Graduation Guidelines: Tech Details
Project description

About Graduation Requirements

The grad requirements project was collaborated on with the IC support team, ADS, Curriculum & Instruction, and Learning Services. The project involves configuring assessments and programs in IC to contain the data which IC requires in order to provide tracking and management for high school students regarding their graduation progress.

Students must meet several requirements before they can graduate, this tracking process used to be done manually and was very laborious and time-consuming for the counselors at the high schools. Rather than tracking student's progress using paperwork and spreadsheets, IC can use grade/transcript data to assess whether students are meeting the correct number and type of credits, and can also use assessment data to track the status of non-course/credit requirements. Students must earn enough credits AND pass at least 1 ELA and Math assessment in order to qualify.

Not all assessments are actual assessments, in some cases the assessments in IC are simply used to track the students' results, such as Capstone Projects or Industry Certifications.

There are 4 areas of responsibility:
- IC graduation program management
- Assessment data manual entry
- Assessment data scripting/imports
- Reporting

IC Graduation Program Management

The IC Support team will work with C&I and Learning Services to configure IC in order to accurately track student graduation progress. This involves configuring courses/credits, and creation and management of the Graduation Requirements program.

Assessment Data Manual Entry

C&I and Learning Services are responsible for entering any assessment data which must be manually entered in IC. They must also provide files/imports for the ADS team for data which will not be entered manually.

Assessment Data Scripting/Imports

The ADS team is responsible for importing assessment data into IC, which includes data which is specifically required for tracking student's graduation requirements.

Data which already exists in IC in a different format must also be scripted to facilitate the use of the IC Assessments module for use with the Graduation Requirements program. This includes duplicating assessment data in a Grad Requirements specific format, and scripting non-assessment data into a format which can be used.

Reporting

There are built in reports in IC which can be used by counselors and other support staff to track graduation requirements, and the ADS team is also responsible for Tableau reporting which allows aggregated analysis by school, grade, etc. IC Grad Requirements reports currently only show a student-by-student record.

Tracking Proficiency Requirements

We will use the **Assessment** tab in Infinite Campus to capture whether students have met ELA and math proficiency. Each option should include a subsequent graduation proficiency of "Y" for *meets* or an "N" for *does not meet*.

Proficiency Options (students must satisfy at least one of the following in ELA **AND** one of the following in Math)	Cut score (minimum score for a "Y" *meets*)	Origin of Data (if new)	Coordinating Department	Test Code in IC
ACT – ELA • Reading • Writing	18	--	Assessment	GRACTELA
ACT - Math	19			GTACTMATH
ACT WorkKeys – ELA • Reading for Information	Bronze	Data import	Assessment	GRACTWKELA
ACT WorkKeys – Math				GRACTWKMATH
Advanced Placement (AP) – ELA • AP English Language & Composition • AP English Literature & Composition	2	--	Assessment	APELA
Advanced Placement (AP) – Math • AP Calculus AB • AP Calculus BC • AP Statistics				APMATH
ASVAB – ELA • Verbal Ability Composite	31	Data import	Assessment	ASVABELA
ASVAB – Math • Math Ability Composite				ASVABMATH
International Baccalaureate (IB) – ELA •	4	Data import	Assessment	IBELA
International Baccalaureate (IB) – Math •				IBMATH
SAT – ELA • Evidence-based Reading & Writing • Writing	430	--	Assessment	SATELA
SAT – Math • Mathematics	460			SATMATH
Concurrent Enrollment (CE) – ELA • Any PPCC or UCCS course with English credit	C- in approved CE class	Script that finds English and math CE classes posted to student transcript.	College & Career Services	CEELA
Concurrent Enrollment (CE) – Math • Any PPCC or UCCS course with Math credit				CEMATH
Industry Certification - ELA	n/a (binary)	Manual entry	College & Career Services	ICELA
Industry Certification - Math				ICMATH
Capstone - ELA	P in Capstone class	Manual entry	Curriculum &	CAPELA

The Infinite Campus Assessment module was used to set up tests for each of the different ELA and math proficiency options using the test codes from the chart above.

- Our proficiency options set up as tests in IC
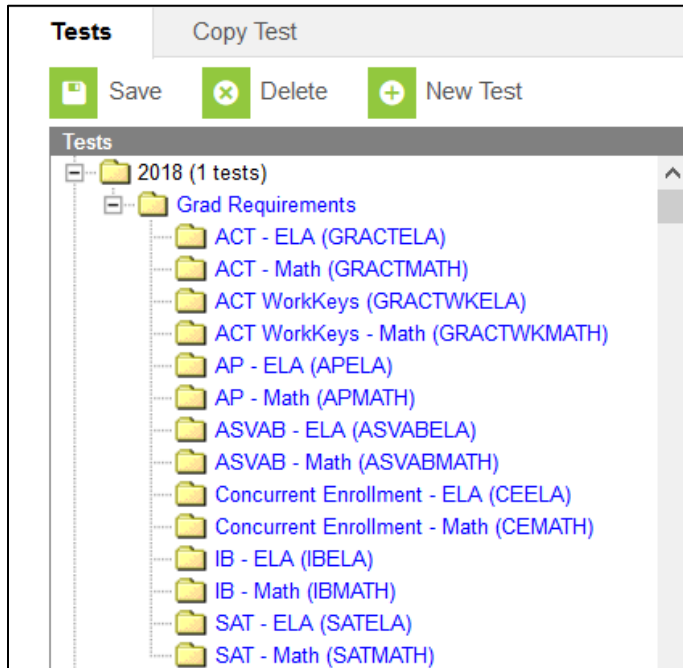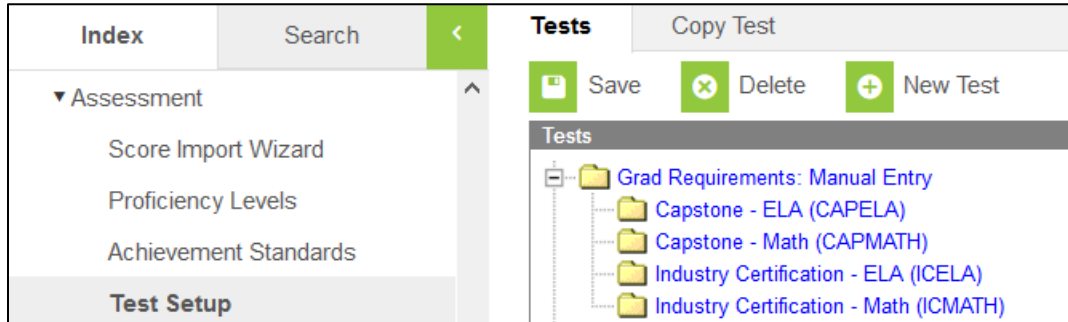
- Creating the tests in IC
  - Path: IC Index → Assessment → Test Setup → New Test
  - First create the parent test to create a folder

Tests	Copy Test

💾 Save   ⊗ Delete   ⊕ New Test

Test Detail

*Name

Grad Requirements

Parent Test

(no parent)

** Parent Test cannot be edited because scores exist for this test.

Code   Year

[   ]   17-18 ∨

Subject

[∨]

Test Grade Level

[∨]

Assessment Type

[∨]

Standards Type

[∨]

Start Date   End Date

[📅] [📅]

Test Type

- ⦿ District Test
- ○ State Test
  - State Code
  - [∨]
- ○ National Test
  - National Code
  - [∨]

☐ Teacher-scored

☐ Display in Portal

☐ Display in Transcripts

Select the score fields you want to use:

☐ Date   ☐ Scale Score   ☐ Raw Score   ☐ Reason Code   ☐ Received Accommodation

☐ Percentile   ☐ Percent   ☐ Normal Curve Equivalent   ☐ Result   ☐ Invalid

Academic Achievement Standard

[∨]

Select the custom fields you want to use:

☐ Growth Percentile   ☐ Grade

Question Label

[   ]

Result Statuses

Value	Result Code	Label	Passing Score
✗ [ ]	[ ]	[ ]	☐

Add Result Status

o   Then set up the individual tests for each proficiency option and link them to the parent test.

Tests   Copy Test

💾 Save   ⊗ Delete   ⊕ New Test

Test Detail

*Name
`ACT - ELA`

Parent Test
`-:Grad Requirements`
** Parent Test cannot be edited because scores exist for this test.

Code   Year
`GRACTELA`   `17-18 ▼`

Subject
`[                    ▼]`

Test Grade Level
`[                 ▼]`

Assessment Type
`[             ▼]`

Standards Type
`[       ▼]`

Start Date   End Date
`[        📅]` `[        📅]`

Test Type

⦿ District Test   ☐ Teacher-scored

○ State Test   ☐ Display in Portal

State Code   ☐ Display in Transcripts
`[                 ▼]`

○ National Test

National Code
`[        ▼]`

Select the score fields you want to use:

☐ Date   ☐ Scale Score   ☐ Raw Score   ☐ Reason Code   ☐ Received Accommodation

☐ Percentile ☐ Percent   ☐ Normal Curve Equivalent ☑ Result   ☐ Invalid

Academic Achievement Standard
`[ ▼]`

Select the custom fields you want to use:

☐ Growth Percentile ☐ Grade

Question Label
`[                              ]`

Result Statuses

	Value	Result Code	Label	Passing Score
✕	`1`	`YES`	`YES`	☑
✕	`0`	`NO`	`NO`	☐

Add Result Status

‹   ›

Tableau visualizations will show proficiency data, most importantly to include a list of high school juniors who have not demonstrated proficiency in at least one ELA and/or Math option, similar to the screen below.

Student Name	Student Number	Math	English Language Arts	School Grade Level
SUSAN				High
YASMINA			—	
STEPHEN		●	●	School
TORREY			—	DISCOVERY CANYON CAMPU...
ATHAN			—	
KACEY			—	Grade
LAYNIE			—	Gr 11
IADISON		●		
BRYCE			—	
HONEY			—	

Academic Plans will be set up in Infinite Campus; these plans can be assigned to students so that school staff (primarily counselors) can run graduation credit and ELA and math proficiency checks without having to pull up Tableau. To create these plans in IC we went to the IC Index → Program Admin → Programs → Graduation Programs tab.

- **ASD20 High School 50 Credits**: Display of students' progress toward required graduation course credits.
  Note: Specialized programs may require a customized Academic Plan to track students' progress toward meeting high school credit requirements for graduation.
  - Specialized plans: Aspen Valley Grad Requirements—Uses the ASD20 High School 50 Credits and **ADDS** College Course, CPR Certification, Job Requirement, Research Paper, Volunteer Hours, and Voter Registration. These additions are manual Yes/No selections on the Assessment tab in IC.
  - Setup in Infinite Campus
    - Creating the Graduation Program (Path: IC Index → Program Admin → Programs → Graduation Programs tab)
      - At a minimum you must give the graduation program a name and attach it to one of your credit groups.

- Adding Credit Requirements
  - The grid will populate with all of the credit types from the credit group that you attached to this graduation program. You can space out the credit requirements over the different grade levels or you can put them all under a single grade level like 12th grade. The goal is to make sure the totals column has the number of credits required for each credit type for graduation.

Credit Requirements Detail - ASD20 High School 50 Credits						
Credit Type	09	10	11	12	Total	Display Seq
Algebra	0	0	0	0	0	
Civics	1	0	0	0	1	7
Core Elective	0	2	2	2	6	11
English	2	2	2	2	8	1
Fine Arts	0	0	1	0	1	3
Geometry	0	0	0	0	0	
Health	1	0	0	0	1	2
Mathematics	2	2	2	0	6	9
Physical Education	1	1	1	0	3	4
Science	2	2	2	0	6	5
Social Science	1	2	0	0	3	6
Unrestricted Elective	2	3	2	4	11	12
US History	0	0	2	0	2	8
World Language	2	0	0	0	2	10
Total	14	14	14	8	50	

- Adding Course Requirements
  - If your district has any specific courses that students are required to take before they can graduate you will enter those course numbers here.
  - Our district requires that a student take two credits of Algebra and two of Geometry for graduation and those courses can be taken in middle school or high school. We set up Algebra and Geometry as course requirements because then the graduation program recognizes that the student has completed the required course whether they took it in middle or high school. If we had set up Algebra and Geometry as credit requirements the graduation program would only recognize credits that were taken in high school, not middle school.

Course Requirements Detail - ASD20 High School 50 Credits								
*Requirement Description	*Transcript/Course Number(s)	Grade	*Min Credits	Min GPA Value	Credit Type this rule applies to	Recommended	Priority
✗ Algebra Course	119998A,119998B,111130,111100A,111110 Preview 0B,111102A,111102B,111065A,111065B, 110066A,110066B,111170A,111170B,111 198A,111198B,111199A,111199B,111170, 111106A,111106B	⌄	2		⌄	☐	☐
✗ Geometry Course	119999A,119999B,116030,116010A,1160 Preview 10B,116020A,116020B,116025A,116025 B,116000A,116000B	⌄	2		⌄	☐	☐

Add CourseRequirement

- We have no Test or GPA requirements for our ASD20 High School 50 Credits graduation program

- **ELA and Math Proficiencies**: Display of students' progress toward ELA and math proficiency requirements for graduation per the menu of options. Specifically, this display shows a list of students who have not yet met ELA and/or math requirements.
  - Unique test codes for each ELA and math proficiency option will be used for this functionality.
  - Our ELA and math proficiencies graduation program only has test requirements in it.
  - Setup in Infinite Campus
    - Adding Test Requirements
      - We created two different test requirements, one for ELA and one for math. Each requirement has the unique test codes we created to account for all of the different ways a student can pass these requirements.
      - Since a student only needs to pass one of the test codes to meet the requirement, we've set the Evaluation Mode to "Count of passing tests = 1".

Test Requirements Detail - ELA and Math Proficiencies

| *Requirement Description	Test Code(s)	State Code(s)	Subject		*Evaluation Mode	Min Raw	Min Scale	Count
✕ ELA	APELA,ASVABELA,CAPELA,CEELA,GRACTELA,GRACTWKELA,IBELA,ICELA,SATELA		⌄	Preview	Count of passing tests ⌄			1
✕ MATH	APMATH,ASVABMATH,CAPMATH,CEMATH,GRACTMATH,GRACTWKMATH,IBMATH,ICMATH,SATMATH		⌄	Preview	Count of passing tests ⌄			1

Add TestRequirement

Credit Overflow values were set up in IC so that if a student has more than they need of one credit type, those extra credits overflow into another credit type. For example if a student has 8 Science credits but only 6 credits of Science are required, we overflow those 2 extra credits into the Core Elective credit type.

- Setting up the credit overflow values
    - Path: IC Index → Grading & Standards → Credit Groups
    - Select one of your credit groups and all of your credit types are listed. Clicking on each credit type brings up the Credit Group Detail record. Enter the type of credit that this credit type should overflow into under the Credit Overflow Rollup field.

SQL Scripting – Assessment Results

A nightly job runs during the school year which finds existing assessment results from IC, and processes and organizes them into the format needed for monitoring Graduation Requirements.
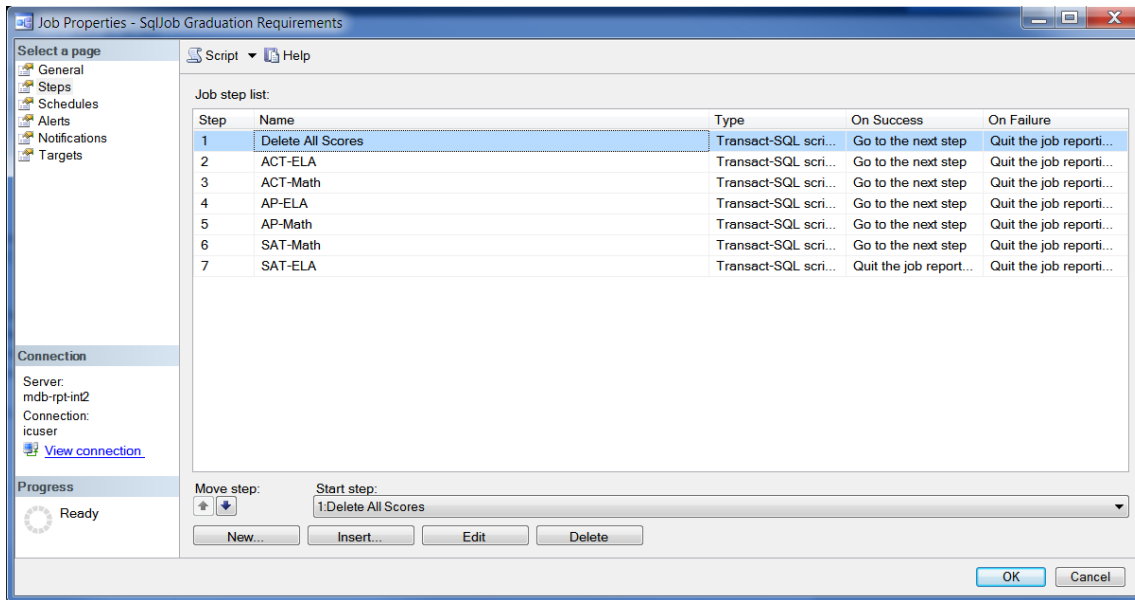
First, all 'Graduation Requirements' scores are deleted from the *TestScore* table for the current year only. This is done by searching for and deleting child and then parent scores which follow the naming scheme 'Grad Requirements'.

Next, each assessment we wish to use in tracking various proficiencies are consolidated and inserted into the IC *TestScore* table in a separate step. This is done by creating a temporary table full of ALL high school students, and then searching for any applicable test scores. If no scores exist for a student, they get a default 'NO' as the result rather than a null record. This is done to explicitly mark a student's proficiency status for this particular assessment, and guarantee that the student appears in our Grad Requirements report rather than being totally missing.

Example of our ACT-ELA proficiency SQL script below:

```
-- Populate the 'Grad Requirements' ACT-ELA assessment.

-- Author: Chris Eggleston

-- Description: This script populates the ACT-ELA assessment for the purpose of reporting which high school students are meeting or not meeting

-- graduation requirements. The first portion gathers ALL current 9th-12th graders and finds any relevant test scores (ACT - English and CO ACT - English).

-- If the student does not have a relevant test score, they still get a test record created which contains a 'NO' result, this is necessary

-- because students who do not have an existing test record are not pulled in the grad requirements report.


IF OBJECT_ID('tempDB..#allhs') IS NOT NULL DROP TABLE #allhs

select distinct stu.personID
```

```sql
into #allhs

from ICSQLDB.Academy20.dbo.Student stu

where stu.grade between '09' and '12'

and stu.endDate is null

and stu.endYear = (select endYear from ICSQLDB.Academy20.dbo.SchoolYear where active = 1)


declare @cutpoint int = 18


IF OBJECT_ID('tempDB..#scores') IS NOT NULL DROP TABLE #scores

select distinct ts.personID, case when ts.scaleScore >= @cutpoint then 'YES' else 'NO' end result

into #scores

from ICSQLDB.Academy20.dbo.TestScore ts

left join ICSQLDB.Academy20.dbo.Test t

    on t.testID = ts.testID

where t.name in ('ACT - English','Colorado ACT - English') -- scores

and ts.scaleScore >= @cutpoint


IF OBJECT_ID('tempDB..#actELA') IS NOT NULL DROP TABLE #actELA

select distinct stu.personID, isnull(s.result, 'NO') result

into #actELA

from #allhs stu

left join #scores s

    on stu.personID = s.personID


declare @parent int = (select top 1 p.testID

        from ICSQLDB.Academy20.dbo.Test p    -- Subquery to get current year's 'Grad Requirements' parentID.

        where p.name = 'Grad Requirements'

        and p.endYear = (select endYear from ICSQLDB.Academy20.dbo.SchoolYear where active = 1))

declare @child int = (select top 1 t.testID    -- Subquery to get current ACT - ELA child test

        from ICSQLDB.Academy20.dbo.Test t

        inner join ICSQLDB.Academy20.dbo.Test p

            on p.testID = t.parentID

        where p.name = 'Grad Requirements'
```

```sql
            and t.name = 'ACT - ELA'

            and p.endYear = (select endYear from ICSQLDB.Academy20.dbo.SchoolYear where active = 1))


-- If parent test does not exist, create it.

insert into ICSQLDB.Academy20.dbo.TestScore (districtID, testID, personID)

select distinct 1 districtID, @parent testID, act.personID

from #actELA act

left join ICSQLDB.Academy20.dbo.TestScore ts

    on ts.personID = act.personID

    and ts.testID = @parent

where ts.testID is null


-- Insert child test records, match to existing parent test.

insert into ICSQLDB.Academy20.dbo.TestScore (districtID, parentID, testID, personID, result)

select distinct 1 districtID, ps.scoreID parentID, @child testID, act.personID, act.result

from #actELA act

inner join ICSQLDB.Academy20.dbo.TestScore ps

    on ps.personID = act.personID

    and ps.testID = @parent

left join ICSQLDB.Academy20.dbo.TestScore ex    -- If a score already exists for this student / test / result combination, do not import duplicates.

    on ex.personID = act.personID

    and ex.testID = @child

    and ex.result = act.result

where ex.scoreID is null
```