

Кафедра ИУ-4
«Проектирование и технология производства ЭС»

Журнал практических работ

**по курсу: «Конструкторско-технологические
базы данных»**

Для студентов приборостроительных специальностей

20__ / __ учебный год

Студент Круглов В. С.
(фамилия, и. о.)

Группа ИУ4-83Б

Преподаватель Власов А. И. Допуск к экзамену (зачету) _____ Подпись _____
(фамилия, и. о.) (число)

Программа
к учебному плану направления подготовки - 11.03.03
«Конструирование и технология электронных средств»,

Виды учебных работ	Объем работ, час.	
	Всего	8 семестр
<u>Выделено на дисциплину</u>	108	108
<u>Аудиторная работа:</u>	52	52
лекции	40	40
семинары	12	12
<u>лабораторные работы</u>	-	-
Самостоятельная работа:	56	56
домашнее задание	30	30
изучение разделов	16	16
подготовка к контрольным мероприятиям	10	10
Контроль знаний	Недели контроля	
Срок сдачи домашнего задания		6, 11
Зачет		-
Экзамен		экзамен

Комплексное задание:

Разработать модуль автоматизированной системы радиотехнического предприятия согласно варианту комплексного задания. Система должна позволять просматривать статистическую информацию по процессу производства изделий электронной техники, обрабатывать ее, составлять сводные отчеты и т.п. Система должна позволять генерировать конструкторско-технологические документы операционные, маршрутные карты, спецификации (и т.п. документы, предусмотренные заданием) в формате PDF, используя введенные в нее данные. Система должна быть реализована с использованием лингвистического обеспечения PHP и развернута на основе СУБД Oracle.

По сумме рейтинговых баллов модулей (М1, М2: 6 практических занятий по 5 баллов, ДЗ – 30 баллов, РК – 10 баллов) в семестре (0-70) и на экзамене (М3:0-30) выставляется итоговая оценка исходя из следующих рейтинговых баллов: 60-75 – удовлетворительно, 76-90 – хорошо, 90 и выше – отлично.

График выполнения контрольных мероприятий*																				
Февраль				Март				Апрель					Май					Июнь		
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17				
6	13	20	27	6	13	20	27	3	10	17	24	31	1	8	15	22	29	5	12	19
					M1					M2		M3								

* - все контрольные мероприятия должны быть сданы не менее чем на минимальную пороговую рейтинговую оценку

<p align="center">Отчет по лабораторной работе № 1 «Разработка интерфейсных модулей АИС на РНР»</p>			
дата	Оценка (max 5)	Бонус за сложность	подпись

ПОДПИСЬ

Цели работы:

Повторение и обобщение принципов разработки пользовательского интерфейса на РНР с функциями доступа к базе данных созданной под управлением СУБД Oracle.

Задачи работы:

- знакомство с работой интерпретатора PHP и вебсервера Apache
- разработка примеров простейших программ на PHP

Задание повышенной сложности (бонус за сложность – 10 баллов):

-разработка интерфейса модуля АИС с расширенными функциями (поиск и т.п.)

Краткий конспект теоретической части (ответы на контрольные вопросы)

Основные понятия РНР

Переменные в PHP

Библиотеки работы с СУБД

Авторизация средствами РНР

Установка и настройка PHP

Сессии и управление ими

Простейшие примеры работы с PHP

Пример 1: Варианты объявления PHP скрипта:

<center>Пример 1: Варианты объявления PHP скрипта</center>

```
<?    echo "1. простейший способ, но возможен конфликт при использовании XML"; ?> <br>
<?php echo("2. Наиболее распространенный способ");           ?> <br>
<%    echo("3. Начиная с PHP 3.0.4 можно факультативно применять ASP-теги");    %> <br>
<script language="php">
    echo("4. используется для лучшей совместимости с HTML редакторами");
</script><br>
```

Вид отображения на экране

Пример 1: Варианты объявления PHP скрипта

1. Простейший способ, но возможен конфликт при использовании XML

2. Наиболее распространённый способ

<% echo("3. Начиная с PHP 3.0.4 можно факультативно использовать ASP-теги"); %>

Пример 2: Объявления переменных:

```
<?php
// объявление переменных
$a = 1;
$b = 2;

/* тело скрипта*/
$c = $a + $b ;
echo ( "результат сложения a=1 и b=2 равен" );
echo $c;
?>
```

Вид отображения на экране

результат сложения a=1 и b=2 равен3

Пример 3: Передача значения из формы:

```
<form action="02.php" method="post">
    Name: <input type="text" name="name"><br>
    <input type="submit">
</form>

<?php
echo ("name=");
echo $_POST["name"]; //используется глобальный массив $_POST["name"]
?>
```

Вид отображения на экране

Name:

name=Valentin Kruglov IU4-83B

Пример 4: Передача нескольких значений из формы:

```
<?php
    echo ("<P>");
    echo ("name= "); echo $_GET["name"];
    echo ("<P>");
    echo ("e-mail= "); echo $_GET["e-mail"];
?>

<P>

<form action=03.php method="get">
    Name:   <input type="text" name="name"><br>
    E-mail: <input type="text" name="e-mail"><br>
           <input type="submit">
</form>
```

Вид отображения на экране
name= Valentin Kruglov IU4-83B
e-mail= kruglov.valentine@gmail.com
Name: <input type="text"/>
E-mail: <input type="text"/>
<input type="submit" value="Отправить запрос"/>

Пример 5: Обработка переключателей и радиокнопок

```
<?
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    echo ("Мой любимый предмет: <i>" . $_POST['kurs'] . "</i><br>");

    $favorite_times = count($times);

    if( $favorite_times <= 1) {
        $times_message = 'не ботан';
    } elseif ($favorite_times > 1 && $favorite_times < 4) {
        $times_message = 'ботаю иногда';
    } else {
        $times_message = 'ботан';
    }

    echo ("Я <i>" . $times_message . "</i><br>");
}
?>

<FORM TARGET="04.php" METHOD="POST">
Любимый предмет:<br>
    <INPUT TYPE=RADIO NAME="kurs" VALUE="КТБД">конструкторско-технологические базы данных<br>
    <INPUT TYPE=RADIO NAME="kurs" VALUE="СФМ">Системы функционального моделирования<br>
    <INPUT TYPE=RADIO NAME="kurs" VALUE="СИИ">Системы искусственного интеллекта<br>

когда вы предпочитаете его изучать:<br>
    <INPUT TYPE=CHECKBOX NAME="times[]" VALUE="m">за завтраком
    <INPUT TYPE=CHECKBOX NAME="times[]" VALUE="n">в обед
    <INPUT TYPE=CHECKBOX NAME="times[]" VALUE="d">за ужином
    <INPUT TYPE=CHECKBOX NAME="times[]" VALUE="l">Поздно ночью

<P>
<INPUT TYPE=HIDDEN NAME="stage" VALUE="results">
<INPUT TYPE=SUBMIT VALUE="Всегда!">
</FORM>
```

Вид отображения на экране
Мой любимый предмет: <i>КТБД</i> я не ботан Любимый предмет: <input checked="" type="radio"/> Конструкторско-технологические базы данных <input type="radio"/> Системы функционального моделирования <input type="radio"/> Системы искусственного интеллекта когда вы предпочитаете его изучать: <input type="checkbox"/> за завтраком <input type="checkbox"/> в обед <input type="checkbox"/> за ужином <input checked="" type="checkbox"/> поздно ночью <input type="submit" value="Всегда!"/>

Задание 1: Разработать PHP модуль, обрабатывающий значения, передаваемые через опции SELECT, “LISTBOX” (выпадающих меню) форм

```
<html>
<body>
  <?php
    $countriesWithCities = [
      "Japan"      => "Tokyo",
      "Mexico"     => "Mexico City",
      "USA"        => "Washington",
      "India"      => "Mumbai",
      "South Korea" => "Seoul",
      "China"      => "Shanghai",
      "Nigeria"   => "Lagos",
      "Brazil"     => "Sao Paulo",
      "Egypt"     => "Cairo",
      "England"   => "London",
    ];
  ?>
  <form action="task_1.php" method="POST">
    <h3>Please choose a country from the list box. </h3>
    <select name="city" id="listBox">
      <?php foreach ($countriesWithCities as $individualCountry => $city) {?>
        <option value=<?php echo($city); ?>><?php echo($individualCountry); ?></option>
      <?php }?>
    </select>
    <input type="submit" name="submitButton" id="submitButton" value="Submit form" />
  </form>

  <?php if (isset($_POST["submitButton"])) {
    echo "You chose " . $_POST['city'];
  }
  ?>
</body>
</html>
```

Please choose a country from the list box.

Japan ▼ Submit form

You chose Washington

Формы представляют собой простейший способ организации внутри HTML-документа обратной связи между пользователем и сервером. Упрощенно формы можно понимать как набор кнопок, флажков, полей ввода, передаваемых сценарию, в качестве входной информации для обработки. Обработка, принятой сервером информации, ничего общего с HTML не имеет и может выполняться самыми разными средствами.

Спецификация дескриптора задания форм		
Дескриптор	Назначение	
<form> </form>	Формуляр	
	Атрибут	Значение
	Action	URL-адрес для отправки заполненного формуляра
	Enctype	кодирование передаваемых данных
	Method	способ передачи формуляра

Типы внутренних конструкций в формах:

- Поля ввода объектов (типы объектов определяются значением атрибута type).
- Поля ввода многострочных текстов.
- Выпадающие меню.
- Поля списков.

Поля ввода.

Спецификация дескриптора задания форм		
Дескриптор	Назначение	
<input> </input>	Поле для ввода строки	
	Атрибут	Значение
	Action=URL	URL адрес для отправки заполненной формы
	Align= <div> <div>Bottom</div> <div>Left</div> <div>Middle</div> <div>Right</div> <div>top</div> </div>	<div>выравнивает нижний край кнопки по базовой линии строки</div> <div>выравнивает кнопку-иллюстрацию по левому краю текста</div> <div>центрирует кнопку-иллюстрацию в текстовой строке</div> <div>выравнивает кнопку-иллюстрацию по правому краю текста</div> <div>выравнивает верхний край кнопки-иллюстрации по верхней линии строки текста</div>
	Checked	Установленный флажок или выбранное положение переключателя
	Maxlength	Максимальная длина вводимых текстов
	Name	Название элемента ввода/управления
	Src	Источник графического файла картинки кнопки
	Type = <div> <div>checkbox</div> <div>file</div> <div>image</div> <div>hidden</div> <div>password</div> <div>radio</div> <div>reset</div> <div>submit</div> <div>text</div> </div>	<div>Тип элемента управления /ввода</div> <div>флажки, независимы друг от друга, их можно установить/сбросить в любой комбинации</div> <div>поле ввода для имени файла,</div> <div>рядом отображается кнопка, "Пролистать/Browse"</div> <div>открывающая стандартное диалоговое окно выбора файла</div> <div>пользовательская командная кнопка вместо стандартных,</div> <div>получаемых с помощью type=submit или type=reset</div> <div>параметры передаваемые на сервер, которые не могут быть изменены пользователем</div> <div>текстовое поле, вводимые данные отображаются "звездочками"</div> <div>селекторные кнопки (переключатели), из группы можно выбрать только одну</div> <div>командная кнопка, возвращает формуляр к исходному состоянию; данные не пересылаются командная кнопка,</div> <div>отправляет на сервер всё внесенное в формуляр</div> <div>однострочное текстовое поле</div>
	Value	Установленное по умолчанию значение

Поля ввода многострочных текстов

Спецификация дескриптора ввода многострочного текстового поля		
Дескриптор	Назначение	
<textarea> </textarea>	Многострочное поле для ввода текста в форме	
	Атрибут	Значение
	Cols=N	Количество символов /столбцов в поле ввода
	Name=char	Имя поля ввода
	Rows=N	Количество строк поля ввода
	Wrap= <div> <div>off</div> <div>physical</div> <div>virtual</div> </div>	<div> <div>верстка не выполняется (сервер получает текст одним куском)</div> <div>автоматическая верстка, с переносом строк по мере достижения правого края (сервер получает текст с разрывами строк)</div> <div>автоматическая верстка, с переносом строк по мере достижения правого края (сервер получает текст одной строкой - без разрывов)</div> </div>

Формирование выпадающих меню и полей списков

Спецификация дескриптора меню или поля списка в форме		
Дескриптор	Назначение	
<select> </select>	Меню или поле списка в форме	
	Атрибут	Значение
	Multiple	Возможность выбора нескольких опций
	Name=char	Имя элемента
	Size=N	Количество одновременно отображаемых элементов

Спецификация дескриптора задания элемента списка/меню в форме		
Дескриптор	Назначение	
<option> </option>	Элемент списка/меню в форме.	
	Атрибут	Значение
	Selection	Выбран по умолчанию
	Value	Параметры элемента

Пример 6: Подключение к базе данных

```
<?php
    if ($c=OCILogon("scott", "tiger", "//localhost/orcl")) {
        echo "successfully connected to oracle.\n";
        OCILogoff($c);
    }else {
        $err = OCIError();
        echo "Oracle Connect Error " . $err[text];
    }
?>
```

Вид отображения на экране



Пример 7: Формирование простейшего отчета из базы данных

```
<?php
$c=OCILogon("scott", "tiger", "//localhost/orcl");
if ( ! $c ) {
    echo "Невозможно подключиться к базе: " . var_dump( OCIError() );
    die();
}

// производим выборку из базы данных
$s = OCIParse($c, "SELECT object_name, object_type FROM user_objects");
OCIExecute($s, OCI_DEFAULT);

echo ("<center><table border=1>
    <tr><td><center>OBJECT_NAME</center></td>
    <td><center>OBJECT_TYPE</center></td>
    </tr>");
while (OCIFetch($s)) {
    echo ("<tr><td>" . ociresult($s, "OBJECT_NAME") . "</td><td>" .
        ociresult($s, "OBJECT_TYPE") . "</td></tr>");
}

echo ("</table></center>" );

// Выполняем commit;
OCICommit($c);

// отключаемся от базы данных
OCILogoff($c);
?>
```

Вид отображения на экране

OBJECT_NAME	OBJECT_TYPE
LOGMNR_SESSION_EVOLVE\$	TABLE
LOGMNR_SESSION_EVOLVE\$_PK	INDEX
LOGMNR_EVOLVE_SEQ\$	SEQUENCE
LOGMNR_SEQ\$	SEQUENCE
LOGMNR_DID\$	SEQUENCE
LOGMNR_UID\$	SEQUENCE
LOGMNR_GLOBAL\$	TABLE
LOGMNR_GT_TAB_INCLUDE\$	TABLE
LOGMNR_GT_USER_INCLUDE\$	TABLE
LOGMNR_GT_VID_INCLUDE\$	TABLE

Пример 8: Вычислительные возможности Oracle

```
<?php
$c=OCILogin("scott", "tiger", "//localhost/orcl");
if ( ! $c ) {
    echo "Невозможно подключиться к базе: " . var_dump( OCIError() );
    die();
}

// Производим выборку из базы данных
$s = OCIParse($c, "SELECT sin(3.14) FROM dual");
OCIExecute($s, OCI_DEFAULT);
while (OCIFetch($s)) {
    echo "sin ( 3.14 ) = " . ociresult($s) . "\n";
}

// Выполняем commit;
OCICommit($c);

// Отключаемся от базы данных
OCILogoff($c);
?>
```

Задание 2: Модифицируйте исходный код так, чтобы выводился в итоговую web страницу результат с точностью до 5 знака, после запятой, реализуйте вычисление других числовых функций.

Исходный код
<pre><?php \$conn = oci_connect("system", "admin", "host.docker.internal:1521/xe"); if (! \$conn) { echo("Невозможно подключиться к базе: " . var_dump(oci_error())); die(); } \$str = oci_parse(\$conn, "SELECT sin(3.14), 2+3 FROM dual"); oci_execute(\$str, OCI_DEFAULT); while (oci_fetch(\$str)) { echo("sin (3.14) = " . round(oci_result(\$str, 1), 5) . "
\n"); echo("2 + 3 = " . oci_result(\$str, 2) . "
\n"); } oci_commit(\$conn); oci_close(\$conn); ?></pre>
Вид отображения на экране
<pre>sin (3.14) = 0.00159 2 + 3 = 5</pre>

Задание 3: Реализуйте шаблон интерфейсной формы АСУ ТП на PHP, модель которой была разработана вами в 8 ЛР 7 семестра. Разместите шаблон в вашем персональном разделе на сервере [http:// host.iu4.bmstu.ru](http://host.iu4.bmstu.ru).

Вклейте экранную форму созданного шаблона

Логин: Пароль:

```
<form action="task_3.php" method="post">
    Логин: <input type="text" name="login" />
    Пароль: <input type="password" name="password" />
    <input type="submit" value="Войти" name="log_in" />
</form>

<?php
    session_start();

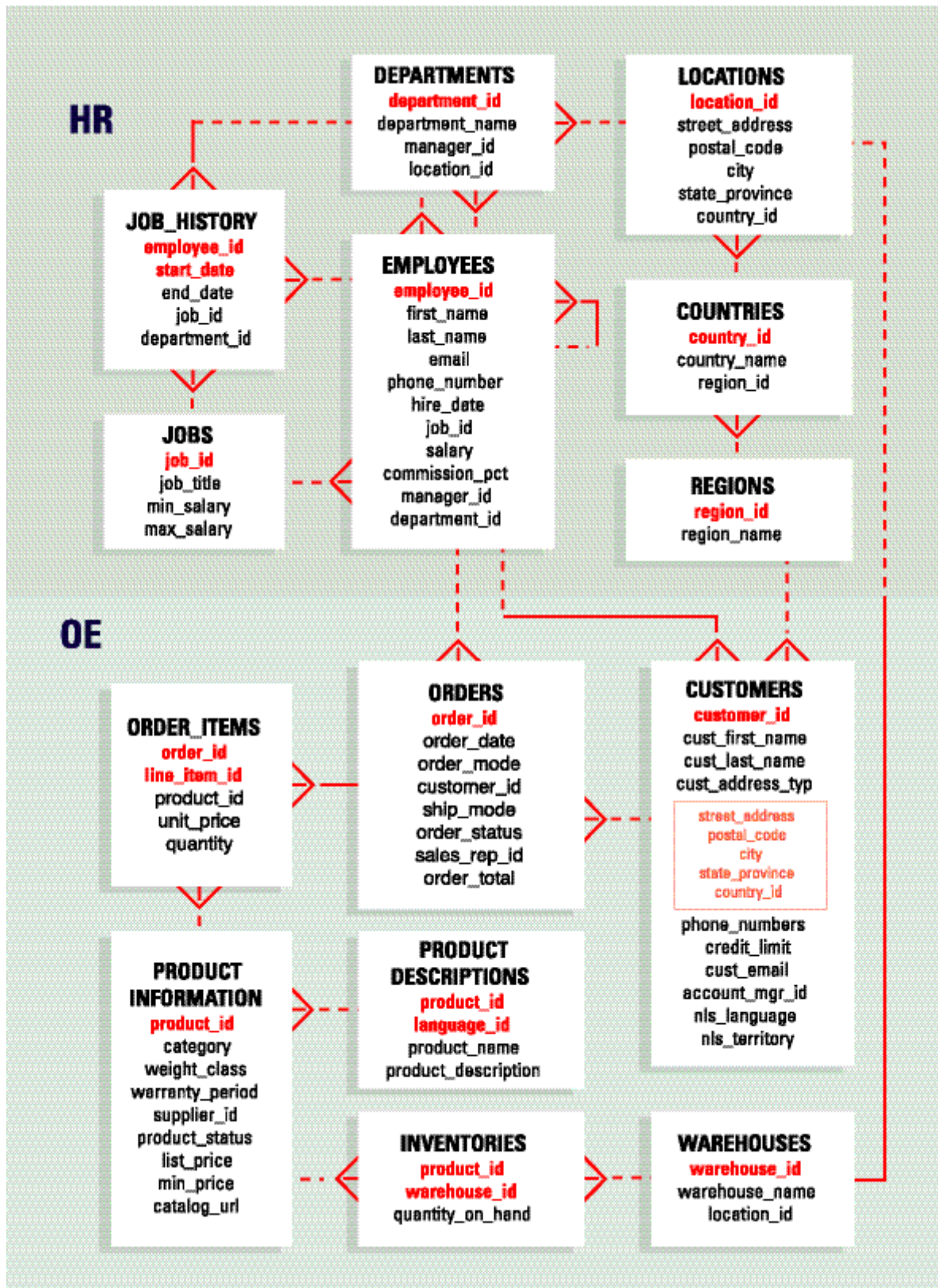
    $conn = oci_connect("system", "admin", "host.docker.internal:1521/xe");
    if (! $conn) {
        echo("Невозможно подключиться к базе: " . var_dump(oci_error()));
        die();
    }

    $str = oci_parse($conn, "SELECT password_hash FROM users WHERE username={$_POST['login']}");
    oci_execute($str, OCI_DEFAULT);

    while (oci_fetch($str)) {
        if (password_verify($_POST['password'], oci_result($str, 1))) {
            $_SESSION['user_id'] = $_POST['login'];
            header('Location: /');
        }
    }
    oci_commit($conn);
    oci_close($conn);

?>
```

Задание 4: Для тестовой схемы HR выполните задания по формированию отчетов, представленных в задании для самоконтроля, результаты отобразите посредством РНР в своем личном каталоге на сервере.



Задания для самоконтроля:

№	Текст задания	Число возвращенных строк
1	Задание: Выбор служащих по диапазону окладов. Выбрать имена, фамилии и оклады служащих, чья зарплата попадает в диапазон от 3000 до 4000 включительно. Упорядочить по окладу в убывающем порядке	
	SQL запрос: SELECT first_name, last_name, salary FROM EMPLOYEES WHERE salary >= 3000 AND salary <= 4000 ORDER BY salary DESC;	
2	Задание: Выбор подчиненных менеджеров Выберите имена, фамилии и код менеджера служащих, у которых код менеджера <105>, <149> или <205>	
	SQL запрос: SELECT first_name, last_name, manager_id FROM employees WHERE manager_id=105 OR manager_id=149 OR manager_id=205;	
3	Задание: Выбор сотрудников по буквам Email Выберите, фамилии, адреса почты и телефоны служащих, у которых в e-mail второй стоит буква "H"	
	SQL запрос: SELECT last_name, email, phone_number FROM employees WHERE INSTR(email, 'H', 2)=2;	
4	Задание: Задача на выборку Выберите фамилии, зарплаты и комиссионные служащих, не имеющих комиссионных. Отсортируйте результат по окладу.	
	SQL запрос:	
5	Задание: Выбор высокооплачиваемых сотрудников отдела Выберите имена, фамилии, оклады и телефоны служащих, работающих в отделе <60> у которых оклад больше 3000. Упорядочите по окладу в убывающем порядке.	
	SQL запрос:	

6	Задание: Выбор коммиссионных и IT сотрудников Выберите имена, фамилии, оклады, должности и коммиссионные служащих, у которых должность начинается с символов <IT> или у которых есть коммиссионные.	
	SQL запрос:	
7	Задание: Выбор сотрудников не подчиняющихся менеджерам. Выберите имена, фамилии и код менеджера служащих, у которых код менеджера отличается от <105>, <149> или <205>. Упорядочите по фамилии.	
	SQL запрос:	
8	Задание: Дата поступления на работу Выберите имена, фамилии и даты поступления на работу служащих и отсортируйте по возрастанию даты поступления на работу.	
	SQL запрос:	
9	Задание: Телефоны служащих Выберите коды отделов, фамилии и телефоны служащих. Отсортируйте по отделам, а в рамках отдела по фамилиям служащих.	
	SQL запрос:	
10	Задание: Взаимосвязь подразделений Выберите коды подразделений и определите их подчиненность.	
	SQL запрос:	

Контрольные вопросы

1. Основные понятия РНР?
2. Переменные в РНР?
3. Библиотеки работы с СУБД?
4. Авторизация средствами РНР?
5. Установка и настройка РНР?
6. Сессии и управление ими?

СПИСОК ЛИТЕРАТУРЫ

1. Норенков И.П. Системы автоматизированного проектирования. - М.: Изд-во МГТУ им.Н.Э.Баумана. 2001.
2. Иванова Г.С. Технология программирования: Учебник для Вузов. – 2-ое издание, стереотипное. М.: Изд-во МГТУ им.Н.Э.Баумана, 2003. – 320 с.; ил. (Сер. Информатика в техническом университете).
3. Власов А.И., Лыткин С.Л., Яковлев В.Л. Краткое практическое руководство по языку PL/SQL - М.: Машиностроение. 2000. 64 с.
4. Сервер Oracle. Справочное руководство по языку SQL / Под ред. А.В.Емельяненко, Н.В. Емельяненко - Протвино, АО РДТех, 1994.
5. Сервер ORACLE. Основные концепции/ Под ред. А.В.Емельяненко, Н.В. Емельяненко - Протвино, АО РДТех, 1996.
6. ГОСТ Р ИСО 10303-1-99. Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Ч.1. Общие представления и основополагающие принципы. - Москва: ИПК Издательство стандартов, 2000.
7. ГОСТ Р ИСО 10303-1-99. Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Ч.21. Методы реализации. Кодирование открытым текстом структуры обмена. - Москва: ИПК Издательство стандартов, 2000.
8. Проектирование и эксплуатация конструкторско-технологических баз данных на основе СУБД Oracle/ Конспект лекций - М.: Москва, МГТУ им. Н.Э. Баумана, 2001, 120 с.
9. Спецификация UML OMG ISO/IEC 19501.