

Attendance Dashboard

This program should be built exclusively using HTML and .js files to be used in a google sheet.

No fake data should be entered when creating the code.

There should be a file with all general functions that can be used across all pages (like submissions that can be used for various tasks)

And one file per page to keep things neat.

A css html file that will hold all the designing code.

The program will have separate URL for:

- Dashboard
- Teacher's Attendance page for BCM
- Teacher's Attendance page for OM

It should be simple, easy on the eye (calm green) (Information shouldn't be justified to the left in rows so the eye doesn't have to scan all the way across the screen when not necessary.), self explanatory and should run as quickly as possible. (maybe using 2D arrays).

Every time a student's name is used in the program, it should show the student ID and use it for comparison. not the actual name (to avoid confusion between students with the same name).

Every time a Family ID is used in the program, it should show the Family ID and Name and use ID for comparison. not the actual name (to avoid confusion between families with the same name).

Any time a date is used it should ONLY use yyyy-mm-dd also when information is being entered into the program.

Any times shown in the dashboard should only display the time without date or seconds

When writing a lot of info or loading info using `getValue()` or `setValues()` to run quicker.

Have the Calendar and Student_Roster stored in browser's memory when dashboard is opened to save time from getting it for finding student's information.

In pages that a lot of data needs to be loaded, use pagination to keep the program running faster.

When Dashboard is opened, The last 30 days from Attendance sheet should go in to the browser's memory to keep things running quickly. Only if user clicks 'load more' when updating attendance should the program look for more saved data.

My Google Sheet Should be build as follows:

Attendance_Data with headers: Student ID | Division | Date | Grade | Name | Arrival Status (On Time / Late / Absent) | Absent Excused (Yes / No) | Late Time | Late Excused (Yes / No) | Leave Time | Leave Excused (Yes / No) | Back Time | Comments | Total Time Missed Excused (In cell formula should calculate this automatically using Hours_Schedule sheet for times) | Total Time Missed Unexcused (In cell formula should calculate this automatically using Hours_Schedule sheet for times) | Total Time Attended (In cell formula should calculate this automatically using Hours_Schedule sheet for times) | Last Updated

Student_Roster with headers: Student ID | Grade | Last Name | First Name | Name | Division | Email | Family ID | Total Excused Hours (Formula should calculate this automatically from Attendance_Data) | Total Unexcused Hours (Formula should calculate this automatically from Attendance_Data) | Yesterday's Total Unexcused Hours

E_Absent with headers: Student ID | Grade | Name | Start Date | Return Date | Reason | Reason Type (Excused / Letter / Other / Unexcused) | Last Updated | Passed | Date Submitted

E_Late with headers: Student ID | Grade | Name | Date | Arrival Time | Reason | Reason Type (Excused / Letter / Other / Unexcused) | Last Updated | Passed | Date Submitted

E_Leave with headers: Student ID | Grade | Name | Date | Time | Return Time | Picked up by | Reason | Reason Type (Excused / L-Recived / Letter / Other / Unexcused) | Last Updated | Passed | Date Submitted

Calendar with headers: Subject | Start Date | Start Time | End Date | End Time | Description | Division (Limited to: All/ BCM/ OM/ PreSchool | Grade | Last Updated | Event Type (First Day / Last Day / No School / Early Dismissal / Late Start / Start Term 2)

Email_Settings with information for warning emails

Email_Log with headers: Student ID | Division | Grade | Name | Hours | Email Type | Date Sent to School | Date Sent to Parents | Approved (Unapproved / Approved / Pending)

Notes with headers: Attention to | Reminder at | Note | Last Updated | Dismissed

Attendance_Report with row 1 indicating Report Period. And headers in row 2 should be Student ID | Division | Grade | Hours Attended | Days Attended | Grade Total Days | Grade Total Hours | Grade Avg Hours per day

Hours_Schedule Sheet:

An auto-generated reference sheet that maps out the entire school year week by week, calculates the total hours per week per grade and also total hours per day per grade with place to record if attendance was taken that day.

Each row is one week (Sunday–Fri).

Column A shows the dates for Sunday- date for Friday (starting with the first week of school and ending with the last week of school. This information is found in Calendar)

On Row 2 Columns B-G have a header of the days of the week with under it color coded — green for regular school, red for no school, yellow for short/early dismissal day.

Row 1 After Column G should be merged and say BCM Total for week(Light pink background) or OM Total for week (Light blue background) as a group indicator for the columns underneath it.

On Row 2 Columns after Column G represents a grade (from BCM and then from OM) and under it will show the total school hours for that grade that week, accounting for no-school days (x), early dismissals(~), late starts(~), division-specific schedules(division that has school's name), and grade-specific exceptions(symbol for majority of the school with a*) (found in Calendar).

Regular days hours are:

For OM:

Grades 6b, 7b & 8b (called Minyan) start at 7:30 and on Sundays at 8:15.

All other Grades in this division start at 8:45 and on Sundays at 9:45

All Grades end school at 16:00, on Sundays at 13:00 and on Fridays at 12:30

For BCM:

All grades start at 8:45 and end at 16:00 (12:30 on Fridays)

After the total for week columns will come the next section with row 1 having the name for each grade (BCM & OM) and then in row 2, each 3 columns should be merged and have a Sun-Fri for OM and Mon-Fri for BCM.

On the left column of each day will be the start time for that day for that grade, the middle column will have the end time for that day for that grade, and on the right column for that day will be a place to put a check if attendance was taken that day for that grade.

Update manually via the "Update Hours Schedule" menu item in the spreadsheet, or via the Recalculate button in the dashboard (MAKING SURE NOT losing which days

attendance was taken). The dashboard reads this sheet on load to power the Total Hours section in Total Attendance, showing each grade's projected yearly hours, expected hours by today, and actual attended hours with a progress bar.

A trigger at 9:15 will check if there is school that day, if yes, check if attendance was taken and if there are any unexcused absences that might need verification with the parents. If yes, it will show a pop up in the dashboard that will make a pleasant ringing sound. And it will also send an email (whether dashboard is open or not) to the office with the information.

The Sheets get their information via:

Attendance_Data From Attendance (Teacher's URL or Dashboard), updated via Dashboard.

Student_Roster Manually entry for most aside from: Total Excused Hours (Formula should calculate this automatically from Attendance_Data) & Total Unexcused Hours (Formula should calculate this automatically from Attendance_Data).

E_Absent & E_Late & E_Leave will get it's information from a form with a separate URL that can be accessed by the parents. & updated from the Dashboard.

Every Wednesday a trigger will go through all the rows that haven't been marked as passed and send an email to the office tepia613@gmail.com with a clear list of all the upcoming absences (including ones that have started already but are continuing in this period of time) for any date (when writing date, include the day of week) between this Wednesday and until next week's Friday. The list should be divided by grade.

It should be worded:

- For Absent: **{Name}** will be absent on **{Date}** for {Reason}
- For Late: **{Name}** will be late on **{Date}** and will arrive at **{Time}** because {Reason}
- For Leave: **{Name}** will leave on **{Date}** at **{Time}** with {Picked up by} (and will be back at {Back Time}) for {Reason}

In addition a trigger will:

- Everyday at 6:00am send any entry entered after the email on Wednesday (Date Submitted) that is for this week (before next wednesday email) to the office to notify them about this update.
- Also at 6:00am Copy the Total Unexcused Hours for each student to Yesterday's Total Unexcused Hours in Student_Roster
- Everyday at 4:30 Mark any rows that their end date has gone by as passed.

Calendar Manually with updates via Dashboard

Email_Settings Manually with updates via Dashboard

Email_Log

Every day at 4:30PM a trigger will call a function that will check all student's total unexcused hours (from Student_Roster) checking if they have gone over a number of hours and should get a notification email.

The Type of email is determined by the amount of unexcused hours:

- Warning Email: will get sent after the student reached more than any of the numbers in Email_Settings B1 separated by ; but less than the next type
- Reaching Limit Email: will get sent after the student reached more than any of the numbers in Email_Settings B2 separated by ; but less than the next type
- Meeting needed Email: will get sent after the student reached the number or more in Email_Settings B3 separated by ;

To check if an email should be sent, the program checks if an email has been sent already by checking if yesterday's total is still within the same range (for example, if a warning email should be sent after 35; 45; 55, and yesterday's total says 36 and today's 40, it still is in the same range) and an email was probably sent yesterday. But if not (for example today's is 45), then a new email should be sent.

If an email should be sent, then a new row is added in Email_Log: Student ID | Division | Grade | Name | Hours (today's total) | Email Type | Date Sent to School | Date Sent to Parents (leave empty) | Approved (Pending).

At 2pm a new trigger will:

- check if any Approved emails have not been sent to Parents, generate a report google doc with all this student's absences from Attendance_Data and sends the applicable email to this parent's email and principal (For BCM students find the email in Email_Settings B4 and for OM in B5) with a link to view the report. The subject for the email and body should be taken from the Email_Settings page accordingly. Once done, the date should be recorded in the Date sent to Parents in the appropriate row.

Notes with headers: Attention to | Reminder at | Note | Last Updated | Dismissed

Attendance_Report: generated from Dashboard

Hours_Schedule Sheet: When Attendance is submitted each grade gets marked that attendance was taken for that day. When an event is added to calendar, it should update this sheet as well (making sure not to lose the data of dates that attendance have been taken!)

This is how the dashboard should work:

All Submit buttons should get disabled once clicked for 5 seconds unless said otherwise.

Left pane: Division Selector (Circle with option of BCM / OM / ALL), Main, Attendance, Update Attendance, Overview, Excuse Absences, Notes & Reminders, Total Attendance, Email Log, Calendar.

Main:

It will show a quick overview that includes:

- Names of students that are marked Absent Unexcused for today (only for this division taken from Attendance_Data) OR a Attendance Was Not Taken warning (Can be checked by Hours_Schedule). (If Attendance was not taken for a specific grade, it should also have a warning for that grade).
- Number of unexcused absences for today
- Number of parents entries for today
- Notes & Reminders

Each of these should be a button leading to the following:

- Update Attendance
- Overview
- Notes & Reminders

Then under all the above will be a list with all the Upcoming Today information:

Upcoming Today:

The information for this section should be taken from E_Absent, E_Late & E_Leave Filtered by Today and division

This section will show 3 sections; Late, Leave & Absent. Each section will not have a fixed height, but will be adjusted by the amount of data in it.

there will be a list showing the information (from the selected division).

For Late (from E_Late): {Grade} {Name} will arrive {Arrival Time} for {Reason} and is {Reason Type}

For Leave (from E_Leave): {Grade} {Name} will Leave at {Time} with {Picked up by} (, will be Back at {Return Time}) for {Reason} and is {Reason Type}.

For Absent (from E_Absent): {Grade} {Name} will be absent {Start Date} (be back on {Return Date}) for {Reason} and is {Reason Type}

Each row will be clickable for a quick edit including: modifying Time (option to remove back time), Excused / Unexcused, add comment, there should be a checkbox to check off if a letter was brought (If checked off, add to the comment: Letter received) and submit.

When submitted, the program will search in Attendance_Data for this student's row on Today's date and modify it according to the information submitted:

For Late: Arrival Status (Change Absent to Late) | Arrival Excused (Clear) | Late Time | Late Excused (Reason Type) | Comments | Last Updated

For Leave, if a row is not found, create a new one putting in all the information needed including: Leave Time | Leave Excused (Reason Type) (If Back was modified also update: | Back Time) | Comments | Last Updated

For Absent: Arrival Excused | Comments | Last Updated

If the checkbox was clicked, the program should also locate this row in its original sheet and change {Reason Type} to L-Recived.

Attendance:

If OM is selected, there should be a button allowing you to select or unselect Minyan.

A place to put the date (automatically should be set for today with option to change)(If this date attendance was taken already, indicated in Hours_Schedule, a warning should come up saying 'Attendance has been taken for grades...') and parallel to it the time (automatically set. On weekdays: BCM / OM 8:45. On Sundays 9:45. Or if date is in calendar sheet and has a late start time, that time should be placed. With option to change). If time is changed, a reason text box should appear under the time with a place to insert the reason for the time change. (when attendance is submitted, this late start should be added to the calendar sheet. If the text box is empty, an error should appear)

Under that there should be an option to select which grades to do attendance for. There should be little buttons in a horizontal row with all the grades from that division (taken from the Student_Roster sheet Column F - Division) and the last one being All. If OM division is selected, there should also be a button that will select all grades from 1b-5b, and a button to select Minyan grades (selecting any grades 6b, 7b & 8b). When Minyan is selected, the

Time In should change to 7:30 & on Sundays to 8:15. But if 1b-5b is selected, it should go back to the automatic time.

Then there should be a button to generate the student's names.

Names should be generated in a list form with their class name listed on the top (this information should be taken from Student_Roster)

Next to each class name there should be a button to mark All Here & a Skip button.

Next to each student there should be 2 buttons: Here / Not here

At the end there should be a submit button that should only allow the submitting if all students were marked as here or not (unless class marked as Skip).

Once the names were generated, the program should look for any Excused absent set for the date that the attendance is being filled in for in E_Absent and save the Reason to be placed in comments.

Once submitted:

A big loading should appear to show that attendance is being processed (maybe even a bar showing progress).

All Not here should be saved as Absent Unexcused unless the program found that this student should be excused this date (as mentioned before).

All 'Not Here' should get saved in Attendance_Data And it should fill in Student ID | Division | Date | Grade | Name | Arrival Status (Absent) | Absent Excused (Yes / No) | (Reason in to Comments) | Last Updated . As long as that information wasn't saved already for that day. If it was, it should overwrite it, not allowing double entry for the same day and student.

Then it should find that date in Hours_Schedule and mark for the grades that attendance was taken for (and not skipped) that attendance was taken.

Update Attendance:

The information for this section should be taken from Attendance_Data Filtered by division

This section will cover: Absent, Late & Leave. There will be 2 buttons side by side at the top: Today / All (Last 30 days). The information will be separated in to 2 sections: Absent/Late & Leave

Top Section Absent / Leave will show a list of students who are marked Absent / Late by {Arrival Status}

There should be an option to sort it by date or grade. And option to hide all excused (hiding the ones that have: Yes {Arrival Excused})

For Absents the rows should be:

{Date} {Grade} {Name} Absent {Arrival Excused} {Comments}

For Late the rows should be:

{Date} {Grade} {Name} Arrived at {Late Time} {Late Excused} {Comments}

Each row will be have 2 buttons: Excuse & Late.

The Excuse button: When clicked the information from that row will pop up in a way that the Arrival Excused & Comments can be modified and a button to submit.

When submitted, the program will search in Attendance_Data for this student's row on {Date}'s date and modify it according to the information submitted.

The Late button: When clicked the information will pop up with an option to add or modify; Late Time, if Excused or Unexcused and comment and a button to submit.

When submitted, the program will search in Attendance_Data for this student's row on {Date}'s date and change: Arrival Status (Change Absent to Late) | Arrival Excused (Clear) | Late Time | Late Excused | Comments | Last Updated.

A +New button will (generate information from Student_Roster and) give an option to select date, family or grade (place holder should be 'select grade'), student name (place holder should be 'select student' with an option to select any student from the grade or family selected before. If Family was selected, there should be an option to select multiple family members), Option for Absent / Late, (if Late place to put in time), Excused/Unexcused, Comment and submit button.

When submitted, the program will search in Attendance_Data for this student's (or students' if more then one was selected) row on {Date}'s date. If data was found, a confirmation should pop up asking 'are you sure you want to overwrite this information'. If yes, the row (or rows) should be updated with the new information. If data was not found it should write a new row (per student selected) with all the relevant information.

Bottom Section Leave: will have a button +New and under it will show a list of students who have data filled in by {Leave Time}

There should be an option to sort it by date or grade. And option to hide all excused (hiding the ones that have: Yes {Leave Excused})

the rows should be:

{Date} {Grade} {Name} Left at {Leave Time} (Back at {Back Time}) {Leave Excused} {Comments}

Each row will be clickable. When clicked the information will pop up with an option to add or modify; Leave Time, if Excused or Unexcused, Back Time and comment and a button to submit.

The +New button will (generate information from Student_Roster and) give an option to select date, Family or grade (place holder should be 'select grade'), student name (place holder should be 'select student' with an option to select any student from the grade or Family selected before. If Family was selected, there should be an option to select multiple family members), Leave Time, Excused/Unexcused, Back Time(Optional), Comment and submit button.

When submitted (both old or new), the program will search in Attendance_Data for this student's (or students' if more then one was selected) row (or rows) on {Date}'s date and change: Leave Time | Leave Excused | Back Time | Comments | Last Updated. If data was not found a new row (per student selected) should be written with all the applicable information marking the student On Time in Arrival Status.

Overview:

This part is very similar to the Update Attendance and can use many overlapping formulas!

The information for this section should be taken from Student_Roster & Attendance_Data Filtered by division.

will show a list of students in the selected division

There should be an option to sort it by name, grade, Unexcused hours, Family ID

(From Student_Roster): {Grade} | {Name} | {Total Unexcused Hours} (If over 20 color red, If over 30 make bold text) | {Family ID}

Each name will be clickable and when clicked it will show (not in a pop up window) all the following data for this student in Attendance_Data:

Date, Arrival Status (If Arrival Excused says Yes make the Arrival Status yellow, if No make it red), Late Time (If Late Excused says Yes make the Late Time yellow, if No make it red), Leave Time (If Leave Excused says Yes make the Leave Time yellow, if No make it red), Back Time, Comments.

Each row will be clickable. When clicked the information will pop up with an option to add or modify all information aside from date and a button to submit (There should also be an option to remove Late / Leave / Back time).

When submitted, the program will search in Attendance_Data for this student's row on {Date}'s date and change any changes made

Excuse Absences:

This part is very similar to the Upcoming Today and can use many overlapping formulas!

The information for this section should be taken from E_Absent, E_Late & E_Leave Filtered by division

There should be 3 tabs spanning across the screen: Letter, Other & All.

Letter tab will show all the rows that have Reason type = Letter / L-Recived.

Other tab will show all the rows that have Reason Type = Other / Excused / Unexcused.

All information from both tabs together

There should be an option to sort by past dates, upcoming dates (today-future), by name & hide Excused

There should be a +Add Excused Absence button that will let to select type Absent / Late / Leave and fill in any data needed. And this info should go in to E_Absent / E_Late / E_Leave correspondingly.

This section will show 3 sections; Absent, Late & Leave. Each section will not have a fixed height, but will be adjusted by the amount of data in it. There will be a list as follows:

For Absent rows with information including (from E_Absent): {Grade} {Name} will be *absent* {Start Date} (be back on {Return Date}) for {Reason} and is {Reason Type}

For Late rows with information including (from E_Late): {Grade} {Name} will be *late* {Date} {Arrival Time} for {Reason} and is {Reason Type}

For Leave rows with information including (from E_Leave): {Grade} {Name} will be *leaving* with {Picked up by} on {Date} {Time} (if Return Time is not null include here: will be *back* {Return Time}) for {Reason} and is {Reason Type}

Each row will be clickable for a quick edit including: modifying the date, Time (option to remove back time), Excused / Unexcused, add comment, there should be a checkbox to check off if a letter was brought (If checked off, add to the comment: Letter received) and submit.

When submitted:

The changes should be updated in the original sheet with the comment added to the existing {Reason} & if Letter received was checked off, {Reason Type} should be saved as L-Recived.

For passed Dates the program should also update the saved data in Attendance_Data as follows:

The program will search in Attendance_Data for this student's row on {date} and modify it according to the information submitted:

For Absent: Arrival Excused | Comments | Last Updated

For Late: Arrival Status (Change Absent to Late) | Arrival Excused (Clear) | Late Time | Late Excused (Reason Type) | Comments | Last Updated

For Leave: Leave Time | Leave Excused (Reason Type) (If Back was modified also update: | Back Time) | Comments | Last Updated

Notes:

(This section is always for all divisions and visible to all divisions)

Should display all saved notes with the option to filter by date or name.

Should have an option to add a note with a pop up giving option to put a name (free text), when is the reminder for (optional) and the note content.

Each note should have an option to edit it or delete it.

A reminder to send parents a note should come up 2 weeks before an event start date saved in calendar.

Total Attendance:

Title: Total days in school

Will show an option to enter a start date (with the default being the first day of school found in Calendar) and an option to enter end date (with today being the default).

then there will be a button to generate the report

The program will rewrite Attendance_Report sheet

Retrieve all student's needed information from Student_Roster, then use Hours_Schedule to check for each grade it's total hours and total days within the time period selected and calculate avg hours per day. Calculate each student's total hours by subtracting from their grade total all the missed hours recorded for that student in Attendance_Data. Then calculate for each student how many days they were in school by dividing their total hours by the grade's avg hours per day (the final number should have 2 numbers past the decimal point).

When done, a link to Attendance_Report will appear in the dashboard.

Title: Report Cards

Will show 2 buttons side by side: 1st Term & 2nd Term

Will show buttons to select grades displayed in 2 rows (1 with all the BCM grades and 1 with all the OM grades) and the last one in each row being All

Will show a button to generate Lates & Absences

When generated, a list will appear with a small space between grades, showing all students from grades selected with their total Unexcused Lates and total Unexcused Absent in that Term (for 1st Term calculate all dates between first day of school until start second term found in Calendar. For 2nd Term calculate all dates between start second term until last day of school found in Calendar) and an option to print the list (this will generate a clear sheet to print with just this list)

Title: Total Hours

This will show a little square per grade with a progress bar showing how many hours they have been in school so far / how many hours they have total scheduled.

This information should be taken from Hours_Scheduled

Email Log:

This information will be taken from Email_Log & Email_Settings

If any rows in Email_Log are marked as Pending in Column H (Approved), a little star will appear by the 'Email Log' on the side bar to show that attention is needed here.

There should be 3 buttons parallel to each other horizontally: Update Warning Email, Update Limit Email & Update Meeting Needed Email

Update Warning Email: Will open a pop up with an option to change the number of hours (saved in B1)(the different sum of hours should be separated by ;), the Subject (saved in B8) & Email Body: (Saved in B10). Before allowing this information to be edited, ask 'are you sure you want to save these changes?'

Update Reaching Limit Email: Will open a pop up with an option to change the number of hours (saved in B2)(the different sum of hours should be separated by ;), the Subject (saved in B14) & Email Body: (Saved in B16). Before allowing this information to be edited, ask 'are you sure you want to save these changes?'

Update Meeting needed Email: Will open a pop up with an option to change the number of hours (saved in B3)(the different sum of hours should be separated by ;), the Subject (saved in B20) & Email Body: (Saved in B22). Before allowing this information to be edited, ask 'are you sure you want to save these changes?'

Under the 3 buttons there should be a list showing any rows that are marked as Pending (In column H) with an option to click the row and change the Pending to Unapproved or Approved with a message 'do you want to approve/unapprove this email?'.

Under that there should be an uneditable list with all the Approved rows listed in the sheet.

Calendar:

Will show a calendar (calendar format) with school's dates (taken from Calendar sheet).
There should be an option to add event here that will be added to the Calendar sheet.

There will also be a separate url (one for OM and one for BCM) for teachers to access only the attendance taking (not checking or modifying). To use this feature a pin will be necessary (1770)

This site needs to run quickly and easily to be done on a phone (shouldn't be too small).