SOAP to REST

Scott Hurrey

SOAP to REST migration: Mapping

The purpose of this document is to provide a mapping from SOAP Web Service calls to REST API Endpoints. This is meant to be a quick-start guide to the migration process. With SOAP being deprecated in the near term future and removed from Learn following the public API Deprecation Policy, it is our intention to provide you the information you need in order to migrate as painlessly as possible. This is a living document and will up updated as the last few remaining gaps in functionality are closed.

New REST APIs are rolled out in new versions of Learn, but never back-ported to previous releases. To be sure the endpoint you require is available, be sure to visit the Developer Portal and check the individual endpoints you need.

Announcements

Use Case	SOAP WS Call	REST Endpoint
Create Course	AnnouncementWS.createCour	rseAPMSTIncements()
Announcement		/learn/api/public/v1/courses/{courseId}/announcements
Read Course	AnnouncementWS.getCourseA	
Announcement		/learn/api/public/v1/courses/{courseId}/announcements/{announcements/
Read All Course	AnnouncementWS.getCourseA	
Announcements	Ţ.	/learn/api/public/v1/courses/{courseId}/announcements
Update Course	AnnouncementWS.updateCou	
Announcement		/learn/api/public/v1/courses/{courseId}/announcements/{annous
Delete Course	AnnouncementWS.deleteCours	
Announcement		/learn/api/public/v1/courses/{courseId}/announcements/{annou
Create Organization	${\bf Announcement WS. create Org A}$	
Announcement		/learn/api/public/v1/courses/{courseId}/announcements
Read Organization	Announcement WS. get Org Ann	, , - , - , , , , , , , , , , , , , , ,
Announcement		/learn/api/public/v1/courses/{courseId}/announcements/{annou
Read All Organization	Announcement WS. get Org Ann	
Announcements		/learn/api/public/v1/courses/{courseId}/announcements
Update Organization	AnnouncementWS.updateOrg.	gAni r oAfficethents()
Announcement		/learn/api/public/v1/courses/{courseId}/announcements/{annou
Delete Organization	${\bf Announcement WS. delete Org A}$	
Announcement		/learn/api/public/v1/courses/{courseId}/announcements/{annou
Create System	$Announcement WS. create Syst \epsilon$	emAnaSun¢benents(pi/public/v1/announcements
Announcement		
Read System	Announcement WS. get System A	
Announcement		/learn/api/public/v1/announcements/{announcementId}
Read All System	Announcement WS. get System/	AnnGhnce/hearts/api/public/v1/announcements
Announcements		
Update System	Announcement WS. update Syst	
Announcement		$/learn/api/public/v1/announcements/\{announcementId\}$
Delete System	${\bf Announcement WS. delete Syste}$	
Announcement		/learn/api/public/v1/announcements/{announcementId}

Calendar

Use Case	SOAP WS Call	REST Endpoint	
Create a Course	CalendarWS.createCourseCalendarItem()CalendarWSOSFeCourseCalendarItem()		
Calendar Item	V	/learn/api/public/v1/calendars/items	
Read a Course	CalendarWS.getCalendarItem()	GET	
Calendar Item	· ·	/learn/api/public/v1/calendars/items/{calendars/	
		type}/{calendar item id}	
Read all Course	CalendarWS.getCalendarItem()	GET /learn/api/public/v1/calendarsGET	
Calendar Items	· ·	/learn/api/public/v1/calendars/items	
Update a Course	CalendarWS.updateCourseCalendarItem()Calend		
Calendar Item	- '	/learn/api/public/v1/calendars/items/{calendars/	
		type}/{calendar item id}	
Delete a Course	CalendarWS.deleteCourseCalendarItem()	DELETE	
Calendar Item	V	/learn/api/public/v1/calendars/items/{calendars/	
		type}/{calendar item id}	
Create an	CalendarWS.createInstitutionalCalendarItem()Ca		
Institutional	V	/learn/api/public/v1/calendars/items	
Calendar Item		, , , , , , , , ,	
Read an	CalendarWS.getCalendarItem()	GET	
Institutional	()	/learn/api/public/v1/calendars/items/{calendars/	
Calendar Item		type}/{calendar item id}	
Read all	CalendarWS.getCalendarItem()	GET /learn/api/public/v1/calendarsGET	
Institutional	()	/learn/api/public/v1/calendars/items	
Calendar Items		/ · · · / · · · / · · · · · · · · · · ·	
Update an	CalendarWS.updateInstitutionalCalendarItem()C	CalendarItem()	
Institutional	1	/learn/api/public/v1/calendars/items/{calendars/	
Calendar Item		type}/{calendar item id}	
Delete an	CalendarWS.deleteInstitutionalCalendarItem()	DELETE	
Institutional	V	/learn/api/public/v1/calendars/items/{calendars/	
Calendar Item		type}/{calendar item id}	
Create a Personal	CalendarWS.createPersonalCalendarItem()Calendar		
Calendar Item		/learn/api/public/v1/calendars/items	
Read a Personal	CalendarWS.getCalendarItem()	GET	
Calendar Item	0 ()	/learn/api/public/v1/calendars/items/{calendars/	
		type}/{calendar item id}	
Read all Personal	CalendarWS.getCalendarItem()	GET /learn/api/public/v1/calendarsGET	
Calendar Items	0.0000000000000000000000000000000000000	/learn/api/public/v1/calendars/items	
Update a Personal	CalendarWS.updatePersonalCalendarItem()CalendarItem()		
Calendar Item	() cure	/learn/api/public/v1/calendars/items/{calendars	
		type}/{calendar item id}	
Delete a Personal	CalendarWS.deletePersonalCalendarItem()	DELETE	
Calendar Item	()	/learn/api/public/v1/calendars/items/{calendars/	
Carcindar 100III		type}/{calendar item id}	

Content

Use		
Case	SOAP WS Call	REST Endpoint
Create	ContentWS.saveCourseTOC()	POST /learn/api/public/v1/courses/{courseId}/contents
Course		
TOC		

Use Case	SOAP WS Call	REST Endpoint
Read Course TOC	ContentWS.getTOCsByCourseId	$\label{eq:contents} $$ $\mathbb{E}T / \frac{public}{v1/courses/\{courseId\}/contents/\{content\ ID\}}$$$
Read All Course TOC	${\bf Content WS. get TOCsBy Course Id}$	$\label{eq:course_def} $$ $$ $$ $$ /\contents $$ $$ /\contents $$$
Update Course TOC	${\bf ContentWS. save Course TOC()}$	PATCH /learn/api/public/v1/courses/{courseId}/contents
Delete Course ΓΟC	${\bf ContentWS. deleteCourseTOCs()}$	DELETE /learn/api/public/v1/courses/{courseId}/contents
Create Content Item	${\bf ContentWS.saveContent()}$	$POST / learn/api/public/v1/courses/\{courseId\}/contents/\{content ID)/children$
Read Content Item	${\bf ContentWS.getFilteredContent()}$	$GET / learn/api/public/v1/courses/\{courseId\}/contents/\{content\ ID\}$
Read All Content Items	${\bf ContentWS.getFilteredContent()}$	GET /learn/api/public/v1/courses/{courseId}/contents/{content ID)/children
Update Content Item	${\bf ContentWS.saveContent()}$	$PATCH / learn/api/public/v1/courses/\{courseId\}/contents$
Delete Content Item	${\bf ContentWS. deleteContents()}$	$DELETE / learn/api/public/v1/courses/\{courseId\}/contents$
Attach File	ContentWS.addContentFile()	POST /learn/api/public/v1/courses/{courseId}/contents/{content ID}/attachment
Remove File	${\bf ContentWS. deleteContentFiles()}$	/learn/api/public/v1/courses/{courseId}/contents/{contentID}/attachment/{attaID}
Update File	Content WS. update Content File Line Line Line Line Line Line Line Lin	hr in
Read Content File	${\bf ContentWS.getContentFiles()}$	GET /learn/api/public/v1/courses/{courseId}/contents/{contentID}/attachment/{atta ID}
Read All Content Files	${\bf ContentWS.getContentFiles()}$	GÉT /learn/api/public/v1/courses/{courseId}/contents/{content ID}/attachment
Download Content File	${\bf ContentWS.getContentFiles()}$	GET /learn/api/public/v1/courses/{courseId}/contents/{contentID}/attachment/{atta ID}/downloads
Create Link	ContentWS.saveLink()	$POST / learn/api/public/v1/courses/\{courseId\}/contents/\{content ID\}/children$
Read Link Read All	, , , , , , , , , , , , , , , , , , ,	CGEFFe(Melont/entiWishglitt/inlk/stbynRede/ferflyspel()}/contents/{content ID} CGEFFe(Melont/entiWishglitt/inlk/stbynRede/ferflyspel()}/contents
Links Update Link	ContentWS.saveLink()	PATCH /learn/api/public/v1/courses/{courseId}/contents/{content ID}

Use		
Case	SOAP WS Call	REST Endpoint
Delete	ContentWS.deleteLink()	DELETE /learn/api/public/v1/courses/{courseId}/contents/{content
Link		ID }
Get Re-	${\bf ContentWS.getReviewStatusBy}$	yC6hEsEId()
viewed		$/learn/api/public/v1/courses/\{courseId\}/contents/\{contentId\}/users/\{userId\}/revieuses/\{contentId\}/users/\{userId\}/revieuses/\{u$
Status		
Set Re-	${\bf Content WS. set Content Reviewe}$	rd(PATCH
viewed		$/ learn/api/public/v1/courses/\{courseId\}/contents/\{contentId\}/users/\{userId\}/reviews/\{contentId\}/users/\{userId\}/reviews/\{us$
Status		

${\bf Context}$

Use Case	SOAP WS Call	REST Endpoint		
Register Tool	ContextWS.registerTool()	N/A		
Emulate a User	ContextWS.emulateUser()	GET /learn/api/public/v1/oauth2/authorizationcodePOST /learn/api/public/v1/oauth2/token		
Extend Session Life	ContextWS.extendSession	LHQST /learn/api/public/v1/oauth2/token		
Get Memberships By	ContextWS.getMembershi	ip&ET /learn/api/public/v1/users/{user id}/courses		
UserID				
Get Current User's	ContextWS.getMyMember	ContextWS.getMyMembers6ips()/learn/api/public/v1/users/me/courses		
Memberships				
Get System	ContextWS.getSystemInstallastidulTd()			
Installation ID				
Login as User	$\operatorname{ContextWS.login}()$	GET /learn/api/public/v1/oauth2/authorizationcodePOST /learn/api/public/v1/oauth2/token		
Login with Access	ContextWS.loginTicket()	N/A		
Ticket	- 0			
Login as Tool	ContextWS.loginTool()	POST /learn/api/public/v1/oauth2/token		
Logout	ContextWS.logout()	N/A		

Courses

	COAD WC C II	DECEMBED 1 : 4
Use Case	SOAP WS Call	REST Endpoint
Create Course	CourseWS.createCourse()	CBOSEN/Sesave Capit sa (blic/v2/courses
Read Course	CourseWS.getCourse()	GET /learn/api/public/v2/courses/{courseId}
Read All Courses	CourseWS.getCourse()	GET /learn/api/public/v2/courses
Update Course	CourseWS.saveCourse()	PATCH /learn/api/public/v2/courses/{courseId}
Update Course	CourseWS.changeCourseB	a PATCOH()/learn/api/public/v2/courses/{courseId}
BatchUID		
Update Course	CourseWS.changeCourseC	CaPeAgDGyBBatchUid()
Category		/learn/api/public/v1/catalog/categories/{categoryType}/{categoryId}
Update Course	CourseWS.changeCourseE	OaPaSDGHeIdearn/api/public/v2/courses/{courseId}
Datasource		
Delete Course	CourseWS.deleteCourse()	DELETE /learn/api/public/v2/courses/{courseId}
Create	CourseWS.createOrg()	POST /learn/api/public/v2/courses
Organization		
Read Organization	CourseWS.getOrg()	GET /learn/api/public/v2/courses/{courseId}
Read All	CourseWS.getOrg()	GET /learn/api/public/v2/courses
Organizations		
Update	CourseWS.saveCourse()	PATCH /learn/api/public/v2/courses/{courseId}
Organization		
Create Organization Read Organization Read All Organizations Update	CourseWS.getOrg() CourseWS.getOrg() CourseWS.getOrg()	POST /learn/api/public/v2/courses GET /learn/api/public/v2/courses/{courseId} GET /learn/api/public/v2/courses

Use Case	SOAP WS Call	REST Endpoint
Update	CourseWS.changeOrgBatchPAd(CH /learn/api/public/v2/courses/{courseId}	
Organization		
BatchUID		
Update	CourseWS.changeOrgCate	
Organization		$/learn/api/public/v1/catalog/categories/\{categoryType\}/\{categoryId\}$
Category		
Update	CourseWS.changeOrgDat	$\textbf{aSPATCEId}() \text{learn/api/public/v2/courses/\{courseId\}}$
Organization		
Datasource	CWC 1-1-+-O()	DELETE /1/
Delete Organization	CourseWS.deleteOrg()	DELETE /learn/api/public/v2/courses/{courseId}
Create Group	CourseWS.saveGroup()	POST /learn/api/public/v1/courses/{courseId}/groups
Read Group	CourseWS.getGroup()	GET /learn/api/public/v1/courses/{courseId}/groups/{group Id}
Read All Groups	CourseWS.getGroup()	GET /learn/api/public/v1/courses/{courseId}/groups GET /learn/api/public/v1/courses/{courseId}/groups
Update Group	CourseWS.saveGroup()	PATCH /learn/api/public/v1/courses/{courseId}/groups/{group
opdate Group	Course w b.saveGroup()	Id}
Delete Group	CourseWS.deleteGroup()	DELETE /learn/api/public/v1/courses/{courseId}/groups/{group
Belete Group	Course W.S. defected Toup()	Id}
Create Term	CourseWS.saveTerm()	POST /learn/api/public/v1/terms
Read Term	· · · · · · · · · · · · · · · · · · ·	urseMIS./deadTe/apB/pCiblirs/stdl/terms/{term id}
Read All Terms		ou CsENVS/learhT/apnisBytNim/ve1 /terms
Update Term	CourseWS.saveTerm()	PATCH /learn/api/public/v1/terms/{term id}
Delete Term	CourseWS.deleteTerm()	DELETE /learn/api/public/v1/terms/{term id}
Add Course to	CourseWS.addCourseToT	Ten MTCH /learn/api/public/v2/courses/{courseId}
Term		
Get Courses in	CourseWS.loadCoursesIn'	Tento / learn/api/public/v2/courses?termId={term id}
Term		
Remove Course	CourseWS.removeCoursel	$\operatorname{Fr}\mathbf{DATGH}() \operatorname{learn/api/public/v2/courses/\{courseId\}}$
From Term		
Create Course	CourseWS.saveCourseCat	terest / learn/api/public/v1/catalog/categories/{categoryType}
Category) CDT
Read Course	CourseWS.getCategories(
Category		/learn/api/public/v1/catalog/categories/{categoryType}/{categoryId}
Read All Course	Course w S. get Categories () GET /learn/api/public/v1/catalog/categories/{categoryType}
Categories Undete Course	Course WC some Course Cot	DATI/NII
Update Course	CourseWS.saveCourseCat	/egomy(у п /learn/api/public/v1/catalog/categories/{categoryType}/{categoryId}
Category Delete Course	CourseWS.deleteCourseC	
Category	Course w.S. defete Course C	/learn/api/public/v1/catalog/categories/{categoryType}/{categoryId}
Create Course	CourseWS.saveCourseCat	
Category	Course W.S.save Course Cat	/learn/api/public/v1/catalog/categories/{categoryType}/{categoryId}/courses/
Membership		/ carry apr/ patric/ v1/ carriags/ carregories/ (carregory 1) pej/ (carregory ary courses/
Read Course	CourseWS.getCourseCate	g 6#V Membership()
Category	0.0000000000000000000000000000000000000	/learn/api/public/v1/catalog/categories/{categoryType}/{categoryId}/courses/
Membership		/learn/api/public/v1/courses/{courseId}/categories
Read All Course	CourseWS.getCourseCate	
Category		/learn/api/public/v1/catalog/categories/{categoryType}/{categoryId}/courses
Memberships		
Update Course	CourseWS.saveCourseCat	
Category		$/learn/api/public/v1/catalog/categories/\{categoryType\}/\{categoryId\}/courses/categoryType\}/$
Membership		
Delete Course	CourseWS.deleteCourseC	
Category		$/learn/api/public/v1/catalog/categories/\{categoryType\}/\{categoryId\}/courses/categoryType\}/$
Membershin		

Membership

Use Case	SOAP WS Call	REST Endpoint
Create	CourseWS.saveOrgCatego	rFOST /learn/api/public/v1/catalog/categories/{categoryType}
Organization		
Category		
Read Organization	CourseWS.getCategories()	
Category		/learn/api/public/v1/catalog/categories/{categoryType}/{categoryId}
Read All	CourseWS.getCategories()	GET /learn/api/public/v1/catalog/categories/{categoryType}
Organization		
Categories		
Update	CourseWS.saveOrgCatego	r y (ATCH
Organization		/learn/api/public/v1/catalog/categories/{categoryType}//{categoryId}
Category		
Delete	CourseWS.deleteOrgCates	g cDALETE
Organization	0	/learn/api/public/v1/catalog/categories/{categoryType}/{categoryId}
Category		
Create	CourseWS.saveOrgCatego	rtNiembership()
Organization	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	/learn/api/public/v1/catalog/categories/{categoryType}/{categoryId}/c
Category		/ / / / / / / / -
Membership		
Read Organization	CourseWS.getOrgCategor	vMFMhershin()
Category	Course W.S. get Org Categor	/learn/api/public/v1/catalog/categories/{categoryType}/{categoryId}/c
Membership		/learn/api/public/v1/cautalog/categories/{category1ypc}/{category1ypc}//category1
Read All	CourseWS.getOrgCategor	
	Course w.s.gerOrgCategor	/learn/api/public/v1/catalog/categories/{categoryType}/{categoryId}/c
Organization		/learn/api/public/v1/catalog/categories/{category1ype}/{category1d}/c
Category		
Memberships	CWCC	
Update	CourseWS.saveOrgCatego	
Organization		/learn/api/public/v1/catalog/categories/{categoryType}/{categoryId}/c
Category		
Membership		DIMERTIN 14 ()
Delete	CourseWS.deleteOrgCateg	
Organization		/learn/api/public/v1/catalog/categories/{categoryType}//{categoryId}/c
Category		
Membership		
Create Cartridge	CourseWS.saveCartridge(
Read Cartridge	CourseWS.getCartridge()	
Jpdate Cartridge	CourseWS.saveCartridge(· · ·
Delete Cartridge	CourseWS.deleteCartridge	V I
Create StaffInfo	CourseWS.saveStaffInfo()	POST /learn/api/public/v1/usersPUT
		$/learn/api/public/v1/courses/\{course\ id\}/users/\{user\ id\}$
Read StaffInfo	CourseWS.getStaffInfo()	
Update StaffInfo	CourseWS.saveStaffInfo()	PATCH /learn/api/public/v1/users/{user id}
Delete StaffInfo	CourseWS.deleteStaffInfo	()DELETE /learn/api/public/v1/users/{user id}
Set Course Banner	CourseWS.setCourseBann	
Get Group Tools	CourseWS.getAvailableGr	
Read	CourseWS.getClassificatio	• () •
Classification	<u> </u>	w

Course Memberships

Use Case	SOAP WS Call	REST Endpoint
Create Course Membership	PUT	
CourseMember-	/learn/api/public/v1/cou	$arses/{courseId}/users/{user}$
shipWS.saveCourseMembership()	$\mathrm{id}\}$	
Read Course Membership	CourseMembershipWS.ge	etCourseMehrHership()
		/learn/api/public/v1/courses/{courseId}/users/{user id}
Read All Course Memberships	${\bf Course Membership WS.ge}$	
		$/learn/api/public/v1/courses/\{courseId\}/users$
Update Course Memberships	CourseMembershipWS.sa	- 0
	~	/learn/api/public/v1/courses/{courseId}/users/{userId}
Delete Course Memberships	CourseMembershipWS.de	- */
	C 1 1 1 1770	/learn/api/public/v1/courses/{courseId}/users/{userId}
Create Organization Membership	CourseMembershipWS.sa	- 0
		/learn/api/public/v1/courses/{courseId}/users/{user
D. 10i. M. M	CMhhWC	id}
Read Organization Membership	CourseMembershipWS.ge	
		/learn/api/public/v1/courses/ $\{courseId\}/users/\{userid\}$
Read All Organization	CourseMembershipWS.ge	,
Memberships	Compensation 1. S. Sc	/learn/api/public/v1/courses/{courseId}/users
Update Organization Memberships	CourseMembershipWS.sa	
opame organization internal	Oddison.2011.0012.11.p	/learn/api/public/v1/courses/{courseId}/users/{userId
Delete Organization Memberships	CourseMembershipWS.de	
2 07000 C - 00000000000000000000000000000	O 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	/learn/api/public/v1/courses/{courseId}/users/{userId
Create Group Membership	CourseMembershipWS.sa	
•	-	/learn/api/public/v1/courses/{courseId}/groups/{group
Read Group Membership	CourseMembershipWS.ge	, , , , , , , , , , , , , , , , , , , ,
-	•	/learn/api/public/v1/courses/{courseId}/groups/{group
Read All Group Memberships	CourseMembershipWS.ge	
-	_	/learn/api/public/v1/courses/{courseId}/groups/{group
Delete Group Memberships	CourseMembershipWS.de	
		/learn/api/public/v1/courses/{courseId}/groups/{group
Get Course Roles	CourseMembershipWS.ge	etCourseR6451) /learn/api/public/v1/courseRoles

$\mathbf{Gradebook}$

Use Case	SOAP WS Call	REST Endpoint
Create an	GradebookWS.saveAt	ttem pts (\$T
Attempt		/learn/api/public/v2/course/{courseId}/gradebook/columns/{columnid}/attemp
Read an	GradebookWS.getAtte	emp G ET
Attempt		/learn/api/public/v2/course/{courseId}/gradebook/columns/{columnid}/attemptid}
Read All	${\it GradebookWS.getAtte}$	emp GÉ T
Attempts		/learn/api/public/v2/course/{courseId}/gradebook/columns/{columnid}/attemp
Update an	GradebookWS.saveAt	
Attempt		/learn/api/public/v2/course/{courseId}/gradebook/columns/{columnid}/attempid}
Delete an	${\it GradebookWS.delete}$	Atternate (CH
Attempt		/learn/api/public/v2/course/{courseId}/gradebook/columns/{columnid}/attempid} (set to 0)
Create a Column	GradebookWS.saveCc	$olum \red{EQST / learn/api/public/v2/course/{courseId}/gradebook/columns}$

Use Case	SOAP WS Call	REST Endpoint
Read a Column	GradebookWS.getGra	$\label{eq:course} $$ \frac{\text{cadeb} \& \text{Columns}()}{\operatorname{learn/api/public/v2/course/\{courseId\}/gradebook/columns/\{columnid\}} $$ id $$$
Read All Columns	${\bf Gradebook WS.get G$	radeb 66EC o/theans/api/public/v2/course/{courseId}/gradebook/columns
Update a Column	GradebookWS.saveCo	$\label{eq:columns} $
Delete a Column	GradebookWS.delete	,
Create a Gradebook Type	GradebookWS.saveG	radelROSTyples()n/api/public/v2/course/{courseId}/gradebook/columns
Read a Gradebook Type	GradebookWS.getGra	$\label{eq:cadebook} $$ \ensuremath{\operatorname{radebook}}(p) = \frac{\operatorname{courseId}}{\operatorname{gradebook}/\operatorname{columns}} = \frac{\operatorname{column}}{\operatorname{id}} $$$
Read All Gradebook Types	GradebookWS.getGra	radeb GET yphes(n)n/api/public/v2/course/{courseId}/gradebook/columns
Update a Gradebook Type	GradebookWS.saveG	$\label{eq:course} $$ \operatorname{ATCHpes}() $$ /\operatorname{learn/api/public/v2/course/{courseId}/gradebook/columns/{columnid}} $$ id $$$
Delete a Gradebook Type	GradebookWS.deletee	
Create a Grade	GradebookWS.saveGr	
Read a Grade	GradebookWS.getGra	$rades (GET / learn/api/public/v2/course/{courseId}/gradebook/columns/{columnid}/attempts/{id}$
Read All Grades	GradebookWS.getGra	$rades (GET / learn/api/public/v2/course/\{courseId\}/gradebook/columns/\{columnid\}/attempts$
Update a Grade	GradebookWS.saveG	$\label{eq:course} $$ \frac{\Pr(ATCH)}{\operatorname{course}}/\operatorname{course}}/\operatorname{course}/\operatorname{columns}/\operatorname{columnid}/\operatorname{attempts}/\operatorname{id}} $$ id $$$
Delete a Grade	GradebookWS.delete	eGradEs(TCH /learn/api/public/v2/course/{courseId}/gradebook/columns/{columnid}/attempts/{id} (set to 0)
Create a Grading Schema	GradebookWS.saveG	radin g Stan /de()rn/api/public/v1/courses/{course id]/gradebook/schemas
Read a Grading Schema	_	rading SÆÆÆn/as() rn/api/public/v1/courses/{course id]/gradebook/schemas/{schema id}
Read All Grading Schemas	GradebookWS.getGra	rading SEE mas (rading See mas from the first of the firs
Update a Grading Schema	GradebookWS.saveGr	radin g Ache H astearn/api/public/v1/courses/{course id]/gradebook/schemas/{schema id}
Delete a Grading Schema	GradebookWS.delete	eGrad DeS ÆEEas(earn/api/public/v1/courses/{course id]/gradebook/schemas/{schema id}

Users

Use Case	SOAP WS Call	REST Endpoint
Create a User	UserWS.saveUser()	POST /learn/api/public/v1/users
Read a User	UserWS.getUser()	GET /learn/api/public/v1/users/{userId}
Read All Users	UserWS.getUser()	GET /learn/api/public/v1/users
Update a User	UserWS.saveUser()	PATCH /learn/api/public/v1/users/{userId}
Delete a User	UserWS.deleteUser()	DELETE /learn/api/public/v1/users/{userId}
Update a User's BatchUid	UserWS.changeUserBatchWtdCH /learn/api/public/v1/users/{userId}	
Update a User's	UserWS.changeUserDatalSAThCeHd(learn/api/public/v1/users/{userId}	
DataSource ID		
Delete User by Role	UserWS.deleteUserByInsNitAtionRole()	
Create a User's Address	UserWS.saveAddressbooRCRTy(Jearn/api/public/v1/users	
Book Entry		
Read a User's Address	UserWS.getAddressboo	ok Ch Ty()earn/api/public/v1/users/{userId}
Book Entry		
Read All a User's Address	UserWS.getAddressboo	ok E ntry(Jearn/api/public/v1/users
Book Entries		
Update a User's Address	UserWS.saveAddressBl	bo RAEAE (ly/vlearn/api/public/v1/users/{userId}
Book Entry		
Delete a User's Address	UserWS.deleteAddress	BobEEntff()/learn/api/public/v1/users/{userId}
Book Entry		
Read All Institution Roles	UserWS.getInstitutionR 6 (45(1) /learn/api/public/v1/institutionRoles	
Read All System Roles	$UserWS.getSystemRoles\\QET/learn/api/public/v1/systemRoles$	
Read All Institution Roles	$UserWS.getUserInstituti\\ \textbf{GHRbl} \\ \textbf{(see)} \\ rn/api/public/v1/users/\{userId\}$	
Assigned to a User		
Create Observer	UserWS.saveObserverAssOdfation()	
Association		/learn/api/public/v1/users/{userId}/observers/{observerId}
Read Observer Association	UserWS.getObservee()	1 1 1 1 1 1 1 1
		$/learn/api/public/v1/users/\{userId\}/observers$
Read All Observer	UserWS.getObservee()	
Association		$/learn/api/public/v1/users/\{userId\}/observers$
Update Observer	UserWS.saveObserverA	V
Association		/learn/api/public/v1/users/{userId}/observers/{observerId}
Delete Observer	UserWS.deleteOverservePASkdZibEion()	
Association		/learn/api/public/v1/users/{userId}/observers/{observerId}

Util

Use Case	SOAP WS Call	REST Endpoint
Read All Datasources	UtilWS.getDataSources()	GET
		/learn/api/public/v1/dataSources
Get Required Entitlements By	UtilWS.getRequiredEntitlements()	N/A
Method	,	,
Check a User's Entitlements	UtilWS.checkEntitlement()	N/A
Create a Setting	UtilWS.saveSetting()	N/A
Load a Setting	UtilWS.loadSetting()	N/A
Update a Setting	UtilWS.saveSetting()	N/A
Delete a Setting	UtilWS.deleteSetting()	N/A
~	0()	•