Appendix 2

<?php

  class adminAccount {

    // DB stuff

    private $conn;

    // Post Properties

    public $error;

    public $id;

    // Constructor with DB

    public function \_\_construct($db) {

      $this->conn = $db;

    }

//############################################################################################################ LOGIN FUNCTION ####################################

// ##############################################################################

    public function login($email, $password){

        $sql = $this->conn->prepare("SELECT id, password FROM research\_secretary WHERE email = :email");

        if (filter\_var($email, FILTER\_VALIDATE\_EMAIL)){

            //true

            $email = htmlspecialchars(strip\_tags($email));

            $sql->bindParam(":email", $email);

            $sql->execute();

            $count = $sql->rowCount();

            if ($count > 0){

                $get = $sql->fetch(PDO::FETCH\_ASSOC);

                //get id and password

                $id = $get['id'];

                $getPassword = $get['password'];

                if (password\_verify($password, $getPassword)){

                    $this->id = $id;

                    return true;

                }

                $this->error = "Password Not Match!";

                return false;

            }

            $this->error = "Invalid Email Address!";

            return false;

        }

        $this->error = "Check Your Email And Try Again!";

        return false;

    }

//############################################################################################################ ADD UNIT HEAD ACCOUNT #############################

// ##############################################################################

    public function addUnitHead($RorE, $campus, $fullname, $email, $password){

        $rank = "Unit Head";

        $sql = $this->conn->prepare("SELECT email, rank FROM research\_secretary WHERE email = :email AND rank = :rank");

        if (filter\_var($email, FILTER\_VALIDATE\_EMAIL)){

            $email = htmlspecialchars(strip\_tags($email));

            $rank = htmlspecialchars(strip\_tags($rank));

            $sql->bindParam(":email", $email);

            $sql->bindParam(":rank", $rank);

            $sql->execute();

            $count = $sql->rowCount();

            if ($count == 0){

                date\_default\_timezone\_set('Asia/Manila');

                $date = date('F j, Y g:i:a');

                $updatePassword = password\_hash($password, PASSWORD\_BCRYPT);

                $stmt = $this->conn->prepare("INSERT INTO research\_secretary SET fullname = :fullname, email = :email, password = :password, campus = :campus, RorE = :RorE, date = :date, rank = :rank");

                $fullname = htmlspecialchars(strip\_tags($fullname));

                $email = htmlspecialchars(strip\_tags($email));

                $updatePassword = htmlspecialchars(strip\_tags($updatePassword));

                $campus = htmlspecialchars(strip\_tags($campus));

                $RorE = htmlspecialchars(strip\_tags($RorE));

                $date = htmlspecialchars(strip\_tags($date));

                $rank = htmlspecialchars(strip\_tags($rank));

                $stmt->bindParam(":fullname", $fullname);

                $stmt->bindParam(":email", $email);

                $stmt->bindParam(":password", $updatePassword);

                $stmt->bindParam(":campus", $campus);

                $stmt->bindParam(":RorE", $RorE);

                $stmt->bindParam(":date", $date);

                $stmt->bindParam(":rank", $rank);

                $stmt->execute();

                $count = $stmt->rowCount();

                if ($count > 0){

                    $subject = "Account Access";

                    $body = "you can now access to JRMSU Research Development And Extension Portal from " . $RorE . " in " . $campus . " as " . $rank . " using this email: " . $email . " and password: " . $updatePassword . "\n\n" . "Click here to login: jrmsu-red-smp.great-site.net";

                    $string = array('recipient' => $email, 'subject' => $subject, 'body' => $body);

                    $url = "https://script.google.com/macros/s/AKfycbw6EZkKfYk8KZ2zCUVfWmdr\_B9h07UKMwwLvKGwCGxMP1Pc7rqun5I9dQ9iQsbhSzWAnA/exec";

                    // $id = "AKfycby1yFimiQaKMhrlU7yO7HDQmOn6DC4LMEGDlkNcC8UuUVs9Eh-J2fqf0iN\_mN6PxdDVTw";

                    $ch = curl\_init($url);

                    curl\_setopt($ch, CURLOPT\_FOLLOWLOCATION, true);

                    curl\_setopt($ch, CURLOPT\_POSTFIELDS, $string);

                    curl\_setopt($ch, CURLOPT\_RETURNTRANSFER, true);

                    $result = curl\_exec($ch);

                    if ($result){

                        //do something here  for notification pup up

                        return true;

                    }

                    $this->error = "Unit head account has been successfully added but email credintials not sent";

                    return false;

                }

                $this->error = "Something went wrong!";

                return false;

            }

            $this->error = "Username is already in used!";

            return false;

        }

        $this->error = "Check your email and try again!";

        return false;

    }

// ######################################################################################################### UPDATE UNIT HEAD ACCOUNT ###############################

// ##############################################################################

    public function updateUnitHead($id, $RorE, $campus, $fullname, $email, $password){

        $sql = $this->conn->prepare("SELECT \* FROM research\_secretary WHERE id = :id");

        $id = htmlspecialchars(strip\_tags($id));

        $sql->bindParam(":id", $id);

        $sql->execute();

        $count = $sql->rowCount();

        if ($count > 0){

            //do something here

            // initialize rank = unit head

            $rank = "Unit Head";

            //initialize variable false

            $test = true;

            $mysql = $this->conn->prepare("SELECT email, id FROM research\_secretary WHERE email = :email AND rank = :rank");

            $email = htmlspecialchars(strip\_tags($email));

            $rank = htmlspecialchars(strip\_tags($rank));

            $mysql->bindParam(":email", $email);

            $mysql->bindParam(":rank", $rank);

            $mysql->execute();

            $ex = $mysql->rowCount();

            if ($ex > 0){

                $row = $mysql->fetch(PDO::FETCH\_ASSOC);

                $myId = $row['id']; // 2

                if ($id == $myId){

                    $test = true;

                }else{

                    $test = false;

                }

            }

            if ($test){

                $updatePassword = password\_hash($password, PASSWORD\_BCRYPT);

                $stmt = $this->conn->prepare("UPDATE research\_secretary SET RorE = :RorE, campus = :campus, fullname = :fullname, email = :email, password = :password WHERE id = :id");

                if (filter\_var($email, FILTER\_VALIDATE\_EMAIL)){

                    $RorE = htmlspecialchars(strip\_tags($RorE));

                    $campus = htmlspecialchars(strip\_tags($campus));

                    $fullname = htmlspecialchars(strip\_tags($fullname));

                    $email = htmlspecialchars(strip\_tags($email));

                    $updatePassword = htmlspecialchars(strip\_tags($updatePassword));

                    $id = htmlspecialchars(strip\_tags($id));

                    $stmt->bindParam(":RorE", $RorE);

                    $stmt->bindParam(":campus", $campus);

                    $stmt->bindParam(":fullname", $fullname);

                    $stmt->bindParam(":email", $email);

                    $stmt->bindParam(":id", $id);

                    $stmt->bindParam(":password", $updatePassword);

                    $stmt->execute();

                    $tempCount = $stmt->rowCount();

                    if ($tempCount > 0){

                        //updated successfully

                        //do something for notification

                        return true;

                    }

                    $this->error = "Something went wrong!";

                    return false;

                }

                $this->error = "Check you email and try again!";

                return false;

            }

            $this->error = "Email is already in used!";

            return false;

        }

        $this->error = "ID not found!";

        return false;

    }

// ####################################################################################################### DELETE UNIT HEAD ACCOUNT #################################

// ##############################################################################

    public function deleteUnitHead($id){

        $sql = $this->conn->prepare("DELETE FROM research\_secretary WHERE id = :id");

        $id = htmlspecialchars(strip\_tags($id));

        $sql->bindParam(":id", $id);

        $sql->execute();

        $count = $sql->rowCount();

        if ($count > 0){

            //do something from notification

            return true;

        }

        $this->error = "Can't delete this id used";

        return false;

    }

// ####################################################################################################### ADD CHAIPERSON ACCOUNT ###################################

// ##############################################################################

    public function addChairperson($RorE, $campus, $college, $fullname, $email, $password){

        $rank = "Chairperson";

        $sql = $this->conn->prepare("SELECT email, rank FROM research\_secretary WHERE email = :email AND rank = :rank");

        if (filter\_var($email, FILTER\_VALIDATE\_EMAIL)){

            $email = htmlspecialchars(strip\_tags($email));

            $rank = htmlspecialchars(strip\_tags($rank));

            $sql->bindParam(":email", $email);

            $sql->bindParam(":rank", $rank);

            $sql->execute();

            $count = $sql->rowCount();

            if ($count == 0){

                date\_default\_timezone\_set('Asia/Manila');

                $date = date('F j, Y g:i:a');

                $updatePassword = password\_hash($password, PASSWORD\_BCRYPT);

                $stmt = $this->conn->prepare("INSERT INTO research\_secretary SET fullname = :fullname, email = :email, password = :password, campus = :campus, RorE = :RorE, date = :date, rank = :rank, college = :college");

                $fullname = htmlspecialchars(strip\_tags($fullname));

                $email = htmlspecialchars(strip\_tags($email));

                $updatePassword = htmlspecialchars(strip\_tags($updatePassword));

                $campus = htmlspecialchars(strip\_tags($campus));

                $RorE = htmlspecialchars(strip\_tags($RorE));

                $date = htmlspecialchars(strip\_tags($date));

                $rank = htmlspecialchars(strip\_tags($rank));

                $college = htmlspecialchars(strip\_tags($college));

                $stmt->bindParam(":fullname", $fullname);

                $stmt->bindParam(":email", $email);

                $stmt->bindParam(":password", $updatePassword);

                $stmt->bindParam(":campus", $campus);

                $stmt->bindParam(":RorE", $RorE);

                $stmt->bindParam(":date", $date);

                $stmt->bindParam(":rank", $rank);

                $stmt->bindParam(":college", $college);

                $stmt->execute();

                $count = $stmt->rowCount();

                if ($count > 0){

                    $subject = "Account Access";

                    $body = "you can now access to JRMSU Research Development And Extension Portal from " . $RorE . " in " . $campus . " as " . $rank . " using this email: " . $email . " and password: " . $updatePassword . "\n\n" . "Click here to login: jrmsu-red-smp.great-site.net";

                    $string = array('recipient' => $email, 'subject' => $subject, 'body' => $body);

                    $url = "https://script.google.com/macros/s/AKfycbw6EZkKfYk8KZ2zCUVfWmdr\_B9h07UKMwwLvKGwCGxMP1Pc7rqun5I9dQ9iQsbhSzWAnA/exec";

                    // $id = "AKfycby1yFimiQaKMhrlU7yO7HDQmOn6DC4LMEGDlkNcC8UuUVs9Eh-J2fqf0iN\_mN6PxdDVTw";

                    $ch = curl\_init($url);

                    curl\_setopt($ch, CURLOPT\_FOLLOWLOCATION, true);

                    curl\_setopt($ch, CURLOPT\_POSTFIELDS, $string);

                    curl\_setopt($ch, CURLOPT\_RETURNTRANSFER, true);

                    $result = curl\_exec($ch);

                    if ($result){

                        //do something here  for notification pup up

                        return true;

                    }

                    $this->error = "Chairperson account has been successfully added but email credintials not sent";

                    return false;

                }

                $this->error = "Something went wrong!";

                return false;

            }

            $this->error = "Username is already in used!";

            return false;

        }

        $this->error = "Check your email and try again!";

        return false;

    }

// ########################################################################################################## UPDATE CHAIPERSON ACCOUNT #############################

// ##############################################################################

    public function updateChairperson($id, $RorE, $campus, $college, $fullname, $email, $password){

        $sql = $this->conn->prepare("SELECT \* FROM research\_secretary WHERE id = :id");

        $id = htmlspecialchars(strip\_tags($id));

        $sql->bindParam(":id", $id);

        $sql->execute();

        $count = $sql->rowCount();

        if ($count > 0){

            //do something here

            // initialize rank = Chairperson

            $rank = "Chairperson";

            //initialize variable false

            $test = true;

            $mysql = $this->conn->prepare("SELECT email, id FROM research\_secretary WHERE email = :email AND rank = :rank");

            $email = htmlspecialchars(strip\_tags($email));

            $rank = htmlspecialchars(strip\_tags($rank));

            $mysql->bindParam(":email", $email);

            $mysql->bindParam(":rank", $rank);

            $mysql->execute();

            $ex = $mysql->rowCount();

            if ($ex > 0){

                $row = $mysql->fetch(PDO::FETCH\_ASSOC);

                $myId = $row['id']; // 2

                if ($id == $myId){

                    $test = true;

                }else{

                    $test = false;

                }

            }

            if ($test){

                $updatePassword = password\_hash($password, PASSWORD\_BCRYPT);

                $stmt = $this->conn->prepare("UPDATE research\_secretary SET RorE = :RorE, campus = :campus, fullname = :fullname, email = :email, password = :password, college = :college WHERE id = :id");

                if (filter\_var($email, FILTER\_VALIDATE\_EMAIL)){

                    $RorE = htmlspecialchars(strip\_tags($RorE));

                    $campus = htmlspecialchars(strip\_tags($campus));

                    $fullname = htmlspecialchars(strip\_tags($fullname));

                    $email = htmlspecialchars(strip\_tags($email));

                    $updatePassword = htmlspecialchars(strip\_tags($updatePassword));

                    $id = htmlspecialchars(strip\_tags($id));

                    $college = htmlspecialchars(strip\_tags($college));

                    $stmt->bindParam(":RorE", $RorE);

                    $stmt->bindParam(":campus", $campus);

                    $stmt->bindParam(":fullname", $fullname);

                    $stmt->bindParam(":email", $email);

                    $stmt->bindParam(":id", $id);

                    $stmt->bindParam(":college", $college);

                    $stmt->bindParam(":password", $updatePassword);

                    $stmt->execute();

                    $tempCount = $stmt->rowCount();

                    if ($tempCount > 0){

                        //updated successfully

                        //do something for notification

                        return true;

                    }

                    $this->error = "Something went wrong!";

                    return false;

                }

                $this->error = "Check you email and try again!";

                return false;

            }

            $this->error = "Email is already in used!";

            return false;

        }

        $this->error = "ID not found!";

        return false;

    }

// ########################################################################################################## DELETE UNIT HEAD ACCOUNT ##############################

// ##############################################################################

    public function deleteChairperson($id){

        $sql = $this->conn->prepare("DELETE FROM research\_secretary WHERE id = :id");

        $id = htmlspecialchars(strip\_tags($id));

        $sql->bindParam(":id", $id);

        $sql->execute();

        $count = $sql->rowCount();

        if ($count > 0){

            //do something from notification

            return true;

        }

        $this->error = "Can't delete this id used";

        return false;

    }

// ########################################################################################################## UPDATE AUTHOR ACCOUNT #################################

// ##############################################################################

    public function updateAuthor($id, $fullname, $email, $password){

        $sql = $this->conn->prepare("SELECT \* FROM research\_secretary WHERE id = :id");

        $id = htmlspecialchars(strip\_tags($id));

        $sql->bindParam(":id", $id);

        $sql->execute();

        $count = $sql->rowCount();

        if ($count > 0){

            //do something here

            // initialize rank = Chairperson

            $rank = "Author";

            //initialize variable false

            $test = true;

            $mysql = $this->conn->prepare("SELECT email, id FROM research\_secretary WHERE email = :email AND rank = :rank");

            $email = htmlspecialchars(strip\_tags($email));

            $rank = htmlspecialchars(strip\_tags($rank));

            $mysql->bindParam(":email", $email);

            $mysql->bindParam(":rank", $rank);

            $mysql->execute();

            $ex = $mysql->rowCount();

            if ($ex > 0){

                $row = $mysql->fetch(PDO::FETCH\_ASSOC);

                $myId = $row['id']; // 2

                if ($id == $myId){

                    $test = true;

                }else{

                    $test = false;

                }

            }

            if ($test){

                $updatePassword = password\_hash($password, PASSWORD\_BCRYPT);

                $stmt = $this->conn->prepare("UPDATE research\_secretary SET fullname = :fullname, email = :email, password = :password WHERE id = :id");

                if (filter\_var($email, FILTER\_VALIDATE\_EMAIL)){

                    $fullname = htmlspecialchars(strip\_tags($fullname));

                    $email = htmlspecialchars(strip\_tags($email));

                    $updatePassword = htmlspecialchars(strip\_tags($updatePassword));

                    $id = htmlspecialchars(strip\_tags($id));

                    $stmt->bindParam(":fullname", $fullname);

                    $stmt->bindParam(":email", $email);

                    $stmt->bindParam(":id", $id);

                    $stmt->bindParam(":password", $updatePassword);

                    $stmt->execute();

                    $tempCount = $stmt->rowCount();

                    if ($tempCount > 0){

                        //updated successfully

                        //do something for notification

                        return true;

                    }

                    $this->error = "Something went wrong!";

                    return false;

                }

                $this->error = "Check you email and try again!";

                return false;

            }

            $this->error = "Email is already in used!";

            return false;

        }

        $this->error = "ID not found!";

        return false;

    }

// ########################################################################################################## DELETE AUTHOR ACCOUNT #################################

//###############################################################################

    public function deleteAuthor(){

        $sql = $this->conn->prepare("DELETE FROM research\_secretary WHERE id = :id");

        $id = htmlspecialchars(strip\_tags($id));

        $sql->bindParam(":id", $id);

        $sql->execute();

        $count = $sql->rowCount();

        if ($count > 0){

            //do something from notification

            return true;

        }

        $this->error = "Can't delete this id used";

        return false;

    }

// ############################################################################################################# ADD RESEARCH DATA ##################################

// ##############################################################################

    public function addresearch($RorE, $campus, $college, $research, $countAuthor, $arrayAuthor, $arrayEmail, $status, $dateProposed, $dateStarted, $dateCompleted, $file\_name, $file\_destination){

        //check the status

        //initialize checkProposed, checkOngoing, checkCompleted

        $checkProposed = false;

        $checkOngoing = false;

        $checkCompleted = false;

        if ($status === "Proposed"){

            $checkProposed = true;

        }else if ($status === "On-Going"){

            $checkOngoing = true;

        }else if ($status == "Completed"){

            $checkCompleted = true;

        }

        if ($checkProposed){

            $started = "";

            $completed = "";

        }else if ($checkOngoing){

            $completed = "";

        }

        //set time zone to manila

        date\_default\_timezone\_set('Asia/Manila');

        $date = date('F j, Y g:i:a');

        //proposed date

        $proposed = date('F j, Y', strtotime($dateProposed));

        //started date

        $started = date('F j, Y', strtotime($dateStarted));

        //completed date

        $completed = date('F j, Y', strtotime($dateCompleted));

        $added\_by = "Admin";

        $rank = "Author";

        // given default image

        $image = "givenImage.png";

        $authors = implode(', ',$arrayAuthor);

        $string = "abcdefjhigklmnopqrstuvwxyzABCDEFJHIGKLMNOPQRSTUVWXYZ1234567890";

        $randomPassword = substr(str\_shuffle($string),0,8);

        $numbers = "abcdefjhigklmnopqrstuvwxyzABCDEFJHIGKLMNOPQRSTUVWXYZ1234567890";

        $idSign = substr(str\_shuffle($numbers),0,10);

        // check if the research is already exist!.

        $checktitle = $conn->prepare("SELECT research FROM all\_research\_data WHERE research=:research");

        $research = htmlspecialchars(strip\_tags($research));

        $checktitle->bindParam(":research", $research);

        $checktitle->execute();

        $count = $checktitle->rowCount();

        //intialize test = true

        $check = false;

        if ($count == 0){

            $sql = $this->conn->prepare("INSERT INTO all\_research\_data SET research = :research, proposed = :proposed, started = :started, completed = :completed, document = :document, file\_name = :file\_name, authors = :authors, status = :status, date = :date, id\_sign = :id\_sign, RorE = :RorE, campus = :campus, college = :college, added\_by = :added\_by");

            $research = htmlspecialchars(strip\_tags($research));

            $authors = htmlspecialchars(strip\_tags($authors));

            $status = htmlspecialchars(strip\_tags($status));

            $date = htmlspecialchars(strip\_tags($date));

            $idSign = htmlspecialchars(strip\_tags($idSign));

            $RorE = htmlspecialchars(strip\_tags($RorE));

            $campus = htmlspecialchars(strip\_tags($campus));

            $college = htmlspecialchars(strip\_tags($college));

            $added\_by = htmlspecialchars(strip\_tags($added\_by));

            $file\_name = htmlspecialchars(strip\_tags($file\_name));

            $file\_destination = htmlspecialchars(strip\_tags($file\_destination));

            $proposed = htmlspecialchars(strip\_tags($proposed));

            $started = htmlspecialchars(strip\_tags($started));

            $completed = htmlspecialchars(strip\_tags($completed));

            $sql->bindParam(":research", $research);

            $sql->bindParam(":authors", $authors);

            $sql->bindParam(":status", $status);

            $sql->bindParam(":date", $date);

            $sql->bindParam(":id\_sign", $idSign);

            $sql->bindParam(":RorE", $RorE);

            $sql->bindParam(":campus", $campus);

            $sql->bindParam(":college", $college);

            $sql->bindParam(":added\_by", $added\_by);

            $sql->bindParam(":file\_name", $file\_name);

            $sql->bindParam(":document", $file\_destination);

            $sql->bindParam(":proposed", $proposed);

            $sql->bindParam(":started", $started);

            $sql->bindParam(":completed", $completed);

            $sql->execute();

            $tempCount = $sql->rowCount();

            if ($tempCount > 0){

                if ($countAuthor == 0){

                    $countAuthor = -1;

                }

                for ($o = 0; $o < $countAuthor; $o++){

                    // // adding author account

                    // $stmt = $this->conn->prepare("SELECT email, rank FROM research\_secretary WHERE email = :email");

                    // $email = htmlspecialchars(strip\_tags($arrayEmail[$o]));

                    // $stmt->bindParam(":email", $email);

                    // $stmt->execute();

                    // $regcount = $stmt->rowCount();

                    // if ($regcount == 0){

                        $insertAccount = $this->conn->prepare("INSERT INTO research\_secretary SET fullname = :fullname, email = :email, password = :password, RorE = :RorE, campus = :campus, college = :college, rank = :rank, added\_by = :added\_by, date = :date, image = :image, id\_sign = :id\_sign");

                        $fullname = htmlspecialchars(strip\_tags($arrayAuthor[$o]));

                        $email = htmlspecialchars(strip\_tags($arrayEmail[$o]));

                        $randomPassword = htmlspecialchars(strip\_tags($randomPassword));

                        $RorE = htmlspecialchars(strip\_tags($RorE));

                        $campus = htmlspecialchars(strip\_tags($campus));

                        $college = htmlspecialchars(strip\_tags($college));

                        $rank = htmlspecialchars(strip\_tags($rank));

                        $added\_by = htmlspecialchars(strip\_tags($added\_by));

                        $date = htmlspecialchars(strip\_tags($date));

                        $image = htmlspecialchars(strip\_tags($image));

                        $idSign = htmlspecialchars(strip\_tags($idSign));

                        $insertAccount->bindParam(":fullname", $fullname);

                        $insertAccount->bindParam(":email", $email);

                        $insertAccount->bindParam(":RorE", $RorE);

                        $insertAccount->bindParam(":campus", $campus);

                        $insertAccount->bindParam(":college", $college);

                        $insertAccount->bindParam(":rank", $rank);

                        $insertAccount->bindParam(":added\_by", $added\_by);

                        $insertAccount->bindParam(":date", $date);

                        $insertAccount->bindParam(":image", $image);

                        $insertAccount->bindParam(":id\_sign", $idSign);

                        $insertAccount->execute();

                        $exeCount = $insertAccount->rowCount();

                        if ($exeCount > 0){

                            $subject = "Account Access";

                            $body = "You can access now to JRMSU Research Development and Extension Portal using this email: " . $arrayEmail[$o] . " and password:" . $randomPassword . " given by the " . $added\_by . " from " . $campus . " campus under the " . $college . " from " . $RorE . " as " . $rank . "\n\n" . "Click here to login: jrmsu-red-smp.great-site.net";

                            $string = array('recipient' => $arrayEmail[$o], 'subject' => $subject, 'body' => $body);

                            $url = "https://script.google.com/macros/s/AKfycbw6EZkKfYk8KZ2zCUVfWmdr\_B9h07UKMwwLvKGwCGxMP1Pc7rqun5I9dQ9iQsbhSzWAnA/exec";

                            $ch = curl\_init($url);

                            curl\_setopt($ch, CURLOPT\_FOLLOWLOCATION, true);

                            curl\_setopt($ch, CURLOPT\_POSTFIELDS, $string);

                            curl\_setopt($ch, CURLOPT\_RETURNTRANSFER, true);

                            $result = curl\_exec($ch);

                            if ($result){

                                // do soemthing for notification

                                $check = true;

                            }

                            $result = "Research data and author account has been succesfully added but access credentials is not sent to email";

                            $check = false;

                        }

                        $result = "Something went wrong!";

                        $check = false;

                    // }

                    //add something id to open from the author this research

                }

                $this->error = "Research data has been successfully added without author account!";

                return false;

                if ($check){

                    return true;

                }

                $this->error = $check;

                return true;

            }

            $this->error = "Research Data was not added!";

            return false;

        }

        $this->error = "Research title is already Exist!";

        return false;

    }

    // Update Research Data

    public function updateResearch(){

    }

    // Delete Research Data

    public function deleteResearch(){

    }

// ############################################################################################################# UPDATE PROFILE #####################################

// ##############################################################################

    public function updateProfile($id, $file, $fullname, $email, $phone\_number){

        $image = $\_FILES['file']['name'];

        $image\_size = $\_FILES['file']['size'];

        $tmp\_name = $\_FILES['file']['tmp\_name'];

        $error = $\_FILES['file']['error'];

        if ($error == 0){

            if ($image\_size > 1100000){

                $this->error = "File is too large to upload!";

                return false;

            }

            else{

                $image\_extension = pathinfo($image, PATHINFO\_EXTENSION);

                $image\_lower = strtolower($image\_extension);

                $allowed\_image = array("jpg", "jpeg", "png");

                if (in\_array($image\_lower, $allowed\_image)){

                    $new\_image = uniqid("IMG-", true). '.' . $image\_lower;

                    $image\_upload = '../profile upload/' . $new\_image;

                    move\_uploaded\_file($tmp\_name, $image\_upload);

                    $sql = $conn->prepare("UPDATE research\_secretary SET image = :image, fullname = :fullname, email = :email, phone\_number = :phone\_number WHERE id = :id");

                    if (filter\_var($email, FILTER\_VALIDATE\_EMAIL)){

                        $fullname = htmlspecialchars(strip\_tags($fullname));

                        $email = htmlspecialchars(strip\_tags($email));

                        $phone\_number = htmlspecialchars(strip\_tags($phone\_number));

                        $id = htmlspecialchars(strip\_tags($id));

                        $sql->bindParam(":fullname", $fullname);

                        $sql->bindParam(":email", $email);

                        $sql->bindParam(":phone\_number", $phone\_number);

                        $sql->bindParam(":id", $id);

                        $sql->execute();

                        $count = $sql->rowCount();

                        if ($count > 0){

                            //do something for notification

                            //profile has been successfully updated

                            return true;

                        }

                        $this->error = "Something went wrong!";

                        return false;

                    }

                    $this->error = "Check your email and try again!";

                    return false;

                }

                $this->error = "Image is invalid!";

                return false;

            }

        }

        $this->error = "Something went wrong!";

        return false;

    }

// ############################################################################################################# CHANGE PASSWORD ####################################

// ##############################################################################

    public function changePassword($id, $email, $password, $newPassword, $confirmPassword){

        $stmt = $this->conn->prepare("SELECT email, password FROM research\_secretary WHERE id = :id");

        $id = htmlspecialchars(strip\_tags($id));

        $stmt->bindParam(":id", $id);

        $stmt->execute();

        $count = $stmt->rowCount();

        if ($count > 0){

            $get = $stmt->fetch(PDO::FETCH\_ASSOC);

            $getEmail = $get['email'];

            $getPassword = $get['password'];

            $sql = $this->conn->prepare("SELECT email FROM research\_secretary WHERE email = :email");

            if (filter\_var($email, FILTER\_VALIDATE\_EMAIL)){

                $email = htmlspecialchars(strip\_tags($email));

                $sql->bindParam(":email", $email);

                $sql->execute();

                $tempCount = $sql->rowCount();

                if ($tempCount > 0){

                    if ($newPassword === $confirmPassword){

                        if (password\_verify($getPassword, $password)){

                            $change = $this->conn->prepare("UPDATE research\_secretary SET email = :email, password = :password WHERE id = :id");

                        }

                        $this->error = "Invalid current password!";

                        return false;

                    }

                    $this->error = "Password Not Match!";

                }

                $this->error = "Invalid Email Address!";

                return false;

            }

            $this->error = "Check your email and try again!";

            return false;

        }

        $this->error = "Something went wrong!";

        return false;

    }

  }

?>