```
<?php
define("NOAUTH", true);

require_once "/var/www/html/redcap/plugins/paw/outcomes_health/scripts/paw_shared.php"; // abstracted SQL queries

$filter = $_GET['filter'];

function calculate_completeness($table, $variable, $project, $filter)
{
	try
	{
		if($table == null || $variable == null)
		{
			throw new Exception("Table or variable is not defined.");
		}
		else
		{
			$calc_completeness = (100 * (round((completeness($table, $variable, $project, $filter) / completeness('demographics', 'warehouse_id', $project, $filter)), 3)));
			return $calc_completeness;
		}
	}
	catch(Exception $e)
	{
		echo "Error: " . $e->getMessage();
	}

}
```

```
try
{
	if($_GET['table'] == null || $_GET['table'] == '')
	{
		throw new Exception("Whoops! Completeness table is undefined.");
	}

	if($_GET['table'] == 'basic')
	{
		$projects = array(1, 2, 3, 5); // projects not including CAPO
		$project_labels = array("HAPPI 1&2", "HAPPI 2.5", "HAPPI 3", "SHARP");
		$wid = array(); // initialize completeness arrays
		$hosp_id = array();
		$hosp_doa = array();
		$ssn = array();
		$firstname = array();
		$lastname = array();
		$dob = array();
		$age = array();
		$race = array();
		$sex = array();
		$overall_completeness = array(1, 1, 1, 1);
		$averages = array();

		$filter_array= explode(" ,", $filter);

		for($i = 0; $i < count($projects); $i++)
		{ // push the rounded completeness values into the array for each project
```

```
                    for($j = 0; $j < count($filter_array); $j++)

                    {


                    array_push($wid, calculate_completeness('demographics', 'warehouse_id',
$projects[$i], $filter_array[$j]));

                    array_push($hosp_id, calculate_completeness('happi_data', 'hospital_id',
$projects[$i], $filter_array[$j]));

                    array_push($hosp_doa, calculate_completeness('hospitalization', 'hosp_doa',
$projects[$i], $filter_array[$j]));

                    array_push($ssn, calculate_completeness('demographics', 'ssn', $projects[$i],
$filter_array[$j]));

                    array_push($firstname, calculate_completeness('demographics', 'firstname',
$projects[$i], $filter_array[$j]));

                    array_push($lastname, calculate_completeness('demographics', 'lastname',
$projects[$i], $filter_array[$j]));

                    array_push($dob, calculate_completeness('demographics', 'dob', $projects[$i],
$filter_array[$j]));

                    array_push($age, calculate_completeness('demographics', 'age', $projects[$i],
$filter_array[$j]));

                    array_push($race, calculate_completeness('demographics', 'race', $projects[$i],
$filter_array[$j]));

                    array_push($sex, calculate_completeness('demographics', 'sex', $projects[$i],
$filter_array[$j]));

                    array_push($averages, (($wid[$i] + $hosp_id[$i] + $hosp_doa[$i] + $ssn[$i] +
$firstname[$i] + $lastname[$i] + $dob[$i] + $age[$i] + $race[$i] + $sex[$i])/10));


                    }


                }


                echo "<table><tr style='background-color:#004963;
color:#ffffff'><th>DATABASE</th><th>Warehouse ID</th><th>Hospital ID</th><th>Hospital
DOA</th><th>SSN</th><th>First Name</th><th>Last
```

```
Name</th><th>DOB</th><th>Age</th><th>Race</th><th>Sex</th><th>Overall Completeness</th></tr>";

                // Create rows of completeness table
                for($j = 0; $j < count($wid); $j++)
                {
                        echo "<tr><td>" . $project_labels[$j] . "</td><td>" . $wid[$j] . "%</td><td>" . $hosp_id[$j] . "%</td><td>" . $hosp_doa[$j] . "%</td><td>" . $ssn[$j] . "%</td><td>" . $firstname[$j] . "%</td><td>" . $lastname[$j] . "%</td><td>" . $dob[$j] . "%</td><td>" . $age[$j] . "%</td><td>" . $race[$j] . "%</td><td>" . $sex[$j] . "%</td><td>" . $averages[$j] . "%</td></tr>";
                }

                // Same calculation and logic as above, but for the whole warehouse
                $wid_paw = calculate_completeness('demographics', 'warehouse_id', null, $filter);
                $hosp_id_paw = calculate_completeness('happi_data', 'hospital_id', null, $filter);
                $hosp_doa_paw = calculate_completeness('hospitalization', 'hosp_doa', null, $filter);
                $ssn_paw = calculate_completeness('demographics', 'ssn', null, $filter);
                $firstname_paw = calculate_completeness('demographics', 'firstname', null, $filter);
                $lastname_paw = calculate_completeness('demographics', 'lastname', null, $filter);
                $dob_paw = calculate_completeness('demographics', 'dob', null, $filter);
                $age_paw = calculate_completeness('demographics', 'age', null, $filter);
                $race_paw = calculate_completeness('demographics', 'race', null, $filter);
                $sex_paw = calculate_completeness('demographics', 'sex', null, $filter);

                $average_paw = (($wid_paw + $hosp_id_paw + $hosp_doa_paw + $ssn_paw + $firstname_paw + $lastname_paw + $dob_paw + $age_paw + $race_paw + $sex_paw) / 10);

                // Add last row to the table
                echo "<tr><td>ALL</td><td>" . $wid_paw . "%</td><td>" . $hosp_id_paw . "%</td><td>" . $hosp_doa_paw . "%</td><td>" . $ssn_paw . "%</td><td>" . $firstname_paw .
```

```
"%</td><td>" . $lastname_paw . "%</td><td>" . $dob_paw . "%</td><td>" . $age_paw . "%</td><td>" . $race_paw . "%</td><td>" . $sex_paw . "%</td><td>" . $average_paw . "%</td></tr>";

                echo "</table>";
                echo "<br />";

        }

        if($_GET["table"] == "clinical")
        {
                $projects = array(1, 2, 3, 5); //array for projects
                $project_labels = array("HAPPI 1&2", "HAPPI 2.5", "HAPPI 3", "SHARP");
                //array for corresponding projects
                $wid = array();                                         //initialize completeness arrays, warehosue identification number
                $exam_sbp = array();                                    //systolic blood pressure for patient
                $exam_dbp = array();                                    //diabolic blood pressure for patient
                $exam_temp = array();                                   //exam temperature for patient
                $ct = array();                                          //chest scan for patient
                $cxr = array();                                         //xray for patient
                $outcome_rh30 = array();                                //outcomes in 30 days
                $outcome_rh6mo = array();                               //outcomes in 6 months
                $outcome_rh1year = array();                             //outcomes in 1 year
                $tcs = array();                                         //days to statistical stability
```

```
$averages = array();

$overall_completentess = array(1, 1, 1, 1);




for($i = 0; $i < count($projects); $i++)

{ //push the rounded completeness values into the array for each project


        array_push($wid, calculate_completeness('demographics', 'warehouse_id',
$projects[$i], $filter));

        array_push($exam_sbp,
calculate_completeness('physical_exam_lab_admission', 'exam_sbp', $projects[$i], $filter));

        array_push($exam_dbp,
calculate_completeness('physical_exam_lab_admission', 'exam_dbp', $projects[$i], $filter));

        array_push($exam_temp,
calculate_completeness('physical_exam_lab_admission', 'exam_temp', $projects[$i], $filter));

        array_push($ct, calculate_completeness('radio_findings', 'ct', $projects[$i],
$filter));

        array_push($cxr, calculate_completeness('radio_findings', 'cxr', $projects[$i],
$filter));

        array_push($outcome_rh30, calculate_completeness('clinical_outcomes',
'outcome_rh30', $projects[$i], $filter));

        array_push($outcome_rh6mo, calculate_completeness('clinical_outcomes',
'outcome_rh6mo', $projects[$i], $filter));

        array_push($outcome_rh1year,calculate_completeness('clinical_outcomes',
'outcome_rh1year', $projects[$i], $filter));

        array_push($tcs, calculate_completeness('clinical_course', 'calculated_tcs',
$projects[$i], $filter));

        array_push($averages, (($wid[$i] + $exam_sbp[$i] + $exam_dbp[$i] +
$exam_temp[$i] + $ct[$i] + $cxr[$i] + $outcome_rh30[$i] + $outcome_rh6mo[$i] +
$outcome_rh1year[$i] + $tcs[$i] + $averages[$i]) / 10));

    }
```

```
echo "<table><tr style='background-color:#004963;
color:#ffffff'><th>DATABASE</th><th>Warehouse ID</th><th>Systolic Blood
<br>Pressure</th><th>Diastolic Blood <br>Pressure</th><th>Temperature</th><th>Chest
Scan</th><th>X-ray</th><th>Outcome<br>in 30 days</th><th>Outcome<br>in 6
months</th><th>Outcome<br>in 1 year</th><th>Days to
Clinical<br>Stability</th><th>Overall<br>Completeness</th></tr>";


for($j = 0; $j < count($wid); $j++)

{

    echo "<tr><td>" . $project_labels[$j] . "</td><td>" . $wid[$j] . "%</td><td>" .
$exam_sbp[$j] . "%</td><td>" . $exam_dbp[$j] . "%</td><td>" . $exam_temp[$j] . "%</td><td>" .
$ct[$j] . "%</td><td>" . $cxr[$j] . "%</td><td>" . $outcome_rh30[$j] . "%</td><td>" .
$outcome_rh6mo[$j] . "%</td><td>" . $outcome_rh1year[$j] . "%</td><td>" . $tcs[$j] . "%</td><td>" .
$averages[$j] .  "%</td></tr>";


}


//same calcuation and logic as above, but for the whole warehouse


$wid_paw = calculate_completeness('demographics', 'warehouse_id', null, $filter);

$exam_sbp_paw = calculate_completeness('physical_exam_lab_admission', 'exam_sbp',
null, $filter);

$exam_dbp_paw = calculate_completeness('physical_exam_lab_admission',
'exam_dbp', null, $filter);

$exam_temp_paw = calculate_completeness('physical_exam_lab_admission',
'exam_temp', null, $filter);

$ct_paw = calculate_completeness('radio_findings', 'ct', null, $filter);

$cxr_paw = calculate_competeness('radio_findings', 'cxr', null, $filter);

$outcome_rh30_paw = calculate_completeness('clinical_outcomes', 'outcome_rh30',
null, $filter);

$outcome_rh6mo_paw = calculate_completness('clinical_outcomes', 'outcome_rh6mo',
null, $filter);

$outcome_rh1year_paw = calculate_completeness('clinical_outcomes',
'outcome_rh1year', null, $filter);
```

```
$tcs_paw = calculate_completeness('clinical_course', 'calculated_tcs', null, $filter);

$averages_paw = (($wid_paw + $exam_sbp_paw + $exam_dbp_paw + $exam_temp_paw + $ct_paw + $cxr_paw + $outcome_rh30_paw + $outcome_rh6mo_paw + $outcome_rh1year_paw + $tcs_paw) / 10) . "%";

echo "<tr><td>ALL</td><td>" . $wid_paw . "</td><td>" . $exam_sbp_paw . "</td><td>" . $exam_dbp_paw . "</td><td>" . $exam_temp_paw . "</td><td>" .$ct_paw. "</td><td>" .$cxr_paw. "</td><td>" . $outcome_rh30_paw . "</td><td>" . $outcome_rh6mo_paw . "</td><td>" . $outcome_rh1year_paw . "</td><td>" . $tcs_paw . "</td><td>" . $averages_paw . "</td></tr>";

echo "</table>";

echo "<br />";

echo "<div>   <font size=\"2\">*CAPO has been removed from this table.<br>    SSN, name, birthday, age and race are not elements of the international study.</div>";

}

if($_GET['table'] == 'psi')

{

$projects = array(1, 2, 3, 5); // projects not including CAPO

$project_labels = array("HAPPI 1&2", "HAPPI 2.5", "HAPPI 3", "SHARP");

$wid = array(); // initialize completeness arrays

$hosp_id = array();

$hosp_doa = array();

$ssn = array();

$firstname = array();

$lastname = array();

$dob = array();

$age = array();

$race = array();
```

```
$sex = array();

$overall_completeness = array(1, 1, 1, 1);

$averages = array();


for($i = 0; $i < count($projects); $i++)

{ // push the rounded completeness values into the array for each project

        array_push($wid, (100 * (round((completeness('demographics', 'warehouse_id', $projects[$i], $filter) / completeness('demographics', 'warehouse_id', $projects[$i], $filter)), 3))));

        array_push($hosp_id, (100 * (round((completeness('happi_data', 'hospital_id', $projects[$i], $filter) / completeness('demographics', 'warehouse_id', $projects[$i], $filter)), 3))));

        array_push($hosp_doa, (100 * (round((completeness('hospitalization', 'hosp_doa', $projects[$i], $filter) / completeness('demographics', 'warehouse_id', $projects[$i], $filter)), 3))));

        array_push($ssn, (100 * (round((completeness('demographics', 'ssn', $projects[$i], $filter) / completeness('demographics', 'warehouse_id', $projects[$i], $filter)), 3))));

        array_push($firstname, (100 * (round((completeness('demographics', 'firstname', $projects[$i], $filter) / completeness('demographics', 'warehouse_id', $projects[$i], $filter)), 3))));

        array_push($lastname, (100 * (round((completeness('demographics', 'lastname', $projects[$i], $filter) / completeness('demographics', 'warehouse_id', $projects[$i], $filter)), 3))));

        array_push($dob, (100 * (round((completeness('demographics', 'dob', $projects[$i], $filter) / completeness('demographics', 'warehouse_id', $projects[$i], $filter)), 3))));

        array_push($age, (100 * (round((completeness('demographics', 'age', $projects[$i], $filter) / completeness('demographics', 'warehouse_id', $projects[$i], $filter)), 3))));

        array_push($race, (100 * (round((completeness('demographics', 'race', $projects[$i], $filter) / completeness('demographics', 'warehouse_id', $projects[$i], $filter)), 3))));

        array_push($sex, (100 * (round((completeness('demographics', 'sex', $projects[$i], $filter) / completeness('demographics', 'warehouse_id', $projects[$i], $filter)), 3))));


        array_push($averages, (($wid[$i] + $hosp_id[$i] + $hosp_doa[$i] + $ssn[$i] + $firstname[$i] + $lastname[$i] + $dob[$i] + $age[$i] + $race[$i] + $sex[$i])/10));

}
```

```
echo "<table><tr style='background-color:#004963; color:#ffffff'><th>DATABASE</th><th>Warehouse ID</th><th>Hospital ID</th><th>Hospital DOA</th><th>SSN</th><th>First Name</th><th>Last Name</th><th>DOB</th><th>Age</th><th>Race</th><th>Sex</th><th>Overall Completeness</th></tr>";


// Create rows of completeness table

for($j = 0; $j < count($wid); $j++)

{

        echo "<tr><td>" . $project_labels[$j] . "</td><td>" . $wid[$j] . "%</td><td>" . $hosp_id[$j] . "%</td><td>" . $hosp_doa[$j] . "%</td><td>" . $ssn[$j] . "%</td><td>" . $firstname[$j] . "%</td><td>" . $lastname[$j] . "%</td><td>" . $dob[$j] . "%</td><td>" . $age[$j] . "%</td><td>" . $race[$j] . "%</td><td>" . $sex[$j] . "%</td><td>" . $averages[$j] . "%</td></tr>";

}



// Same calculation and logic as above, but for the whole warehouse

$wid_paw = (100 * (round((completeness('demographics', 'warehouse_id', null, $filter) / completeness('demographics', 'warehouse_id', null, $filter)), 3))) . "%";

$hosp_id_paw = (100 * (round((completeness('happi_data', 'hospital_id', null, $filter) / completeness('happi_data', 'warehouse_id', null, $filter)), 3))) . "%";

$hosp_doa_paw = (100 * (round((completeness('hospitalization', 'hosp_doa', null, $filter) / completeness('hospitalization', 'warehouse_id', null, $filter)), 3))) . "%";

$ssn_paw = (100 * (round((completeness('demographics', 'ssn', null, $filter) / completeness('demographics', 'warehouse_id', null, $filter)), 3))) . "%";

$firstname_paw = (100 * (round((completeness('demographics', 'firstname', null, $filter) / completeness('demographics', 'warehouse_id', null, $filter)), 3))) . "%";

$lastname_paw = (100 * (round((completeness('demographics', 'lastname', null, $filter) / completeness('demographics', 'warehouse_id', null, $filter)), 3))) . "%";

$dob_paw = (100 * (round((completeness('demographics', 'dob', null, $filter) / completeness('demographics', 'warehouse_id', null, $filter)), 3))) . "%";

$age_paw = (100 * (round((completeness('demographics', 'age', null, $filter) / completeness('demographics', 'warehouse_id', null, $filter)), 3))) . "%";

$race_paw = (100 * (round((completeness('demographics', 'race', null, $filter) / completeness('demographics', 'warehouse_id', null, $filter)), 3))) . "%";
```

```
$sex_paw = (100 * (round((completeness('demographics', 'sex', null, $filter) /
completeness('demographics', 'warehouse_id', null, $filter)), 3))) . "%";


$average_paw = (($wid_paw + $hosp_id_paw + $hosp_doa_paw + $ssn_paw + $firstname_paw +
$lastname_paw + $dob_paw + $age_paw + $race_paw + $sex_paw) / 10) . "%";


// Add last row to the table

echo "<tr><td>ALL</td><td>" . $wid_paw . "</td><td>" . $hosp_id_paw . "</td><td>" . $hosp_doa_paw
. "</td><td>" . $ssn_paw . "</td><td>" . $firstname_paw . "</td><td>" . $lastname_paw . "</td><td>" .
$dob_paw . "</td><td>" . $age_paw . "</td><td>" . $race_paw . "</td><td>" . $sex_paw . "</td><td>" .
$average_paw . "</td></tr>";


echo "</table>";

echo "<br />";

}


if($_GET['table'] == 'common')

{


$projects = array(1, 2, 3, 5);
                        //array for the projects

$projects_labels = array("HAPPI 1&2", "HAPPI 2.5", "HAPPI 3", "SHARP");                 //labels
for the projects

$hosp_vent = array();
                                //hospital ventilation

$hx_renalfailure = array();
                                //renal failure

$hx_liver = array();
                                //liver disease

$hx_cirrhosis = array();
                        //cirrhosis

$abx_30 = array();
                                        //antibiotics within 30 days
```

```
$cardio_systemicsteroids = array();
                        //systemic steroids

$cardio_antithrombotic = array();
                        //antithrombotic

$micro_bacter = array();
                                //persistent bacteria

$prevent_smokinghistory = array();
                        //smoking history of patient

$hosp_dischargedate = array();
                        //discharge date at hospital


for($i = 0; $i < count($projects); $i++)

{


array_push($hosp_vent, calculate_competeness(completeness('hospitalization', 'hosp_vent',
$projects[$i], $filter);

array_push($hx_renalfailure, calculate_completeness('patient_history', 'hx_renalfailure', $projects[$i],
$filter);

array_push($hx_liver, clculate_completeness('patient_history', 'hx_liver', $projects[$i], $filter);

array_push($hx_cirrhosis, calculate_completeness('patient_history', 'hx_cirrhosis', $projects[$i], $filter);

array_push($abx_30, calculate_completeness('antimicrobial_therapy', 'abx_30', $projects[$i], $filter);

array_push($cardio_systemicsteroids, calculate_completeness('cardiovascular',
'cardio_systemicsteroids', $projects[$i], $filter);

array_push($cardio_antithrombotic, calculate_completeness('cardiovascular', 'cardio_antithrombotic',
$projects[$i], $filter);

array_push($micro_bacter, calculate_completeness('antimicrobial_therapy', 'micro_bacter',
$projects[$i], $filter);

array_push($prevent_smokinghistory, calculate_completeness('prevention', 'prevent_smokinghistory',
$projects[$i], $filter);

array_push($hosp_dischargedate, calculate_completeness('hospitalization', 'hosp_dischargedate',
$projects[$i], $filter);

array_push(($hosp_vent[$i] + $hx_renal_failure + $hx_liver[$i] + $hx_cirrhosis[$i] + $abx_30[$i] +
$cardio_systemicsteroids[$i] + $cardio_antithrombotic[$i] + $micro_bacter[$i] +
$prevent_smokinghistory[$i] + $hosp_dischargedate[$i]) / 10));
```

```
}


echo "<table><tr style='background-color:#004963; color:#ffffff'><th>Hospital
Ventilation</th><th>Renal Failure</th><th>Liver Disease</th><th>Hospital
DOA</th><th>Cirrhosis</th><th>Antibiotics administered Within 30 Days</th><th>Systemic
Steroids</th><th>Antithrombitc</th><th>Persistent Bacteria</th><th>Smoking History</th><th>Date
of Hospital Discharge</th><th>Average Completeness</th></tr>";


for($j = 0; $j < count($projects); $j++)

{

        echo "<tr><td>" . $hosp_vent[$j] . "</td><td>" . $hx_renalfailure[$j] . "%</td><td>" .
$hx_liver[$j] . "%</td><td>" . $hosp_doa[$j] . "%</td><td>" . $hx_cirrhosis[$j] . "%</td><td>" .
$abx_30[$j] . "%</td><td>" . $cardio_systemicsteroids[$j] . "%</td><td>" . $cardio_antithrombotic[$j] .
"%</td><td>" . $micro_bacter[$j] . "%</td></tr>";

}


$hosp_vent_paw = calculate_completeness('hospitalization', 'hosp_vent', null, $filter);

$hx_renalfailure_paw = calculate_completeness('hospitalization', 'hosp_vent', null, $filter);

$hx_liver_paw = calculate_completeness('hospitalization', 'hosp_vent', null, $filter);

$hosp_doa_paw = calculate_completeness('hospitalization', 'hosp_doa', null, $filter);

$hx_cirrhosis_paw = calculate_completeness('hospitalization', 'hosp_vent', null, $filter);

$abx_30_paw = calculate_completeness('hospitalization', 'hosp_vent', null, $filter);

$cardio_systemicsteroids_paw = calculate_completeness('hospitalization', 'hosp_vent', null, $filter);

$cardio_antithrombotic_paw = calculate_completeness('hospitalization', 'hosp_vent', null, $filter);

$micro_bacter_paw = calculate_completeness('hospitalization', 'hosp_vent', null, $filter);

$prevent_smokinghistory_paw = calculate_completeness('prevention', 'prevent_smokinghistory', null,
$filter);

$hosp_dischargedate_paw = calculate_completeness('hospitalization', 'hosp_dischargedate', null,
$filter);

$average_paw = (($hosp_vent + $hx_renalfailure_paw + $hx_liver_paw + $hosp_doa_paw +
$hx_cirrhosis_paw + $abx_30_paw + $cardio_systemicsteroids_paw + $cardio_antithrombotic_paw +
$micro_bacter_paw + $prevent_smokinghistory_paw + $hosp_dischargedate_paw) / 10));
```

```
echo "<tr><td>ALL</td><td>" . $hosp_vent_paw . "</td><td>" . $hx_renalfailure_paw . "</td><td>" .
$hx_liver_paw . "</td><td>" . $hosp_doa_paw . "</td><td>" . $hx_cirrhosis_paw . "</td><td>" .
$abx_30_paw . "</td><td>" . $cardio_systemicsteroids_paw . "</td><td>" .
$cardio_antithrombotic_paw . "</td><td>" . $micro_bacter_paw . "</td><td>" .
$prevent_smokinghistory_paw . "</td><td>" . $hosp_dischargedate_paw . "</td><td>" .$average_paw.
"</td></tr>";


echo "</table>";

echo "<br />";


}


}


catch(Exception $e)

{

        echo "Error: " . $e->getMessage();

}


?>
```