

Zugriffsbeschränkungen im Expertenmodus

Einstieg in die Expertenregeln



Die Einstellungen in den Tabs "Sichtbarkeit" und "Zugang" stehen nur in herkömmlichen Kursen zur Verfügung. Der Expertenmodus kann also nur in herkömmlichen Kursen nicht in "Lernpfad Kursen" verwendet werden.

Bei vielen Kursbausteinen können Sie unter den Tabs Sichtbarkeit und Zugang weitere Einstellungen vornehmen. Sie können den Kursbaustein beispielsweise für Lernende sperren, nur für bestimmte Gruppen zugänglich machen oder ihn datumsabhängig freischalten.

Bei komplizierten Sichtbarkeits- und Zugangsregeln können Sie den "Expertenmodus" verwenden. Dies erlaubt Ihnen, die Sichtbarkeit und den Zugang eines Kursbausteins frei nach Ihren Bedürfnissen zu konfigurieren. Beispielsweise können Sie den Zugang zu einem Kursbaustein nur für bestimmte Benutzernamen freischalten, mehrere Einschränkungen miteinander verknüpfen oder mit relativen Daten arbeiten. Ein Beispiel soll dies erläutern:

Sie wollen einen Kursfragebogen erst in der letzten Kurswoche frei schalten, möchten diese Option aber schon mal einrichten, damit Sie es später nicht vergessen.

Sie schalten also den Kursbaustein „Umfrage“ datumsabhängig frei, damit Sie sich im Kursverlauf nicht mehr darum kümmern müssen. In den Tabs *Sichtbarkeit* und *Zugang* des „Fragebogens“ können Sie hierfür im einfachen Modus das Anfangs- und Enddatum eingeben. Sie können Ihren Fragebogen auch nur für eine bestimmte Teilnehmergruppe zugänglich machen. Wählen Sie hierfür unter Sichtbarkeit bzw. Zugang ergänzend "Gruppenabhängig". So könnten Sie z.B. bei institutionsübergreifenden Online-Kursen zwei unterschiedliche Fragebögen verwenden. Voraussetzung ist lediglich, dass Sie die Kursteilnehmenden in (zwei) unterschiedliche Gruppen geteilt haben die sie nun zuweisen können.

Expertenregeln dienen in erster Linie dazu, Ihnen Arbeit abzunehmen oder sie zu erleichtern und auch komplexe Szenarien umzusetzen. Wie bei einer Sprache folgen die Expertenregeln einer Syntax oder Satzbauregel. Sollte man einen syntaktischen Fehler machen, dann weist OpenOLAT Sie darauf hin. Dies hilft einem vor allem am Anfang, wenn man keine oder nur geringe Programmierkenntnisse hat. Eine Expertenregel überprüft, ob ein bestimmtes Attribut WAHR oder FALSCH ist.

Als Einstieg in die Syntax der Expertenregel empfiehlt es sich, zunächst im einfachen Modus eine Regel zu definieren. Sie erstellen einen Kursbaustein, z. B. eine „Einzelne Seite“ und klicken unter dem Tab „Zugang“ auf „Für Lernende gesperrt“.

Dann klicken Sie auf „Expertenmodus anzeigen“ und sehen Ihre erste Expertenregel in Syntax Form:

```
( ( isCourseCoach(0) | isCourseAdministrator(0) ) )
```

Der gesamte Ausdruck ist doppelt eingeklammert. Die beiden äußeren Klammern kann man in diesem Fall auch weglassen. Probieren Sie es einfach aus. CourseCoach ist der Kursbetreuer und CourseAdministrator der Kursbesitzer. Der senkrechte Strich in der Mitte „|“ steht für den Booleschen Operator ODER. Diese Expertenregel ist WAHR für den Betreuer ODER den Besitzer. Nur diese beiden haben Zugang auf den Kursbaustein.

Nun ändern Sie den Booleschen Operator in „&“:

```
isCourseCoach(0) & isCourseAdministrator(0)
```

Dies bedeutet, dass nur Betreuer, die gleichzeitig auch Besitzer sind, Zugriff haben. Diese Einstellung ist nur im Expertenmodus möglich.

Sie können beliebig viele Szenarien durchspielen und weitere Attribute und Operatoren einfügen. Um sich in die Funktion der Regeln besser einzuarbeiten, finden Sie in diesem Kapitel weitere Attribute und Beispiele, deren Auswirkungen näher erläutert werden.

Konfiguration von Expertenregeln

Eine Expertenregel prüft, ob ein Attribut einen bestimmten Wert besitzt.

Attribut	Beschreibung	Beispiel Expertenregel
isGuest	nur für Gäste zugänglich	isGuest(0)
isCourseCoach	nur für Betreuer sichtbar	isCourseCoach(0)
isUser	nur für einen bestimmten Benutzer verfügbar	isUser("pmuster")

Arbeiten mit den Konstanten "TRUE" und "FALSE"

Die Konstanten „true“ und „false“ prüfen das Vorhandensein („true“ bzw. „1“) oder Nicht-Vorhanden-Sein („false“ bzw. „0“) eines Attributs. Man spricht von einer sogenannten Booleschen Variablen (benannt nach George Boole dem Begründer der Booleschen Algebra). Dabei handelt es sich um eine Variable, die nur endlich viele Werte oder Zustände einnehmen kann. In diesem Fall kann die Variable nur zwei Zustände oder Werte annehmen („true“ = „1“ = wahr, vorhanden oder „false“ = „0“ = falsch, nicht-vorhanden).

Zur praktischen Erläuterung im OLAT-Kontext soll uns eine Expertenregel für den Zugang zu einem Kursbaustein bzw. Bereich innerhalb eines Kurses dienen.

Fall 1: Nur Gast-Nutzer sollen Zugang zu einem Baustein erhalten, beispielsweise um Bereiche für Gäste und OLAT Benutzer voneinander zu trennen.

Der jeweilige Nutzer erhält also Zugang, wenn er das Attribut „isGuest“ wahr ist. Für diese Expertenregel gibt es drei Alternativen:

isGuest(0) oder **isGuest(0)=1** oder **isGuest(0)=true**

Fall 2: Hier sollen alle Nutzer außer den Gast-Nutzern einen Zugang erhalten. Der jeweilige Nutzer erhält also Zugang, wenn das Attribut „isGuest“ nicht wahr bzw. nicht vorhanden ist. Für diese Expertenregel gibt es zwei Alternativen:

isGuest(0)=0 oder **isGuest(0)=false**

Eine ausführliche Liste aller wichtigen Bestandteile für die Expertenregeln finden Sie in der nachfolgenden Box.

Typ	Syntax	Bedeutung
Konstanten	<i>TRUE</i> oder <i>1</i>	Wahr
	<i>FALSE</i> oder <i>0</i>	Falsch
	<i>ANY_COURSE</i>	Abfrage soll für jeden Kurs gelten (nur für isCourseAdministrator(), isCourseCoach(), isCourseParticipant())
Variablen	<i>now</i>	Momentane Server-Systemzeit
Funktionen	<i>date("[date]")</i>	Datum abfragen
	<i>inLearningGroup("[string]")</i>	Gibt TRUE für alle Mitglieder der Lerngruppe [string]
	<i>inRightGroup("[string]")</i>	Gibt TRUE für alle Mitglieder der Rechtegruppe [string]
	<i>isLearningGroupFull("[string]")</i>	Gibt für die angegebene Lerngruppe den Boolean TRUE (=voll) oder FALSE (=nicht voll) zurück.
	<i>isUser("[string]")</i>	Gibt TRUE für den Benutzer mit dem Benutzernamen [string]
	<i>inLearningArea("[string]")</i>	Gibt TRUE für alle Mitglieder der Gruppen im Lernbereich [string]
	<i>isGlobalAuthor(0)</i>	Gibt TRUE für alle Mitglieder der OLAT-Autorengruppe
	<i>isCourseAdministrator(0)</i>	Gibt TRUE für alle Besitzer Ihres Kurses (Lernressource)
	<i>isCourseAdministrator(ANY_COURSE)</i>	Gibt TRUE für alle Benutzer die Besitzer eines beliebigen Kurses auf dem System sind (Lernressource)
	<i>isCourseCoach(0)</i>	Gibt TRUE für alle Benutzer, die eine Lerngruppe oder den gesamten Kurs betreuen
	<i>isCourseCoach(ANY_COURSE)</i>	Gibt TRUE für alle Benutzer, die eine Lerngruppe eines beliebigen Kurses oder einen beliebigen gesamten Kurs betreuen
	<i>isCourseParticipant(0)</i>	Gibt TRUE für alle Teilnehmer des Kurses
	<i>isCourseParticipant(ANY_COURSE)</i>	Gibt TRUE für alle Benutzer, die in einem beliebigen Kurs als Teilnehmer eingetragen sind
	<i>isGuest(0)</i>	Gibt TRUE für alle Benutzer, die OLAT als Gäste besuchen
	<i>hasAttribute("[AttrName]" ; [string])</i>	Gibt TRUE, wenn [string] dem Wert des AAI-Attributs [AttrName] des jeweiligen Benutzers entspricht. AAI - Generelle Informationen AAI-Attribute Spezifikation der AAI-Attribute (pdf-Datei)
	<i>isInAttribute("[AttrName]" ; [substring])</i>	Gibt TRUE, wenn [substring] einem Teil des Wertes des AAI-Attributs [AttrName] des jeweiligen Benutzers entspricht.

	<code>getUserProperty("[userPropertyname]")</code>	Gibt den Wert des spezifizierten Benutzerattributes zurück. Mit "=" kann dieser Wert mit einem fixen Wert verglichen werden.
	<code>getPassed([integer] ")</code>	Gibt vom Kursbaustein mit spezifizierter ID den Boolean TRUE (=Bestanden) oder FALSE (=Nicht bestanden) zurück
	<code>getScore([integer] ")</code>	Gibt vom Kursbaustein mit spezifizierter ID die Anzahl Punkte zurück
	<code>getAttempts([integer] ")</code>	Gibt vom Kursbaustein mit spezifizierter ID die Anzahl abgeschlossener Versuche zurück. Kann auf Kursbausteine vom Typ <i>Test</i> , <i>Selbsttest</i> , <i>Fragebogen</i> angewendet werden.
	<code>getLastAttemptDate([integer] ")</code>	Gibt vom Kursbaustein mit spezifizierter ID das Datum des letzten Versuches zurück. Die Anwendung ist gleich wie die getAttempts Methode.
	<code>getInitialEnrollmentDate([integer] ")</code>	Gibt vom Kursbaustein <i>Einschreibung</i> mit spezifizierter ID das Datum des erstmaligen Einschreibens des betreffenden Kursteilnehmers zurück.
	<code>getRecentEnrollmentDate([integer] ")</code>	Gibt vom Kursbaustein <i>Einschreibung</i> mit spezifizierter ID das Datum des letzten Einschreibens des betreffenden Kursteilnehmers zurück.
	<code>getInitialCourseLaunchDate(0)</code>	Gibt das Datum des erstmaligen Kursbesuchs des betreffenden Kursteilnehmers zurück.
	<code>getRecentCourseLaunchDate(0)</code>	Gibt das Datum des letzten Kursbesuchs des betreffenden Kursteilnehmers zurück.
	<code>getPassedWithCourseId([integer-1] ; [integer-2] ")</code>	Gibt vom Kursbaustein mit ID=[integer-2] des Kurses mit ID=[integer-1] den Boolean TRUE (=Bestanden) oder FALSE (=Nicht bestanden) zurück
	<code>getScoreWithCourseId([integer-1] ; [integer-2] ")</code>	Gibt vom Kursbaustein mit ID=[integer-2] des Kurses mit ID=[integer-1] die Anzahl Punkte zurück
	<code>hasUserProperty("[userPropertyname]", "[string]")</code>	Gibt TRUE, wenn [string] dem Wert des Benutzerattribut [userPropertyname] des jeweiligen Benutzers entspricht.
	<code>userPropertyStartswith(" [userPropertyname]", "[substring]")</code>	Gibt TRUE, wenn das Benutzerattribut [userPropertyname] des jeweiligen Benutzers mit [substring] beginnt.
	<code>userPropertyEndswith(" [userPropertyname]", "[substring]")</code>	Gibt TRUE, wenn das Benutzerattribut [userPropertyname] des jeweiligen Benutzers mit [substring] endet.
	<code>isInUserProperty(" [userPropertyname]", "[substring]")</code>	Gibt TRUE, wenn das Benutzerattribut [userPropertyname] des jeweiligen Benutzers [substring] enthält.
	<code>isNotInUserProperty(" [userPropertyname]", "[substring]")</code>	Gibt TRUE, wenn das Benutzerattribut [userPropertyname] des jeweiligen Benutzers [substring] nicht enthält.
	<code>hasNotUserProperty(" [userPropertyname]", "[string]")</code>	Gibt TRUE, wenn der jeweilige Benutzer das Benutzerattribut [userPropertyname] nicht besitzt.
	<code>hasLanguage("de")</code>	Gibt TRUE, wenn der jeweilige Benutzer Deutsch als Systemsprache eingestellt hat. Für Englisch "de" durch "en" ersetzen.
Einheiten	<i>min</i>	Minuten
	<i>h</i>	Stunden
	<i>d</i>	Tage
	<i>w</i>	Wochen
	<i>m</i>	Monate
Operatoren	=	gleich
	>	größer als
	<	kleiner als
	>=	größer gleich
	<=	kleiner gleich
	*	Multiplikation
	/	Division
	+	Addition
-	Subtraktion	
Booleans	&	Logisches UND

Verschiedene Expertenregeln benötigen Benutzerattribute um Einschränkungen vornehmen zu können. Dies ermöglicht bspw. die Einschränkung von Benutzern nach Name, Geschlecht, Adresse, Studienfach und weitere. Diese Benutzerattribute sind in der Regel im Benutzerprofil enthalten. OpenOLAT stellt standardisierte Begriffe für diese Attribute bereit. Folgende Expertenregeln benötigen Benutzerattribute:

- `getUserProperty("[userPropertyname]")`
- `hasUserProperty("[userPropertyname]", "[string]")`
- `userPropertyStartswith("[userPropertyname]", "[substring]")`
- `userPropertyEndswith("[userPropertyname]", "[substring]")`
- `isInUserProperty("[userPropertyname]", "[substring]")`
- `isNotInUserProperty("[userPropertyname]", "[substring]")`
- `hasNotUserProperty("[userPropertyname]", "[string]")`

Die folgenden Benutzerattribute stehen in OpenOLAT zur Verfügung. Bitte beachten Sie, dass eine Einschränkung mittels Benutzerattributen nur dann erfolgreich sein kann, wenn diese in ihrem System auch genutzt werden, und standardmässig ausgefüllt sind. Überprüfen Sie im persönlichen Menü unter Konfiguration/Profil die verfügbaren Benutzerattribute. Bei Fragen kontaktieren Sie ihren Systemadministrator.

Benutzerdaten		Kontaktdaten		Adressdaten	
userName	Benutzername	telPrivate	Telefon privat	street	Strasse
firstName	Vorname	telMobile	Telefon Mobil	extendedAddress	Adresszusatz
lastName	Nachname	telOffice	Telefon Geschäft	poBox	Postfach
email	E-Mail-Adresse	skype	Skype ID	zipCode	PLZ
creationDateDisplayProperty	Erstelldatum des Benutzers	xing	Xing	region	Region / Kanton
lastloginDateDisplayProperty	Letzter Login des Benutzers	homepage	Homepage	city	Stadt
birthDay	Geburtsdatum			country	Land
gender	Geschlecht			countryCode	Länderkürzel
Organisation		Berufliche Kontaktdaten		Verschiedenes	
institutionalName	Institution	department	Dienststelle / Firma	typeOfUser	Art von Benutzer
institutionalUserIdentifier	Institutionsnummer (Matrikelnummer)	officeStreet	Dienstadresse - Strasse	rank	Dienstgrad / Amtsbezeichnung
institutionalEmail	Institutions E-Mail	extendedOfficeAddress	Dienstadresszusatz	socialSecurityNumber	Sozialversicherungsnummer
orgUnit	Organisationseinheit / Studiengruppe	officeZipCode	Dienst-Postleitzahl	degree	Akademischer Grad
studySubject	Studienfach	officeCity	Dienstadresse - Stadt	position	Funktion / Stellung
graduation	Abschlussjahr	officeCountry	Dienstadresse - Land	userInterests	Expertise
		officeMobilePhone	Dienstmobiltelefon		

Beispiele für die Anwendung:

- Es sollen nur Kursteilnehmer eines bestimmten Studienganges Zugang erhalten:

```
getUserProperty("studySubject") = "Maschinenbau"
```

Nun muss, wer Zugang haben möchte, in seinem Profil, im Feld Studienfach Maschinenbau eingetragen haben.

- Sollen nur Kursteilnehmer Zugang erhalten, die nichts im Feld Studiengang eingetragen haben, lautet die Regel:

```
getUserProperty("studySubject") = ""
```

- Sollen nur Kursteilnehmer Zugang erhalten, die irgendeinen Studiengang eingetragen haben, muss die Regel lauten:

```
getUserProperty("studySubject") = "" = false
```

oder

```
getUserProperty("studySubject") = "" = 0
```

Es gibt verschiedene Möglichkeiten, die einzelnen Regeln miteinander zu verknüpfen. Die beiden wichtigsten Operatoren zur Verknüpfung von Attributen sind:

- UND-Verknüpfung: &
- ODER-Verknüpfung: |

Bitte beachten Sie, dass eine ODER-Verknüpfung vor einer UND-Verknüpfung gemacht wird. Damit die UND-Verknüpfung zuerst gemacht wird, müssen Klammern gesetzt werden.

Beispiel: Die Expertenregel (inGroup("Teilnehmende Intensivkurs") | isCourseCoach(0)) lässt entweder Teilnehmende des Intensivkurses oder alle Gruppenbetreuer auf den Kursbaustein zugreifen.

Nachfolgend sind einige Beispiele aufgeführt, die Ihnen zeigen, wie Sie die Expertensyntax verwenden können.

Beispiele für Expertenregeln in den Tabs «Sichtbarkeit», «Zugang» und «Punkte» (Struktur-Baustein)

Beispiele für Expertenregeln in den Tabs «Sichtbarkeit», «Zugang» und «Punkte» (Struktur-Baustein)

inLearningGroup("Anfänger") = 0

Mit Ausnahme der Gruppe «Anfänger» ist der Kursbaustein für alle Kursteilnehmer sichtbar.

(now >= date("22.07.2018 12:00")) & (now <= date("23.12.2018 18:00")) | inLearningGroup("Betreuer")

Der Kursbaustein ist zwischen dem 22.07.2018 und 23.12.2018 für alle Kursteilnehmer sichtbar, während er für Mitglieder der Lerngruppe «Betreuer» jederzeit sichtbar ist.

(now >= date("03.09.2018 00:00")) & (now <= date("13.10.2018 00:00")) & inRightGroup("Assessoren") | isUser("schmidt")

Der Kursbaustein ist zwischen dem 03.09.2018 und 13.10.2018 für alle Kursteilnehmer der Rechtegruppe «Assessoren» sichtbar, während er für die Person mit dem Benutzernamen «schmidt» jederzeit sichtbar ist.

hasAttribute("swissEduPersonStudyBranch3", "6200")

Ausschliesslich Studierende der Humanmedizin haben Zugriff auf den Kursbaustein.

Siehe auch:

[AAI-Attribute](#)

[Spezifikation der AAI-Attribute \(pdf-Datei\)](#)

hasAttribute("swissEduPersonHomeOrganization", "uzh.ch")

Ausschliesslich Studierende der Universität Zürich haben Zugriff auf den Kursbaustein.

Siehe auch:

[AAI-Attribute](#)

[Spezifikation der AAI-Attribute \(pdf-Datei\)](#)

isInAttribute("surname", "Mue")

Gibt TRUE für alle Personen, deren Attribut *surname* die Buchstabenfolge "Mue" enthält. Gibt z.B. TRUE für den Wert "Mueller" oder "Muehlebacher"

Siehe auch:

[AAI-Attribute](#)

[Spezifikation der AAI-Attribute \(pdf-Datei\)](#)

isInAttribute("eduPersonEntitlement", "http://vam.uzh.ch")

Gibt TRUE für alle Personen, deren Attribut *eduPersonEntitlement* den Wert "<http://vam.uzh.ch>" enthält. Gibt z.B. auch TRUE für den Wert "<http://vam.uzh.ch/surgery>"

Siehe auch:

[AAI-Attribute](#)

[Spezifikation der AAI-Attribute \(pdf-Datei\)](#)

(getUserProperty("orgUnit") = "Sales")

Prüft ob eine Person in der Organisationseinheit "Sales" ist. Sinnvoll z.B. wenn die Daten automatisiert aus LDAP übernommen werden.

(getPassed("69742969114730") | getPassed("69742969115733") | getPassed("69742969118009")) * 10

Diese Regel wird im Tab «*Punkte*» -> «*Punkte verarbeiten*» des Bausteins *Struktur* gesetzt. Der Baustein *Struktur* zeigt 10 Punkte, wenn einer der Tests (Kursbaustein-IDs "69742969114730", "69742969115733" oder "69742969118009") bestanden wurde, sonst 0 Punkte.

(getScore("69742969114730") + getScore("69742969115733") + getScore("69742969118009")) >= 140 | getPassed("69978845384688")

Diese Regel wird im Tab «*Punkte*» -> «*Bestanden wenn*» des Bausteins *Struktur* gesetzt. Der Baustein *Struktur* zeigt ein «*Bestanden*», wenn in allen Tests zusammen minimal 140 Punkte erzielt werden oder wenn manuell ein «*Bestanden*» gesetzt wird (Kursbaustein *Bewertung* mit ID "69978845384688").

getAttempts("70323786958847") > 0

Gibt TRUE, sobald der betreffende Kursteilnehmer den Test mit spezifizierter ID ein erstes Mal abgeschlossen hat, also sobald ein Versuch erfolgte unabhängig von der Punktzahl.

getLastAttemptDate("70323524635734") + 24h < now

Gibt TRUE, wenn der letzte Testversuch 24 Stunden zurückliegt.

getInitialEnrollmentDate("70323786958847") <= date("26.5.2018 18:00")

Gibt TRUE für diejenigen Kursteilnehmer, die sich vor 18 Uhr des 26. Mai 2018 über den Kursbaustein *Einschreibung* mit spezifizierter ID in eine zur Auswahl stehende Gruppe eingeschrieben haben.

getInitialEnrollmentDate("70323786958847") + 2h > now

Gibt TRUE während zwei Stunden ab Einschreibzeitpunkt für diejenigen Kursteilnehmer, die sich über den Kursbaustein *Einschreibung* mit spezifizierter ID in eine zur Auswahl stehende Gruppe eingeschrieben haben. So kann abgebildet werden, dass jeder Kursteilnehmer nur während einer bestimmten Zeitdauer z.B. ein Skript bearbeiten kann.

(getInitialCourseLaunchDate(0) >= never) | (getInitialCourseLaunchDate(0) + 2h > now)

Gibt TRUE, wenn der Kursteilnehmer den Kurs noch nicht besucht hat oder während zwei Stunden seit dem ersten Kursbesuch. So kann abgebildet werden, dass jeder Kursteilnehmer nur während einer bestimmten Zeitdauer den Kurs sehen kann.

(getRecentCourseLaunchDate(0) + 10min < now)

Gibt TRUE, wenn sich der Benutzer seit mehr als 10 Minuten im Kurs bewegt.

(getCourseBeginDate(0) <= today) & (getCourseEndDate(0) >= today)

Gibt den Wert TRUE zurück, wenn das heutige Datum zwischen Beginn- und Enddatum des Durchführungszeitraums des Kurses liegt.

isAssessmentMode(0)

Gibt TRUE sobald der Kurs innerhalb eine Prüfung ist.

hasUserProperty("email", "john.doe@openolat.org")

Gibt TRUE, wenn der Kursteilnehmer in OpenOLAT mit der eingetragenen E-Mail-Adresse registriert ist.

userPropertyEndswith("email", "@openolat.org")

Gibt TRUE, wenn die E-Mail-Adresse des Kursteilnehmers auf @openolat.org endet.

isInUserProperty("email", "doe@openo")

Gibt TRUE, wenn der Begriff *doe@openo* ein Teil der E-Mail-Adresse des Kursteilnehmers ist.

isNotInUserProperty("email", "doe@openo")

Gibt FALSE, wenn der Begriff *doe@openo* ein Teil der E-Mail-Adresse des Kursteilnehmers ist.

isNotInUserProperty("email", "doe@openo")

Gibt FALSE, wenn der Begriff *doe@openo* ein Teil der E-Mail-Adresse des Kursteilnehmers ist.



Bitte beachten Sie, dass die oben erwähnten Kursbaustein-IDs Beispiele sind. Wenn Sie Ihren Kurs erstellen, müssen Sie jeweils die Nummern referenzieren, die auf dem ersten Tab «*Titel und Beschreibung*» des gewünschten Kursbausteins sichtbar sind.

Einsatz von AAI-Attributen

Wenn Sie sich an einer Schweizer Hochschule befinden, oder einer anderen Institution die auf eine AAI - Infrastruktur Zugriff hat, können Sie mit AAI-Attributen im Kurs Zugriffsregeln setzen, damit nur Kursteilnehmer mit bestimmten Benutzerattributen (z.B. Teilnehmer, die einer bestimmten Institution angehören) auf das Kursmaterial zugreifen können. Die Abkürzung AAI steht für „Authentication and Authorization Infrastructure“ und ermöglicht es Angehörigen einer Hochschule, mit nur einem Benutzernamen und Passwort Zugriff auf Systeme aller teilnehmenden Hochschulen zu erhalten. Weitere Informationen zu AAI finden Sie z.B. bei [Switch](#) oder dem [deutschen Forschungsnetz](#).

Verfügbare Attribute und mögliche Werte sind in der AAI Attribute Specification bei [Switch](#) (Dokument in englischer Sprache) und [DFN-AAI](#) beschrieben. Zwei an schweizer Hochschulen häufig gebrauchte Attribute und Beispiele der entsprechenden Expertenregeln finden Sie in der folgenden Tabelle:

Attribut	Beschreibung	Beispiel Expertenregel und Erklärung
swissEduPerson-HomeOrganization	Universität oder Heimorganisation	hasAttribute ("swissEduPersonHomeOrganization", "uzh.ch"): Nur Angehörige der Universität Zürich sind zugelassen.
swissEduStudyBranch3	Studienrichtung 3. Klassifikation	hasAttribute ("swissEduPersonStudyBranch3", "6400"): Nur Studierende der Studienrichtung Veterinärmedizin sind zugelassen.

Anwendung

Sie können AAI-Attribute mit der Syntax

hasAttribute("[AttrName]", "[string] ") oder ***isInAttribute***("[AttrName]", "[substring] ") abfragen.

Dabei gilt folgendes:

[AttrName]	ist der Attributnamen, den Sie in der nachfolgenden Tabelle und auch in der Spezifikation der AAI-Attribute (pdf-Datei) (Spalte <i>LDAP Namen</i>) auf Seite 5 vorfinden.
[string]	ist der Wert des AAI-Attributes mit Namen [AttrName].
[substring]	ist ein beliebig grosser Teil von [string].

AAI Beispiel

Variable Sie können AAI-Attribute mit der Syntax <i>hasAttribute</i> ('[AttrName]', '[string]') oder <i>isInAttribute</i> ('[AttrName]', '[substring]') abfragen.	Beispiel Wert [string]	Beschreibung
swissEduPersonUniqueID	845938727494@uzh.ch	Eindeutige persönliche Identifikationsnummer
surname	Muster	Nachname
givenName	Hans	Vorname
mail	hans.muster@uzh.ch	Bevorzugte E-Mail-Adresse
swissEduPersonHomeOrganization	uzh.ch	Heimorganisation/Universität
swissEduPersonHomeOrganizationType	university	Art der Heimorganisation
eduPersonAffiliation	student	Funktion innerhalb der Heimorganisation
swissEduPersonStudyBranch1	4	Studienrichtung 1. Klassifikation
swissEduPersonStudyBranch2	42 (=Naturwissenschaften)	Studienrichtung 2. Klassifikation
swissEduPersonStudyBranch3	4600 (=Chemie)	Studienrichtung 3. Klassifikation
swissEduPersonStudyLevel	15	Beschreibung des Studien-Fortschrittes
eduPersonEntitlement	http://vam.uzh.ch/surgery	Zugriffsrecht auf Ressource
employeeNumber	01-234-567	Matrikelnummer (nur für Studierende der Universität Zürich)
organizationalUnit	1	Einheit der Heimorganisation z.B. Fakultät (nur für Mitarbeiter)

<code>isInAttribute("surname", "ust")</code>	true
<code>hasAttribute("swissEduPersonStudyBranch3", "4600")</code>	true
<code>hasAttribute("swissEduPersonStudyBranch3", "1200")</code>	false
<code>isInAttribute("eduPersonEntitlement", "http://vam.uzh.ch")</code>	true
<code>isInAttribute("eduPersonEntitlement", "http://vam.uzh.ch/ophthalmology")</code>	false
<code>hasAttribute("employeeNumber", "01-234-567")</code>	true

Einen Link zur Liste der möglichen Attribut-Werte finden Sie im Appendix der Spezifikation der AAI-Attribute (pdf-Datei) ab Seite 20. [Spezifikation der AAI-Attribute \(pdf-Datei\)](#)

Für weitere Informationen zu Werten oder der Anwendung von AAI-Attributen, wenden Sie sich in der Schweiz bitte an [Switch](#), und in Deutschland an das [Deutsche Forschungsnetz](#).



Verwenden Sie die AAI Attribute nur dann, wenn Sie sicher sind, dass alle Teilnehmer Ihres Kurses sich über eine AAI Struktur einwählen. Ansonsten greifen die Parameter nicht!