

Returns clean details of student session marks for chosen academic year, date, and session.

Usage

```
hhs_attendance_student_session(academicYear, goDate, session = NULL)
```

Arguments

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

Examples

```
hhs_attendance_student_session(2020, "2020-07-01")
hhs_attendance_student_session(2020, "2020-07-01", "am")
```

hhs_attendance_student_session_range

Get clean student session marks over a range of dates.

Description

Returns clean student session marks over a range of dates for chosen academic year.

Usage

```
hhs_attendance_student_session_range(
  academicYear,
  goDateStart,
  goDateEnd,
  session = NULL
)
```

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.
session	session as string.

Examples

```
hhs_attendance_student_session_range(2021, "2021-07-12", "2021-07-16")
hhs_attendance_student_session_range(2021, "2021-07-12", "2021-07-16", session = "am")
```

hhs_attendance_student_summary

Get clean student attendance summary.

Description

Returns clean details of student attendance summary for chosen academic year.

Usage

```
hhs_attendance_student_summary(academicYear)
```

Arguments

academicYear	academic year as integer.

Examples

```
hhs_attendance_student_summary(2020)
```

---

hhs_behaviour_events_range
Get clean behaviour events.

Description

Returns clean behaviour events over a range of dates within in the chosen academic year.

Usage

```
hhs_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

```
hhs_behaviour_events_range(2021, "2021-07-12", "2021-07-16")
```

---

hhs_class_list_student
Get student-focused class list.

Description

Returns clean details of all classes in the chosen academic year for chosen student.

Usage

```
hhs_class_list_student(academicYear, student)
```

Arguments

academicYear	academic year as integer.
student	student search string.

Examples

```
hhs_class_list_student(2020, "smith")
```

hhs_class_list_teacher

Get teacher-focused class list.

Description

Returns clean details of all classes in the chosen academic year for chosen staff. If staff code is not provided, class list for all staff are returned.

Usage

```
hhs_class_list_teacher(
  academicYear,
  staffCode = NULL,
  yearGroupFrom = "7",
  yearGroupTo = "11"
)
```

Arguments

academicYear	academic year as integer.
staffCode	case-sensitive staff code as string.
yearGroupFrom	starting year group as string. Defaults to "7".
yearGroupTo	ending year group as string. Defaults to "11".

Examples

```
hhs_class_list_teacher(2020, "ACH")
hhs_class_list_teacher(2020)
```

hhs_detentions   *Get clean school detention data.*

Description

Returns clean school detention data for chosen academic year and date range, along with a line plot.

Usage

```
hhs_detentions(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

```
hhs_detentions(2022, "2021-09-01", "2021-09-30")
```

---

hhs_exam_results   *Get clean external examination results.*

Description

Returns clean details of external examination results in the chosen academic year for chosen year group.

Usage

```
hhs_exam_results(academicYear, yearGroup, type = NULL)
```

Arguments

academicYear	academic year as integer.
yearGroup	year group as string.
type	qualification type as string.

Examples

```
hhs_exam_results(2020, "11")
hhs_exam_results(2020, "11", type = "gcse")
hhs_exam_results(2020, "11", type = c("gcse", "btec"))
```

---

hhs_exclusions   *Get clean school exclusion data.*

Description

Returns clean school exclusion data for chosen academic year and date range, along with a line plot.

Usage

```
hhs_exclusions(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

```
hhs_exclusions(2022, "2021-09-01", "2021-09-30")
```

---

hhs_markbook_attainment

Get clean markbook attainment.

Description

Returns clean details of attainment in the chosen academic year for all markbooks.

Usage

```
hhs_markbook_attainment(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
hhs_markbook_attainment(2022)
```

---

hhs_raw_student_details_general

Get raw general student details.

Description

Returns raw general student details for chosen academic year.

Usage

```
hhs_raw_student_details_general(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
hhs_student_details_general(2021)
```

---

hhs_readability   *Get clean readability data.*

Description

Returns clean readability data for texts in a given directory.

Usage

```
hhs_readability(text_dir, return_full = FALSE)
```

Arguments

text_dir   path to directory containing texts as string.

return_full   return full readability statistics as logical. Default FALSE.

Examples

```
hhs_readability("C:/Users//User/Texts", TRUE)
```

---

hhs_school   *Get school details.*

Description

Returns details of the school and the available academic years.

Usage

```
hhs_school()
```

---

hhs_school_population   *Get clean school population data.*

Description

Returns clean details of school population for all available academic years, along with a line plot.

Usage

```
hhs_school_population()
```

Examples

```
hhs_school_population()
```

---

hhs_setup *Run set up for hhs functions.*

Description

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

Usage

```
hhs_setup(api_key)
```

Arguments

api_key   API key as string.

Examples

```
hhs_setup("a96650481f0245eea396726f85ac7049")
```

---

hhs_student_details_general
Get clean general student details.

Description

Returns clean general student details for chosen academic year.

Usage

```
hhs_student_details_general(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
hhs_student_details_general(2021)
```

---

hhs_student_send_search
                                Get clean SEND notes search.

Description

Returns clean student details based on SEND notes search for chosen academic year.

Usage

```
hhs_student_send_search(academicYear, notesSearch)
```

Arguments

academicYear   notesSearch as string.

Examples

```
hhs_student_send_search(2021, "extra time")
hhs_student_send_search(2021, "irlen")
```

---

hhs_targets   *Get clean targets.*

Description

Returns clean details of targets in the chosen academic year and subject for all year groups.

Usage

```
hhs_targets(academicYear, subject = NULL)
```

Arguments

academicYear   academic year as integer.

subject   subject as string. . Defaults to null.

Examples

```
hhs_targets(2022, "Combined Science 9-1")
```

---

hhs_timetable *Get clean timetable data.*

Description

Returns clean timetable data for chosen academic year.

Usage

```
hhs_timetable(academicYear)
```

Arguments

academicYear   academic year as integer.

Examples

```
hhs_timetable(2021)
```

Index