

**Федеральное государственное образовательное бюджетное учреждение
высшего образования
«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ
ПРАВИТЕЛЬСТВЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»
(Финансовый университет)**

**Департамент анализа данных,
принятия решений и финансовых технологий**

УТВЕРЖДАЮ

Проректор

по развитию

образовательных программ

Каменева Е.А. Каменева
«*26*» *ноября* 20 *19* г.

Кондрашов Ю.Н.

Программирование на языках Python и SQL

Рабочая программа дисциплины

для студентов, обучающихся по направлению подготовки
09.03.03 «Прикладная информатика»
профиль «ИТ – сервисы и технологии обработки данных в экономике и
финансах»

*Рекомендовано Ученым советом
факультета прикладной математики и информационных технологий
(протокол № 19 от 19.11.2019 г.)*

*Одобрено Советом учебно-научного департамента анализа данных,
принятия решений и финансовых технологий
(протокол № 3 от 15.10.2019 г.)*

Москва 2019

**Федеральное государственное образовательное бюджетное учреждение
высшего образования
«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ
ПРАВИТЕЛЬСТВЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»
(Финансовый университет)**

**Департамент анализа данных,
принятия решений и финансовых технологий**

Кондрашов Ю.Н.

Программирование на языках Python и SQL

Рабочая программа дисциплины

для студентов, обучающихся по направлению подготовки
09.03.03 «Прикладная информатика»
профиль «IT – сервисы и технологии обработки данных в экономике и
финансах»

Москва 2019

**Федеральное государственное образовательное бюджетное учреждение
высшего образования
«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ
ПРАВИТЕЛЬСТВЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»
(Финансовый университет)**

**Департамент анализа данных,
принятия решений и финансовых технологий**

УТВЕРЖДАЮ

Проректор

по развитию

образовательных программ

_____ Е.А. Каменева

26.11.2019 г.

Кондрашов Ю.Н.

Программирование на языках Python и SQL

Рабочая программа дисциплины

для студентов, обучающихся по направлению подготовки
09.03.03 «Прикладная информатика»
профиль «ИТ – сервисы и технологии обработки данных в экономике и
финансах»

*Рекомендовано Ученым советом
факультета прикладной математики и информационных технологий
(протокол № 19 от 19.11.2019 г.)*

*Одобрено Советом учебно-научного департамента анализа данных,
принятия решений и финансовых технологий
(протокол № 3 от 15.10.2019 г.)*

Москва 2019

УДК 004(073)
ББК 32.972.134

К64

Рецензент: В.И. Завгородний, д.э.н., профессор
департамента анализа данных, принятия
решений и финансовых технологий

Кондрашов Ю.Н. «Программирование на языках Python и SQL».
Рабочая программа дисциплины для студентов, обучающихся по
направлению подготовки 09.03.03 «Прикладная информатика», профиль
«ИТ – сервисы и технологии обработки данных в экономике и финансах»,
очная и очно-заочная формы обучения (программа подготовки бакалавра)
– М.: Финансовый университет при Правительстве Российской
Федерации, департамент «Анализ данных, принятия решений и
финансовых технологий», 2019.- 34 с.

Дисциплина «Программирование на языках Python и SQL» относится
к Модулю дисциплин по выбору, углубляющих освоение профиля.

В рабочей программе дисциплины представлены цели и задачи
дисциплины, требования к результатам освоения дисциплины,
содержание дисциплины, тематика практических занятий и технология их
проведения, формы самостоятельной работы студентов, система
оценивания, учебно-методическое и информационное обеспечение
дисциплины.

УДК 004(073)
ББК 32.972.134

Учебное издание
Кондрашов Юрий Николаевич
Программирование на языках Python и SQL
Рабочая программа дисциплины

Компьютерный набор, верстка

Ю.Н. Кондрашов

Формат 60х90/16. Гарнитура Times New Roman
Усл. п.л. ____ . Изд. № ____ . Тираж - ____ экз.
Заказ №
Отпечатано в Финуниверситете

© Ю.Н. Кондрашов, 2019
© Финансовый университет, 2019

Оглавление

1. Наименование дисциплины	5
2. Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине	5
3. Место дисциплины в структуре образовательной программы	7
4. Объем дисциплины в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся	7
5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий.....	8
5.1. Содержание дисциплины	8
5.2. Учебно - тематический план.....	10
5.3. Содержание практических и семинарских занятий.....	12
6.Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине.....	15
6.1.Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы.....	15
6.2. Перечень вопросов, заданий, тем для подготовки к текущему контролю	16
7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине	25
8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины	31
9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины	32
10. Методические указания для обучающихся по освоению дисциплины.	33
11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем.....	34
12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.....	34

1. Наименование дисциплины

Программирование на языках Python и SQL

2. Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (владения, умения и знания), соотнесенные с компетенциями/индикаторами достижения компетенции
ПКП-3	Способность применять методы разработки приложений в сфере экономики и финансов на платформе корпоративных информационных систем	1. Демонстрирует знание назначения и функционал типовых модулей корпоративных информационных систем, основные методы разработки приложений на их платформе.	Знать: <ul style="list-style-type: none">• возможности языков программирования Python и SQL для реализации функционала типовых модулей корпоративных информационных систем и разработки на их основе приложений в сфере экономики и финансов Уметь: <ul style="list-style-type: none">• встраивать алгоритмы обработки экономической и финансовой информации на языках программирования Python и SQL в типовые модули корпоративных информационных системы
		2. Владеет методологией разработки приложений в сфере экономики и финансов на платформе корпоративных информационных систем.	Знать: <ul style="list-style-type: none">• методологию разработки приложений в сфере экономики и финансов на базе языков программирования Python и SQL Уметь: <ul style="list-style-type: none">• реализовывать экономические и финансовые приложения на языках программирования Python и SQL и интегрировать их в корпоративные информационные системы.

ПКП-5	Способность применять технологии разработки настольных, мобильных и Web – приложений в сфере экономики и финансов	1. Демонстрирует знание технологии разработки настольных, мобильных и web-приложений.	Знать: <ul style="list-style-type: none"> • технологии разработки настольных, мобильных и web-приложений на языках программирования Python и SQL Уметь: <ul style="list-style-type: none"> • использовать технологии разработки настольных, мобильных и web-приложений на базе языков программирования Python и SQL
		2. Владеет навыками разработки настольных приложений в сфере экономики и финансов.	Знать: <ul style="list-style-type: none"> • возможности разработки настольных приложений в сфере экономики и финансов на языках программирования Python и SQL. Уметь: <ul style="list-style-type: none"> • разрабатывать настольные приложения в сфере экономики и финансов на языках программирования Python и SQL.
		3. Владеет навыками разработки мобильных приложений в сфере экономики и финансов.	Знать: <ul style="list-style-type: none"> • возможности разработки мобильных приложений в сфере экономики и финансов на языках программирования Python и SQL. Уметь: <ul style="list-style-type: none"> • разрабатывать мобильные приложения в сфере экономики и финансов на языках программирования Python и SQL.
		4. Владеет навыками разработки web-приложений в сфере экономики и финансов.	Знать: <ul style="list-style-type: none"> • возможности разработки web-приложений в сфере экономики и финансов на языках программирования Python и SQL. Уметь: <ul style="list-style-type: none"> • разрабатывать web-приложения в сфере экономики и финансов на языках программирования Python и SQL.

3. Место дисциплины в структуре образовательной программы

Дисциплина «Программирование на языках Python и SQL» относится к Модулю дисциплин по выбору, углубляющих освоения профиля «ИТ-сервисы и технологии обработки данных в экономике и финансах», направление подготовки 09.03.03 «Прикладная информатика».

Дисциплине «Программирование на языках Python и SQL» предшествуют следующие дисциплины, необходимые для ее изучения: «Организация вычислительных систем», «Современные технологии программирования».

4. Объем дисциплины в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся

Общая трудоёмкость дисциплины составляет 5 зачётных единиц.

Вид промежуточной аттестации – зачет.

Вид текущего контроля – контрольная работа.

Очная /очно-заочная форма обучения, 2019 год приема

Вид учебной работы по дисциплине	Всего (в з/ед. и часах)	Семестр 4/5 (в часах)
Общая трудоемкость дисциплины	5 /180	180
Контактная работа - Аудиторные занятия	68/34	68/34
<i>Лекции</i>	16/16	16/16
<i>Семинары, практические занятия</i>	52/18	52/18
Самостоятельная работа	112/146	112/146
Вид текущего контроля	Контрольная работа	Контрольная работа
Вид промежуточной аттестации	Зачет	Зачет

Общая трудоёмкость дисциплины составляет 4 зачётных единиц.

Вид промежуточной аттестации – зачет.

Вид текущего контроля – контрольная работа.

Очная /очно-заочная форма обучения, 2020 год приема

Вид учебной работы по дисциплине	Всего (в з/ед. и часах)	Семестр 4/5 (в часах)
Общая трудоемкость дисциплины	4/144	144
Контактная работа - Аудиторные занятия	50/34	50/34
Лекции	16/16	16/16
Семинары, практические занятия	34/18	34/18
Самостоятельная работа	94/110	94/110
Вид текущего контроля	Контрольная работа	Контрольная работа
Вид промежуточной аттестации	Зачет	Зачет

5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий

5.1. Содержание дисциплины

Тема 1. Использование языков Python и SQL в информационных системах обработки экономической и финансовой информации.

Современные тенденции развития систем обработки экономической и финансовой информации. Языки Python и SQL как отражение современных тенденций развития технологий хранения, обработки и анализа данных. Области применения языков Python и SQL.

Тема 2. Типы и структуры данных, основные алгоритмические конструкции.

Типы данных (бинарные, числовые, символьные, дата/время, финансовые, «большие объекты», иерархические, специальные типы данных), особенности и рекомендации по использованию, преобразование типов данных. Переменные, массивы, списки, указатели, иерархии, многомерные структуры. Типовые структуры данных и алгоритмы обработки (ветвление, множественное ветвление, циклы, обработка строк, списков, работа с файлами, сортировка, рекурсия). Использование процедур и функций, параметры и аргументы функций. Объектные-ориентированные

возможности в языках программирования.

Тема 3. Программирование на языке Python.

Среда программирования Python. Типы данных в Python, использование и преобразование. Реализация типовых алгоритмических конструкций и использование процедур и функций в Python. Использование Python для описания типовых структур данных и алгоритмов их обработки. Объектно-ориентированное программирование в Python. Стандартные библиотеки в Python для решения прикладных задач. Работа со списками и словарями в Python. Создание словаря. Операции над словарями. Перебор элементов словаря. Методы для работы со словарями.

Организация интерфейса. Доступ к базам данных из приложений на Python.

Использование языка Python в анализе данных, машинном обучении и обработке больших данных.

Тема 4. Программирование на языке SQL.

Стандарт SQL, этапы развития и реализация. Классификация типов задач.

Среда программирования на языке SQL. Типы данных в SQL, использование и преобразование. Реализация типовых алгоритмических конструкций и использование процедур и функций в SQL.

Обработка информации в БД на языке SQL. Создание объектов БД на языке SQL (таблицы, ограничения, связи).

Программирование запросов (условия выбора, использование группировок и агрегатов данных, соединений, подзапросов, табличных выражений и наследуемых таблиц, операций над множествами, рекурсий, представлений, редактирования с условиями, триггеров, разворачивания и сворачивания данных, наборов группирования). Динамический SQL.

Доступ к базам данных из приложений.

Использование языка SQL в анализе данных, машинном обучении и

обработке больших данных. Язык MDX.

Тема 5. Эффективное использования языка SQL в приложениях для обработки экономической и финансовой информации.

Оптимизации организации выполнения операторов на языке SQL в приложениях. Непроцедурность языка SQL и оптимизация выполнения запросов. Критерии и аргументы процедур оптимизации. Использование оптимальных планов выполнения в приложениях для обработки экономической и финансовой информации.

Организация параллельного доступа к информации (транзакции, использование транзакций для организации параллельной работы, оптимистическое и пессимистическое управление параллелизмом, использование блокировок и управление уровнями изоляции). Транзакции в распределенных системах.

5.2. Учебно - тематический план

Очная /очно-заочная форма обучения, 2019 год приема

№ п/п	Наименование тем (разделов) дисциплины	Трудоемкость в часах						Формы текущего контроля успевае- мости
		Всего	Аудиторная работа				Самост оательн ая работа	
			Общая, в т.ч.:	Лекции	Семинары, практическ ие занятия	Занятия в интерак- тивных формах		
1.	Использование языков Python и SQL в информационных системах обработки экономической и финансовой информации.	14/14	4/4	2/2	2/2	2/2	10/10	Обсуждение, опрос
2.	Типы и структуры	22/26	10/4	2/2	8/2	4/2	12/22	Опрос, выполнение

	данных, основные алгоритмические конструкции.							индивидуальных заданий
3.	Программирование на языке Python.	32/36	14/8	4/4	10/4	6/4	18/28	Опрос, выполнение индивидуальных заданий
4.	Программирование на языке SQL.	58/54	20/10	4/4	16/6	8/4	38/44	Опрос, выполнение индивидуальных заданий
5.	Эффективное использование языка SQL в приложениях для обработки экономической и финансовой информации.	54/50	20/8	4/4	16/4	8/4	34/42	Опрос, выполнение индивидуальных заданий
	В целом по дисциплине	180	68/34	16/16	52/18	28/16	112/146	Контрольная работа
	Итого в %					41%/47%		

Очная /очно-заочная форма обучения, 2020 год приема

№ п/п	Наименование тем (разделов) дисциплины	Трудоемкость в часах						Формы текущего контроля успевае- мости
		Всего	Аудиторная работа				Самост оятельн ая работа	
			Общая, в т.ч.:	Лекции	Семинары, практическ ие занятия	Занятия в интерак- тивных формах		
1.	Использование языков Python и SQL в информационн ых системах обработки экономической и финансовой информации.	14/8	6/4	2/2	4/2	2/2	8/4	Обсуждение, опрос

2.	Типы и структуры данных, основные алгоритмические конструкции.	18/20	8/4	2/2	6/2	4/2	10/16	Опрос, выполнение индивидуальных заданий
3.	Программирование на языке Python.	26/28	10/8	4/4	6/4	4/4	16/20	Опрос, выполнение индивидуальных заданий
4.	Программирование на языке SQL.	44/46	12/10	4/4	8/6	6/4	32/36	Опрос, выполнение индивидуальных заданий
5.	Эффективное использование языка SQL в приложениях для обработки экономической и финансовой информации.	42/42	14/8	4/4	10/4	8/4	28/34	Опрос, выполнение индивидуальных заданий
	В целом по дисциплине	144	50/34	16/16	34/18	24/16	94/110	Контрольная работа
	Итого в %					48/47%		

5.3. Содержание семинаров, практических занятий

Наименование тем (разделов) дисциплины	Перечень вопросов для обсуждения на семинарских, практических занятиях, рекомендуемые источники из разделов 8,9 (указывается раздел и порядковый номер источника)	Формы проведения занятий
Тема 1. Использование языков Python и SQL в информационных системах обработки экономической и финансовой информации.	1. Современные тенденции развития систем обработки экономической и финансовой информации. Области применения языков Python и SQL. <i>Рекомендуемые источники: п.8, [6], [8]; п.9, [13]</i>	Интерактивная форма, коллективное обсуждение результатов анализа

<p>Тема 2. Типы и структуры данных, основные алгоритмические конструкции.</p>	<p>1. Типы данных, преобразование типов данных. Переменные, массивы, списки, указатели, иерархии, многомерные структуры. 2. Типовые структуры данных и алгоритмы обработки (ветвление, множественное ветвление, циклы, обработка строк, списков, работа с файлами, сортировка, рекурсия). 3. Использование процедур и функций, параметры и аргументы функций. 4. Объектные-ориентированные возможности в языках программирования. <i>Рекомендуемые источники: п.8, [1], [2]; п.9, [1].</i></p>	<p>Интерактивная форма, практикум по решению задач по тематике занятия и коллективное обсуждение решений</p>
<p>Тема 3. Программирование на языке Python.</p>	<p>1. Среда программирования Python. Стандартные библиотеки в Python для решения прикладных задач. Типы данных. Реализация типовых алгоритмических конструкций <i>Рекомендуемые источники: п.8, [3], [9]; п.9, [1], [14]</i> 2. Использование процедур и функций в Python. Реализация типовых структур данных и алгоритмов их обработки. <i>Рекомендуемые источники: п.8, [3], [9]; п.9, [1], [14]</i> 3. Объектно-ориентированное программирование в Python. <i>Рекомендуемые источники: п.8, [2], [9]; п.9, [1]</i> 4. Работа со списками и словарями в Python. <i>Рекомендуемые источники: п.8, [3], [10]; п.9, [1]</i> 5. Организация интерфейса. Доступ к базам данных из приложений на Python. <i>Рекомендуемые источники: п.8, [3], [7]; п.9, [1], [14]</i></p>	<p>Интерактивная форма, практикум по решению задач по тематике занятия и коллективное обсуждение решений</p>

<p>Тема 4. Программирование на языке SQL.</p>	<p>1. Среда программирования. Использование и преобразование типов данных. Реализация типовых алгоритмических конструкций, использование процедур и функций. <i>Рекомендуемые источники: п.8, [4], [5]; п.9, [1], [2]</i></p> <p>2. Создание объектов БД на языке SQL (таблицы, ограничения, связи). <i>Рекомендуемые источники: п.8, [4], [5]; п.9, [1], [2]</i></p> <p>3. Программирование запросов (условия выбора, использование группировок и агрегатов данных, соединений). <i>Рекомендуемые источники: п.8, [4], [5]; п.9, [1], [2]</i></p> <p>4. Программирование запросов (условия выбора, использование подзапросов, табличных выражений и наследуемых таблиц, операций над множествами, рекурсий, представлений). <i>Рекомендуемые источники п.8, [4], [5]; п.9, [1], [2]</i></p> <p>5. Программирование запросов (использование редактирования с условиями, хранимых процедур и функций, триггеров, разворачивания и сворачивания данных, наборов группирования). <i>Рекомендуемые источники: п.8, [4], [5]; п.9, [1], [2]</i></p> <p>6. Доступ к базам данных из приложений. <i>Рекомендуемые источники: п.8, [4], [5]; п.9, [1], [2]</i></p> <p>7. Динамический SQL. <i>Рекомендуемые источники: п.8, [4], [5]; п.9, [1], [2]</i></p> <p>8. Язык SQL в анализе данных, машинном обучении и обработке больших данных. Язык MDX. <i>Рекомендуемые источники: п.8, [4], [5]; п.9, [1], [2]</i></p>	<p>Интерактивная форма, практикум по решению задач по тематике занятия и коллективное обсуждение решений</p>
---	---	--

Тема 5. Эффективное использование языка SQL в приложениях для обработки экономической и финансовой информации.	<p>1. Интерфейс анализа планов выполнения запросов. Анализ планов выполнения запросов. <i>Рекомендуемые источники: п.8, [5]; п.9, [1]</i></p> <p>2. Анализ влияния индексирования таблиц и селективности запросов. <i>Рекомендуемые источники: п.8, [5]; п.9, [1]</i></p> <p>3. Анализ влияния последовательности обработки таблиц и техник обработки соединений. <i>Рекомендуемые источники: п.8, [5]; п.9, [1]</i></p> <p>4. Управление процессом оптимизации, использование опций оптимизатора. Инструменты для редактирования стратегии оптимизатора. <i>Рекомендуемые источники: п.8, [5]; п.9, [1]</i></p> <p>5. Организация транзакций в операторах SQL. <i>Рекомендуемые источники: п.8, [5]; п.9, [1]</i></p> <p>6. Использование вложенных транзакций. <i>Рекомендуемые источники: п.8, [5]; п.9, [1]</i></p> <p>7. Пессимистическое управление параллелизмом (использование уровней изоляции) <i>Рекомендуемые источники: п.8, [5]; п.9, [1]</i></p> <p>8. Оптимистическое управление параллелизмом (использование версий строк и временных меток) <i>Рекомендуемые источники: п.8, [5]; п.9, [1]</i></p>	Интерактивная форма, практикум по решению задач по тематике занятия и коллективное обсуждение решений
---	--	---

6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

6.1. Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы

Наименование тем (разделов) дисциплины	Перечень вопросов, отводимых на самостоятельное освоение	Формы внеаудиторной самостоятельной работы
Тема 1. Использование языков Python и SQL в информационных системах обработки экономической и финансовой информации.	Современные тенденции развития систем обработки экономической и финансовой информации.	Работа с учебной литературой. Решение типовых задач. Разбор вопросов по теме занятия. Выполнение домашних заданий к каждому занятию
Тема 2. Типы и структуры данных, основные алгоритмические конструкции.	Многомерные структуры. Специальные типы данных, особенности и рекомендации по использованию.	Работа с учебной литературой. Решение типовых задач. Разбор вопросов по теме занятия. Выполнение домашних заданий к каждому занятию

Тема 3. Программирование на языке Python.	Объектно-ориентированное программирование в Python.	Работа с учебной литературой. Решение типовых задач. Разбор вопросов по теме занятия. Выполнение домашних заданий к каждому занятию
Тема 4. Программирование на языке SQL.	Использование языка SQL в анализе данных, машинном обучении и обработке больших данных.	Работа с учебной литературой. Решение типовых задач. Разбор вопросов по теме занятия. Выполнение домашних заданий к каждому занятию Выполнение и защита контрольной работы.
Эффективное использование языка SQL в приложениях для обработки экономической и финансовой информации.	Транзакции в распределенных системах.	Работа с учебной литературой. Решение типовых задач. Разбор вопросов по теме занятия. Выполнение домашних заданий к каждому занятию.

6.2. Перечень вопросов, заданий, тем для подготовки к текущему контролю

Примеры заданий контрольной работы

Язык Python.

В строке содержащей последовательность слов, разделенных запятыми удалить все нечетные слова. Ответ представить в виде строки.

Пример: строка 'SIX,SEVEN,EIGHT,NINE,TEN' будет преобразована в: 'SIX,EIGHT,TEN'.

Язык SQL.

Контрольная работа выполняется на языке SQL в среде управления SQL Server Management Studio.

Для заданной структуры базы данных (пример):

1. Сотрудники (Код сотрудника, ФИО, Возраст, Пол, Адрес, Телефон, Паспортные данные, Код должности).

2. Должности (Код должности, Наименование должности, Оклад, Обязанности, Требования).

3. Исполнители (Код исполнителя, Имя, Описание).

4. Жанры (Код жанра, Наименование, Описание).

5. Записи (Код записи, Наименование, Код исполнителя, Альбом, Год, Код жанра, Дата записи, Длительность, Рейтинг).

6. График работы (Код графика, Дата, Код сотрудника, Время начала, Время окончания, Код записи 1, Код записи 2, Код записи 3).

Изменить схему базы данных для задания любого количества записей в графике работы.

Пример преобразования:

Исходная таблица (первичные ключи подчеркнуты):

Сотрудники (Код сотрудника, Фамилия, Номер Телефона1, Номер Телефона2, Номер Телефона3)

Преобразование:

Сотрудники (Код сотрудника, Фамилия)

Телефоны сотрудников (Код сотрудника, Код телефона, Номер Телефона)

Пример данных:

Сотрудники:

<u>Код сотрудника</u>	Фамилия
1	Иванов
2	Петров
3	Сидоров

Телефоны сотрудников

<u>Код сотрудника</u>	<u>Код телефона</u>	Номер Телефона
1	1	12345678
1	2	87654321
2	1	33445566
3	1	22233345
3	2	66677789

Написать запрос для создания двух связанных таблиц, с использованием ограничений для задания счетчика для первичного ключа

одной из таблиц, проверки вводимых значений на принадлежность диапазону и возможных значений, определения связи между таблицами с каскадным удалением связанных значений.

Написать запрос с использованием внутреннего соединения, группировки и использования статистической функции. Запрос должен иметь содержательный смысл.

Написать запрос с использованием внешнего соединения. Запрос должен иметь содержательный смысл.

Для запроса с использованием внешнего соединения написать альтернативный запрос с использованием подзапроса.

Примеры индивидуальных заданий

Язык Python.

Перевернуть число

Перевод из десятичной системы счисления в двоичную без использования встроенной функции bin().

Сумма и произведение цифр числа

Вычисление факториала циклом

Нахождение факториала рекурсией

Вычисление n-го числа ряда Фибоначчи с помощью цикла while

Рекурсивное вычисление n-го числа ряда Фибоначчи

Программа нахождения НОД делением

Программа нахождения НОД вычитанием

Программа двоичного поиска элемента

Программа проверка простоты числа перебором делителей

Программа сортировка выбором

Программа сортировка «пузырьком»

В строке заменить пробелы или несколько пробелов символом *

Выбрать из строки числа

Определить длину самого короткого слова

Определить процент строчных и прописных букв в строке

Отсортировать слова по возрастанию их длины

Написать функцию `arithmetic`, принимающую 3 аргумента: первые 2 - числа, третий - операция, которая должна быть произведена над ними. Если третий аргумент `+`, сложить их; если `-`, то вычесть; `*` — умножить; `/` — разделить (первое на второе). В остальных случаях вернуть строку "Неизвестная операция".

Написать функцию `is_year_leap`, принимающую 1 аргумент — год, и возвращающую `True`, если год високосный, и `False` иначе.

Написать функцию `square`, принимающую 1 аргумент — сторону квадрата, и возвращающую 3 значения (с помощью кортежа): периметр квадрата, площадь квадрата и диагональ квадрата.

Написать функцию `season`, принимающую 1 аргумент — номер месяца (от 1 до 12), и возвращающую время года, которому этот месяц принадлежит (зима, весна, лето или осень).

Пользователь делает вклад в размере `a` рублей сроком на `years` лет под 10% годовых (каждый год размер его вклада увеличивается на 10%. Эти деньги прибавляются к сумме вклада, и на них в следующем году тоже будут проценты). Написать функцию `bank`, принимающая аргументы `a` и `years`, и возвращающую сумму, которая будет на счету пользователя.

Написать функцию `is_prime`, принимающую 1 аргумент — число от 0 до 1000, и возвращающую `True`, если оно простое, и `False` - иначе.

Написать функцию `date`, принимающую 3 аргумента — день, месяц и год. Вернуть `True`, если такая дата есть в нашем календаре, и `False` иначе.

Написать функцию `XOR_cipher`, принимающая 2 аргумента: строку, которую нужно зашифровать, и ключ шифрования, которая возвращает строку, зашифрованную путем применения функции XOR (^) над символами строки с ключом. Написать также функцию `XOR_uncipher`,

которая по зашифрованной строке и ключу восстанавливает исходную строку.

Программа определения количества строк, слов и символов в файле.

Программа расчета количества определенных слов в файле.

Язык SQL

Объявление переменных

Объявить переменную Perem1 типа денежный, а переменную Perem2 типа число с целой частью равной 8 и дробной частью равной 2.

Объявить переменную Perem1 типа строка длиной 100, а переменную Perem2 типа длинное целое.

Объявить переменную Perem1 типа динамическая строка с максимальной длиной 1000, а переменную Perem2 типа целое число.

Объявить переменную Perem1 типа строка длиной 30, а переменную Perem2 типа число с целой частью равной 10 и дробной частью равной 5.

Объявить переменную Perem1 типа дата/ время, а переменную Perem2 типа число в диапазоне от 0 до 255.

Работа с датой и временем

Определить переменную Date1 типа дата/время. Присвоить ей значение даты 31.12.2009 в формате dd.mm.yyyy.

Определить переменную Date1 типа дата/время. Присвоить ей значение даты 31.12.2009 в формате mm.dd.yyyy.

Определить переменную Date1 типа дата/время. Присвоить ей значение даты 31.12.2009 в формате yyyy.mm.dd.

Преобразование типов переменных

Объявить переменные типа FLOAT, CHAR, TINYINT, BIT. Присвоить значения, соответствующие типам. Выполнить преобразование переменных типа FLOAT, CHAR, TINYINT, BIT в INT, DATETIME, BIT, FLOAT соответственно и вывести результат на экран.

Объявить переменные типа INT, DATETIME, NUMERIC, VARCHAR.

Присвоить значения, соответствующие типам. Выполнить преобразование переменных типа INT, DATETIME, NUMERIC, VARCHAR в FLOAT, CHAR, FLOAT, INT соответственно и вывести результат на экран.

Использование функций для работы с типом дата/время

Вывести на экран название текущего месяца и текущее время.

Разобрать на отдельные составляющие текущую дату и время и вывести значения на экран в следующем порядке (вместо многоточий): "Сегодня: День = ..., Месяц = ..., Год = ..., Часов = ..., Минут = ..., Секунд = ..."

В исходный текст, сохраненный в переменной Perem, после слова " время " вставить текущее время. Результат сохранить в той же переменной Perem и вывести на экран.

Создать таблицу и заполнить одно из ее полей датами от даты вашего рождения до даты, отличающейся на количество дней, равных дню Вашего рождения (дата рождения 15 марта 1990, от этой даты заполнить 15 дней). Количество дней получить из заданной даты рождения.

Ветвления

Использую простое выражение Case, произвести вывод измененных значений одного из полей таблицы базы данных своего варианта (использовать не менее 3 альтернатив изменения и предусмотреть использование ELSE). При необходимости изменить значения полей в используемой таблице.

Использую поисковое выражение Case, произвести замену значений одного из полей таблицы базы данных своего варианта (использовать не менее 3 альтернатив изменения и предусмотреть использование ELSE). При необходимости изменить значения полей в используемой таблице.

Условная конструкция IF

Подсчитать количество записей в одной из таблиц базы данных своего варианта. Если их в таблице от ?? до ?? (задать значения), то ничего не

сообщать, в противном случае вывести сообщение вида "В таблице ... записей" (вместо многоточия поставить точное количество записей).

Подсчитать сумму значений одного из полей таблицы базы данных своего варианта. Если полученная сумма в диапазоне от ?? до ?? (заданы значения), то ничего не сообщать, в противном случае вывести сообщение вида "Сумма по полю ????? = ..." (вместо многоточия поставить точную сумму).

Цикл WHILE

Вычислить сумму членов последовательности. Формула общего члена последовательности определяется номером варианта. Вычисление вести до заданной точности вычисления (задается в программе). Точность вычисления определяется как разность двух членов последовательности.

Обработка строк символов

Удалить в тексте лишние пробелы. Лишними считаются те, которые идут непосредственно за пробелом. Подсчитать количество исправлений.

По правилам оформления машинописных текстов перед знаками .,!?:; пробелы не ставятся. Удалите лишние пробелы. Подсчитать количество исправлений.

Удалить из базового текста слово, стоящее на заданном месте. В качестве разделителей между словами используются символы пробела, запятой, тире, восклицательного знака и точки.

Переставить слова в обратном порядке. Разделитель между словами символ пробела.

Использование курсоров

Создать статический курсор по одной из таблиц базы данных своего варианта с 2-3 полями.

Создать динамический курсор по одной из таблиц базы данных своего варианта с 2-3 полями.

Создать динамический курсор по двум таблицам базы данных своего

варианта с 4-6 полями.

Создать статический курсор по одной из таблиц базы данных своего варианта с 2-3 полями. Вывести данные 3-й записи. После работы закрыть и удалить курсор из памяти.

Создать статический курсор по одной из таблиц базы данных своего варианта с 2-3 полями. Написать операторы, обеспечивающие переход к последней строке курсора, к третьей строке курсора, на две строки вперед от текущей, на три строки назад от текущей. Вывести все записи курсора и указанные записи по переходу. После работы закрыть и удалить курсор из памяти.

Создать курсор по одной из таблиц базы данных своего варианта. Перебрать все записи таблицы, просуммировать значения произведений двух полей таблицы, результат сохранить в заданной переменной, которую после суммирования вывести на экран. Закрыть и удалить из памяти курсор.

Создать курсор по одной из таблиц базы данных своего варианта. Изменить значения полей третьей записи. Закрыть и удалить из памяти курсор.

Для своего варианта базы данных написать следующие запросы:

1. Запрос с использованием внутреннего соединения двух таблиц с использованием и без использования ключевого слова JOIN. Запрос должен иметь содержательный смысл.

2. Запрос с использованием внутреннего соединения на выбор соответствующей информации из трех и более таблиц. Описать семантику запроса. Задать вычисляемое поле на основе значений полей из разных таблиц. Запрос должен иметь содержательный смысл.

3. Запрос с использованием внутреннего соединения и группировкой по двум полям. Описать семантику запроса. Запрос должен иметь содержательный смысл.

4. Запрос с составным внутренним соединением. Запрос должен иметь содержательный смысл.

5. Выполнить внутреннее самосоединение таблицы PEOPLE. Вывести только информацию о людях, которые имеют родителей.

6. Запрос с использованием внешнего соединения. Запрос должен иметь содержательный смысл.

7. Для запроса с использованием внешнего соединения добавить условие where для полей дополнительной таблицы. Прокомментировать полученный результат.

8. Добавить к базе данных своего варианта таблицу с последовательными значениями дат. Написать запрос для вывода дат, когда не выполнялось какое-либо событие.

9. Выполнить внешнее самосоединение таблицы PEOPLE. Вывести информацию о людях, которые не имеют родителей.

Создание и использование процедур.

На базе имеющихся запросов (код SQL запросов нужно изменить таким образом, чтобы в них можно было передавать значения полей, по которым осуществляется поиск) создать:

Процедуру без параметров.

Процедуру с входными параметрами

Процедуру с входными параметрами и выходным параметром

Процедуру с входным параметром и использованием RETURN.

Создать процедуру с использованием типа данных cursor в параметре OUTPUT. Произвести обработку курсора после вызова процедуры.

Создать вложенную хранимую процедуру

Создание и использование функций.

Создать скалярную функцию и использовать обращение к ней в запросе.

Создать линейную табличную функцию и обращение к ней.

Создать многооператорную табличную функцию и обращение к ней.

Создание и использование триггеров.

Создать DDL - триггер в базе данных варианта задания для события изменения таблицы. Продемонстрировать работу триггера на примере изменения таблицы базы данных.

Создать DML – триггер для проверки бизнес-правила в базе данных варианта задания. Триггер должен проверять возможность изменения записей одной таблицы на основе проверки записей другой таблицы. При невыполнении условия происходит откат транзакции. Продемонстрировать работу триггера в случае выполнения и не выполнения бизнес-правила при редактировании записей таблицы.

Критерии бальной оценки различных форм текущего контроля успеваемости

Критерии бальной оценки различных форм текущего контроля успеваемости содержится в соответствующих методических рекомендациях Департамента анализа данных, принятия решений и финансовых технологий.

7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по данной дисциплине

Перечень компетенций с указанием результатов обучения в процессе освоения образовательной программы содержится в разделе 2. *«Перечень планируемых результатов освоения образовательной программы с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине».*

**Типовые контрольные задания или иные материалы, необходимые для
оценки индикаторов достижения компетенций, знаний и умений**

Код компетенций	Наименование компетенций	Примеры заданий для оценки индикаторов сформированности компетенций
ПКП-3	Способность применять методы разработки приложений в сфере экономики и финансов на платформе корпоративных информационных систем	<p>1. Демонстрирует знание назначения и функционал типовых модулей корпоративных информационных систем, основные методы разработки приложений на их платформе.</p> <p align="center">Задание 1.</p> <p>Напишите запрос для вывода дат, когда не выполнялись заказы (структура таблиц задана).</p> <p>2. Владеет методологией разработки приложений в сфере экономики и финансов на платформе корпоративных информационных систем.</p> <p align="center">Задание 2.</p> <p>Напишите функцию bank, принимающая аргументы а (процент по вкладу) и years (количество лет) и возвращающую сумму, которая будет на счету пользователя.</p>
ПКП-5	Способность применять технологии разработки настольных, мобильных и Web – приложений в сфере экономики и финансов	<p>1. Демонстрирует знание технологии разработки настольных, мобильных и web-приложений.</p> <p align="center">Задание 1.</p> <p>Напишите запрос с использованием внутреннего соединения (структура таблиц задана).</p> <p>2. Владеет навыками разработки настольных приложений в сфере экономики и финансов.</p> <p align="center">Задание 2.</p> <p>Напишите программу для вычисления факториала циклом.</p> <p>3. Владеет навыками разработки мобильных приложений в сфере экономики и финансов.</p> <p align="center">Задание 3.</p> <p>Напишите функцию date, принимающую 3 аргумента — день, месяц и год. Вернуть True, если такая дата есть в нашем календаре, и False иначе.</p>

		<p>4. Владеет навыками разработки web-приложений в сфере экономики и финансов.</p> <p>Задание 4.</p> <p>Разберите на отдельные составляющие текущую дату и вывести значения на экран в следующем порядке (вместо многоточий): "Сегодня: День = ..., Месяц = ..., Год =</p>
--	--	--

Примеры тестовых заданий

1. Запрос, вычисляющий общую массу каждого блюда
 - SELECT КодБлюда, Avg (ВесГр) FROM Состоит GROUP BY КодБлюда;
 - SELECT КодБлюда, Count(ВесГр) FROM Состоит GROUP BY КодБлюда;
 - *SELECT КодБлюда, Sum(ВесГр) FROM Состоит GROUP BY КодБлюда;
 - SELECT КодБлюда, Sum(ВесГр) FROM Состоит GROUP BY КодБлюда HAVING Sum(ВесГр)>100;
2. Значение, подходящее под условие: [Фамилия] Like "*ов" AND [Отчество] Like "*ич"
 - Шилова Тамара Васильевна
 - Петренко Иван Иванович
 - *Семенов Роман Николаевич
 - Бонд Джеймс
3. При левом внешнем объединении таблиц в результирующее отношение включаются
 - все записи из правой таблицы и только тех записей из левой таблицы, в которых связанные поля совпадают
 - *все записи из левой таблицы и только тех записей из правой таблицы, в которых связанные поля совпадают
 - все записи из левой таблицы и все записи из правой таблицы
 - все записи, в которых связанные поля обеих таблиц совпадают
4. Фрагмент оператора создания таблицы

Сорт VARCHAR(50) NOT NULL

CHECK (Сорт in('первый','второй','третий')).

определяет Поле Сорт, который...

 - получает значения по умолчанию

- объявляется первичным ключом
- объявляется внешним ключом
- *может принимать только перечисленные значения

5. Инструкция SELECT является:

- процедурной
- *непроцедурной
- зависит от наличия в инструкции условия
- зависит от наличия в инструкции группировки

6. В запросе «SELECT DISTINCT Клиент.Фирма FROM Клиент» ключевое слово DISTINCT используется для...

- *удаления повторяющихся строк
- удаления неопределённых значений
- группировки повторяющихся строк
- группировки строк, имеющих значения полей по умолчанию

7. Какие ошибки здесь допущены?

```
def factorial(n):
```

```
    if n == 0:
```

```
        return 1
```

```
    else:
```

```
        return n * factorial(n - 1)
```

```
print factorial(5)
```

- Функция не может вызывать сама себя
- В коде нет никаких ошибок
- *Необходимо указать тип возвращаемого значения
- Функция всегда будет возвращать 1

Примеры практико-ориентированных (ситуационных) заданий.

1. При помощи стека (можно использовать любую реализацию стека) проверить что в строке содержащей открывающие и закрывающие скобки трех типов: (), [], {}. Соблюдается корректный баланс и вложенность скобок
2. Рекурсивно реализовать функцию fib(n) вычисляющую значение n-го

числа Фибоначи. Числа Фибоначчи — элементы числовой последовательности в которой каждое последующее число равно сумме двух предыдущих чисел (Числа Фибоначчи: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, ...).

3. Инвертировать последовательность слов, разделенных запятыми. Пример: строка 'SIX, SEVEN, EIGHT, NINE, TEN' будет преобразована в: 'TEN, NINE, EIGHT, SEVEN, SIX'.
4. Определить общее количество покупок, сделанных покупателями по датам покупки и их общую стоимость. Используются таблицы Покупатель, Покупка, Товар (схема таблиц задана).
5. Имеется таблица Сотрудники (схема таблиц задана). Написать запрос для получения неповторяющихся пар сотрудников.
6. Используются таблицы Поставщик, Товар (схема таблиц задана). Написать два альтернативных запроса с использованием внешнего соединения и подзапроса для вывода списка поставщиков, которые не поставляли товаров.

Примерные вопросы для подготовки к зачету

Язык Python.

1. Встроенные числовые типы языка Python.
2. Списки. Создание, основные операции.
3. Основные методы списка.
4. Кортежи. Создание, основные методы и операции.
5. Словари. Создание, основные операции.
6. Методы для работы со словарями.
7. Множества. Создание, основные методы и операции.
8. Переменные. Правила именования переменных.
9. Динамическая типизация.
10. Операторы сравнения и логические операторы.
11. Инструкция if...else.
12. Инструкция цикла while.
13. Инструкция цикла for.
14. Создание и вызов функции.
15. Передача аргументов функцию.
16. Модули. Инструкции import и from.
17. Базовые принципы объектно-ориентированного программирования.

18. Класс, метод класса, атрибут класса. Определение класса и создание экземпляра класса.

19. Наследование.

20. Абстрактные методы класса.

21. Статические методы класса.

22. Свойства класса.

23. Событие. Обработчик события. Цикл обработки событий

24. Элемент Кнопка. Создание и настройка.

25. Элементы Надпись и Текстовое поле. Создание и настройка.

Метод get().

26. Элемент Флажок. Создание, настройка, получение статуса флажка.

27. Элемент переключатель. Создание, настройка, доступ к значению.

Язык SQL.

28. Типы данных. Рекомендации по использованию.

29. Преобразование типов данных.

30. Функций для работы с типом дата/время.

31. Функции для работы со строковыми переменными.

32. Простое выражение Case.

33. Поисковое выражение CASE.

34. Условная конструкция IF.

35. Цикл WHILE.

36. Курсоры. Объявление курсора и операторы для работы с курсором

37. Создание и использование хранимых процедур.

38. Создание и использование функций.

39. Создание и использование триггеров.

40. Создание таблиц и связей. Ограничения. Рекурсивная связь.

41. Ввод данных в таблицы.

42. Использование в запросах группировки и статистических функций.

43. Использование в запросах операций соединения.

44. Составные соединения.

45. Самосоединение таблиц.

46. Использование подзапросов (подзапросы возвращающие одно, множество значений).

47. Связанные (коррелированные) подзапросы.

48. Табличный подзапрос (табличные выражения).

49. Альтернативное использование соединений и подзапросов в запросах.

50. Операции над множествами

51. Использование операции UNION ALL для организации рекурсивных запросов:

52. Создание и использование представлений.

53. Редактирование данных в таблицах с условиями.

8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. Колдаев, В.Д. Структуры и алгоритмы обработки данных: учебное пособие / В.Д. Колдаев. – Москва: ИЦ РИОР: НИЦ ИНФРА-М, 2014. – 296 с. – ЭБС ZNANIUM.com. - URL: <http://znanium.com/catalog/product/418290> (дата обращения: 27.12.2019). – Текст : электронный.

2. Кондрашов, Ю.Н. Язык SQL. Сборник ситуационных задач по дисциплине «Базы данных»: учебное пособие/ Ю.Н. Кондрашов. - Москва: РУСАЙНС, 2018. - 126 с. - Текст : непосредственный. – То же. – ЭБС BOOK.ru. – URL: <https://www.book.ru/book/929715> (дата обращения: 27.12.2019). – Текст : электронный.

3. Кондрашов, Ю.Н. Эффективное использование СУБД MS SQL Server: учебное пособие/ Ю.Н. Кондрашов. - Москва: РУСАЙНС, 2017. -128 с. – Текст : непосредственный. – То же. – ЭБС BOOK.ru. – URL: <https://www.book.ru/book/927673> (дата обращения: 27.12.2019). – Текст : электронный.

4. Кондрашов, Ю.Н. Анализ данных и машинное обучение на платформе MS SQL Server: учебное пособие / Ю.Н. Кондрашов. — Москва: Русайнс, 2020. — 303 с. — Текст : непосредственный.— То же. — ЭБС BOOK.ru. — URL: <https://book.ru/book/933497> (дата обращения: 27.12.2019). — Текст : электронный.

Дополнительная литература:

5. Маккинни, У. Python и анализ данных / У. Маккинни. - Москва: ДМК-Пресс, 2015. - 482 с. - Текст: непосредственный. - То же. - ЭБС ZNANIUM.com.
- URL: <http://znanium.com/catalog/product/1027796> (дата обращения: 27.12.2019).
- Текст : электронный.

9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. Информационно-образовательный портал Финансового университета при Правительстве Российской Федерации
<http://portal.ufrf.ru/>
2. Сайт департамента анализа данных, принятия решений и финансовых технологий.
3. Электронная библиотека Финансового университета (ЭБ)
<http://elib.fa.ru/> (<http://library.fa.ru/files/elibfa.pdf>)
4. Электронная библиотека Финансового университета (ЭБ)
<http://elib.fa.ru/> (<http://library.fa.ru/files/elibfa.pdf>)
5. Электронно-библиотечная система BOOK.RU <http://www.book.ru>
6. Электронно-библиотечная система «Университетская библиотека ОНЛАЙН» <http://biblioclub.ru/>
7. Электронно-библиотечная система Znanium
<http://www.znanium.com>
8. «Деловая онлайн библиотека» издательства «Альпина Паблишер»
<http://lib.alpinadigital.ru/en/library>
9. Издательство «Открытые системы»
<http://www.osp.ru/os/2012/02/13014107>
10. Электронно-библиотечная система издательства «Лань»
<https://e.lanbook.com/>

11. Электронно-библиотечная система издательства «ЮРАЙТ»
<https://www.biblio-online.ru/>
12. Научная электронная библиотека eLibrary.ru <http://elibrary.ru>
13. . Python Data Analysis Library [Электронный ресурс]: сайт. – Режим доступа: <http://pandas.pydata.org/>
14. Python Documentation [Электронный ресурс]: сайт. – Режим доступа: <http://python.org/doc/>
15. Python Standard Library [Электронный ресурс]: сайт. – Режим доступа: <https://docs.python.org/2/library/>
16. Scikit-learn Machine Learning in Python [Электронный ресурс]: сайт. – Режим доступа: <http://scikit-learn.org>
17. Официальный сайт продукта <https://www.python.org/>

10. Методические указания для обучающихся по освоению дисциплины

Изучение дисциплины ориентировано прежде всего на формирование навыков использования современных языков Python и SQL для разработки приложений в сфере экономики и финансов на платформе корпоративных информационных систем. Необходимые знания теории и практические навыки использования современных языков Python и SQL формируются как во время лекционных занятий, так и на семинарских занятиях. Для успешного усвоения лекционного материала необходима регулярная работа с рекомендованными методическими материалами.

Значительная часть семинарских занятий проводятся в интерактивном режиме с подробным обсуждением изучаемых тем. Активная работа компьютерных классов и самостоятельная работа являются обязательным условием формирования знаний, умений и навыков самостоятельного проектирования и сопровождения баз данных.

Большое значение в образовательном процессе придается выполнению индивидуальных заданий на основе вариантов и контрольной работы. Результаты выполнения индивидуальных заданий завершается представлением результатов преподавателю.

11.Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем

11. 1. Комплект лицензионного программного обеспечения:

1. Windows, Microsoft Office.
2. Антивирус ESET Endpoint Security

11.2. Современные профессиональные базы данных и информационные справочные системы

1. Информационно-правовая система «Гарант»
2. Информационно-правовая система «Консультант Плюс»
3. Электронная энциклопедия: <http://ru.wikipedia.org/wiki/Wiki>
4. Система комплексного раскрытия информации «СКРИН» - <http://www.skrin.ru/>

11.3. Сертифицированные программные и аппаратные средства защиты информации

– не используются

1. СУБД MS SQL Server с компонентами Analysis Services.
2. Microsoft Visual Studio.
3. Дистрибутив языка Python 3.4 (или более поздней версии) Anaconda 3.
4. Текстовый процессор MS Word.

12.Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

Лекционные занятия проводятся в мультимедийных аудиториях, семинарские занятия в компьютерных классах.