

МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Астраханский государственный университет имени В. Н. Татищева»  
(Астраханский государственный университет им. В. Н. Татищева)

СОГЛАСОВАНО  
Руководитель ОПОП

Е. О. Вострикова

«04» апреля 2024 г.

УТВЕРЖДАЮ  
Заведующий кафедрой экономической  
теории

Е. О. Вострикова

«04» апреля 2024 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)  
«Программирование на PYTHON»**

Составитель(и)	<b>Федорова Е. П., доцент, к.э.н., доцент кафедры экономической теории;</b>
Согласовано с работодателями:	<b>Чурсов Д.С., начальник отдела корпоративных продаж Астраханского филиала АО «СОГАЗ»; Козлова И.А., заместитель управляющего по корпоративному блоку ООО «РОО «Астраханский» Филиал №2351 ВТБ (ПАО)»; 38.03.01 ЭКОНОМИКА</b>
Направление подготовки / специальность	<b>Экономика и управление</b>
Направленность (профиль) / специализация ОПОП	<b>бакалавр</b>
Квалификация (степень)	<b>очная, очно-заочная</b>
Форма обучения	<b>2024</b>
Год приёма	<b>4 (по очной форме) 4 (по очно-заочной форме) /</b>
Курс	<b>7 (по очной форме) / 7 (по очно-заочной форме) /</b>
Семестр(ы)	

Астрахань – 2024

## 1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

**1.1. Целями освоения дисциплины (модуля) «Программирование на PYTHON»** являются овладение навыками программирования на языке Python, овладение методами сбора и обработки данных для решения экономическо-управленческих задач.

### **1.2. Задачи освоения дисциплины (модуля):**

- изучение теории и практики бизнес-планирования и управления проектами в - формирование и развитие навыков программирования на языке Python;
- формирование у обучаемых практических знаний для самостоятельного создания и использования сложных структур данных;
- разработка архитектуры, алгоритмических и программных решений системного и прикладного программного обеспечения;
- развитие и использование математических и информационных инструментальных средств, автоматизированных систем в практической деятельности.

## 2. МЕСТО ДИСЦИПЛИНЫ (МОДУЛЯ) В СТРУКТУРЕ ОПОП

**2.1. Учебная дисциплина (модуль) «Программирование на PYTHON»** относится к части, формируемой участниками образовательных отношений и осваивается в 7 семестре.

Программа курса строится на предпосылке, что студенты владеют базовыми положениями экономики, знакомы с основными положениями статистики. Одновременно курс создает предпосылки для более глубокого освоения методов программирования, а также дальнейшего расширения практического и теоретического арсенала обучающегося и понимания целесообразности применения инструментов программирования для оценки поведения экономических агентов.

**2.2. Для изучения данной учебной дисциплины (модуля) необходимы следующие знания, умения, навыки, формируемые предшествующими учебными дисциплинами (модулями):**

– *Статистика, Введение в информационные технологии*

Знания: теоретические основы функционирования экономики.

Умения: рассчитывать основные экономические показатели.

Навыки: самостоятельная работа с экономической литературой, применение инструментов экономического анализа.

**2.3. Последующие учебные дисциплины (модули) и (или) практики, для которых необходимы знания, умения, навыки, формируемые данной учебной дисциплиной (модулем):**

– *Управление проектами, Бизнес-аналитика*

## 3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

Процесс освоения дисциплины (модуля) направлен на формирование элементов следующей(их) компетенции(ий) в соответствии с ФГОС ВО и ОПОП ВО по данному направлению подготовки / специальности:

*в) профессиональной(ых) (ПК).*

**ПК-3:** Способен анализировать и интерпретировать финансовую, бухгалтерскую и иную информацию, содержащуюся в отчетности предприятий различных форм собственности, организаций, ведомств и т.д., использовать полученные сведения для принятия управленческих решений, выявления тенденции изменения социально-экономических показателей, а также подготовить информационный обзор и/или аналитический отчет.

**ПК-5:** Способен собирать, анализировать исходные данные, рассчитывать экономические и социально-экономические показатели деятельности экономических агентов, а также обосновывать и представлять результаты расчетов в соответствии с принятыми стандартами и нормативно-правовой базой.

**Таблица 1. Декомпозиция результатов обучения**

Код компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине (модулю)		
		Знать (1)	Уметь (2)	Владеть (3)
ПК-3	ПК-3.2	-Методы анализа и прогнозирования состояния конъюнктуры различных рынков	- Анализировать и прогнозировать состояния конъюнктуры различных рынков	- Навыками анализа и прогнозирования состояния конъюнктуры различных рынков
ПК-5	ПК-5.3	- Современные технические средства и информационные технологии	- Использовать для решения аналитических и исследовательских задач современные технические средства и информационные технологии	- Навыками решения аналитических и исследовательских задач

#### 4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Общая трудоемкость дисциплины в соответствии с учебным планом составляет 4 зачетные единицы (144 часа).

Трудоемкость отдельных видов учебной работы студентов очной, очно-заочной и заочной форм обучения приведена в таблице 2.1.

**Таблица 2.1. Трудоемкость отдельных видов учебной работы по формам обучения**

Вид учебной и внеучебной работы	для очной формы обучения	для очно-заочной формы обучения
Объем дисциплины в зачетных единицах	4	4
Объем дисциплины в академических часах	144	144
Контактная работа обучающихся с преподавателем (всего), в том числе (час.):	72	72
- занятия лекционного типа, в том числе:	36	36
- практическая подготовка (если предусмотрена)	-	-
- занятия семинарского типа (семинары, практические, лабораторные), в том числе:	36	36
- практическая подготовка (если предусмотрена)	-	-
- в ходе подготовки и защиты курсовой работы	-	-

Вид учебной и внеучебной работы	для очной формы обучения	для очно-заочной формы обучения
- консультация (предэкзаменационная)	-	-
- промежуточная аттестация по дисциплине	-	-
Самостоятельная работа обучающихся (час.)	72	72
Форма промежуточной аттестации обучающегося (зачет/экзамен), семестр (ы)	зачет – 7 семестр	зачет – 7 семестр

Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий и самостоятельной работы, для каждой формы обучения представлено в таблице 2.2.

**Таблица 2.2. Структура и содержание дисциплины (модуля)**

*для очной формы обучения*

Раздел, тема дисциплины (модуля)	Контактная работа, час.							СР, час	Итого часов	Форма текущего контроля успеваемости, форма промежуточной аттестации [по семестрам]
	Л		ПЗ		ЛР		КР / КП			
	Л	в т.ч. ПП	ПЗ	в т.ч. ПП	ЛР	в т.ч. ПП				
<b>Семестр 7.</b>										
Тема 1. Введение: особенности, функции и управляющие конструкции Python	12				12			24		Опрос по теме
Тема 2. Классы и объектно-ориентированное программирование	12				12			24		Лабораторная работа
Тема 3. Сетевое программирование	12				12			24		Лабораторная работа
<b>Консультации</b>										
<b>Контроль промежуточной аттестации</b>										<b>Диф. зачет</b>
<b>ИТОГО за семестр:</b>	<b>36</b>		<b>36</b>					<b>72</b>	<b>144</b>	
<b>Итого за весь период</b>	<b>36</b>		<b>36</b>					<b>72</b>	<b>144</b>	

*для очно-заочной формы обучения*

Раздел, тема дисциплины (модуля)	Контактная работа, час.				СР, час	Итого часов	Форма текущего
	Л	ПЗ	ЛР	КР			

	Л	В т.ч. ПП	ПЗ	В т.ч. ПП	ЛР	В т.ч. ПП	/ КП	.		контроля успеваемости, форма промежуточной аттестации [по семестрам ]
<b>Семестр 7.</b>										
Тема 1. Введение: особенности, функции и управляющие конструкции Python	12				12			24		Опрос по теме
Тема 2. Классы и объектно- ориентированное программирование	12				12			24		Лабораторн ая работа
Тема 3. Сетевое программирование	12				12			24		Лабораторн ая работа
<b>Консультации</b>										
<b>Контроль промежуточной аттестации</b>										<b>Диф. зачет</b>
<b>ИТОГО за семестр:</b>	<b>36</b>		<b>36</b>					<b>72</b>	<b>144</b>	
<b>Итого за весь период</b>	<b>36</b>		<b>36</b>					<b>72</b>	<b>144</b>	

**Таблица 3. Матрица соотнесения разделов, тем учебной дисциплины (модуля) и формируемых компетенций**

Раздел, тема дисциплины (модуля)	Кол-во часов	Код компетенции		Общее количество компетенций
		ПК-3	ПК-5	
Тема 1. Введение: особенности, функции и управляющие конструкции Python	48	+	+	2
Тема 2. Классы и объектно- ориентированное программировани е	48	+	+	2
Тема 3. Сетевое программировани е	48	+	+	2
<b>Итого</b>	<b>144</b>			

**Краткое содержание каждой темы дисциплины (модуля)**

**Тема 1. Введение: особенности, функции и управляющие конструкции Python**

Краткая история языка программирования Python. В Python все является объектом. В Python нет переменных. Интернированные (intern) объекты в Python. Глобальная блокировка интерпретатора. Подходы к сборке мусора в Python. Встроенные типы данных Python. Основные операторы в Python. Использование отступов в Python. Комментарии. Правила именования переменных (имен). Оператор if и match. Цикл while. Цикл for. Различные способы написания циклов. Итерации и включения. Источники документации Python. Области видимости. Аргументы функции. Возвращение результатов выполнения функцией. Рекурсия. Аннотация функций. Лямбда-функции (выражения). Декораторы. Генераторы.

### **Тема 2. Классы и объектно-ориентированное программирование**

Определение класса. Имена (переменные) экземпляров класса. Методы экземпляра класса. Имена (переменные) класса. Статические методы. Методы класса. Приватные методы и переменные. Наследование. Множественное наследование. Абстрактные классы и переопределение методов. Перегрузка операций. Вложенные классы и пространство имен. Перечисления (Enum).

### **Тема 3. Сетевое программирование**

Архитектура "клиент-сервер". Модуль socket. Пример клиента и сервера, работающих без установления соединения. Пример клиента и сервера, работающих с установлением соединения. Фреймворк для сетевых серверов - socketserver.

## **5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПРЕПОДАВАНИЮ И ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)**

### **5.1. Указания для преподавателей по организации и проведению учебных занятий по дисциплине (модулю)**

Курс строится на лекционных и лабораторных занятиях. Лабораторные занятия являются аудиторными. Осмысление и обобщение сложных разделов дисциплины теоретического материала осуществляется студентами самостоятельно и заключается лекционной презентацией. Для проведения лабораторных занятий необходимо использование компьютеров.

### **5.2. Указания для обучающихся по освоению дисциплины (модулю)**

**Таблица 4. Содержание самостоятельной работы обучающихся**

#### *для очной формы обучения*

Вопросы, выносимые на самостоятельное изучение	Кол-во часов	Форма работы
Тема 1. Введение: особенности, функции и управляющие конструкции Python	24	Подготовка к опросу
Тема 2. Классы и объектно-ориентированное программирование	24	Лабораторная работа
Тема 3. Сетевое программирование	24	Лабораторная работа

#### *для очно-заочной формы обучения*

Вопросы, выносимые на самостоятельное изучение	Кол-во часов	Форма работы
Тема 1. Введение: особенности, функции и управляющие конструкции Python	24	Подготовка к опросу
Тема 2. Классы и объектно-ориентированное программирование	24	Лабораторная работа
Тема 3. Сетевое программирование	24	Лабораторная работа

### 5.3. Виды и формы письменных работ, предусмотренных при освоении дисциплины (модуля), выполняемые обучающимися самостоятельно

Особое значение придается организации самостоятельной работы студентов. Обучающийся должен изучить учебный план и программу дисциплины для того, чтобы своевременно понять и правильно оценить ее роль в учебном процессе. Студенту необходимо осуществить выбор тактики и стратегии получения знаний в полном объеме по осваиваемой дисциплине.

Темы дисциплины должны изучаться последовательно. Самостоятельная работа студентов (аудиторная и внеаудиторная) позволяет расширить приобретенные на знания, научиться их прикладному применению, и эффективному проведению работы с нормативной базой и рекомендуемой литературой.

Для своевременной помощи обучающимся при изучении дисциплины кафедрами организуются индивидуальные и групповые консультации, устанавливается время приема выполненных работ.

Самостоятельная работа студентов состоит из следующих видов:

- поиск и обзор литературы и электронных источников информации,
- подготовка домашнего задания.

## 6. ОБРАЗОВАТЕЛЬНЫЕ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

### 6.1. Образовательные технологии

**Таблица 5. Образовательные технологии, используемые при реализации учебных занятий**

Раздел, тема дисциплины (модуля)	Форма учебного занятия		
	Лекция	Практическое занятие, семинар	Лабораторная работа
Тема 1. Введение: особенности, функции и управляющие конструкции Python	Обзорная лекция	Не предусмотрено	Выполнение практических заданий
Тема 2. Классы и объектно-ориентированное программирование	Обзорная лекция	Не предусмотрено	Выполнение практических заданий
Тема 3. Сетевое программирование	Обзорная лекция	Не предусмотрено	Выполнение практических заданий

### 6.2. Информационные технологии

При реализации различных видов учебной и внеучебной работы используются следующие информационные технологии:

- использование электронных учебников и различных сайтов (например, электронные библиотеки, журналы и т.д.) как источник информации
- использование возможностей электронной почты преподавателя
- использование средств представления учебной информации (электронных учебных пособий и практикумов, применение новых технологий для проведения очных (традиционных) лекций и семинаров с использованием презентаций и т.д.)
- виртуальная обучающая среда (или система управления обучением LMS Moodle «Электронное образование»).

### 6.3. Программное обеспечение, современные профессиональные базы данных и информационные справочные системы

#### 6.3.1. Программное обеспечение

Наименование программного обеспечения	Назначение
Adobe Reader	Программа для просмотра электронных документов
Платформа дистанционного обучения LMS Moodle	Виртуальная обучающая среда
Mozilla FireFox	Браузер
Microsoft Office 2013, Microsoft Office Project 2013, Microsoft Office Visio 2013	Пакет офисных программ
7-zip	Архиватор
Microsoft Windows 10 Professional	Операционная система
Kaspersky Endpoint Security	Средство антивирусной защиты
Google Chrome	Браузер
Notepad++	Текстовый редактор
OpenOffice	Пакет офисных программ
Opera	Браузер
Microsoft Security Assessment Tool. Режим доступа: <a href="http://www.microsoft.com/ru-ru/download/details.aspx?id=12273">http://www.microsoft.com/ru-ru/download/details.aspx?id=12273</a> (Free) Windows Security Risk Management Guide Tools and Templates. Режим доступа: <a href="http://www.microsoft.com/en-us/download/details.aspx?id=6232">http://www.microsoft.com/en-us/download/details.aspx?id=6232</a> (Free)	Программы для информационной безопасности
VLC Player	Медиапроигрыватель
Sofa Stats	Программное обеспечение для статистики, анализа и отчётности
WinDjView	Программа для просмотра файлов в формате DJV и DjVu
LibreOffice	Пакет офисных программ.

#### 6.3.2. Современные профессиональные базы данных и информационные справочные системы

<i>Наименование современных профессиональных баз данных, информационных справочных систем</i>
<a href="http://dlib.eastview.com">Универсальная справочно-информационная полнотекстовая база данных периодических изданий ООО «ИВИС»</a> <a href="http://dlib.eastview.com">http://dlib.eastview.com</a> Имя пользователя: AstrGU Пароль: AstrGU
Электронные версии периодических изданий, размещённые на сайте информационных ресурсов <a href="http://www.polpred.com">www.polpred.com</a>
Электронный каталог Научной библиотеки АГУ на базе MARK SQL НПО «Информ-систем» <a href="https://library.asu.edu.ru/catalog/">https://library.asu.edu.ru/catalog/</a>
Электронный каталог «Научные журналы АГУ» <a href="https://journal.asu.edu.ru/">https://journal.asu.edu.ru/</a>
Корпоративный проект Ассоциации региональных библиотечных консорциумов (АРБИКОН) «Межрегиональная аналитическая роспись статей» (МАРС) – сводная база данных, содержащая полную аналитическую роспись 1800 названий журналов по разным отраслям знаний. Участники проекта предоставляют друг другу электронные копии

<i>Наименование современных профессиональных баз данных, информационных справочных систем</i>
отсканированных статей из книг, сборников, журналов, содержащихся в фондах их библиотек. <a href="http://mars.arbicon.ru">http://mars.arbicon.ru</a>
Справочная правовая система КонсультантПлюс. Содержится огромный массив справочной правовой информации, российское и региональное законодательство, судебную практику, финансовые и кадровые консультации, консультации для бюджетных организаций, комментарии законодательства, формы документов, проекты нормативных правовых актов, международные правовые акты, правовые акты, технические нормы и правила. <a href="http://www.consultant.ru">http://www.consultant.ru</a>

## **7. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

### **7.1. Паспорт фонда оценочных средств**

При проведении текущего контроля и промежуточной аттестации по дисциплине (модулю) «Бизнес-планирование» проверяется сформированность у обучающихся компетенций, указанных в разделе 3 настоящей программы. Этапность формирования данных компетенций в процессе освоения образовательной программы определяется последовательным освоением дисциплин (модулей) и прохождением практик, а в процессе освоения дисциплины (модуля) – последовательным достижением результатов освоения содержательно связанных между собой разделов, тем.

**Таблица 6. Соответствие разделов, тем дисциплины (модуля), результатов обучения по дисциплине (модулю) и оценочных средств**

№ п/п	Контролируемый раздел, тема дисциплины (модуля)	Код контролируемой компетенции (компетенций)	Наименование оценочного средства
1	Тема 1. Введение: особенности, функции и управляющие конструкции Python	ПК-3, ПК-5	Опрос по теме
2	Тема 2. Классы и объектно-ориентированное программирование	ПК-3, ПК-5	Опрос по теме, лабораторная работа
3	Тема 3. Сетевое программирование	ПК-3, ПК-5	Опрос по теме, лабораторная работа

### **7.2. Описание показателей и критериев оценивания компетенций, описание шкал оценивания**

**Таблица 7. Показатели оценивания результатов обучения в виде знаний**

Шкала оценивания	Критерии оценивания
5 «отлично»	демонстрирует глубокое знание теоретического материала, умение обоснованно излагать свои мысли по обсуждаемым вопросам, способность

Шкала оценивания	Критерии оценивания
	полно, правильно и аргументированно отвечать на вопросы, приводить примеры
4 «хорошо»	демонстрирует знание теоретического материала, его последовательное изложение, способность приводить примеры, допускает единичные ошибки, исправляемые после замечания преподавателя
3 «удовлетворительно»	демонстрирует неполное, фрагментарное знание теоретического материала, требующее наводящих вопросов преподавателя, допускает существенные ошибки в его изложении, затрудняется в приведении примеров и формулировке выводов
2 «неудовлетворительно»	демонстрирует существенные пробелы в знании теоретического материала, не способен его изложить и ответить на наводящие вопросы преподавателя, не может привести примеры

**Таблица 8. Показатели оценивания результатов обучения в виде умений и владений**

Шкала оценивания	Критерии оценивания
5 «отлично»	демонстрирует способность применять знание теоретического материала при выполнении заданий, последовательно и правильно выполняет задания, умеет обоснованно излагать свои мысли и делать необходимые выводы
4 «хорошо»	демонстрирует способность применять знание теоретического материала при выполнении заданий, последовательно и правильно выполняет задания, умеет обоснованно излагать свои мысли и делать необходимые выводы, допускает единичные ошибки, исправляемые после замечания преподавателя
3 «удовлетворительно»	демонстрирует отдельные, несистематизированные навыки, испытывает затруднения и допускает ошибки при выполнении заданий, выполняет задание по подсказке преподавателя, затрудняется в формулировке выводов
2 «неудовлетворительно»	не способен правильно выполнить задания

### 7.3. Контрольные задания и иные материалы, необходимые для оценки результатов обучения по дисциплине (модулю)

#### Тема 1. Введение: особенности, функции и управляющие конструкции Python

Вопросы:

1. Почему в Python нет переменных?
2. Для чего используется счетчик ссылок?
3. В чем разница между сильными и слабыми ссылками?
4. Что такое интернированные объекты?
5. Как можно интернировать строку, содержащую не ASCII-символы?
6. Что такое GIL? В чем его особенность?
7. Как работает алгоритм подсчета ссылок?
8. Опишите принцип работы Garbage Collector.
9. Зачем используется подход с поколениями при работе Garbage Collector?
10. Для чего можно использовать слабые ссылки?
11. Назовите четыре основных типа данных Python.
12. Что означает «неизменяемость» и какие три основных типа Python считаются неизменяемыми?

неизменяемыми?

13. Что означает «последовательность» и какие три типа входят в эту категорию?

## Тема 2. Классы и объектно-ориентированное программирование

Лабораторная работа:

1. Напишите класс, реализующий все арифметические операции над двумя значениями (a и b).
2. Напишите класс, описывающий такой объект, как автомобиль. Продумайте, какие методы и переменные он должен иметь.
3. Напишите класс, описывающий такой объект, как прямоугольник. Перегрузите у реализованного класса методы сравнения (сравнивать по площади), после чего создайте два экземпляра класса и проверьте, как работают перегруженные методы.
4. Напишите класс, описывающий такой объект, как автомобиль. У него может быть различное количество состояний, реализуемых посредством перечислений. Добавьте методы, позволяющие экземпляру класса менять свое текущее состояние (например: остановка, движение, поворот налево и т. д.).
5. Напишите класс, который подсчитывает текущее количество его экземпляров в приложении. Для корректного отображения этого числа перегрузите у класса метод `__del__` и напишите необходимую логику.
6. Напишите базовый класс, задающий интерфейс и часть характеристик

## Тема 3. Сетевое программирование

Лабораторная работа:

1. Напишите программу (клиентскую и серверную часть), позволяющую общаться пользователям внутри локальной сети. Используйте при этом протокол UDP.
2. Напишите клиент-серверное приложение, позволяющее передавать файл по сети. Используйте при этом протокол UDP.
3. Напишите клиент-серверное приложение, позволяющее передавать файл по сети. Используйте при этом протокол TCP.
4. Напишите программу, позволяющую играть в «Крестики-нолики» по сети.
5. Напишите программу, позволяющую играть в «Морской бой» по сети.
6. Напишите сетевой калькулятор. На клиентской стороне пользователь вводит выражение, а само его вычисление производится на серверной части.
7. Напишите клиент-серверное приложение, где на клиентской стороне вход в основное окно приложения осуществляется только после его авторизации на сервере, с возможностью регистрации нового пользователя. На серверной стороне хеш связки «логин:пароль» пользователей хранится в отдельном файле

### Перечень вопросов и заданий, выносимых на зачет

1. Почему в Python нет переменных?
2. Для чего используется счетчик ссылок?
3. В чем разница между сильными и слабыми ссылками?
4. Что такое интернированные объекты?
5. Как можно интернировать строку, содержащую не ASCII-символы?
6. Что такое GIL? В чем его особенность?
7. Как работает алгоритм подсчета ссылок?
8. Опишите принцип работы Garbage Collector.
9. Зачем используется подход с поколениями при работе Garbage Collector?
10. Для чего можно использовать слабые ссылки?
11. Назовите четыре основных типа данных Python.
12. Что означает «неизменяемость» и какие три основных типа Python считаются неизменяемыми?
13. Что означает «последовательность» и какие три типа входят в эту категорию?

14. Какие существуют способы взаимодействия между программами по сети?
15. В каких случаях лучше использовать TCP?
16. В каких случаях лучше использовать UDP?
17. Приведите пример логики работы серверной части, работающей без установления соединения.
18. Приведите пример логики работы клиентской части, работающей без установления соединения.
19. Приведите пример логики работы серверной части, работающей с установлением соединения.
20. Приведите пример логики работы клиентской части, работающей с установлением соединения.
21. Для чего используется модуль socket?
22. Какие исключения определены в модуле socket?
23. Для чего используется модуль socketserver и каков принцип его работы?

**Таблица 9. Примеры оценочных средств с ключами правильных ответов**

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
<p><b>ПК-3:</b> Способен анализировать и интерпретировать финансовую, бухгалтерскую и иную информацию, содержащуюся в отчетности предприятий различных форм собственности, организаций, ведомств и т.д., использовать полученные сведения для принятия управленческих решений, выявления тенденции изменения социально-экономических показателей, а также подготовить информационный обзор и/или аналитический отчет.</p>				
1.	Задание закрытого типа	Какой из следующих операторов используется для сравнения двух значений в Python? a) = b) == c) := d) ===	b	2 минуты
2.		Какой метод используется для добавления элемента в конец списка в Python? a) append() b) add() c) insert() d) push()	a	2 минуты
3.		Какой из следующих типов данных является неизменяемым в Python? a) list b) dict c) set d) tuple	d	2 минуты
4.		Какой оператор используется для выполнения условного выражения в Python? a) if b) when c) case	a	3 минуты

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
5.		<p>d) switch</p> <p>Какой из следующих способов используется для определения длины строки в Python? Почему не подходят другие?</p> <p>a) length() b) str.length() c) len() d) size()</p>	<p>С,</p> <p>a) length() — такой функции нет в Python. b) str.length() — это невалидный синтаксис для Python. d) size() — такой функции также нет для строк в Python.</p>	5 минут
6.	Задание открытого типа	<p>Объясните, что такое списки в Python и как они отличаются от кортежей. Приведите примеры использования обоих типов данных.</p>	<p>Списки в Python — это изменяемые (mutable) последовательности, которые могут содержать элементы различных типов. Они определяются с помощью квадратных скобок []. Кортежи, с другой стороны, являются неизменяемыми (immutable) последовательностями и определяются с помощью круглых скобок ().</p> <p>Пример списка: <code>my_list = [1, 2, 3, 'four', 5.0]</code>  <code>my_list.append(6)</code> # Добавление элемента  <code>print(my_list)</code> # Вывод: <code>[1, 2, 3, 'four', 5.0, 6]</code></p> <p>Пример кортежа: <code>my_tuple = (1, 2, 3, 'four', 5.0)</code>  <code># my_tuple[0] = 10</code> # Это вызовет ошибку, так как кортежи неизменяемы  <code>print(my_tuple)</code> # Вывод: <code>(1, 2, 3, 'four', 5.0)</code></p>	5-7 минут
7.		<p>Как работает механизм обработки исключений в Python? Напишите пример кода, который демонстрирует использование конструкции <code>try</code>, <code>except</code>, и объясните, как она работает.</p>	<p>Механизм обработки исключений в Python позволяет обрабатывать ошибки, которые могут возникнуть во время выполнения программы. Конструкция <code>try</code> используется для</p>	5 минут

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
			<p>обертывания кода, который может вызвать исключение, а <code>except</code> — для обработки этого исключения.</p> <p>Пример кода:</p> <pre>try:     result = 10 / 0 # Это вызовет деление на ноль except ZeroDivisionError:     print("Ошибка: Деление на ноль!")</pre> <p>В этом примере, если произойдет ошибка деления на ноль, программа не завершится аварийно, а выполнит код в блоке <code>except</code>, выводя сообщение об ошибке.</p>	
8.		<p>Что такое функции в Python и как они определяются?</p>	<p>Функции в Python — это блоки кода, которые выполняют определенную задачу и могут быть вызваны из других частей программы. Они определяются с помощью ключевого слова <code>def</code>.</p>	8 минут
9.		<p>Объясните, что такое генераторы в Python и как они отличаются от обычных функций. Приведите пример генератора и объясните, как он работает.</p>	<p>Генераторы в Python — это специальные функции, которые используют ключевое слово <code>yield</code> для возврата значений по одному за раз, вместо того чтобы возвращать все значения сразу, как это делает обычная функция. Генераторы позволяют экономить память и обрабатывать большие объемы данных.</p> <pre>def count_up_to(n):     count = 1     while count &lt;= n:         yield count         count += 1</pre>	5 минут

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
			# Пример использования for number in count_up_to(5): print(number) # Вывод: 1, 2, 3, 4, 5 В этом примере генератор count_up_to возвращает числа от 1 до n по одному за раз.	
10.		Как вы можете работать с файлами в Python?	В Python для работы с файлами используется встроенная функция open(). Файлы можно открывать в различных режимах, таких как чтение ('r'), запись ('w'), добавление ('a') и другие.	3 минуты
ПК-5: Способен собирать, анализировать исходные данные, рассчитывать экономические и социально-экономические показатели деятельности экономических агентов, а также обосновывать и представлять результаты расчетов в соответствии с принятыми стандартами и нормативно-правовой базой.				
11.	Задание закрытого типа	Какой из следующих вариантов является правильным способом создания списка в Python? a) list = {1, 2, 3} b) list = [1, 2, 3] c) list = (1, 2, 3) d) list = <1, 2, 3>	b	
12.		Какой из следующих вариантов правильно определяет функцию, которая возвращает квадрат числа? a) def square(x): return x * x b) function square(x): return x * x c) def square(x) return x * x d) square(x) = x * x	a	
13.		Какой из следующих вариантов кода создает кортеж в Python? a) my_tuple = [1, 2, 3] b) my_tuple = (1, 2, 3) c) my_tuple = {1, 2, 3} d) my_tuple = <1, 2, 3>	b	
14.		Какой из следующих вариантов кода правильно проверяет, является ли	a	

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		переменная $x$ положительным числом? a) if $x > 0$ : b) if ( $x > 0$ ) c) if $x > 0$ then: d) if $x > 0$ {}		
15.		Какой из следующих вариантов кода правильно импортирует модуль <code>math</code> в Python? Объясните свой ответ a) <code>import math()</code> b) <code>import math</code> c) <code>from math import *</code> d) <code>require math</code>	b, Он позволяет использовать все функции и константы, определенные в модуле <code>math</code> , с префиксом <code>math..</code> Например, чтобы использовать функцию <code>sqrt</code> для вычисления квадратного корня, нужно будет написать <code>math.sqrt(4)</code> .	
16.	Задание открытого типа	Напишите функцию для вычисления факториала	<pre>def factorial(n):     if n &lt; 0:         raise         ValueError("Факториал не определен для отрицательных чисел")     if n == 0 or n == 1:         return 1     return n * factorial(n - 1)  # Пример использования print(factorial(5)) # Вывод: 120</pre>	
17.		Напишите функцию <code>find_max(lst)</code> , которая принимает список чисел и возвращает максимальное значение в этом списке.	<pre>def find_max(lst):     if not lst:         raise         ValueError("Список не должен быть пустым")     max_value = lst[0]     for num in lst:         if num &gt; max_value:             max_value = num     return max_value  # Пример использования print(find_max([3, 1, 4, 1, 5, 9, 2])) # Вывод: 9</pre>	
18.		Напишите функцию <code>is_palindrome(s)</code> , которая принимает строку <code>s</code> и возвращает <code>True</code> , если строка является палиндромом	<pre>def is_palindrome(s):     s = s.lower().replace(" ",         "") # Приводим к нижнему регистру и убираем пробелы</pre>	

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		(читается одинаково слева направо и справа налево), и False в противном случае.	<pre>return s == s[::-1]</pre> <p># Пример использования  <pre>print(is_palindrome("A man a plan a canal Panama"))</pre> # Вывод: True</p>	
19.		Напишите функцию <code>sort_list(lst)</code> , которая принимает список чисел и возвращает новый список, отсортированный по возрастанию.	<pre>def sort_list(lst):     return sorted(lst)</pre> <p># Пример использования  <pre>print(sort_list([5, 3, 8, 1, 2]))</pre> # Вывод: [1, 2, 3, 5, 8]</p>	
20.		Напишите функцию <code>count_vowels(s)</code> , которая принимает строку <code>s</code> и возвращает количество гласных букв в этой строке.	<pre>def count_vowels(s):     vowels = "aeiouAEIOU"     count = 0     for char in s:         if char in vowels:             count += 1     return count</pre> <p># Пример использования  <pre>print(count_vowels("Hello, World!"))</pre> # Вывод: 3</p>	

Полный комплект оценочных материалов по дисциплине (модулю) (фонд оценочных средств) хранится в электронном виде на кафедре, утверждающей рабочую программу дисциплины (модуля).

#### 7.4. Методические материалы, определяющие процедуры оценивания результатов обучения по дисциплине (модулю)

*[Методические материалы составляют систему текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины (модуля), закрепляют виды и формы текущего контроля, сроки проведения, а также виды промежуточной аттестации по дисциплине (модулю), её сроки и формы проведения (устный зачёт / экзамен, письменный зачёт / экзамен и т. п.). В системе контроля указывается процедура оценивания результатов обучения по дисциплине (модулю) при использовании балльно-рейтинговой системы, показывается механизм получения оценки (из чего складывается оценка по дисциплине (модулю) в соответствии с балльно-рейтинговой системой), указывается система бонусов и штрафов, примерный набор дополнительных показателей.]*

*Раздел может быть представлен в табличной форме (курсивом в таблицах 10, 11 выделены примерные варианты заполнения)]*

**Таблица 10. Технологическая карта рейтинговых баллов по дисциплине (модулю)**

№ п/п	Контролируемые мероприятия	Количество мероприятий / баллы	Максимальное количество баллов	Срок представления
<b>Основной блок</b>				

№ п/п	Контролируемые мероприятия	Количество мероприятий / баллы	Максимальное количество баллов	Срок представления
1.	<i>Ответ на занятия</i>	1/10	10	В течение семестра
2.	<i>Лабораторная работа</i>	2/40	80	В конце семестра
<b>Всего</b>			<b>90</b>	-
<b>Блок бонусов</b>				
3.	<i>Посещение занятий</i>			
4.	<i>Своевременное выполнение всех заданий</i>	1/10	10	
<b>Всего</b>			<b>10</b>	-
<b>Дополнительный блок**</b>				
5.	<i>Дифференцированный зачет</i>		10	
<b>Всего</b>			<b>10</b>	-
<b>ИТОГО</b>			<b>100</b>	-

**Таблица 11. Система штрафов (для одного занятия)**

Показатель	Балл
<i>Опоздание на занятие (два и более)</i>	-2
<i>Нарушение учебной дисциплины</i>	-2
<i>Неготовность к занятию</i>	-1
<i>Пропуск занятия без уважительной причины</i>	-1

**Таблица 12. Шкала перевода рейтинговых баллов в итоговую оценку за семестр по дисциплине (модулю)**

Сумма баллов	Оценка по 4-балльной шкале
90–100	5 (отлично)
85–89	4 (хорошо)
75–84	
70–74	
65–69	3 (удовлетворительно)
60–64	
Ниже 60	2 (неудовлетворительно)

При реализации дисциплины (модуля) в зависимости от уровня подготовленности обучающихся могут быть использованы иные формы, методы контроля и оценочные средства, исходя из конкретной ситуации.

## **8. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)**

### **8.1. Основная литература**

Чернышев, С. А. Основы программирования на Python: учебник для вузов / С. А. Чернышев. — 2-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2024. - URL: <https://urait.ru/bcode/567821> (Образовательная платформа ЮРАЙТ)

### **8.2. Дополнительная литература**

Федоров, Д. Ю. Программирование на python: учебное пособие для вузов / Д. Ю. Федоров. — 6-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2025. - URL: <https://urait.ru/bcode/556864> (Образовательная платформа ЮРАЙТ)

### 8.3. Интернет-ресурсы, необходимые для освоения дисциплины (модуля)

<i>Наименование ЭБС</i>
<p><b>Цифровой образовательный ресурс IPRsmart:</b>            - ЭОР № 1 – программа для ЭВМ «Автоматизированная система управления цифровой библиотекой IPRsmart»;            - ЭОР № 2 – электронно-образовательный ресурс для иностранных студентов «<b>РУССКИЙ КАК ИНОСТРАННЫЙ</b>»  <a href="http://www.iprbookshop.ru">www.iprbookshop.ru</a></p>
<p><b>Электронно-библиотечная система BOOK.ru</b>  <a href="https://book.ru">https://book.ru</a></p>
<p><b>Образовательная платформа ЮРАЙТ,</b>  <a href="https://urait.ru/">https://urait.ru/</a></p>
<p><b>Электронная библиотека «Астраханский государственный университет» собственной генерации на платформе ЭБС «Электронный Читальный зал – БиблиоТех»</b>  <a href="https://biblio.asu.edu.ru">https://biblio.asu.edu.ru</a>  <i>Учётная запись образовательного портала АГУ</i></p>
<p><b>Электронно-библиотечная система (ЭБС) ООО «Политехресурс» «Консультант студента»</b>            Многопрофильный образовательный ресурс «Консультант студента» является электронной библиотечной системой, предоставляющей доступ через Интернет к учебной литературе и дополнительным материалам, приобретённым на основании прямых договоров с правообладателями. Каталог содержит более 15 000 наименований изданий.  <a href="http://www.studentlibrary.ru">www.studentlibrary.ru</a>  <i>Регистрация с компьютеров АГУ</i></p>

### 9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Практические занятия:

- Аудитория, оснащенная презентационной техникой: проектор, экран, компьютер (ноутбук), звуковые колонки, микрофон (в случае количества студентов более 80 человек);
- Рабочее место преподавателя, оснащенное компьютером с доступом в Интернет
- Для самостоятельной работы студенту предоставляется доступ к библиотеке, читальному залу, залу открытого доступа к сети Интернет, ПК.

### 10. ОСОБЕННОСТИ РЕАЛИЗАЦИИ ДИСЦИПЛИНЫ (МОДУЛЯ) ПРИ ОБУЧЕНИИ ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

Рабочая программа дисциплины (модуля) при необходимости может быть адаптирована для обучения (в том числе с применением дистанционных образовательных технологий) лиц с ограниченными возможностями здоровья, инвалидов. Для этого требуется заявление обучающихся, являющихся лицами с ограниченными возможностями здоровья, инвалидами, или их законных представителей и рекомендации психолого-медико-педагогической комиссии. При обучении лиц с ограниченными возможностями здоровья учитываются их индивидуальные психофизические особенности. Обучение инвалидов осуществляется также в соответствии с индивидуальной программой реабилитации инвалида (при наличии).

Для лиц с нарушением слуха возможно предоставление учебной информации в визуальной форме (краткий конспект лекций; тексты заданий, напечатанные увеличенным

шрифтом), на аудиторных занятиях допускается присутствие ассистента, а также сурдопереводчиков и тифлосурдопереводчиков. Текущий контроль успеваемости осуществляется в письменной форме: обучающийся письменно отвечает на вопросы, письменно выполняет практические задания. Доклад (реферат) также может быть представлен в письменной форме, при этом требования к содержанию остаются теми же, а требования к качеству изложения материала (понятность, качество речи, взаимодействие с аудиторией и т. д.) заменяются на соответствующие требования, предъявляемые к письменным работам (качество оформления текста и списка литературы, грамотность, наличие иллюстрационных материалов и т. д.). Промежуточная аттестация для лиц с нарушениями слуха проводится в письменной форме, при этом используются общие критерии оценивания. При необходимости время подготовки к ответу может быть увеличено.

Для лиц с нарушением зрения допускается аудиальное предоставление информации, а также использование на аудиторных занятиях звукозаписывающих устройств (диктофонов и т. д.). Допускается присутствие на занятиях ассистента (помощника), оказывающего обучающимся необходимую техническую помощь. Текущий контроль успеваемости осуществляется в устной форме. При проведении промежуточной аттестации для лиц с нарушением зрения тестирование может быть заменено на устное собеседование по вопросам.

Для лиц с ограниченными возможностями здоровья, имеющих нарушения опорно-двигательного аппарата, на аудиторных занятиях, а также при проведении процедур текущего контроля успеваемости и промежуточной аттестации могут быть предоставлены необходимые технические средства (персональный компьютер, ноутбук или другой гаджет); допускается присутствие ассистента (ассистентов), оказывающего обучающимся необходимую техническую помощь (занять рабочее место, передвигаться по аудитории, прочитать задание, оформить ответ, общаться с преподавателем).