

```

/*
*   Manager Dashboard
*   https://docs.google.com/spreadsheets/d/1C3dDc-C7LvRL7gioAAiDFj-dfFOVCb6XemK6hvF\_aM0/edit#gid=2117167135
*
*   Employee Dashboard
*   https://docs.google.com/spreadsheets/d/1gUSe3CvPbocMjeGift4tKAKBrVxfMvJOkkw60S4JxB0/edit#gid=120178601
*
*   Scripts to manage employee schedules, Office hours (CHAT), record Project time, case backups, and full systems team reporting
*
*   team rosters, schedules, employee data - all derived from the Employee Data tab, everything can be altered from there
*   data feeds from Manager dashboard to the employee dashboards and back, sending data between when directly input by employee
*
*   Nightly script resets the scheduling and clears the daily activity for backups, chats, etc
*
*   Intricate Google sheets formulas used to maintain lists of teams, scheduling, calls by name/manager, etc. - this was done to create a
*   centralized (one tab) location for editing, updating and managing teams and employee information
*
*   backup systems were created to record activity for further reporting and statistical analysis across the entire team based on work done,
*   time spent and projects taken
*
*   Buttons were created for ease of use by the user (employee) to quickly reset daily activities or submit requests for assistance
*/

```

```

function lateShift(e)
{
    if (e.value != "LATE" && e.value != "REM-LATE")
        return;

    var mgrCaseSheet =
    SpreadsheetApp.getActiveSpreadsheet().getSheetByName("Case Assigning Sheet");
    var mgrEmpSheet =
    SpreadsheetApp.getActiveSpreadsheet().getSheetByName("Employee Data");
    var lastEmpRow = mgrEmpSheet.getLastRow();
    var empName;
    var empSheetID;
    var empSheet;
    var empDashboard;

```

```

// check 'Status' column for "LATE" status
for (late = 3; late < 46; late++)
{
    if (mgrCaseSheet.getRange(late,2).getValue() == "LATE" ||
mgrCaseSheet.getRange(late,2).getValue() == "REM-LATE")
    {
        empName = mgrCaseSheet.getRange(late,1).getValue();
        for (r = 2; r <= lastEmpRow; r++)
        {
            if (empName == mgrEmpSheet.getRange(r,1).getValue())
            {
                empSheetID = mgrEmpSheet.getRange(r,4).getValue();
                break;
            }
        }

        empSheet = SpreadsheetApp.openById(empSheetID);
        empDashboard = empSheet.getSheetByName("Dashboard");
        empDashboard.getRange(4,3,1,49).clearContent();
        empDashboard.getRange(4,3,1,12).setValue(6);
        empDashboard.getRange(4,19,1,4).setValue(4);
        empDashboard.getRange(4,37,1,1).setValue(1);
    }
}
};

function checkReminder()
{
    // get the spreadsheet object
    var spreadsheet = SpreadsheetApp.getActiveSpreadsheet();
    // set the first sheet as active
    SpreadsheetApp.setActiveSheet(spreadsheet.getSheetByName("Facilitator's
Sheet"));
    // fetch this sheet
    var sheet = spreadsheet.getActiveSheet();

    // figure out what the last row is
    var lastRow = sheet.getLastRow();

    // the rows are indexed starting at 1, and the first row
    // is the headers, so start with row 2
    var startRow = 2;

    // collect values for employee email and 'backups worked' status
    var remRange = sheet.getRange(2,7,lastRow-startRow+1,1);
    var addRange = sheet.getRange(2,4,lastRow-startRow+1,1);

```

```

    var numRows = remRange.getNumRows();
    var reminder = remRange.getValues();
    var address = addRange.getValues();
    var msg = "Hello, this is a reminder to check your Back up Case responsibilities
for the day.";

    // Loop through the team to check for 'backups worked' status and send a
reminder email
    for (var i = 0; i <= numRows - 1; i++)
    {
        var confirm = reminder[i][0];
        if (confirm == "Yes")
        {
            var email = "ebscofacilitator@gmail.com," + address[i][0];
            MailApp.sendEmail(email,"Courtesy Backup Reminder", msg);
        }
    }
    Browser.msgBox('Reminder Sent Successfully.');
```

```

};

function checkQueue()
{
    // get the spreadsheet object
    var spreadsheet = SpreadsheetApp.getActiveSpreadsheet();
    // set the first sheet as active
    SpreadsheetApp.setActiveSheet(spreadsheet.getSheetByName("Facilitator's
Sheet"));
    // fetch this sheet
    var sheet = spreadsheet.getActiveSheet();

    // figure out what the last row is
    var lastRow = sheet.getLastRow();

    // the rows are indexed starting at 1, and the first row
    // is the headers, so start with row 2
    var startRow = 2;

    // collect values for employee email and 'in queue' status
    var remRange = sheet.getRange(2,8,lastRow-startRow+1,1);
    var addRange = sheet.getRange(2,4,lastRow-startRow+1,1);
    var numRows = remRange.getNumRows();
    var reminder = remRange.getValues();
    var address = addRange.getValues();
    var msg = "Hello, this is a Low Queue Warning, please check if you are
supposed to be in the Queue.";

    // Loop through the team to check for 'in queue' status and send a reminder email

```

```

        for (var i = 0; i <= numRows - 1; i++)
        {
            var confirm = reminder[i][0];
            if (confirm == "Yes")
            {
                var email = "ebscofacilitator@gmail.com," + address[i][0];
                MailApp.sendEmail(email,"Low Queue Warning", msg);
            }
        }
        Browser.msgBox('Queue Reminder Sent Successfully.');
```

```

};

function clearInOutBoard()
{
    var mgrSheet =
    SpreadsheetApp.openById("1graglcVXxtD8cEvtui_JsxKC8GusSjixE5xczwUEFwl");
    var mgrEmpSheet = mgrSheet.getSheetByName("Employee Data");
    var mgrCaseAssign = mgrSheet.getSheetByName("Case Assigning Sheet");
    var mgrBackup = mgrSheet.getSheetByName("Back up Cases");
    var mgrBkpArchive = mgrSheet.getSheetByName("Backup Archive");
    var empSheet; // used to reference Employee Dashboard sheet later
    var lastRow = mgrEmpSheet.getLastRow();
    var lastBkpRow = mgrBackup.getLastRow();
    var lastArcRow = mgrBkpArchive.getLastRow();

    var r = 4;
        var column = 3;
    var status;
    var assign;
    var bandwidth;
    var row = lastArcRow + 1;
    var archive;

    // archive backup daily backup cases
    for (bkp = 3; bkp <= lastBkpRow; bkp++)
    {
        archive = mgrBackup.getRange(bkp,1,1,7).getValues();
        if (archive[0][0] != "")
        {
            mgrBkpArchive.getRange(row,1).setValue(new Date());
            mgrBkpArchive.getRange(row,2,1,7).setValues(archive);
            mgrBackup.getRange(bkp,1,1,4).clearContent();
            row++;
        }
    }
}

// clear Case Assigning Sheet daily values

```

```

for (clr = 3; clr < 60; clr++)
{
    //clear each editable column row by row
    status = mgrCaseAssign.getRange(clr,2).getValue();
    assign = mgrCaseAssign.getRange(clr,5).getValue();
    bandwidth = mgrCaseAssign.getRange(clr,7).getValue();

    if (status != "EARLY" && status != "Status")
    {
        mgrCaseAssign.getRange(clr,2).clearContent();
    }

    if (assign != "None" && assign != "CRM Assigned")
    {
        mgrCaseAssign.getRange(clr,5).clearContent();
    }

    if (bandwidth != "Bandwidth" && bandwidth != "N/A")
    {
        mgrCaseAssign.getRange(clr,7).clearContent();
    }
}

// reset shifts to default values for employees
for (emp = 2; emp <= lastRow; emp++)
{
    var row = emp;
    var sheetID = mgrEmpSheet.getRange(row,4).getValue();
    var empTitle = mgrEmpSheet.getRange(row,8).getValue();

    var startShift = mgrEmpSheet.getRange(row,9).getDisplayValue();
    var startFill = 1;
    var endShift = 1;
    var offPhone = mgrEmpSheet.getRange(row,10).getDisplayValue();
    var offFill = 1;

    empSheet = SpreadsheetApp.openById(sheetID);
    var empDashboard = empSheet.getSheetByName("Dashboard");

    switch(startShift)
    {
        case "07:00:00":
        {
            startFill = -8;
            offFill = 3;    // 9:00 am column number
            break;
        }
    }
}

```

```
case "08:30:00":
{
    startFill = -2;
    break;
}
case "09:00:00":
{
    startFill = 0;
    break;
}
case "09:30:00":
{
    startFill = 2;
    break;
}
case "10:00:00":
{
    startFill = 4;
    break;
}
}
```

endShift = startFill + 34;

```
switch(offPhone)
{
    case "09:00:00":
    {
        offFill = 3;
        break;
    }
    case "10:00:00":
    {
        offFill = 7;
        break;
    }
    case "11:00:00":
    {
        offFill = 11;
        break;
    }
    case "12:00:00":
    {
        offFill = 15;
        break;
    }
    case "01:00:00":
```

```

    {
        offFill = 19;
        break;
    }
    case "02:00:00":
    {
        offFill = 23;
        break;
    }
    case "03:00:00":
    {
        offFill = 27;
        break;
    }
    case "04:00:00":
    {
        offFill = 31;
        break;
    }
}

```

// clear backup case Status and Action Taken fields for each employee Dashboard sheet

```

// RANGE = F11:I22
if (sheetID != "1Nn-mvrnjtonOtS3BRpX0e1VjNNLzBVW8ygc3hOtRAVI")
{
    for (R = 11; R <= 22; R++)
    {
        empDashboard.getRange(R,6).clearContent();
        empDashboard.getRange(R,9).clearContent();
    }

    for (C = 39; C < 50; C++)
    {
        if (empDashboard.getRange(C,1) == "")
        {
            empDashboard.getRange(C,11).clearContent();
            empDashboard.getRange(C,16).clearContent();
            empDashboard.getRange(C,21).clearContent();
            empDashboard.getRange(C,26).clearContent();
            empDashboard.getRange(C,28).clearContent();
        }
    }
}

```

```

        for (c = column;c < 49;c++)
        {
            if (empTitle == "ATSR" || empTitle == "TSR")

```

```

        {
            empDashboard.getRange(r,c).clearContent();
        }
        else
        {
            empDashboard.getRange(r,c).setValue(4);
        }

        empDashboard.getRange(r,offFill,1,4).setValue(4);
        if (startFill > 0)
        {
            empDashboard.getRange(r,3,1,startFill).setValue(6);
        }
        empDashboard.getRange(r,endShift+3,1,46-endShift).setValue(6);
    }
}
};

function archiveCHAT()
{
    var mgrSheets = SpreadsheetApp.getActiveSpreadsheet();
    var mgrChat = mgrSheets.getSheetByName("CHAT Requests");
    var mgrChatArchive = mgrSheets.getSheetByName("CHAT Archive");
    var mgrEmployee = mgrSheets.getSheetByName("Employee Data");
    var empSheetID;
    var empDashboard;
    var lastEmpRow = mgrEmployee.getLastRow();
    var lastChatRow = mgrChat.getLastRow();
    var lastChatArcRow = mgrChatArchive.getLastRow();
    var archive;
    var rowChat = lastChatArcRow + 1;

    for (arc = 2; arc <= lastChatRow; arc++)
    {
        archive = mgrChat.getRange(arc,1,1,16).getValues();

        if (archive[0][3] == "")
        {
            continue;
        }

        if (archive[0][0] != "" && archive[0][3] == "Yes")
        {
            mgrChatArchive.getRange(rowChat,1,1,16).setValues(archive);

```

```
/*
```



```

        for (emp = 2; emp <= lastEmpRow; emp++)
        {
            if (archive[0][2] == mgrEmployee.getRange(emp,
1).getValue())
            {
                empSheetID = mgrEmployee.getRange(emp,
4).getValue());
                break;
            }
        }
        empDashboard = SpreadsheetApp.openById(empSheetID);
        empDashboard.getRange("K39:AB49").clearContent();
*/

        mgrChat.getRange(arc,1,1,11).clearContent();
        rowChat++;
    }
}

/*
var blanks = mgrChat.getRange(2,1,102,11).getValues();
// Browser.msgBox('Low Queue Warning',blanks[1][0],Browser.Buttons.OK);
// return;

for (count = 99;count > 95;count--)
{
    if (blanks[count][0] == "")
    {
        continue;
    }
    else
    {
        //Browser.msgBox('Low Queue Warning',blanks[count][0],Browser.Buttons.OK);
        // reaches this, put 'Test Name' in row 98 or so for testing

        for (line = 1; line > 100; line++)
        {
            Browser.msgBox('Low Queue Warning',count,Browser.Buttons.OK);
            return;

            if (blanks[count][0] != "" && blanks[line][0] == "")
            {
                mgrChat.getRange(line,1,1,11).setValues(blanks[count][0]);
                mgrChat.getRange(count,1,1,11).clearContent();
            }
        }
    }
}
}

```

```
    }  
    */  
};
```

```
function clearTime()  
{  
    var sheet =  
SpreadsheetApp.getActiveSpreadsheet().getSheetByName("Dashboard");  
    var data_sheet = SpreadsheetApp.getActiveSpreadsheet().getSheetByName("Data  
Sheet");  
    var r = 4;  
    var column = 3;  
    var response = Browser.msgBox('Reset Daily Breaks','Are you sure you want to clear  
your breaks for the day?',Browser.Buttons.YES_NO);  
  
    var startShift = data_sheet.getRange(2,4).getDisplayValue();  
    var startFill = 1;  
    var endShift = 1;  
    var offPhone = data_sheet.getRange(2,5).getDisplayValue();  
    var offFill = 1;  
  
    switch(startShift)  
    {  
        case "07:00:00":  
        {  
            startFill = -8;  
            offFill = 3;    // 9:00 am column number  
            break;  
        }  
        case "08:30:00":  
        {  
            startFill = -2;  
            break;  
        }  
        case "09:00:00":  
        {  
            startFill = 0;  
            break;  
        }  
        case "09:30:00":  
        {  
            startFill = 2;  
            break;  
        }  
        case "10:00:00":
```

```

    {
        startFill = 4;
        break;
    }
}

endShift = startFill + 34;

switch(offPhone)
{
    case "09:00:00":
    {
        offFill = 3;
        break;
    }
    case "10:00:00":
    {
        offFill = 7;
        break;
    }
    case "11:00:00":
    {
        offFill = 11;
        break;
    }
    case "12:00:00":
    {
        offFill = 15;
        break;
    }
    case "01:00:00":
    {
        offFill = 19;
        break;
    }
    case "02:00:00":
    {
        offFill = 23;
        break;
    }
    case "03:00:00":
    {
        offFill = 27;
        break;
    }
    case "04:00:00":
    {

```

```

        offFill = 31;
        break;
    }
}

// Process the user's response.
if (response == 'yes')
{
    sheet.getRange(r,3,1,46).clearContent();
    sheet.getRange(r,offFill,1,4).setValue(4);
    if (startFill > 0)
    {
        sheet.getRange(r,3,1,startFill).setValue(6);
    }

    sheet.getRange(r,endShift+3,1,46-endShift).setValue(6);
}
else
{
    return;
}
}

function submitCHAT()
{
    // Sheet information
    var empSheets = SpreadsheetApp.getActiveSpreadsheet();
    var empDashboard = empSheets.getSheetByName("Dashboard");
    var mgrSheets =
SpreadsheetApp.openById("1graglcVXxtD8cEvtui_JsxKC8GusSjixE5xczwUEFwl");
    var mgrCHAT = mgrSheets.getSheetByName("CHAT Requests");

    // function variables
    var empCases;        // retrieve list of backup cases
    var chatLastRow;
    var chatRequest = empDashboard.getRange(1,2).getValue(); // Employee Name

    for (r = 0; r < 11; r++)
    {
        if (empDashboard.getRange((r+25),1).isBlank())
        {
        }
        else
        {
            for (R = 2; R < 101; R++)
            {
                if(mgrCHAT.getRange(R,5).getValue() == "")

```

```

    {
        chatLastRow = R;
        break;
    }
}

```

```

var row = chatLastRow;
var chatCase = empDashboard.getRange(r+25,1).getValue();
var chatSubject = empDashboard.getRange(r+25,2).getValue();
var chatCreate = empDashboard.getRange(r+25,7).getValue();
var chatDate = empDashboard.getRange(r+25,9).getValue();
var chatProduct = empDashboard.getRange(r+25,11).getValue();
var chatSupport = empDashboard.getRange(r+25,16).getValue();
var chatType = empDashboard.getRange(r+25,21).getValue();
var chatPriority = empDashboard.getRange(r+25,26).getValue();

```

```

mgrCHAT.getRange(row,5).setValue(chatCase);    // Case Number
mgrCHAT.getRange(row,7).setValue(chatSubject); // Case Subject
mgrCHAT.getRange(row,8).setValue(chatCreate);  // Date Created
mgrCHAT.getRange(row,9).setValue(chatDate);    // Today's Date
mgrCHAT.getRange(row,10).setValue(chatProduct); // Product/Interface
mgrCHAT.getRange(row,11).setValue(chatSupport); // Area of Support
mgrCHAT.getRange(row,2).setValue(chatType);    // CHAT Request Type
mgrCHAT.getRange(row,6).setValue(chatPriority); // Priority
mgrCHAT.getRange(row,1).setValue(chatRequest); // Request

```

```

// Clear Employee Dashboard cells
empDashboard.getRange((r+25),1).clearContent();
empDashboard.getRange((r+25),2).clearContent();
empDashboard.getRange((r+25),7).clearContent();
empDashboard.getRange((r+25),9).clearContent();
empDashboard.getRange((r+25),11).clearContent();
empDashboard.getRange((r+25),16).clearContent();
empDashboard.getRange((r+25),21).clearContent();
empDashboard.getRange((r+25),26).clearContent();
empDashboard.getRange((r+25),28).clearContent();

```

```

    }
}
}

```