Министерство науки и высшего образования Российской Федерации

федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет инфокоммуникационных технологий

ЛАБОРАТОРНАЯ РАБОТА №4 «Запросы на выборку и модификацию данных, представления и индексы в PostgreSQL»

Выполнил:
студент : Аль-Мошки Исмаил Абдулвахаб
группа: К32401

Проверили:
Говорова Марина Михайловна

Санкт-Петербург
2023

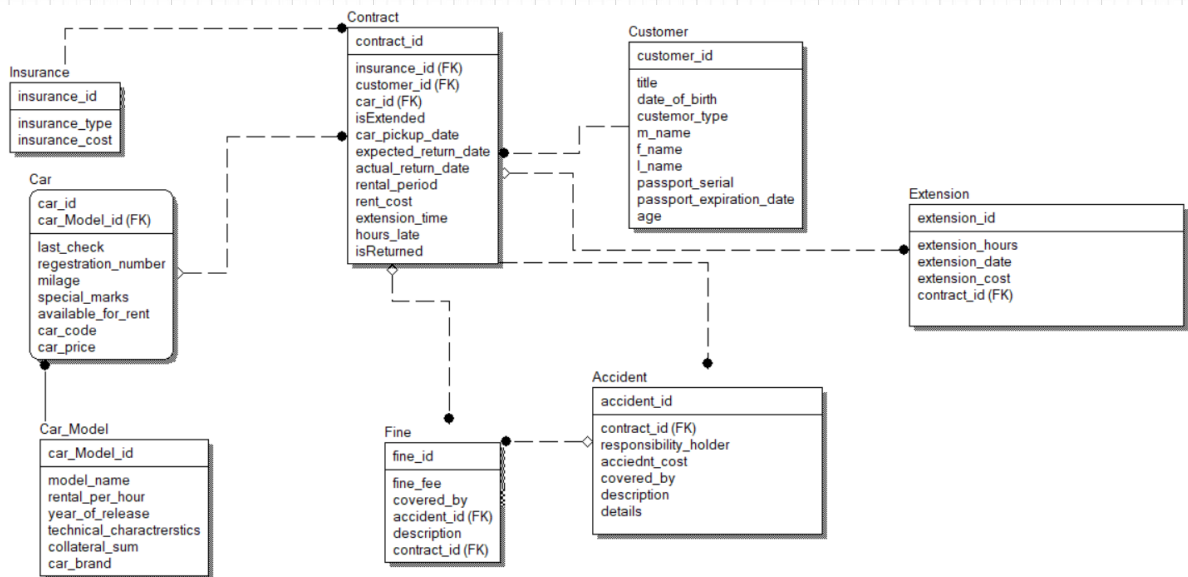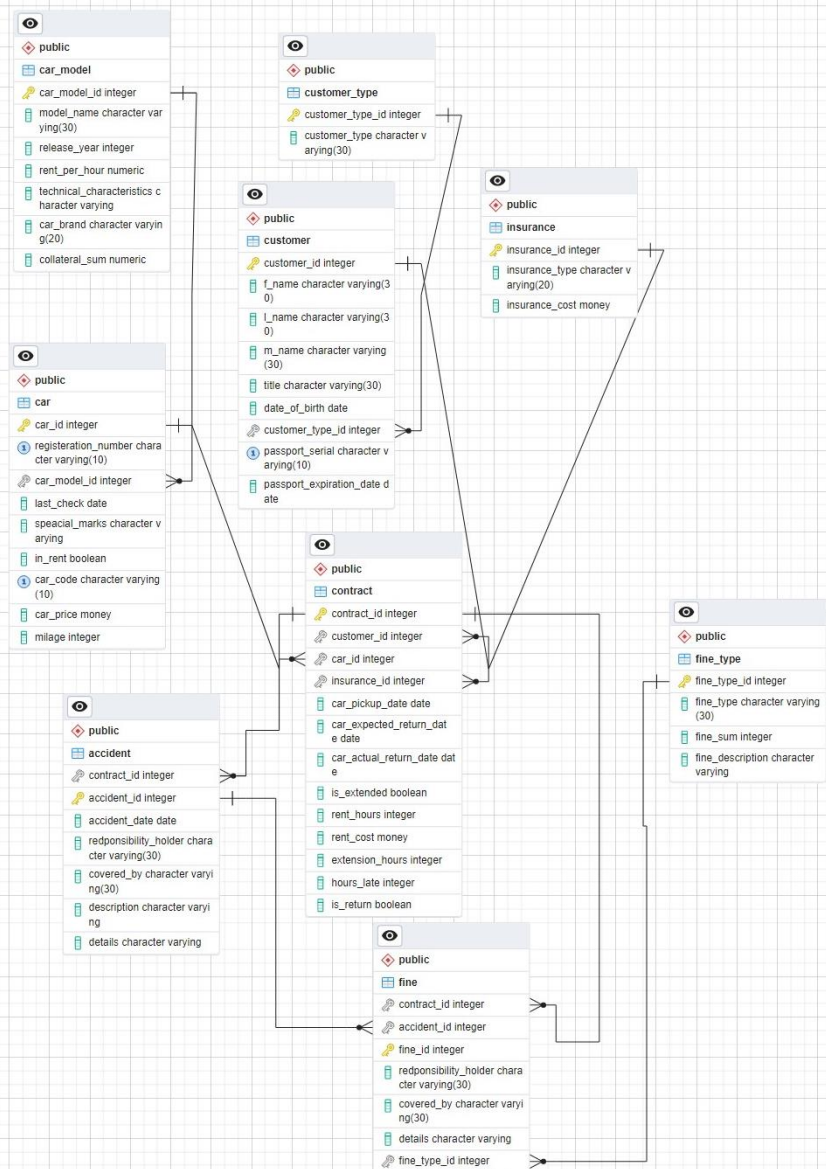**Цель работы:** овладеть практическими навыками разработки форм для вставки, выборки и редактирования данных.

Оборудование: компьютерный класс.

Программное обеспечение: среда разработки XAMPP 8 (PHP 8) или аналогичная, СУБД PostgreSQL 1X, pgadmin 4.

Практическое задание: Минимальный уровень: создать скрипты для простой вставки, редактирования, добавления и удаления данных в таблице.

1. Минимальный уровень: создать скрипты для простой вставки, редактирования, добавления и удаления данных в таблице.
2. Базовый уровень: создать формы для вставки, редактирования, удаления и выборки данных базовыми средствами.
3. Повышенный уровень: создать формы для вставки, редактирования, удаления и выборки данных с использованием библиотеки PDO.

Схема базы данных:

car_model (public)
- car_model_id integer
- model_name character varying(30)
- release_year integer
- rent_per_hour numeric
- technical_characteristics character varying
- car_brand character varying(20)
- collateral_sum numeric

customer_type (public)
- customer_type_id integer
- customer_type character varying(30)

customer (public)
- customer_id integer
- f_name character varying(30)
- l_name character varying(30)
- m_name character varying(30)
- title character varying(30)
- date_of_birth date
- customer_type_id integer
- passport_serial character varying(10)
- passport_expiration_date date

insurance (public)
- insurance_id integer
- insurance_type character varying(20)
- insurance_cost money

car (public)
- car_id integer
- registration_number character varying(10)
- car_model_id integer
- last_check date
- speacial_marks character varying
- in_rent boolean
- car_code character varying(10)
- car_price money
- milage integer

contract (public)
- contract_id integer
- customer_id integer
- car_id integer
- insurance_id integer
- car_pickup_date date
- car_expected_return_date date
- car_actual_return_date date
- is_extended boolean
- rent_hours integer
- rent_cost money
- extension_hours integer
- hours_late integer
- is_return boolean

fine_type (public)
- fine_type_id integer
- fine_type character varying(30)
- fine_sum integer
- fine_description character varying

accident (public)
- contract_id integer
- accident_id integer
- accident_date date
- redponsibility_holder character varying(30)
- covered_by character varying(30)
- description character varying
- details character varying

fine (public)
- contract_id integer
- accident_id integer
- fine_id integer
- redponsibility_holder character varying(30)
- covered_by character varying(30)
- details character varying
- fine_type_id integer

Insurance
- insurance_id
- insurance_type
- insurance_cost

Contract
- contract_id
- insurance_id (FK)
- customer_id (FK)
- car_id (FK)
- isExtended
- car_pickup_date
- expected_return_date
- actual_return_date
- rental_period
- rent_cost
- extension_time
- hours_late
- isReturned

Customer
- customer_id
- title
- date_of_birth
- custemor_type
- m_name
- f_name
- l_name
- passport_serial
- passport_expiration_date
- age

Car
- car_id
- car_Model_id (FK)
- last_check
- registration_number
- milage
- special_marks
- available_for_rent
- car_code
- car_price

Extension
- extension_id
- extension_hours
- extension_date
- extension_cost
- contract_id (FK)

Car_Model
- car_Model_id
- model_name
- rental_per_hour
- year_of_release
- technical_charactrerstics
- collateral_sum
- car_brand

Fine
- fine_id
- fine_fee
- covered_by
- accident_id (FK)
- description
- contract_id (FK)

Accident
- accident_id
- contract_id (FK)
- responsibility_holder
- acciednt_cost
- covered_by
- description
- details

Листинги скриптов для работы с БД и скриншоты результатов:

Setting.php : the connection code to the database:

```php
<?php

// $db = pg_connect("host=localhost dbname='Car Rental'
// user=postgres password ='Abdallah9090' ");

$pdo = new PDO("pgsql:host=localhost ;dbname='Car Rental';
user=postgres;password='Abdallah9090' ");
$pdo->setAttribute(PDO::ATTR_ERRMODE,PDO::ERRMODE_EXCEPTION);
?>
```

Index.php: the reading page:

```php
<?php
require_once('setting.php');
$search =$_GET["search"]??'';
if($search){
$search_value = $_GET['search'];
$statement = $pdo -> prepare("SELECT * FROM contract JOIN customer
 USING (customer_id) JOIN car USING(car_id) JOIN car_model USING
(car_model_id)  WHERE contract_id =:contract_id");
$statement ->bindValue(":contract_id",$search_value);
}
else{

  $statement = $pdo -> prepare("SELECT * FROM contract JOIN customer
 USING (customer_id) JOIN car USING(car_id) JOIN car_model USING (car_model_id) ");
}
$statement -> execute();
$contracts = $statement -> fetchAll(PDO::FETCH_ASSOC);
?>

<!DOCTYPE html>
<html lang="en">


<script>
$(document).ready(function(){
  $('[data-toggle="tooltip"]').tooltip();
});
</script>
<head>
<meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="style.css"/>
```

```html
  <script src="https://code.iconify.design/iconify-icon/1.0.7/iconify-
icon.min.js"></script>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-
KK94CHFLLe+nY2dmCWGMq91rCGa5gtU4mk92HdvYe+M/SXH301p5ILy+dN9+nJOZ"
crossorigin="anonymous">
  </head>
<body>

  <nav class="navbar bg-body-tertiary">
  <div class="container-fluid">
    <a class="navbar-brand">CARry</a>
    <form class="d-flex" method="get">
      <input class="form-control me-2" name='search' type="search"
placeholder="Search" aria-label="Search">
      <button class="btn btn-outline-success" type="submit">Search</button>
    </form>
  </div>
</nav>
    <div class="container">
        <div class="table-responsive">
            <div class="table-wrapper">
                <div class="table-title">
                    <div class="row">
                        <div class="col-sm-8"><h2>Contract <b>List</b></h2></div>
                        <div class="col-sm-4">

                        </div>
                    </div>
                </div>
                <table class="table table-striped table-hover table-bordered">
                    <thead>
                        <tr>
                            <th>#</th>
                            <th>First Name <i class="fa fa-sort"></i></th>
                            <th>Last Name</th>
                            <th>Car Num. <i class="fa fa-sort"></i></th>
                            <th>Contract Num.</th>
                            <th> <i class="fa fa-sort"></i></th>

                        </tr>
                    </thead>
                    <tbody>
                        <?php
    foreach($contracts as $i => $contract){?>
    <tr>
  <th scope="row"><?PHP echo $i+1?> </th>
    <td><?php echo $contract['f_name'];?></td>
    <td><?php echo $contract['l_name'];?></td>
    <td><?php echo $contract['registeration_number'];?></td>
    <td><?php echo $contract['contract_id'];?></td>
    <td>
```

```html
                                        <form method="POST" action="./update.php"
style="display:inline-block">
                                            <button href="./update.php" class="edit"
type="submit" title="Edit" data-toggle="tooltip"><svg
xmlns="http://www.w3.org/2000/svg" width="24" height="24" viewBox="0 0 24 24"><path
fill="currentColor" d="M5 18.08V19h.92l9.06-9.06l-.92-.92z" opacity=".3"/><path
fill="currentColor" d="M20.71 7.04a.996.996 0 0 0 0-1.41l-2.34-2.34c-.2-.2-.45-.29-
.71-.29s-.51.1-.7.29l-1.83 1.83l3.75 3.75l1.83-1.83zM3 17.25V21h3.75L17.81 9.94l-
3.75-3.75L3 17.25zM5.92 19H5v-.92l9.06-9.06l.92.92L5.92 19z"/></svg></button>
                                            <input name="id" type="hidden" value=<?php echo
$contract['contract_id'] ?> />
                                        </form>

                                        <form method="POST" action="./delete.php"
style="display:inline-block" >
                                            <button href="./delete.php"  type="submit"
class="delete" title="Delete" data-toggle="tooltip"><svg
xmlns="http://www.w3.org/2000/svg" width="24" height="24" viewBox="0 0 24 24"><path
fill="currentColor" d="M16 9v10H8V9h8m-1.5-6h-5l-1 1H5v2h14V4h-3.5l-1-1zM18 7H6v12c0
1.1.9 2 2 2h8c1.1 0 2-.9 2-2V7z"/></svg></button>
                                            <input name="id" type="hidden" value=<?php echo
$contract['contract_id'] ?> />
                                        </form>
                                        </td>



        </tr>
<?php }
?>

                            </tr>
                        </tbody>
                    </table>

                </div>
            </div>
        </div>
        <a href="./create.php" class="create_button">Create</a>
</body>
</html>
```

Delete.php :

```php
<?php
require_once('./setting.php');
$id = $_POST['id']??null;
if(!$id){
  // header("Location:index.php");
}
$statement = $pdo -> prepare("DELETE FROM contract WHERE contract_id = :id");
$statement->bindValue(':id',$id);
$statement -> execute();
```

```php
header("Location:index.php");
?>
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Delete</title>
</head>
<body>
  <h2>Item Deleted</h2>

</body>
</html>
```

Create.php:

```php
<?php
require_once('./setting.php');
$Error=[];
$Insert_errors=[];
if($_SERVER["REQUEST_METHOD"]=="POST"){
  $f_name = $_POST['f_name'];
  $l_name = $_POST['l_name'];
  $car_registeration = $_POST['car_registeration'];
  $insurance_id = $_POST['insurance_id'];
  $pickup_date = $_POST['pickup_date'];
  $return_date = $_POST['return_date'];
  $rent_hours = $_POST['rent_hours'];


if(!$f_name){
  $Error[] = "Client's First Name is not provided ";
}
if(!$l_name){
  $Error[] = "Client's Last Name is not provided ";
}
if(!$insurance_id){
  $Error[] = "Insurance Type  is not provided ";
}
if(!$pickup_date){
  $Error[] = "Pickup Date is not provided ";
}
if(!$return_date){
  $Error[] = "Return Date is not provided ";
}
if(!$rent_hours  or ($rent_hours<=0) ){
  $Error[] = "Value not valid for rent hours";
}

//Chech foriegn keys car_id customer_id
```

```php
$check_car = true;
$check_customer = true;


if(empty($Errors) and $check_car and $check_customer){
  //check the type of the query result
    $highest_id_staement = $pdo ->  prepare("SELECT max(contract_id) FROM contract");
    $highest_id_staement -> execute();

    //get the hiegst id then pop after it
  $highest_id = $highest_id_staement -> fetch();


  $contract_id = $highest_id['max'] +1;

  //find the customer by their name
    $customer_id_statement = $pdo -> prepare("SELECT customer_id FROM customer WHERE
f_name = :f_name AND l_name = :l_name");
  $customer_id_statement->bindValue(":f_name",$f_name);
  $customer_id_statement->bindValue(":l_name",$l_name);
  $customer_id_statement->execute();
  // var_dump($customer_id_statement);
  $customer_id = $customer_id_statement -> fetch();
    var_dump($customer_id[0]);

  if($customer_id){
    $customer_id = $customer_id[0];
    // var_dump($customer_id);
  }
  else{
    $Insert_errors[] = 'Customer Does not exist in the system';
  }


  // find out rent cost :-
    //find the car_id by the car_registeration.
$car_id_statement = $pdo -> prepare("SELECT car_id FROM car WHERE
registeration_number=:registeration_number");
$car_id_statement-> bindValue(":registeration_number",$car_registeration);
$car_id_statement-> execute();
$car_id = $car_id_statement -> fetch();
if($car_id){
  $car_id = $car_id[0];

  //if the car_id exists, then :
  //find the rent_per_hour for the car_id
$rent_per_hour_statement= $pdo -> prepare("SELECT rent_per_hour FROM car
JOIN car_model USING (car_model_id) WHERE car.car_id = :car_id ");
$rent_per_hour_statement ->  bindValue(":car_id",$car_id);
$rent_per_hour_statement -> execute();
$rent_per_hour = $rent_per_hour_statement -> fetch();
$rent_per_hour = $rent_per_hour['rent_per_hour'];
$rent_cost = $rent_hours*$rent_per_hour;
```

```php
}
else{
  $Insert_errors[] = "no car holds such registeration number";
}



if(empty($Insert_errors)){
  $pdo->setAttribute( PDO::ATTR_ERRMODE, PDO::ERRMODE_WARNING );
  $statement = $pdo -> prepare("INSERT INTO contract
  (contract_id,customer_id,car_id,insurance_id,car_pickup_date,
  car_expected_return_date,car_actual_return_date,is_extended,
  rent_hours, rent_cost,extension_hours,hours_late,is_return)
   VALUES (:contract_id,:customer_id,:car_id ,
  :insurance_id, :pickup_date , :return_date,NULL,false,
  :rent_hours,:rent_cost,0,0,false)");
  $statement-> bindValue(":contract_id",$contract_id);
  $statement-> bindValue(":customer_id",$customer_id);
  $statement-> bindValue(":car_id",$car_id);
  $statement-> bindValue(":insurance_id",$insurance_id);
  $statement-> bindValue(":pickup_date",$pickup_date);
  $statement-> bindValue(":return_date",$return_date);
  $statement-> bindValue(":rent_hours",$rent_hours);
  $statement-> bindValue(":rent_cost",$rent_cost);
  $statement -> execute();

  // header("LOCATION: index.php");
}


}


}

?>
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-
KK94CHFLLe+nY2dmCWGMq91rCGa5gtU4mk92HdvYe+M/SXH301p5ILy+dN9+nJOZ"
crossorigin="anonymous">
  <link rel="stylesheet" href="style.css"/>
  <title>Create</title>
</head>
<body>
  <h1>New Contract</h1>
  <div class="errors">
  <?php
  foreach($Error as $Error){ ?>
```

```php
    <div class="alert alert-danger" role="alert">
    <?php echo $Error   ?>
    </div>;
    <?php }  ?>
  </div>


  <?php
  foreach($Insert_errors as $Error){ ?>
    <div class="alert alert-danger" role="alert">
    <?php echo $Error   ?>
    </div>;
    <?php }  ?>
  </div>



  <form method="POST"  class=form >
    <div >
    <label for="f_name" class="form-label">Customer's First Name</label>
    <input class="form-control form-control-lg" name="f_name" type="text"
placeholder="Ivan " aria-label=".form-control-lg example">
</div>
<div >
    <label for="l_name" class="form-label">Customer's Last Name</label>
    <input class="form-control form-control-lg" name="l_name" type="text"
placeholder="Ivanovich" aria-label=".form-control-lg example">
</div>

<div >
    <label for="car_registeration" class="form-label">Car registeration
Number</label>
    <input class="form-control form-control-lg" name="car_registeration" type="text"
placeholder="Car Registeration Number" aria-label=".form-control-lg example">
</div>

<div class="from-item">
<label class="form-label" for="insurance_id">insurance type </label>
<select class="form-select" name="insurance_id" aria-label="Default select example">

  <option value="1">Casual Insurance</option>
  <option value="2">Full Insurance</option>
  <option value="3">Premium Insurance</option>
</select>
</div>

<div class="">
  <label for="pickup_date" class="form-label">Pickup Date</label>
    <input class="form-control form-control-lg" name="pickup_date" type="date"  aria-
label=".form-control-lg example">
</div>

<div class="">
```

```html
    <label for="return_date" class="form-label">Return Date</label>
      <input class="form-control form-control-lg" name="return_date" type="date"  aria-
label=".form-control-lg example">
</div>

<div class="">
  <label for="rent_hours" class="form-label">Rent Hours</label>
    <input class="form-control form-control-lg" name="rent_hours"
type="number"  aria-label=".form-control-lg example">
</div>
<div><button  type="submit" class="btn btn-primary submit_button">
  Submit
</button>
</div>

  </form>

</body>
</html>
```

Update.php:

```php
<?php
$contract_id = $_POST["id"];
require_once('./setting.php');
$statement = $pdo -> prepare("SELECT * FROM contract  WHERE contract_id =
:contract_id "  );
$statement ->bindValue(":contract_id",$contract_id);
$statement -> execute();
$contract = $statement -> fetch();

if($_SERVER["REQUEST_METHOD"]=="GET"){
  $return_date = $_GET["return_date"];
  $extension_hours = $_GET["extension_hours"];
  $hours_late = $_GET["hours_late"];
  $contract_id = $_GET["contract_id"];
  $update_statement = $pdo -> prepare("UPDATE contract SET car_return_actual_date
=:return_date
  , extension_hours = :extemsion_hours , hours_late = : hours_late  WHERE contract_id
= :contract_id");
  $update_statement ->bindValue(":contract_id",$contract_id);
  $update_statement ->bindValue(":return_date ",$return_date );
  $update_statement ->bindValue(":extemsion_hours",$extemsion_hours);
  $update_statement ->bindValue(":hours_late",$hours_late);
  $update_statement -> execute();

 //add is_return and extension automations
}

?>

<!DOCTYPE html>
<html lang="en">
```

```html
<head>
<meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="stylesheet" href="style.css"/>
  <script src="https://code.iconify.design/iconify-icon/1.0.7/iconify-
icon.min.js"></script>
   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-
KK94CHFLLe+nY2dmCWGMq91rCGa5gtU4mk92HdvYe+M/SXH301p5ILy+dN9+nJOZ"
crossorigin="anonymous">
  </head>
<body>
  <h2>Update</h2>

<form method="GET"  class=form >
<div class="">
  <label for="return_date" class="form-label">Actual Return Date</label>
    <input class="form-control form-control-lg" name="return_date" type="date"
value=<?Php echo $contract["car_actual_return_date"] ?> aria-label=".form-control-lg
example">
</div>


<div class="">
  <label for="extenstion_hours" class="form-label">Extension Hours</label>
    <input class="form-control form-control-lg" name="extension_hours" type="number"
value=<?Php echo $contract["extension_hours"] ?> aria-label=".form-control-lg
example">
</div>



    <input class="form-control form-control-lg" name="contract_id" type="hidden"
value=<?Php echo $contract_id ?> aria-label=".form-control-lg example">




<div class="">
  <label for="hours_late" class="form-label">Hours Late</label>
    <input class="form-control form-control-lg" name="hours_late" type="number"
value=<?Php echo $contract["hours_late"] ?>  aria-label=".form-control-lg example">
</div>
<div><button  type="submit" class="btn btn-primary submit_button">
  Submit
</button>
</div>


  </form>
</body>
```

```
</html>
```

Вывод:
 В этой лабораторной работе я освоил команды DML SQL в postgresql и научился
создавать сложные запросы и подзапросы.

```
</html>
```