



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Магнитогорский государственный технический университет им. Г.И. Носова»



УТВЕРЖДАЮ  
Директор ИЭиАС  
С.И. Лукьянов

26.02.2020 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

***БАЗЫ ДАННЫХ В АСУ ТП***

Направление подготовки (специальность)  
27.03.04 УПРАВЛЕНИЕ В ТЕХНИЧЕСКИХ СИСТЕМАХ

Направленность (профиль/специализация) программы  
Системы и средства автоматизации технологических процессов

Уровень высшего образования - бакалавриат  
Программа подготовки - академический бакалавриат

Форма обучения  
очная

Институт/ факультет	Институт энергетики и автоматизированных систем
Кафедра	Автоматизированных систем управления
Курс	4
Семестр	8

Магнитогорск  
2020 год

Рабочая программа составлена на основе ФГОС ВО по направлению подготовки 27.03.04 УПРАВЛЕНИЕ В ТЕХНИЧЕСКИХ СИСТЕМАХ (уровень бакалавриата) (приказ Минобрнауки России от 20.10.2015 г. № 1171)

Рабочая программа рассмотрена и одобрена на заседании кафедры Автоматизированных систем управления 12.02.2020, протокол № 6

Зав. кафедрой  С.М. Андреев

Рабочая программа одобрена методической комиссией ИЭиАС 26.02.2020 г. протокол № 5

Председатель  С.И. Лукьянов

Рабочая программа составлена:

доцент кафедры АСУ, канд. техн. наук  М.Ю. Рябчиков

Рецензент: зам. директора  ЗАО "Консом СКС" , канд. техн. наук Ю.Н. Волшуков



## Лист актуализации рабочей программы

---

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2021 - 2022 учебном году на заседании кафедры Автоматизированных систем управления

Протокол от \_\_\_\_\_ 20\_\_ г. № \_\_\_\_  
Зав. кафедрой \_\_\_\_\_ С.М. Андреев

---

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2022 - 2023 учебном году на заседании кафедры Автоматизированных систем управления

Протокол от \_\_\_\_\_ 20\_\_ г. № \_\_\_\_  
Зав. кафедрой \_\_\_\_\_ С.М. Андреев

---

Рабочая программа пересмотрена, обсуждена и одобрена для реализации в 2023 - 2024 учебном году на заседании кафедры Автоматизированных систем управления

Протокол от \_\_\_\_\_ 20\_\_ г. № \_\_\_\_  
Зав. кафедрой \_\_\_\_\_ С.М. Андреев

### **1 Цели освоения дисциплины (модуля)**

развитие профессиональных компетенций в области: поиска, хранения, обработки и анализ информации из различных источников и баз данных; представления информации в системах АСУ ТП; использования навыков работы с компьютером при работе с базами данных; сбора и анализа исходных данных для расчета и проектирования систем и средств автоматизации и управления.

Для достижения поставленных целей в дисциплине «Базы данных в АСУ ТП» решаются задачи:

- изучение роли и места баз данных в автоматизированных системах, назначения и основных характеристиках различных систем управления базами данных, их функциональных возможностей;
- изучение особенностей структурной организации реляционных баз данных и основными функциями системы управления базами данных (СУБД);
- изучение методик проектирования реляционных базы данных;
- изучение практических приемов работы с СУБД и языка структурированных запросов SQL;
- изучение основ встроенного и модульного программирования на SQL;
- изучение особенностей работы автоматизированных систем диспетчерского управления и сбора данных (SCADA) с базами данных.

### **2 Место дисциплины (модуля) в структуре образовательной программы**

Дисциплина Базы данных в АСУ ТП входит в вариативную часть учебного плана образовательной программы.

Для изучения дисциплины необходимы знания (умения, владения), сформированные в результате изучения дисциплин/ практик:

Системы автоматизации и управления

Информатика и информационные технологии

Интегрированные системы проектирования и управления

Знания (умения, владения), полученные при изучении данной дисциплины будут необходимы для изучения дисциплин/практик:

Подготовка к защите и защита выпускной квалификационной работы

### **3 Компетенции обучающегося, формируемые в результате освоения дисциплины (модуля) и планируемые результаты обучения**

В результате освоения дисциплины (модуля) «Базы данных в АСУ ТП» обучающийся должен обладать следующими компетенциями:

Структурный элемент компетенции	Планируемые результаты обучения
ОПК-6	способностью осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий
Знать	- структуры данных, используемые для представления типовых информационных объектов, типовые алгоритмы обработки данных; модели представления данных; - основные операторы языков запросов SQL, T-SQL, PL/SQL;

Уметь	- создавать отчеты с применением механизмов динамического формирования SQL запросов; - разрабатывать и программно реализовывать механизмы обеспечения целостности данных в созданной базе за счет применения подпрограмм и триггеров;
Владеть	- языками программирования SQL, T-SQL, PL/SQL
ОПК-9 способностью использовать навыки работы с компьютером, владеть методами информационных технологий, соблюдать основные требования информационной безопасности	
Знать	- компьютерные приложения для взаимодействия с СУБД разных производителей; - средства проектирования баз данных. - принципы работы сред программирования встроенных в SCADA Intouch и WinCC и возможности по их применению в задачах автоматизированного сбора, обработки и хранения информации;
Уметь	- использовать стандартные пакеты прикладных программ для решения практических задач формирования запросов к базам; - осуществлять проектирование базы данных с применением CASE систем проектирования (ERWin); - строить диалоговые окна SCADA систем Intouch или WinCC взаимодействующие с СУБД;
Владеть	- навыками работы с современными программными средствами исследования и проектирования систем управления; навыками работы с техническими и программными средствами автоматизированного сбора, хранения и обработки данных - языками оболочек WinCC или Intouch на уровне достаточном для организации связи с СУБД и создания эффективного интерфейса взаимодействия с технологом – оператором и СУБД;
ПК-5 способностью осуществлять сбор и анализ исходных данных для расчета и проектирования систем и средств автоматизации и управления	
Знать	- интерфейсы взаимодействия разрабатываемых приложений с СУБД; - возможности, связанные с применением технологий OLAP и Data mining в СУБД с реляционной и многомерной моделью представления информации; - методы, формы и этапы проектирования баз данных;
Уметь	- применять принципы и законы реляционной алгебры при самостоятельном проектировании реляционных баз данных; - настраивать и программировать интерфейсы взаимодействия с базами данных; - разрабатывать клиентские приложения с применением объектных интерфейсов подключения ADO и DAO; создавать приложения обработки данных на основе модульного программирования; решать исследовательские и проектные задачи с использованием баз данных;
Владеть	- методами построения современных проблемно-ориентированных прикладных программных средств/

#### 4. Структура, объём и содержание дисциплины (модуля)

Общая трудоемкость дисциплины составляет 3 зачетных единиц 108 акад. часов, в том числе:

- контактная работа – 56,2 акад. часов;
- аудиторная – 55 акад. часов;
- внеаудиторная – 1,2 акад. часов
- самостоятельная работа – 51,8 акад. часов;

Форма аттестации - зачет

Раздел/ тема дисциплины	Семестр	Аудиторная контактная работа (в акад. часах)			Самостоятельная работа студента	Вид самостоятельной работы	Форма текущего контроля успеваемости и промежуточной аттестации	Код компетенции
		Лек.	лаб. зан.	практ. зан.				
1. Модели представления данных и принципы реляционной алгебры								
1.1 Структуры данных, используемые для представления типовых информационных объектов. Типовые алгоритмы обработки данных. Модели представления данных	8	1			4	Самостоятельное изучение учебной и научно литературы. Поиск дополнительной материалов по теме	Устный опрос	ОПК-6, ПК-5
1.2 Взаимосвязь реляционной алгебры с модификациями языка запросов SQL		1			4	Самостоятельное изучение учебной и научно литературы. Поиск дополнительной материалов по теме	Устный опрос	ОПК-6, ПК-5
Итого по разделу		2			8			
2. Проектирование баз данных								
2.1 Методы, формы и этапы проектирования баз данных. Средства проектирования баз данных	8	2			4	Самостоятельное изучение учебной литературы, подготовка к контрольной работе	Опрос по выполненной самостоятельной работе «Проектирование базы данных»	ОПК-9, ПК-5
2.2 Метод нормальных форм и метод сущность-связь		2			4	Самостоятельное изучение учебной литературы, подготовка и выполнение контрольной работы	Опрос по выполненной самостоятельной работе «Проектирование базы данных»	ОПК-9, ПК-5

Итого по разделу		4			8			
3. Языки запросов SQL, T-SQL, PL/SQL								
3.1 Базовые операторы языка SQL и языка T-SQL для Microsoft SQL Server	8	4	6/2И		4	Самостоятельное изучение учебной литературы, подготовка к лабораторным занятиям, выполнение индивидуального задания по работе	Устный опрос по работе №1 Устный опрос по работе №2 Устный опрос по работе №3	ОПК-6, ОПК-9
3.2 Операторы PL/SQL для Oracle		2	4/2И		4	Самостоятельное изучение учебной литературы, подготовка к лабораторным занятиям, выполнение индивидуального задания по работе	Устный опрос по работе №10	ОПК-6, ОПК-9
Итого по разделу		6	10/4И		8			
4. Модульное и встроенное программирование с применением курсоров								
4.1 Понятие курсора, триггера. Принципы организации SQL-программы и операторы SQL, используемые при программировании	8	2	4/2И		4	Самостоятельное изучение учебной литературы, подготовка к лабораторным занятиям, выполнение индивидуального задания по работе	Устный опрос по работе №9 Устный опрос по работе №12 Устный опрос по работе №13	ОПК-6, ПК-5
4.2 Модульное и встроенное программирование с использованием PL/SQL		2	4/2И		4	Самостоятельное изучение учебной литературы, подготовка к лабораторным занятиям, выполнение индивидуального задания по работе	Устный опрос по работе №7 Устный опрос по работе №8 Устный опрос по работе №11	ОПК-6, ПК-5
Итого по разделу		4	8/4И		8			
5. Интерфейсы доступа к серверам баз данных								
5.1 Объектные интерфейсы ADO, DAO	8	1		2/2И	4	Самостоятельное изучение учебной литературы, подготовка к практическим занятиям, выполнение индивидуального задания по работе	Устный опрос по работе №4	ПК-5

5.2 Низкоуровневый интерфейс ODBC		1		2	4	Самостоятельное изучение учебной литературы, подготовка к практическим занятиям, выполнение индивидуального задания по работе	Устный опрос по работе №5	ПК-5
Итого по разделу		2		4/2И	8			
6. Взаимодействие баз данных с системами диспетчерского управления и сбора данных								
6.1 Средства доступа SCADA к базам данных	8	1		3/2И	4	Самостоятельное изучение учебной литературы, подготовка к практическим занятиям, выполнение индивидуального задания по работе	Устный опрос по работе №5	ОПК-9, ПК-5
6.2 Системы ведения архивов технологической информации, ее представления и обработки в SCADA		1		4	4	Самостоятельное изучение учебной литературы, подготовка к практическим занятиям, выполнение индивидуального задания по работе	Устный опрос по работе №5	ОПК-9, ПК-5
Итого по разделу		2		7/2И	8			
7. Технологии OLAP и Data mining в задачах обработки данных технологических процессов								
7.1 Многомерное представление данных. Язык запросов MDX	8	1			2	Самостоятельное изучение учебной и научно-технической литературы. Поиск дополнительной литературы по теме	Устный опрос	ПК-5
7.2 Принципы, стандарты и средства OLAP и Data mining		1	4		1,8	Самостоятельное изучение учебной литературы, подготовка к лабораторным занятиям, выполнение индивидуального задания по работе	Устный опрос по работе №6	ПК-5, ОПК-6, ОПК-9
Итого по разделу		2	4		3,8			
Итого за семестр		22	22/8И	11/4И	51,8		зачёт	



Итого по дисциплине	22	22/8И	11/4И	51,8		зачет	ОПК-6,ПК-5,ОПК-9
---------------------	----	-------	-------	------	--	-------	------------------

## 5 Образовательные технологии

Для реализации предусмотренных видов учебной работы в качестве образовательных технологий в преподавании дисциплины «Базы данных в АСУ ТП» используются:

Традиционные образовательные технологии – информационная лекция (вводную лекцию, где дает первое представление о предмете и знакомство студентов с назначением и задачами курса); лекции – консультации, изложение нового материала сопровождается постановкой вопросов и дискуссией в поисках ответов на эти вопросы; лабораторные работы.

Технологии проблемного обучения – проблемные лекции является результатом усвоения полученной информации посредством постановки проблемного вопроса и поиска путей его решения; лабораторные работы с использованием проблемного обучение, которое заключается в стимулировании студентов к самостоятельной «добыче» знаний, необходимых для решения конкретной проблемы.

Информационно-коммуникационные образовательные технологии – в ходе проведения лекционных занятий предусматривается использование электронного демонстрационного материала (лекции-визуализации), использование Интернет ресурсов для промежуточных аттестаций и проверки остаточных знаний

Лекционный материал закрепляется в ходе лабораторных и практических работ, на которых выполняются групповые или индивидуальные задания по пройденной теме.

Лабораторные занятия проводятся в интерактивной форме с использованием следующих методов интерактивного обучения:

- актуализация познавательной деятельности учащихся путем побуждения к осмыслению логики и последовательности проведения научного исследования, к выделению в нем главных и наиболее существенных этапов; при этом определяется конечная цель исследования, а пути его проведения и формы представления результата обучающийся выбирает сам;

- отсутствие жестко регламентированного порядка выполнения работы по обработке экспериментальных данных, когда студент оперирует вспомогательной информацией о способах поиска необходимых программных средств, функций, протоколов передачи и обработки данных, что вырабатывает способность к познанию;

- при постановке и анализе результатов исследования для достижения поставленных целей обучающиеся должны делать сравнения, сопоставлять новые факты, приемы использованные другими участниками группы, обращать внимание на причины, вызывающие то или иное явление и быть способными продемонстрировать индивидуальность своего подхода к решению задачи;

- проведение занятий в форме поиска причин допущенных ошибок при проведении исследования, причин несовпадения результатов с полученными другими группами обучающихся, побуждение к стремлению находить и устранять чужие и свои ошибки.

Самостоятельная работа стимулирует студентов к самостоятельной проработке в процессе выполнения контрольных работ, а также в процессе подготовки к устному опросу, тестированию и итоговой аттестации.

В ходе проведения лекционных занятий предусматривается:

- использование электронного демонстрационного материала по современной измерительной технике;

- использование электронных учебников по отдельным темам занятий;

- активные и интерактивные формы обучения: вариативный опрос, дискуссии, устный опрос, тестовый опрос, индивидуальная «защита» лабораторных работ и т.д.

## **6 Учебно-методическое обеспечение самостоятельной работы обучающихся**

Представлено в приложении 1.

## **7 Оценочные средства для проведения промежуточной аттестации**

Представлены в приложении 2.

## **8 Учебно-методическое и информационное обеспечение дисциплины (модуля)**

### **а) Основная литература:**

1. Советов, Б. Я. Базы данных : учебник для вузов / Б. Я. Советов, В. В. Цехановский, В. Д. Чертовской. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2020. — 420 с. — (Высшее образование). — ISBN 978-5-534-07217-4. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/449940> (дата обращения: 20.09.2020).

2. Стружкин, Н. П. Базы данных: проектирование: учебник для вузов / Н. П. Стружкин, В. В. Годин. — Москва : Издательство Юрайт, 2020. — 477 с. — (Высшее образование). — ISBN 978-5-534-00229-4. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/450165> (дата обращения: 20.09.2020).

### **б) Дополнительная литература:**

1. Илюшечкин, В. М. Основы использования и проектирования баз данных : учебник для вузов / В. М. Илюшечкин. — Москва : Издательство Юрайт, 2020. — 213 с. — (Высшее образование). — ISBN 978-5-534-03617-6. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/449679> (дата обращения: 20.09.2020).

2. Гордеев, С. И. Организация баз данных в 2 ч. Часть 1 : учебник для вузов / С. И. Гордеев, В. Н. Волошина. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2020. — 310 с. — (Высшее образование). — ISBN 978-5-534-04469-0. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/452928> (дата обращения: 20.09.2020).

3. Гордеев, С. И. Организация баз данных в 2 ч. Часть 2 : учебник для вузов / С. И. Гордеев, В. Н. Волошина. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2020. — 513 с. — (Высшее образование). — ISBN 978-5-534-04470-6. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/454122> (дата обращения: 20.09.2020).

4. Парфенов, Ю. П. Постреляционные хранилища данных : учебное пособие для вузов / Ю. П. Парфенов. — Москва : Издательство Юрайт, 2020. — 121 с. — (Высшее образование). — ISBN 978-5-534-09837-2. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/453758> (дата обращения: 20.09.2020).

### **в) Методические указания:**

1. Рябчиков, М. Ю. Базы данных и информационные системы в АСУ ТП : учебное пособие / М. Ю. Рябчиков, Е. С. Рябчикова. - Магнитогорск : МГТУ, 2013. - 1 электрон. опт. диск (CD-ROM). - Загл. с титул. экрана. - URL: <https://magtu.informsystema.ru/uploader/fileUpload?name=940.pdf&show=dcatalogues/1/1118971/940.pdf&view=true> (дата обращения: 14.05.2020). - Макрообъект. - Текст : электронный. - Сведения доступны также на CD-ROM.

2. Маркин, А. В. Программирование на SQL в 2 ч. Часть 1 : учебник и практикум для вузов / А. В. Маркин. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2020. — 403 с. — (Высшее образование). — ISBN 978-5-534-12256-5. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/452357> (дата обращения: 20.09.2020).

3. Маркин, А. В. Программирование на SQL в 2 ч. Часть 2 : учебник и практикум

для вузов / А. В. Маркин. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2020. — 340 с. — (Высшее образование). — ISBN 978-5-534-12258-9. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: <https://urait.ru/bcode/451185> (дата обращения: 20.09.2020).

4. Рябчиков, М. Ю. Программирование системы диспетчерского управления : учебное пособие / М. Ю. Рябчиков, Е. С. Рябчикова ; МГТУ. - Магнитогорск : МГТУ, 2016. - 1 электрон. опт. диск (CD-ROM). - Загл. с титул. экрана. - URL: <https://magtu.informsystema.ru/uploader/fileUpload?name=2734.pdf&show=dcatalogues/1/1132625/2734.pdf&view=true> (дата обращения: 14.05.2020). - Макрообъект. - Текст : электронный. - Сведения доступны также на CD-ROM.. – Макрообъект.

Методические рекомендации по выполнению лабораторных и практических работ приведены также в приложении 3.

#### г) Программное обеспечение и Интернет-ресурсы:

##### Программное обеспечение

Наименование ПО	№ договора	Срок действия лицензии
MS Office 2003 Professional	№ 135 от 17.09.2007	бессрочно
MS Windows XP Professional(для классов)	Д-1227-18 от 08.10.2018	11.10.2021
Oracle Open JDK	свободно распространяемое ПО	бессрочно
Oracle SQL Developer	свободно распространяемое ПО	бессрочно
MS SQL Server Management Studio	свободно распространяемое ПО	бессрочно
Oracle My SQL Workbench Community Edition	свободно распространяемое ПО	бессрочно
Oracle SQL Developer Data Modeler	свободно распространяемое ПО	бессрочно
MS Office Access Prof 2007(для классов)	Д-1227-18 от 08.10.2018	11.10.2021
FAR Manager	свободно распространяемое ПО	бессрочно

### Профессиональные базы данных и информационные справочные системы

Название курса	Ссылка
Электронная база периодических изданий East View Information Services, ООО «ИВИС»	<a href="https://dlib.eastview.com/">https://dlib.eastview.com/</a>
Национальная информационно-аналитическая система – Российский индекс научного цитирования	URL: <a href="https://elibrary.ru/project_risc.asp">https://elibrary.ru/project_risc.asp</a>
Поисковая система Академия Google (Google Scholar)	URL: <a href="https://scholar.google.ru/">https://scholar.google.ru/</a>
Информационная система - Единое окно доступа к информационным ресурсам	URL: <a href="http://window.edu.ru/">http://window.edu.ru/</a>
Федеральное государственное бюджетное учреждение «Федеральный институт промышленной собственности»	URL: <a href="http://www1.fips.ru/">http://www1.fips.ru/</a>
Российская Государственная библиотека. Каталоги	<a href="https://www.rsl.ru/ru/4readers/catalogues/">https://www.rsl.ru/ru/4readers/catalogues/</a>
Электронные ресурсы библиотеки МГТУ им. Г.И. Носова	<a href="http://magtu.ru:8085/marcweb2/Default.asp">http://magtu.ru:8085/marcweb2/Default.asp</a>
Федеральный образовательный портал – Экономика. Социология. Менеджмент	<a href="http://ecsocman.hse.ru/">http://ecsocman.hse.ru/</a>
Университетская информационная система РОССИЯ	<a href="https://uisrussia.msu.ru">https://uisrussia.msu.ru</a>
Международная наукометрическая реферативная и полнотекстовая база данных научных изданий «Web of science»	<a href="http://webofscience.com">http://webofscience.com</a>
Международная реферативная и полнотекстовая справочная база данных	<a href="http://scopus.com">http://scopus.com</a>
Международная база полнотекстовых журналов Springer Journals	<a href="http://link.springer.com/">http://link.springer.com/</a>
Международная коллекция научных протоколов по различным отраслям знаний	<a href="http://www.springerprotocols.com/">http://www.springerprotocols.com/</a>

Международная база научных материалов в области физических наук и инжиниринга	<a href="http://materials.springer.com/">http://materials.springer.com/</a>
Международная база справочных изданий по всем отраслям знаний SpringerReference	<a href="http://www.springer.com/references">http://www.springer.com/references</a>
Международная реферативная база данных по чистой и прикладной математике zbMATH	<a href="http://zbmath.org/">http://zbmath.org/</a>
Международная реферативная и полнотекстовая справочная база данных научных изданий «Springer Nature»	<a href="https://www.nature.com/siteindex">https://www.nature.com/siteindex</a>
Архив научных журналов «Национальный электронно-информационный конкорциум» (НП НЭИКОН)	<a href="https://archive.neicon.ru/xmlui/">https://archive.neicon.ru/xmlui/</a>

### **9 Материально-техническое обеспечение дисциплины (модуля)**

Материально-техническое обеспечение дисциплины включает:

1. Учебные аудитории для проведения занятий лекционного типа  
Мультимедийные средства хранения, передачи и представления информации
2. Учебная аудитория для проведения практических занятий и лабораторных работ:  
компьютерный класс  
Персональные компьютеры с пакетом MS Office, выходом в Интернет и с доступом в электронную информационно-образовательную среду университета
3. Помещения для самостоятельной работы обучающихся  
Персональные компьютеры с пакетом MS Office, выходом в Интернет и с доступом в электронную информационно-образовательную среду университета
4. Учебные аудитории для групповых и индивидуальных консультаций, текущего контроля и промежуточных консультаций  
Доска, мультимедийный проектор, экран
5. Помещение для хранения и профилактического обслуживания учебного оборудования  
Стеллажи для хранения учебно-методической документации

### Учебно – методическое обеспечение самостоятельной работы обучающихся

По дисциплине «Базы данных в АСУ ТП» предусмотрена аудиторная и внеаудиторная самостоятельная работа обучающихся.

Аудиторная самостоятельная работа студентов предполагает выполнение и защиту лабораторных и практических работ, решение индивидуальных задач.

<b>Перечень лабораторных и практических работ</b>	<b>Вопросы к защите</b>
№1. Язык запросов SQL. Оператор Select (порядок выполнения в [1] раздела методических указаний)	1. Понятие целостности данных. Ограничения целостности. 2. Основные виды связей между различными типами объектов. 3. Структура оператора SELECT. 4. Основные операции реляционной алгебры. 5. Реляционная модель представления данных.
№2. Язык запросов SQL. Агрегатные функции и составные запросы (порядок выполнения в [1] раздела методических указаний)	1. Структура оператора SELECT с учетом операций группировки и отбора групп. 2. Выполните обзор агрегатных функций. 3. Особенности применения агрегатных функций без группировки. 4. Как применить агрегатную функцию к результату работы другой агрегатной функции. 5. Какие типы составных SQL операторов с SELECT Вы знаете?
№3. Язык запросов SQL. Команды изменения объектов базы данных (порядок выполнения в [1] раздела методических указаний)	1. Команды работы с записями. 2. Команды создания, удаления и модификации объектов базы, кроме записей. 3. Порядок назначения первичных и внешних ключей. 4. Порядок удаления записей с первичными и внешними ключами. 5. Команды назначения значений полей по умолчанию и установки прочих ограничений.
№4. Создание приложений, использующих систему подключения ADO (порядок выполнения в [1] раздела методических указаний)	1. Организация подключения к базе через ADO. Синтаксис строки подключения. 2. Объекты и методы объекта Connection. 3. Объекты, методы и свойства объекта RecordSet. 4. Понятие курсора и команды управления курсором ADO. 5. Дайте сравнительную оценку интерфейсам ADO и DAO.
№5. Работа SCADA Intouch с базами данных (порядок выполнения в [2] раздела методических указаний)	1. Способы взаимодействия Intouch с базами данных. 2. Организация сбора экспериментальной информации в условиях крупного производства, управляемого распределенной системой включающей контроллеры и станции SCADA систем. 3. Способы получения данных с применением SCADA системы Intouch. 4. Динамический SQL в Intouch с применением ODBC. 5. Классификаций функций - упрощенных аналогов SQL в Intouch.
№6. Автоматизированная работа Intouch с	1. Задачи автоматического контроля качества процесса. 2. Технологии OLAP при управлении технологическим процессом. 3. Задачи Data mining.

<b>Перечень лабораторных и практических работ</b>	<b>Вопросы к защите</b>
базами данных через утилиту SPC PRO (порядок выполнения в [2] раздела методических указаний)	4. Назначение диаграмм Парето? 5. Функции инструмента SPC PRO Intouch?
№7. Программные блоки PL/SQL (порядок выполнения в приложении)	1. Структура программного блока PL/SQL. 2. Способы объявления переменных и констант Oracle. 3. Условное управление в PL/SQL. 4. Особенности применения оператора Select в программных блоках 5. Особенности работы с WEB интерфейсом сервера Oracle.
№8. Пакеты и курсоры PL/SQL (порядок выполнения в приложении)	1. Структура и название пакета Oracle. 2. Порядок создания и отладки пакета. 3. Синтаксис объявления курсора. Варианты объявления. 4. Применение переменной в теле курсора. 5. Курсорные циклы Oracle.
№9. Триггеры PL/SQL (порядок выполнения в приложении)	1. Назначение триггера уровня записи. 2. События, обрабатываемые триггерами. 3. Ограничения триггера. 4. Рекурсивные триггеры. 5. Определение условий применения триггера.
№10. Динамические запросы PL/SQL (порядок выполнения в приложении)	1. Понятие и назначение динамического запроса. 2. Порядок выполнения динамического SQL оператора без оператора SELECT. 3. Порядок выполнения динамического SQL оператора с оператором SELECT. 4. Обзор функций пакета DBMS_SQL. 5. Отличия модульного и встроенного SQL при программировании.
№11. Язык транзакций T-SQL (порядок выполнения в [1] раздела методических указаний)	1. Особенности выполнения последовательности SQL операторов на T-SQL. 2. Порядок объявления и способы присвоения значений переменным на T-SQL. 3. Глобальные переменные T-SQL. 4. Оператор EXISTS на T-SQL. 5. Вывод данных в программах T-SQL.
№12. Работа с процедурами и триггерами T-SQL (порядок выполнения в приложении)	1. Особенности объявления и использования процедуры, с выходными и входными параметрами. 2. Процедура с параметром – курсором. 3. Как вызвать процедуру? 4. Как выполнить шифрование кода триггера? 5. Может ли триггер обращаться к другим базам?
№13. Работа с курсорами T-SQL (порядок выполнения в [1] раздела методических указаний)	1. Типы курсоров T-SQL. 2. Понятие API курсора. 3. В чем отличие динамического курсора от статического? 4. Синтаксис объявления курсора на PL-SQL. 5. Назначение параметра SCROLL при объявлении курсора.



## Примеры вариантов заданий на самостоятельную работу

### Проектирование базы данных

#### *Порядок выполнения:*

1. Дополнить список атрибутов, заданных в методических указаниях [1] десятью своими атрибутами.
2. Выделить отдельные сущности. Графически изобразить связи между ними. Параметры связей принять самостоятельно исходя из предметной области.
3. В словесной форме описать значение связей.
4. Согласно правилам метода сущность-связь сформировать набор таблиц.
5. Проверить и обеспечить соответствие таблиц третьей-усиленной нормальной форме.
6. Графические представить итоговые таблицы базы с учетом связей.
7. Привести команды создания на SQL таблиц и команды добавления в таблицы 2-3 записей.
8. Реализовать разработанную базу с использованием CASE средства проектирования (ER-WIN или Access).

### Варианты исходных данных:

#### **Вариант 1**

Необходимо составить базу размещения поставок изделий по складам. В каждой поставке может быть несколько видов изделий, каждая поставка организуется одним поставщиком. У одного поставщика может быть несколько поставок.

Наименование полей: номер склада, фамилия и инициалы заведующего, телефон, номер поставки, номер вида изделия в поставке, код поставщика, название поставщика, город поставщика, шифр изделия, название изделия, дата поставки, размер поставки, цена изделия в поставке.

#### **Вариант 2**

Необходимо составить базу цен на изделия, выпускаемые предприятиями различных отраслей промышленности.

Наименование полей: код предприятия, название предприятия, код отрасли, название отрасли, ФИО лица ответственного за отрасль, адрес предприятия, название изделия, шифр изделия, цена изделия квартала 1, цена изделия квартала 2, цена изделия квартала 3, цена изделия квартала 4, годовой выпуск изделия.

#### **Вариант 3**

Необходимо составить базу списка школ разных районов города и обучающихся в них учеников.

Наименование полей: район, код района, телефон отдела образования, номер школы, телефон школы, год открытия школы, количество учителей школы, количество учеников школы, номер ученика, фамилия ученика, домашний адрес ученика.

#### **Вариант 4**

Необходимо составить базу персонала организации и списка материально ответственных за средства, распределенные по различным подразделениям. Каждое средство может числиться только за одним подразделением.

Наименование полей: код подразделения, название подразделения, код сотрудника - материально ответственного, фамилия ответственного, должность ответственного, телефон ответственного, зарплата ответственного, инвентарный номер средства, наименование средства, стоимость, дата оприходования, норма амортизации в процентах годовая,

### **Вариант 5**

Необходимо составить базу набранных в университет студентов.

Наименование полей: код института, краткое название института, телефон, код студента, фамилия и инициалы, пол, дата рождения, средний балл по аттестату, шифр группы студентов, код специальности студентов, название специальности студентов, код старосты группы.

### **Вариант 6**

Необходимо составить базу данных по оказанию услуг сотрудниками разных отделов одной организации с различными клиентами.

Наименование полей: номер отдела, фамилия и инициалы начальника, название отдела, число сотрудников отдела, номер сотрудника, фамилия сотрудника, пол, должность, дата поступления, зарплата, дата оказания услуг, уплаченная клиентом сумма, код клиента, фамилия клиента, адрес клиента.

### **Вариант 7**

Необходимо составить базу данных товаров магазина, поставок на склад и продаж товаров клиентам.

Наименование полей: код магазина, название магазина, телефон магазина, название товара, шифр товара, цена, количество товара на складе, дата последнего поступления, количество товара в последней поставке, дата продажи товара, количество проданного товара, номер чека выданного клиенту.

### **Вариант 8**

Необходимо составить базу данных работы фирмы доставки посылок от поставщиков потребителям (адресат) через временные склады, расположенные в районе потребителей.

Наименование полей: код адресата, адрес адресата, фамилия адресата, телефон адресата, шифр посылки, дата отгрузки адресату с временного склада, цена посылки, номер временного склада, адрес поставщика, код поставщика, фамилия поставщика, способ доставки, адрес временного склада, текущее число свободных мест на складе, дата прибытия посылки на временный склад.

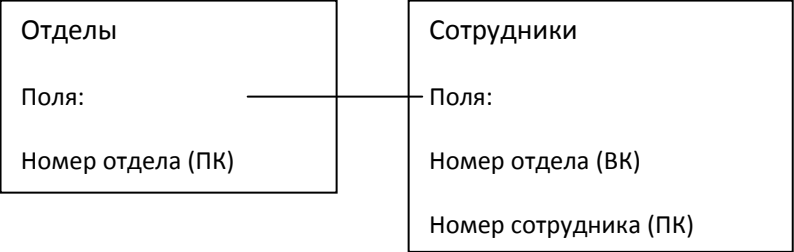
Приложение 2

**Оценочные средства для проведения промежуточной аттестации  
по дисциплине «Базы данных в АСУ ТП»**

**а) Планируемые результаты обучения и оценочные средства для проведения промежуточной аттестации:**

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
<b>ОПК-6 - способностью осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий</b>		
Знать	– структуры данных, используемые для представления типовых информационных объектов, типовые алгоритмы обработки данных; модели представления данных; – основные операторы языков запросов SQL, T-SQL, PL/SQL;	<ol style="list-style-type: none"> <li>1. Понятие целостности данных. Ограничения целостности.</li> <li>2. Основные виды связей между различными типами объектов.</li> <li>3. Структура оператора SELECT.</li> <li>4. Основные операции реляционной алгебры.</li> <li>5. Реляционная модель представления данных.</li> <li>6. Структура оператора SELECT с учетом операций группировки и отбора групп.</li> <li>7. Выполните обзор агрегатных функций.</li> <li>8. Особенности применения агрегатных функций без группировки.</li> <li>9. Как применить агрегатную функцию к результату работы другой агрегатной функции.</li> <li>10. Какие типы составных SQL операторов с SELECT Вы знаете?</li> <li>11. Команды работы с записями.</li> <li>12. Команды создания, удаления и модификации объектов базы, кроме записей.</li> <li>13. Порядок назначения первичных и внешних ключей.</li> <li>14. Порядок удаления записей с первичными и внешними ключами.</li> <li>15. Команды назначения значений полей по умолчанию и установки прочих ограничений.</li> <li>16. Понятие и назначение динамического запроса.</li> <li>17. Порядок выполнения динамического SQL оператора без оператора SELECT.</li> <li>18. Порядок выполнения динамического SQL оператора с оператором SELECT.</li> <li>19. Обзор функций пакета DBMS_SQL.</li> </ol>

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
		20. Отличия модульного и встроенного SQL при программировании.
Уметь	<ul style="list-style-type: none"> <li>– создавать отчеты с применением механизмов динамического формирования SQL запросов;</li> <li>– разрабатывать и программно реализовывать механизмы обеспечения целостности данных в созданной базе за счет применения подпрограмм и триггеров;</li> </ul>	<ol style="list-style-type: none"> <li>1. Написать функцию на PL/SQL, которая создает заданный по имени синоним и удаляет заданный по имени синоним, для заданной по имени таблицы.</li> <li>2. С использованием динамически формируемого оператора на PL-SQL удалить таблицу с заданным в переменной именем.</li> <li>3. Требуется создать две таблицы согласно заданному варианту задания и заполнить их данными по 1 записи в каждой из таблиц. Наименование полей первой таблицы: район, код района, телефон отдела образования. Наименование полей второй таблицы: код района, номер школы, телефон, год открытия, количество учителей, количество учеников.</li> <li>4. Сделать триггер, который при добавлении нового покупателя делает названия городов проживания всех покупателей такими же, как и их имена.</li> <li>5. Создать программу на VBA, содержащую две формы. Первая форма используется для добавления в базу новых покупателей. Вторая форма используется для вывода информации о покупателях, чей рейтинг меньше заданного.</li> </ol>
Владеть	языками программирования SQL, T-SQL, PL/SQL	Лабораторные работы №1,2,3,7,8,10,11
<b>ОПК-9 - способностью использовать навыки работы с компьютером, владеть методами информационных технологий, соблюдать основные требования информационной безопасности</b>		
Знать	<ul style="list-style-type: none"> <li>– компьютерные приложения для взаимодействия с СУБД разных производителей;</li> <li>– средства проектирования баз данных.</li> <li>– принципы работы сред программирования встроенных в SCADA Intouch и WinCC и возможности по их</li> </ul>	<ol style="list-style-type: none"> <li>1. Особенности работы с WEB интерфейсом сервера Oracle.</li> <li>1. Функции инструмента SPC PRO Intouch?</li> <li>2. Порядок проектирования создания базы в ER-WIN.</li> <li>3. Порядок создания базы в среде Access.</li> <li>4. Динамический SQL в Intouch с применением ODBC.</li> <li>5. Классификаций функций - упрощенных аналогов SQL в Intouch.</li> <li>6. Назначение триггера уровня записи.</li> <li>7. События, обрабатываемые триггерами.</li> <li>8. Ограничения триггера.</li> </ol>

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
	<p>применению в задачах автоматизированного сбора, обработки и хранения информации;</p>	<p>9. Рекурсивные триггеры.  10. Определение условий применения триггера.  11. Особенности объявления и использования процедуры, с выходными и входными параметрами.  12. Процедура с параметром – курсором.  13. Как вызвать процедуру?  14. Как выполнить шифрование кода триггера?  15. Может ли триггер обращаться к другим базам?  16. Типы курсоров T-SQL.  17. Понятие API курсора.  18. В чем отличие динамического курсора от статического?  19. Синтаксис объявления курсора на PL-SQL.  20. Назначение параметра SCROLL при объявлении курсора.</p>
<p>Уметь</p>	<ul style="list-style-type: none"> <li>– использовать стандартные пакеты прикладных программ для решения практических задач формирования запросов к базам;</li> <li>– осуществлять проектирование базы данных с применением CASE систем проектирования (ERWin);</li> <li>– строить диалоговые окна SCADA систем Intouch или WinCC взаимодействующие с СУБД; создавать системы сбора экспериментальных данных с применением открытых интерфейсов, серверов ввода-вывода данных и систем диспетчерского управления (Intouch, WinCC), а также определять требуемый при этом состав</li> </ul>	<p>1. С использованием клиента базы Oracle вывести максимальный возраст сотрудников для каждого отдела в момент поступления на работу. Для расчета возраста в днях на конкретную дату использовать разницу между датой поступления и датой рождения.</p> <div style="text-align: center; margin: 10px 0;">  <pre> graph LR     A[Отделы] --- B[Сотрудники]     A --- A1[Поля: Номер отдела (ПК)]     B --- B1[Поля: Номер отдела (ВК), Номер сотрудника (ПК)] </pre> </div> <p>2. В среде ER-Win спроектировать методом сущность-связь БД. Имеется три сущности Водители такси, Машины и Гаражи. Определить классы принадлежности сущностей и степень их связи и указать в словесной форме,</p>

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
	прикладного программного обеспечения;	<p>что означают принятые ограничения.</p> <p>3. В среде Intouch реализовать окно для вывода информации о студентах из базы данных Access:</p> <div data-bbox="1227 596 1785 890" style="border: 1px solid black; padding: 5px; margin: 10px auto; width: fit-content;"> <p style="text-align: right;">Журнал</p> <p>Номер события = #</p> <p>Режим = #</p> <p>Фамилия = #</p> <p>Группа = #</p> <p>Год набора = #</p> <p style="text-align: center;"> <input type="button" value="Считать"/> <input type="button" value="Первая"/> <input type="button" value="Следующая"/> </p> <p>Заданная фамилия = #</p> </div>
Владеть	<p>– навыками работы с современными программными средствами исследования и проектирования систем управления; навыками работы с техническими и программными средствами автоматизированного сбора, хранения и обработки данных</p> <p>– языками оболочек WinCC или Intouch на уровне достаточном для организации связи с СУБД и создания эффективного интерфейса взаимодействия с технологом – оператором и СУБД;</p>	<p>Лабораторные работы №1,2,3,7,8,10,11</p> <p>Практическая работа №5.</p> <p>Контрольная работа «Проектирование базы данных».</p>

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
<b>ПК-5 - способностью осуществлять сбор и анализ исходных данных для расчета и проектирования систем и средств автоматизации и управления</b>		
Знать	<ul style="list-style-type: none"> <li>– интерфейсы взаимодействия разрабатываемых приложений с СУБД;</li> <li>– возможности, связанные с применением технологий OLAP и Data mining в СУБД с реляционной и многомерной моделью представления информации;</li> <li>– методы, формы и этапы проектирования баз данных;</li> </ul>	<ol style="list-style-type: none"> <li>1. Поясните правила метода “Сущность-связь” для проектирования баз данных.</li> <li>2. Поясните требования к нормальным формам.</li> <li>3. Что такое транзитивная зависимость?</li> <li>4. Что такое третья усиленная нормальная форма.</li> <li>5. Что такое степень связи и класс принадлежности сущности?</li> <li>6. Структура программного блока PL/SQL.</li> <li>7. Способы объявления переменных и констант Oracle.</li> <li>8. Условное управление в PL/SQL.</li> <li>9. Особенности применения оператора Select в программных блоках</li> <li>10. Структура и название пакета Oracle.</li> <li>11. Порядок создания и отладки пакета.</li> <li>12. Синтаксис объявления курсора. Варианты объявления.</li> <li>13. Применение переменной в теле курсора.</li> <li>14. Курсорные циклы Oracle.</li> <li>15. Особенности выполнения последовательности SQL операторов на T-SQL.</li> <li>16. Порядок объявления и способы присвоения значений переменным на T-SQL.</li> <li>17. Глобальные переменные T-SQL.</li> <li>18. Оператор EXISTS на T-SQL.</li> <li>19. Вывод данных в программах T-SQL.</li> <li>20. Организация подключения к базе через ADO. Синтаксис строки подключения.</li> <li>21. Объекты и методы объекта Connection.</li> <li>22. Объекты, методы и свойства объекта RecordSet.</li> <li>23. Понятие курсора и команды управления курсором ADO.</li> <li>24. Дайте сравнительную оценку интерфейсам ADO и DAO.</li> <li>25. Способы взаимодействия Intouch с базами данных.</li> <li>26. Организация сбора экспериментальной информации в условиях крупного производства, управляемого распределенной системой включающей контроллеры и станции SCADA систем.</li> </ol>

Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства						
		27. Способы получения данных с применением SCADA системы Intouch. 28. Задачи автоматического контроля качества процесса. 29. Технологии OLAP при управлении технологическим процессом. 30. Задачи Data mining. 31. Назначение диаграмм Парето?						
Уметь	<ul style="list-style-type: none"> <li>– применять принципы и законы реляционной алгебры при самостоятельном проектировании реляционных баз данных;</li> <li>– настраивать и программировать интерфейсы взаимодействия с базами данных;</li> <li>– разрабатывать клиентские приложения с применением объектных интерфейсов подключения ADO и DAO;</li> <li>– создавать приложения обработки данных на основе модульного программирования;</li> <li>– решать исследовательские и проектные задачи с использованием баз данных;</li> </ul>	<ol style="list-style-type: none"> <li>1. Спроектировать методом сущность-связь БД. Имеется три сущности Бензин, Бензоколонки и Заправки. Определить классы принадлежности сущностей и степень их связи и указать в словесной форме, что означают принятые ограничения. Самостоятельно придумать названия первичных ключей сущностей, а остальные поля показать как столбец - Прочие.</li> <li>2. В среде Intouch организовать подключение к базе Access через интерфейс ODBC и вывести информацию об ошибках подключения:           <div style="border: 1px solid black; padding: 5px; margin: 10px 0;"> <table style="width: 100%; border-collapse: collapse;"> <tr> <td style="border: 1px solid black; padding: 5px; text-align: center;">Подключение</td> <td style="padding: 5px;"><b>Номер соединения = #</b></td> </tr> <tr> <td style="border: 1px solid black; padding: 5px; text-align: center;">Отключение</td> <td style="padding: 5px;"><b>Код ошибки = #</b></td> </tr> <tr> <td style="padding: 5px;"></td> <td style="padding: 5px;"><b>Описание ошибки = #</b></td> </tr> </table> </div> </li> <li>3. В среде VBA для Excel выполнить подключения к базе данных Access через интерфейс ADO и сформировать команды для добавления в базу новых покупателей и вывода информации о покупателях, чей рейтинг меньше заданного.</li> <li>4. На PL-SQL создать пакет work_employee содержащий функцию перевода сотрудника (заданного по номеру) в другой отдел.</li> <li>5. С использованием SQL рассчитать среднее значение столбца данных.</li> <li>6. С использованием SQL рассчитать дисперсию по заданному столбцу данных.</li> </ol>	Подключение	<b>Номер соединения = #</b>	Отключение	<b>Код ошибки = #</b>		<b>Описание ошибки = #</b>
Подключение	<b>Номер соединения = #</b>							
Отключение	<b>Код ошибки = #</b>							
	<b>Описание ошибки = #</b>							



Структурный элемент компетенции	Планируемые результаты обучения	Оценочные средства
Владеть	методами построения современных проблемно-ориентированных прикладных программных средств;	Лабораторные работы №6, 9, 12, 13. Практические работы №4,5

**б) Порядок проведения промежуточной аттестации, показатели и критерии оценивания:**

Промежуточная аттестация по дисциплине «Базы данных в АСУ ТП» включает теоретические вопросы, позволяющие оценить уровень усвоения обучающимися знаний, и практические задания, выявляющие степень сформированности умений и владений, проводится в форме зачета.

Показатели и критерии оценивания зачета:

Оценка	Критерии
Зачтено	<ol style="list-style-type: none"><li>1. Сформированы систематические знания об основных задачах в области поиска, хранения, обработки и анализ информации из различных источников и баз данных.</li><li>2. Сформированы навыки владения языком запросов SQL.</li><li>3. Получены навыки владения компьютерными приложениями для взаимодействия с СУБД разных производителей и средствами проектирования баз данных.</li><li>4. Выполнена самостоятельная работа, подтверждающая знания и умения в области проектирования баз данных.</li><li>5. Продемонстрированы навыки в области организации взаимодействия с базами данных клиентских приложений через интерфейс ADO, а также в области настройки систем диспетчерского управления для организации сбора и представления информации.</li></ol>
Не зачтено	<ol style="list-style-type: none"><li>1. Усвоено основное содержание материала, но изложено фрагментарно, не всегда последовательно.</li><li>2. Определения и понятия даны не чётко.</li><li>3. Допущены ошибки при формировании структуры SQL оператора.</li><li>4. Практические навыки слабые.</li><li>5. Основное содержание учебного материала не раскрыто.</li><li>6. Не даны ответы на дополнительные вопросы преподавателя.</li><li>7. Допущены грубые ошибки в определениях.</li><li>8. Отсутствуют навыки владения компьютерными приложениями для взаимодействия с СУБД разных производителей и средствами проектирования баз данных.</li></ol>

**Методические указания к выполнению лабораторных и практических работ****Работа №7. Программные блоки PL/SQL****Общие сведения****1. Присваивания переменным**

Вы можете присваивать переменным значения двумя способами.

Первый способ использует оператор присваивания := (двоеточие, за которым следует знак равенства).

```
starting_empno CONSTANT NUMBER(4) := 7902;
```

Второй способ присвоить значение переменной - это извлечь в нее значение из базы данных посредством фразы INTO предложения SELECT или FETCH.

```
SELECT sal, mgr INTO salary, mgr_num FROM .....
```

**2. Атрибуты %TYPE и %ROWTYPE**

Атрибут %TYPE представляет тип данных переменной, константы или столбца. Он особенно полезен при объявлении переменной, которая ссылается на столбец из таблицы базы данных. Например, таблица emp содержит столбец с именем sal. Чтобы дать переменной salary тот же тип данных, что у столбца sal объявите salary с использованием атрибута %TYPE:

```
salary emp.sal %TYPE;
```

Атрибут %ROWTYPE обозначает тип записи, представляющей строку в таблице.

**3. Операторы цикла WHILE - LOOP**

Общая структура оператора loop:

```
WHILE условие LOOP
```

```
-- поместите сюда последовательность предложений, которые выполняются пока условие верно
```

```
...
```

```
END LOOP;
```

**4. Пример программного блока**

В примере отыскивается первый сотрудник, имеющий жалование не менее \$4000, по иерархии начальников служащего 7902 и его ФИО и зарплата копируется в предварительно созданную таблицу temp:

```

DECLARE
    salary          emp.sal%TYPE;
    mgr_num         emp.mgr%TYPE;
    last_name      emp.ename%TYPE;
    starting_empno CONSTANT NUMBER(4) := 7902;
BEGIN
    SELECT sal, mgr INTO salary, mgr_num FROM emp
        WHERE empno = starting_empno;
    WHILE salary < 4000 LOOP
        SELECT sal, mgr, ename INTO salary, mgr_num, last_name
            FROM emp
            WHERE empno = mgr_num;
    END LOOP;

    INSERT INTO temp VALUES (null, salary, last_name);
    COMMIT;
END;

```

### Порядок выполнения работы

1. Запустить Oracle Navigator и создать таблицу temp(NUMBER, NUMBER, VARCHAR2).
2. Проверить работу примера.
3. В таблицу temp вывести зарплаты и фамилии самого высокооплачиваемого и низкооплачиваемого сотрудников.
4. В таблицу temp с использованием цикла записать значения X и функции  $Y=3X+X^2$ . При этом X изменяется от 1 до значения, при котором Y становится больше 200.

## Работа №8. Пакеты и курсоры PL/SQL

### Общие сведения

Если вы объявляете курсор, извлекающий, скажем, фамилию, жалование, дату приема и должность сотрудника, то PL/SQL позволяет вам создать запись, содержащую такую же информацию. Вы делаете это с помощью атрибута %ROWTYPE. Предположим, вы записали следующие объявления курсора и записи:

```

DECLARE
    CURSOR c1 IS SELECT ename, sal, hiredate, job FROM emp;
    emp_rec c1%ROWTYPE;
    ...

```

При выполнении предложения "FETCH c1 INTO emp\_rec;" значение столбца ename таблицы emp будет присвоено полю ename записи emp\_rec, значение столбца sal будет присвоено полю sal, и т.д.

Запись %ROWTYPE

Запись emp\_rec

emp_rec.ename	JAMES
emp_rec.sal	950.00
emp_rec.hiredate	03-DEC-81

emp\_rec.job            CLERK

Пример использования курсора для выгрузки имен всех сотрудников имеющих (зарплату+200) более 2000 в отдельную таблицу:

```
DECLARE
  CURSOR my_cursor IS SELECT sal+200 wages, ename
    FROM emp;
  my_rec my_cursor%ROWTYPE;
BEGIN
  OPEN my_cursor;
  LOOP
    FETCH my_cursor INTO my_rec;
    EXIT WHEN my_cursor%NOTFOUND;
    IF my_rec.wages > 2000 THEN
      INSERT INTO temp VALUES (null, my_rec.wages,
        my_rec.ename);
    END IF;
  END LOOP;
  CLOSE my_cursor;
  COMMIT;
END;
```

### Варианты заданий к работе

#### *Задание 1*

##### **Вариант 1.**

Создать пакет work\_employee содержащий функцию удаления сотрудника (заданного по фамилии).

##### **Вариант 2.**

Создать пакет work\_employee содержащий функцию изменения кода начальника сотрудника (заданного по номеру).

##### **Вариант 3.**

Создать пакет work\_employee содержащий функцию изменения зарплаты сотрудника (заданного по фамилии).

##### **Вариант 4.**

Создать пакет work\_employee содержащий функцию перевода сотрудника (заданного по номеру) в другой отдел.

##### **Вариант 5.**

Создать пакет work\_employee содержащий функцию добавления нового сотрудника.

##### **Вариант 6.**

Создать пакет work\_employee содержащий функцию изменяющую фамилию сотрудника (заданного по фамилии).

#### *Задание 2*

##### **Вариант 1.**

Найти три самые большие зарплаты и записать в таблицу temp сотрудников, которые их получают и сами зарплаты.

##### **Вариант 2.**

Вывести в таблицу temp названия двух самых высокооплачиваемых отделов (сумма зарплат сотрудников отдела).

**Вариант 3.**

Отсортировать сотрудников в порядке возрастания оклада sal и выгрузить фамилии и зарплаты каждого второго сотрудника в предварительно созданную таблицу базы данных SAL.

**Вариант 4.**

Вывести сотрудников в алфавитном порядке в таблицу temp.

**Вариант 5.**

Вывести в таблицу temp названия двух самых высокооплачиваемых отделов (сумма зарплат сотрудников отдела).

**Вариант 6.**

Удалить каждого второго сотрудника из тех, у кого нет подчиненных.

**Вариант 7.**

Удалить трех сотрудников, зарплата которых больше всего соответствует средней по всем сотрудникам.

**Вариант 8.**

Удалить два отдела, где работает больше всего сотрудников.

**Вариант 9.**

Перевести всех сотрудников двух отделов, где меньше всего сотрудников в тот отдел, где больше всего сотрудников.

**Вариант 10.**

Отсортируйте всех сотрудников в алфавитном порядке имен. Если зарплата каждого второго сотрудника меньше предыдущего, то удалите его.

## **Работа №9. Триггеры PL/SQL**

### **Общие сведения**

ТРИГГЕР БАЗЫ ДАННЫХ - это хранимая программная единица PL/SQL, ассоциированная с конкретной таблицей базы данных. ORACLE исполняет триггер базы данных автоматически каждый раз, когда данная операция SQL воздействует на эту таблицу. В отличие от подпрограмм, которые должны вызываться явно, триггер базы данных вызывается неявно. Можно использовать триггеры базы данных для:

- \* аудита (отслеживания) модификаций данных
- \* прозрачной журнализации (регистрации) событий
- \* реализации комплексных организационных правил
- \* автоматического вычисления значений столбцов
- \* осуществления сложных процедур защиты
- \* поддержки дублированных таблиц

С каждой таблицей можно ассоциировать до 12 триггеров базы данных. Для создания триггера базы данных требуется иметь привилегии CREATE TRIGGER, а также либо владеть ассоциированной таблицей, либо иметь привилегии ALTER для ассоциированной таблицы, либо иметь привилегии ALTER ANY TABLE.

Триггер базы данных состоит из трех частей: события триггера, необязательного ограничения триггера и действия триггера. Когда происходит событие триггера, триггер базы данных исполняется, и анонимный блок PL/SQL выполняет предписанное действие. Триггеры базы данных исполняются с привилегиями владельца, а не текущего пользователя. Поэтому владелец должен иметь должный доступ ко всем объектам, вовлекаемым в действие триггера.

Следующий пример иллюстрирует прозрачную журнализацию событий. Триггер базы данных с именем `georder` обеспечивает, что товар заказывается заново каждый раз, когда его имеющееся на складе количество (`qty_on_hand`) падает ниже пороговой точки.

```
CREATE TRIGGER reorder
/* событие триггера */
AFTER UPDATE OF qty_on_hand ON inventory -- таблица
FOR EACH ROW
/* ограничение триггера */
WHEN (new.reorderable = 'T')
BEGIN
/* действие триггера */
IF :new.qty_on_hand < :new.reorder_point THEN
    INSERT INTO pending_orders
        VALUES (:new.part_no, :new.reorder_qty, SYSDATE);
END IF;
END;
```

Имя в фразе `ON` идентифицирует таблицу базы данных, ассоциированную с триггером базы данных. Событие триггера специфицирует предложение манипулирования данными SQL, которое воздействует на таблицу. В данном случае это предложение `UPDATE`. Если предложение триггера сбивается, оно откатывается. Ключевое слово `AFTER` указывает, что триггер базы данных исполняется после того, как обновление выполнено.

По умолчанию, триггер базы данных выполняется один раз на всю таблицу. Необязательная фраза `FOR EACH ROW` указывает, что триггер должен выполняться один раз на каждую строку. Для того, чтобы триггер выполнился, однако, требуется, чтобы булевское выражение в фразе `WHEN` давало значение `TRUE`.

Префикс `:new` представляет собой **КОРРЕЛИРУЮЩЕЕ ИМЯ**, которое отсылает к вновь измененному значению столбца. Внутри триггера базы данных можно обращаться как к новому, так и к старому (`:old`) значениям столбцов в измененных строках. Заметьте, что в фразе `WHEN` двоеточие не используется. Можно использовать фразу `REFERENCING` (здесь не показана), чтобы заменить `:new` и `:old` другими коррелирующими именами.

За исключением предложений управления транзакциями, таких как `COMMIT` и `ROLLBACK`, в блоке `BEGIN..END` может появляться любое предложение SQL или процедурное предложение, включая вызовы подпрограмм. Триггер базы данных может также иметь секции `DECLARE` и `EXCEPTION`.

### Варианты заданий к работе

1. Создать триггер, который при изменении параметров сотрудника записывает в таблицу `temp` его фамилию и номер (`empno`).
2. Создать триггер, который при удалении сотрудника записывает в таблицу `temp` фамилии и номера (`empno`) всех его подчиненных.
3. Создать триггер, который при удалении сотрудника выводит сообщение:  
Сотрудник <ФИО> – удален  
(\(\n  
(=\*.\*)

(,,(")("")

4. При добавлении нового отдела в dept перевести в него сотрудников, которые работают в отделе, номер которого наименее отличается от номера нового отдела.
5. При удалении сотрудника сохранить имя его начальника в таблицу temp.
6. При изменении зарплаты сотрудника сохраняет в temp название всех отделов и суммарные зарплаты сотрудников по каждому отделу.
7. При изменении зарплаты, если новая зарплата меньше старой, заносит в temp надпись:  
“Имя сотрудника – повезло”, а в противном случае:  
“Имя сотрудника – не повезло”
8. При изменении названия отдела, если новое название начинается с буквы “Ы”, переносит в него всех сотрудников, имена которых начинаются с буквы “S”.

## Работа №10. Динамические запросы PL/SQL

### Общие сведения

В Oracle v7 отсутствуют такие типы данных в DBMS\_SQL как VARCHAR2S и VARCHAR2\_TABLE, которые появились начиная с версии v8 и соответственно отсутствует функция DEFINE ARRAY и PARSE с использованием VARCHAR2\_TABLE. Механизмы обработки в Oracle v7 предусматривают выгрузку полученных после выполнения SELECT значений в обычные переменные или обычные таблицы с указанием индекса.

В приведенном ниже примере выбираются непосредственно подчиненные сотруднику заданному по номеру начальнику:

```
DECLARE
v_mgr NUMBER := 7839;
v_statement1 VARCHAR2(256) := 'SELECT ename FROM emp WHERE ' ;
TYPE tb_type IS TABLE OF VARCHAR2(256) INDEX BY
BINARY_INTEGER;
v_statement1_tb1 tb_type;
v_emp_tb1 tb_type;
PROCEDURE run_qry ( p_statement1_tb1 tb_type,
                   p_mgr IN NUMBER,
                   p_emp_tb1 IN OUT tb_type)

IS
v_cid NUMBER;
v_rownum NUMBER;
v_arraysize NUMBER := 10;
v_ttt VARCHAR2(10);
v_st VARCHAR2(256);
BEGIN
v_cid:=DBMS_SQL.OPEN_CURSOR;
v_st := p_statement1_tb1(1) || p_statement1_tb1(2);
DBMS_SQL.PARSE(v_cid, v_st, DBMS_SQL.V7);
IF p_statement1_tb1(2) IS NOT NULL THEN
```



```

DBMS_SQL.BIND_VARIABLE(v_cid, ':mgr', p_mgr);
END IF;
DBMS_SQL.DEFINE_COLUMN(v_cid, 1, v_ttt, v_arraysize);
v_rownum := DBMS_SQL.EXECUTE(v_cid);
v_rownum := 1;
DBMS_OUTPUT.PUT_LINE('Помошники : ');
WHILE v_rownum = 1 LOOP
v_rownum := DBMS_SQL.FETCH_ROWS(v_cid);
DBMS_SQL.COLUMN_VALUE(v_cid, 1, v_ttt);
IF v_rownum = 1 THEN
DBMS_OUTPUT.PUT_LINE('ФИО = ' || v_ttt);
END IF;
END LOOP;
DBMS_SQL.CLOSE_CURSOR(v_cid);
END run_qry;
BEGIN
v_statement1_tb1(1) := v_statement1;
v_statement1_tb1(2) := 'mgr = :mgr';
run_qry(v_statement1_tb1, v_mgr, v_emp_tb1);
END;

```

При выборе данных с использованием динамического запроса следует придерживаться следующей последовательности действий:

1. Сформировать строку запроса в котором условия заданы неявно, например как: 'mgr = :mgr'.
2. Открыть курсор (DBMS\_SQL.OPEN\_CURSOR).
3. Выполнить синтаксический разбор оператора (запроса) DBMS\_SQL.PARSE.
4. Выполнить подстановку значений переменных в текст запроса. (DBMS\_SQL.BIND\_VARIABLE).
5. Указать какие переменные будут использоваться для приема данных каждого столбца (DBMS\_SQL.DEFINE\_COLUMN).
6. Выполнить запрос.
7. Извлечь строки (DBMS\_SQL.FETCH\_ROWS).
8. Считать в текущей строке значение каждого столбца (DBMS\_SQL.COLUMN\_VALUE).
9. Закрывать курсор (DBMS\_SQL.CLOSE\_CURSOR).

### **Задания по вариантам**

#### **Часть 1. Динамическое исполнение операторов без SELECT**

1. Создать синоним с заданным именем для заданной таблицы (заданные значения должны храниться в переменных).

2. Написать функцию, которая создает таблицу с заданным именем, содержащую заданное число полей (поля называются как имя + № поля) типа VARCHAR2 заданного одного размера.
3. Создать функцию, которая создает представление с заданным именем для заданного запроса с одним задаваемым в виде параметра условием.
4. Создать функцию, которая записывает фамилии сотрудников в заданное поле заданной таблицы согласно одному заданному условию отбора сотрудников.

## **Часть 2. Динамические запросы с SELECT**

1. Вывести зарплаты сотрудников, удовлетворяющие двум каким-либо критериям.
2. Вывести названия отделов, в которых число сотрудников больше заданного.
3. Вывести те должности из emp, которые занимают меньше заданного числа сотрудников.
4. Вывести такие сделки (табл. SALES\_ORDER – дата и кем и с кем заключена), где поле TOTAL находится в заданном диапазоне.
5. Вывести такие продукты (таблицы ITEM и PRODUCT) и их количество на складе, стоимость которых находится в заданных пределах.
6. Вывести фирмы (CUSTOMER – название и город) заключившие больше указанного количества сделок по указанному товару.