

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кубанский государственный университет»
Факультет компьютерных технологий и прикладной математики

УТВЕРЖДАЮ:

Проректор по учебной работе,
качеству образования – первый
проректор

Иванов А.Г.



« 01 » июля

2016 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.В.ДВ.10.02 РАЗРАБОТКА WEB-ПРИЛОЖЕНИЙ С ИСПОЛЬЗОВАНИЕМ DJANGO

(код и наименование дисциплины в соответствии с учебным планом)

Направление подготовки/специальность 02.03.03 «Математическое
обеспечение и администрирование информационных систем»
(код и наименование направления подготовки/специальности)

Направленность (профиль) / специализация «Технология программирования»
(наименование направленности (профиля) специализации)

Программа подготовки академическая
(академическая /прикладная)

Форма обучения очная
(очная, очно-заочная, заочная)

Квалификация (степень) выпускника бакалавр
(бакалавр, магистр, специалист)

Краснодар 2016

Рабочая программа дисциплины РАЗРАБОТКА WEB-ПРИЛОЖЕНИЙ С ИСПОЛЬЗОВАНИЕМ DJANGO составлена в соответствии с федеральным государственным образовательным стандартом высшего образования (ФГОС ВО) по направлению подготовки 02.03.03 «Математическое обеспечение и администрирование информационных систем»

код и наименование направления подготовки

Программу составил(и):

Н.Ю. Добровольская, доцент, к. п. н., доцент

И.О. Фамилия, должность, ученая степень, ученое звание



подпись

А.В. Харченко, ст. преподаватель

И.О. Фамилия, должность, ученая степень, ученое звание



подпись

Рабочая программа дисциплины РАЗРАБОТКА WEB-ПРИЛОЖЕНИЙ С ИСПОЛЬЗОВАНИЕМ DJANGO утверждена на заседании кафедры информационных технологий протокол № 8 от «23»мая 2016 г.

Заведующий кафедрой (разработчик) Кольцов Ю.В

фамилия, инициалы



подпись

Рабочая программа обсуждена на заседании кафедры интеллектуальных информационных систем протокол № 4 от «20»апреля 2016 г.

Заведующий кафедрой (выпускающей) Костенко К.И.

фамилия, инициалы



подпись

Утверждена на заседании учебно-методической комиссии факультета компьютерных технологий и прикладной математики протокол № 7 от «29» июня 2016г

Председатель УМК факультета Малыхин К.В.

фамилия, инициалы



подпись

Рецензенты:

Рубцов С. Е., доцент кафедры математического моделирования ФГБОУ «КубГУ»

Бегларян М. Е., заведующий кафедрой СГЕНД СКФ ФГБОУВО «Российский государственный университет правосудия»,
канд. физ-мат. наук, доцент

1 Цели и задачи изучения дисциплины

1.1 Цель освоения дисциплины

Курс посвящен современному фреймворку языка программирования Python – Django, о его базовых возможностях, особенностях, способах интеграции с другими веб-библиотеками. Раскрываются подходы использования современной архитектуры построения веб-каркасов MVC. Описываются способы использования сторонних пакетов при проектировании приложений на Django. Особое внимание уделяется практическим вопросам использования базовых возможностей фреймворка.

Воспитательная цель: формирование свободного и творческого подхода к программированию на современных языках высокого уровня, интереса к наблюдению за тенденциями и новостями в области средств разработки программного обеспечения.

1.2 Задачи дисциплины

Использование многопроцессорных вычислительных систем предполагает практическое освоение следующих разделов параллельного программирования:

- изучение Python как мультипарадигменного языкового средства, достаточно полно отражающего современные концепции разработки ПО;
- формирование навыков создания web-приложений на языке Python;
- совершенствование и углубление навыков объектно-ориентированного и функционального программирования;
- знакомство с основами создания приложений для взаимодействия с базами данных на основе технологии DBAPI 2.0;

Отбор материала основывается на необходимости ознакомить студентов со следующей современной научной информацией:

- об особенностях и последних достижениях в области разработки кроссплатформенного ПО;
- о положительных и отрицательных чертах подхода к программированию, реализованному в языке Python.

Содержательное наполнение дисциплины обусловлено общими задачами в подготовке бакалавра.

Научной основой для построения программы данной дисциплины является теоретико-прагматический подход в обучении.

1.3 Место дисциплины в структуре образовательной программы

Дисциплина «Программирование на основе API» относится к вариативной части блока 1 дисциплины учебного плана.

Дисциплина «Разработка Web-приложений с использованием Django» логически и содержательно-методически связана с такими дисциплинами как «Алгоритмы. Теория и практическое применение», «Программирование на основе API», «Разработка приложений в MS VisualStudio». Является логически связанной с математическими дисциплинами, рассматривает объекты таких дисциплин как «Фундаментальные дискретные модели» с точки зрения программирования.

Входными знаниями для освоения данной дисциплины являются знания, умения и опыт, накопленный студентами в процессе изучения дисциплин «Фундаментальные дискретные модели», «Программирование на основе API». Обучающийся должен:

- иметь базовые навыки в написании программ на процедурных и объектно-ориентированных языках;

– быть знакомым с наиболее часто встречающимися структурами данных, уметь ими пользоваться и знать внутреннюю организацию.

1.4 Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы.

Изучение данной учебной дисциплины направлено на формирование у обучающихся профессиональных компетенций (ПК)

№ п.п.	Индекс компетенции	Содержание компетенции (или её части)	В результате изучения учебной дисциплины обучающиеся должны		
			знать	уметь	владеть
1.	ПК-5	готовностью к использованию современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ	классификацию и архитектуру современных операционных систем; - тенденции развития сервисных программ	использовать и классифицировать современные программные средства при решении различных прикладных задач; различать сетевые оболочки и сервисные программы по типам назначения и вариантам использования ;	навыками использования и администрирования современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ; навыками работы в команде по построению и разработке информационных систем и программного обеспечения; навыками организации процесса работы и администрирования операционных систем и оболочек

2. Структура и содержание дисциплины.

2.1 Распределение трудоёмкости дисциплины по видам работ.

Общая трудоёмкость дисциплины составляет Зач.ед. (108 часов), их распределение по видам работ представлено в таблице(для студентов ОФО).

Вид учебной работы	Всего	Семестры
--------------------	-------	----------

		часов	(часы)			
			5	_____		
Контактная работа, в том числе:						
Аудиторные занятия (всего):		54	54			
Занятия лекционного типа		18	18	-	-	-
Лабораторные занятия		36	36	-	-	-
Занятия семинарского типа (семинары, практические занятия)		-	-	-	-	-
		-	-	-	-	-
Иная контактная работа:						
Контроль самостоятельной работы (КСР)		6	6			
Промежуточная аттестация (ИКР)		0,2	0,2			
Самостоятельная работа, в том числе:						
Курсовая работа		-	-	-	-	-
Проработка учебного (теоретического) материала		18	18	-	-	-
Выполнение индивидуальных заданий		25	25	-	-	-
Реферат		-	-	-	-	-
Подготовка к текущему контролю		4,8	4,8	-	-	-
Контроль:						
Подготовка к экзамену		-	-			
Общая трудоемкость	час.	108	108	-	-	-
	в том числе контактная работа	60,2	60,2			
	зач. ед	3	3			

2.2 Структура дисциплины:

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.

Разделы дисциплины, изучаемые в 5 семестре (очная форма)

№	Наименование разделов	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	СРС
1	2	3	4	5	6	7
1.	Введение в Python	12	2	-	4	6
2.	Архитектура веб-приложений. Модель MVC	10	2	-	4	4
3.	Введение в фреймворк Django. Реализация концепции MVC в Django.	14	2	-	4	8
4.	Модели и шаблоны в Django.	16	4	-	6	6
5.	Представления в Django	19	4	-	8	7
6.	Механизмы сессий, авторизации и cookie	12	2	-	4	6
7.	Работа со static и media файлами	12	2	-	4	6
8.	Обзор изученного материала и прием зачета	6,8	-	-	2	4,8
9.	Контроль самостоятельной работы (КСР)	6				
10.	Промежуточная аттестация (ИКР)	0,2				
	<i>Итого по дисциплине:</i>	108	18	-	36	47,8

Примечание: Л – лекции, ПЗ – практические занятия / семинары, ЛР – лабораторные занятия, СРС – самостоятельная работа студента

2.3 Содержание разделов дисциплины:

2.3.1 Занятия лекционного типа.

№	Наименование раздела	Содержание раздела	Форма текущего контроля
1	2	3	4
1.	Введение в Python	Введение. Классификация, сферы применения, парадигма. Синтаксис. Форматирование блоков кода. Числовые, булевы, строковые литералы. Основные операции над простыми типами. Простые управляющие конструкции. Декларация функций. Модули (импортирование). Соглашения по оформлению кода (PEP8). Организация кода в модули и пакеты. Синтаксические конструкции, циклы, условия. Коллекции: списки, строки, словари, кортежи, множества. Объектно-ориентированное программирование, элементы функционального программирования. Генераторы. Системные библиотеки math, time, os, shutil, sys, random.	<i>T</i>
2.	Архитектура веб-приложений. Модель MVC	Сеть интернет. OSI-модель. Эволюция web-приложений. Архитектура web-приложения. HTTP-протокол. MVC-модель Запрос, ответ, параметры, GET, POST запросы. Анализ GET и POST запросов с помощью средств разработки в браузерах.	<i>K</i>
3.	Введение в фреймворк Django. Реализация концепции MVC в Django.	Структура приложений на Django. Установка и настройка Django. Работа с URL в Django. Базовые модели, view, шаблоны. Класс HttpResponse.	<i>K</i>
4.	Модели и шаблоны в Django.	Реляционные модели. ORM-подход. Сравнение подходов. Класс models.Model, реализация основных полей в Django. Использование миграций и fixtures. Работа с формами в Django. Модельные формы. Основы языка шаблонов в Django. Подключение шаблонов. Пути к шаблонам. Сбор шаблонов. Основные операции в шаблонах. Наследование шаблонов. Передача параметров в шаблоны. Фильтры.	<i>T</i>
5.	Представления в Django	Представления как реализация контроллера в MVC-модели. Обработка запроса через urls.py, вызов представления. Обработка POST и GET данных. Редирект. Представления в виде функций. Именованные и позиционные аргументы представления. Параметр request. Метод	<i>K</i>

		render, класс HttpResponseRedirect. Представления, основанные на классах.	
6.	Механизмы сессий, авторизации и cookie	Авторизация, аутентификация, регистрация. Принцип работы сессий и куки. Механизмы авторизации в Django. Права доступа. Декораторы. Доступ к сессии и cookie с помощью request.	<i>T</i>
7.	Работа со static и media файлами	Языки HTML, CSS, JS. Подключение static-файлов, сбор static-файлов, пути их поиска. Загрузка файлов, пути к медиа-файлам. Работа с версткой и JS. AJAX-запрос.	<i>K</i>

2.3.2 Занятия семинарского типа.

Занятия семинарского типа не предусмотрены.

2.3.3 Лабораторные занятия.

№	Наименование лабораторных работ	Форма текущего контроля
1	3	4
1.	Знакомство с понятием динамической типизации. Обзор популярных языков с динамической типизацией.	Отчет по лабораторной работе
2.	Структура приложения. Типы, переменные. Управление потоком выполнения. Отладка. Стиль написания кода. Документирование и оценка производительности.	Отчет по лабораторной работе
3.	Введение. Классификация, сферы применения, парадигма. Синтаксис. Форматирование блоков кода. Числовые, булевы, строковые литералы. Основные операции над простыми типами. Простые управляющие конструкции. Декларация функций. Модули (импортирование). Соглашения по оформлению кода (PEP8). Организация кода в модули и пакеты. Синтаксические конструкции, циклы, условия. Коллекции: списки, строки, словари, кортежи, множества. Объектно-ориентированное программирование, элементы функционального программирования. Генераторы. Системные библиотеки math, time, os, shutil, sys, random.	Отчет по лабораторной работе
4.	Сеть интернет. OSI-модель. Эволюция web-приложений. Архитектура web-приложения. HTTP-протокол. MVC-модель Запрос, ответ, параметры, GET, POST запросы. Анализ GET и POST запросов с помощью средств разработки в браузерах.	Отчет по лабораторной работе
5.	Структура приложений на Django. Установка и настройка Django. Работа с URL в Django. Базовые модели, view, шаблоны. Класс HttpResponse.	Отчет по лабораторной работе
6.	Реляционные модели. ORM-подход. Сравнение подходов. Класс	Отчет по

	models.Model, реализация основных полей в Django. Использование миграций и fixtures. Работа с формами в Django. Модельные формы. Основы языка шаблонов в Django. Подключение шаблонов. Пути к шаблонам. Сбор шаблонов. Основные операции в шаблонах. Наследование шаблонов. Передача параметров в шаблоны. Фильтры.	лабораторной работе
7.	Представления как реализация контроллера в MVC-модели. Обработка запроса через urls.py, вызов представления. Обработка POST и GET данных. Редирект. Представления в виде функций. Именованные и позиционные аргументы представления. Параметр request. Метод render, класс HttpResponseRedirect. Представления, основанные на классах.	Отчет по лабораторной работе
8.	Авторизация, аутентификация, регистрация. Принцип работы сессий и куки. Механизмы авторизации в Django. Права доступа. Декораторы. Доступ к сессии и cookie с помощью request.	Отчет по лабораторной работе
9.	Языки HTML, CSS, JS. Подключение static-файлов, сбор static-файлов, пути их поиска. Загрузка файлов, пути к медиа-файлам. Работа с версткой и JS. AJAX-запрос.	Отчет по лабораторной работе

Защита лабораторной работы (ЛР), выполнение курсового проекта (КП), курсовой работы (КР), расчетно-графического задания (РГЗ), написание реферата (Р), эссе (Э), коллоквиум (К), тестирование (Т) и т.д.

2.3.4 Примерная тематика курсовых работ (проектов)

Курсовые работы не предусмотрены.

2.4 Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

№	Вид СРС	Перечень учебно-методического обеспечения дисциплины по выполнению самостоятельной работы
1	2	3
1	Решение индивидуальных задач	Стандарты оформления исходного кода программ и современные интегрированные среды разработки программного обеспечения: учеб.-метод.пособие/ Ю.В.Кольцов [и др.]. – Краснодар:Кубанскийгос.ун-т, 2015.-111с., утвержденные кафедрой информационных технологий, протокол № 7 от 09 апреля 2015 г..
2	Отчет по лабораторной работе	Стандарты оформления исходного кода программ и современные интегрированные среды разработки программного обеспечения: учеб.-метод.пособие/ Ю.В.Кольцов [и др.]. – Краснодар:Кубанскийгос.ун-т, 2015.-

		111с., утвержденные кафедрой информационных технологий, протокол № 7 от 09 апреля 2015 г.
--	--	---

Учебно-методические материалы для самостоятельной работы обучающихся из числа инвалидов и лиц с ограниченными возможностями здоровья (ОВЗ) предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа,

Для лиц с нарушениями слуха:

- в печатной форме,
- в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

- в печатной форме,
- в форме электронного документа,

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

3. Образовательные технологии

При проведении занятий по дисциплине используются следующие образовательные технологии:

- Технология разноуровневого обучения (дифференцированное обучение);
- Технология коллективного взаимодействия (организованный диалог, коллективный способ обучения);
- Технология адаптивного обучения (индивидуализированное обучение).

Для лиц с ограниченными возможностями здоровья предусмотрена организация консультаций с использованием электронной почты.

4. Оценочные средства для текущего контроля успеваемости и промежуточной аттестации.

4.1 Фонд оценочных средств для проведения текущего контроля.

Задачи для самостоятельного решения

Перечень части компетенций, проверяемых оценочным средством

ПК-5 - Готовностью к использованию современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ.

Введение в Python

1. Дана последовательность натуральных чисел, завершающаяся числом 0. Определите, какое наибольшее число подряд идущих элементов этой последовательности равны друг другу.
2. Дана последовательность натуральных чисел, завершающаяся числом 0. Определите наибольшую длину монотонного фрагмента последовательности (то есть такого фрагмента, где все элементы либо больше предыдущего, либо меньше).
3. Дана последовательность натуральных чисел, завершающаяся числом 0. Определите количество строгих локальных максимумов в этой последовательности. (*Элемент последовательности называется локальным максимумом, если он строго больше предыдущего и последующего элемента последовательности. Первый и последний элемент последовательности не являются локальными максимумами.*)

4. Определите наименьшее расстояние между двумя локальными максимумами последовательности натуральных чисел, завершающейся числом 0. Если в последовательности нет двух локальных максимумов, выведите число 0.
5. В списке все элементы различны. Поменяйте местами минимальный и максимальный элемент этого списка.

Архитектура веб-приложений. Модель MVC

Создать Django приложение, реализующее механизмы регистрации, авторизации пользователей, также позволяющее: Вести свой блог (модель поста) и оставлять комментарии к блогам других пользователей (модель комментария)

Введение в фреймворк Django. Реализация концепции MVC в Django.

Создать Django приложение, реализующее механизмы регистрации, авторизации пользователей, также позволяющее: Вести свою личную страницу с указанием информации о себе (модель информация) и писать другим пользователям личные сообщения (модель сообщения)

Модели и шаблоны в Django.

Создать Django приложение, реализующее механизмы регистрации, авторизации пользователей, также позволяющее: Вести свой блог (модель поста) и оставлять отметки "нравится" к блогам других пользователей, с которыми пользователь находится в отношении "друг" (модель друга)

Представления в Django

Создать простой сервис проведения тестирования по каким-либо темам. Т.е. есть тесты с вариантами ответов, один или несколько вариантов должны быть правильными. Тесты группируются в наборы тестов, которые затем пользователь может проходить и видеть свой результат. Функциональные части сервиса: Регистрация пользователей, Аутентификация пользователей.

Механизмы сессий, авторизации и cookie

Создать простой сервис проведения тестирования по каким-либо темам. Т.е. есть тесты с вариантами ответов, один или несколько вариантов должны быть правильными. Тесты группируются в наборы тестов, которые затем пользователь может проходить и видеть свой результат. Функциональные части сервиса:

- Регистрация пользователей
- Аутентификация пользователей
- Зарегистрированные пользователи могут
- Проходить любой из тестовых наборов
- Последовательный ответ на все вопросы, каждый вопрос должен выводиться на новой странице с отправкой формы (перескакивать через тесты или оставлять неотмеченными нельзя)
- После завершения тестирования смотреть результат:
- количество правильных/неправильных ответов
- процент правильных ответов

Работа со static и media файлами

Админка Django. Разделы:

- Стандартный раздел пользователей
- Раздел с наборами тестов
- Возможность на странице набора тестов добавлять вопросы/ответы к вопросам/отмечать правильные ответы

- Валидация на то, что должен быть хотябы 1 правильный вариант
- Валидация на то, что все варианты не могут быть правильными
- Удаление вопросов/вариантов ответов/изменение правильных решений при редактировании тестового набора

4.2 Фонд оценочных средств для проведения промежуточной аттестации

Зачетно-экзаменационные материалы для промежуточной аттестации (зачет)

Перечень части компетенций, проверяемых оценочным средством

ПК-5 - Готовностью к использованию современных системных программных средств: операционных систем, операционных и сетевых оболочек, сервисных программ.

Вопросы для подготовки к зачету

1. Введение в серверное Web-программирование
2. Язык программирования Python
3. Библиотека Django
4. Создание проекта и приложения Django
5. Модели Django
6. Контроллеры Django. Регулярные выражения
7. Простые шаблоны Django
8. Постраничный вывод данных. Пагинатор Django
9. Вывод на основе классов. Классы-контроллеры Django
10. Простые формы Django
11. Выгрузка файлов на Web-сайт
12. Разграничение доступа
13. Комментарии Django
14. Статичные страницы Django
15. Создание и вывод миниатюр. Библиотека easy-thumbnails
16. Привязка тегов к данным. Библиотека django-taggit
17. Создание web-сайта. Планирование и предварительные действия
18. Главная страница
19. Гостевая книга
20. Список новостей. Хранилище изображений
21. Список категорий товаров
22. Список товаров
23. Блог
24. Генерирование каналов новостей RSS и Atom
25. Рассылка электронной почты
26. Журналирование
27. Настройка встроенного административного сайта Django
28. Публикация Web-сайта

Компонентом промежуточного контроля по дисциплине являются решение задачи из списка индивидуальных задач и ответа на теоретический вопрос. Максимальное количество баллов, которые студент может получить за правильное решение задачи, составляет 2 балла. Максимальное количество баллов, которые студент может получить за ответ на контрольный вопрос, составляет 10 баллов.

Рекомендации по оцениванию задачи

Описание	Баллы
Предоставлен работоспособный программный код, студент может пояснить ход решения, знает назначение команд, может изменить некоторые условия по	2

просьбе преподавателя.	
Программный код может быть не работоспособен, однако алгоритм решения задачи корректный, студент может пояснить ход решения, знает назначение некоторых команд	1
Программный код не работает, алгоритм решения не верный, студент не знает назначения отдельных команд	0

Рекомендации по оцениванию ответа на контрольный вопрос

Описание	Баллы
Студент владеет теоретическими знаниями по данному разделу, что подтверждается его ответами на дополнительные вопросы; студент умеет правильно объяснять теоретический материал, иллюстрируя его примерами;	8-10
Студент владеет теоретическими знаниями по данному разделу, что подтверждается его ответами на дополнительные вопросы, при ответе студент допускает незначительные ошибки; студент умеет правильно объяснять теоретический материал;	4-7
Теоретический материал не усвоен или усвоен частично, студент не может предоставить четкий ответ на поставленный вопрос; студент затрудняется привести примеры, поясняющие ответы на вопросы;	0-3

Критерии оценки:

- **оценка «зачтено»:** студент получил не менее 4 баллов за контрольный вопрос, студент получил от 7 до 10 баллов за индивидуальные задачи, при этом по каждой из них он получил хотя бы один балл.

- **оценка «не зачтено»:** студент получил менее 4 баллов за контрольный вопрос, или менее 1 балла хотя бы за одну из 7 индивидуальных задач.

Оценка	
Не зачтено	Зачтено
- студент получил менее 4 баллов за контрольный вопрос - менее 1 балла хотя бы за одну из 7 индивидуальных задач	- студент получил не менее 4 баллов за контрольный вопрос; - не менее 1 балла за каждую из 7 индивидуальных задач;

Оценочные средства для инвалидов и лиц с ограниченными возможностями здоровья выбираются с учетом их индивидуальных психофизических особенностей.

– при необходимости инвалидам и лицам с ограниченными возможностями здоровья предоставляется дополнительное время для подготовки ответа на экзамене;

– при проведении процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья предусматривается использование технических средств, необходимых им в связи с их индивидуальными особенностями;

– при необходимости для обучающихся с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения по дисциплине может проводиться в несколько этапов.

Процедура оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в печатной форме увеличенным шрифтом,
- в форме электронного документа.

Для лиц с нарушениями слуха:

- в печатной форме,

– в форме электронного документа.

Для лиц с нарушениями опорно-двигательного аппарата:

– в печатной форме,

– в форме электронного документа.

Данный перечень может быть конкретизирован в зависимости от контингента обучающихся.

5 Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

5.1 Основная литература:

1. Буйначев, С.К. Основы программирования на языке Python : учебное пособие / С.К. Буйначев, Н.Ю. Боклаг ; Министерство образования и науки Российской Федерации, Уральский федеральный университет имени первого Президента России Б. Н. Ельцина. - Екатеринбург : Издательство Уральского университета, 2014. - 92 с. : табл., ил. - Библиогр. в кн. - ISBN 978-5-7996-1198-9 ; [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=275962>

2. Северенс, Ч. Введение в программирование на Python / Ч. Северенс. - 2-е изд., испр. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 231 с.: схем., ил.; [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=429184>

3. Хахаев, И.А. Практикум по алгоритмизации и программированию на Python : курс / И.А. Хахаев. - 2-е изд., исправ. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 179 с. : ил. - Библиогр. в кн.; [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=429256>

4. Столбовский, Д.Н. Основы разработки Web-приложений на ASP.NET : учебное пособие / Д.Н. Столбовский ; Национальный Открытый Университет "ИНТУИТ". - Москва : Интернет-Университет Информационных Технологий, 2009. - 304 с. - (Основы информационных технологий). - ISBN 978-5-94774-991-5 ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=233488>

Для освоения дисциплины инвалидами и лицами с ограниченными возможностями здоровья имеются издания в электронном виде в электронно-библиотечной системе «Biblioclub».

5.2 Дополнительная литература:

1. Сузи, Р.А. Язык программирования Python : курс / Р.А. Сузи. - 2-е изд., испр. - Москва : Интернет-Университет Информационных Технологий, 2007. - 327 с. - (Основы информационных технологий). - ISBN 978-5-9556-0109-0; [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=233288>

2. Богданов, М.Р. Разработка клиентских приложений Web-сайтов : курс / М.Р. Богданов. - Москва : Интернет-Университет Информационных Технологий, 2010. - 228 с. : ил. ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=233745>

3. Sweigart, A. Разработка компьютерных игр на языке Python / A. Sweigart. - 2-е изд., испр. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 505 с. : ил ; [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=429009>

5.3. Периодические издания:

1. Прикладная информатика
2. Проблемы передачи информации
3. Программные продукты и системы
4. Программирование
5. COMPUTATIONAL NANOTECHNOLOGY (ВЫЧИСЛИТЕЛЬНЫЕ НАНОТЕХНОЛОГИИ)
6. COMPUTERWORLD РОССИЯ
7. WINDOWS IT PRO / RE

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

1. Python Software Foundation. – URL: <http://www.python.org>
2. Сузи, Р.А. Язык программирования Python : курс / Р.А. Сузи. - 2-е изд., испр. – М.: Интернет-Университет Информационных Технологий, 2007. - 327 с. - (Основы информационных технологий). - ISBN 978-5-9556-0109-0 ; [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=233288>

7. Методические указания для обучающихся по освоению дисциплины

После прослушивания лекции рекомендуется выполнить упражнения, приводимые в лекции для самостоятельной работы, а также выполнить на компьютере с использованием среды Python задачи, приводимые в лекции в качестве примеров.

При самостоятельной работе студентов необходимо изучить литературу, приведенную в перечнях выше, для осмысления вводимых понятий, анализа предложенных подходов и методов разработки параллельных программ. Разрабатывая решение новой задачи, студент должен уметь выбрать методы решения задачи с учетом целевой аппаратной платформы, проводить отладку и профилирование программы на языке Python.

В качестве систем программирования для решения задач и изучения методов и алгоритмов, приведенных в лекциях, рекомендуется использовать на практических занятиях и при самостоятельной работе стандартную реализацию языка Python в связке со средой разработки IDLE.

Раздел дисциплины	Форма СР	Сроки выполнения	Формы контроля
Введение. Классификация, сферы применения, парадигма. Синтаксис. Форматирование блоков кода. Числовые, булевы, строковые литералы. Основные операции над простыми типами. Простые управляющие конструкции. Декларация функций. Модули (импортирование). Соглашения по оформлению кода (PEP8). Организация кода в модули и пакеты. Синтаксические конструкции, циклы, условия. Коллекции: списки, строки, словари, кортежи,	Проработка теоретического материала. Решение индивидуальных задач.	15.09	Сдача индивидуальных задач

множества. Объектно-ориентированное программирование, элементы функционального программирования. Генераторы. Системные библиотеки math, time, os, shutil, sys, random.			
Сеть интернет. OSI-модель. Эволюция web-приложений. Архитектура web-приложения. HTTP-протокол. MVC-модель Запрос, ответ, параметры, GET, POST запросы. Анализ GET и POST запросов с помощью средств разработки в браузерах.	Проработка теоретического материала. Решение индивидуальных задач.	01.10	Сдача индивидуальных задач
Структура приложений на Django. Установка и настройка Django. Работа с URL в Django. Базовые модели, view, шаблоны. Класс HttpResponseRedirect.	Проработка теоретического материала. Решение индивидуальных задач.	15.10	Сдача индивидуальных задач
Реляционные модели. ORM-подход. Сравнение подходов. Класс models.Model, реализация основных полей в Django. Использование миграций и fixtures. Работа с формами в Django. Модельные формы. Основы языка шаблонов в Django. Подключение шаблонов. Пути к шаблонам. Сбор шаблонов. Основные операции в шаблонах. Наследование шаблонов. Передача параметров в шаблоны. Фильтры.	Проработка теоретического материала. Решение индивидуальных задач.	01.11	Сдача индивидуальных задач
Представления как реализация контроллера в MVC-модели. Обработка запроса через urls.py, вызов представления. Обработка POST и GET данных. Редирект. Представления в виде функций. Именованные и позиционные аргументы представления. Параметр request. Метод render, класс HttpResponseRedirect. Представления, основанные на классах.	Проработка теоретического материала. Решение индивидуальных задач.	10.11	Сдача индивидуальных задач
Авторизация, аутентификация, регистрация. Принцип работы сессий	Проработка теоретического	20.11	Сдача индивидуальных

и куки. Механизмы авторизации в Django. Права доступа. Декораторы. Доступ к сессии и cookie с помощью request.	материала. Решение индивидуальных задач.		задач
Языки HTML, CSS, JS. Подключение static-файлов, сбор static-файлов, пути их поиска. Загрузка файлов, пути к медиа-файлам. Работа с версткой и JS. AJAX-запрос.	Проработка теоретического материала. Решение индивидуальных задач.	01.12	Сдача индивидуальных задач
Введение. Классификация, сферы применения, парадигма. Синтаксис. Форматирование блоков кода. Числовые, булевы, строковые литералы. Основные операции над простыми типами. Простые управляющие конструкции. Декларация функций. Модули (импортирование). Соглашения по оформлению кода (PEP8). Организация кода в модули и пакеты. Синтаксические конструкции, циклы, условия. Коллекции: списки, строки, словари, кортежи, множества. Объектно-ориентированное программирование, элементы функционального программирования. Генераторы. Системные библиотеки math, time, os, shutil, sys, random.	Проработка теоретического материала. Решение индивидуальных задач.	10.12	Сдача индивидуальных задач
Сеть интернет. OSI-модель. Эволюция web-приложений. Архитектура web-приложения. HTTP-протокол. MVC-модель. Запрос, ответ, параметры, GET, POST запросы. Анализ GET и POST запросов с помощью средств разработки в браузерах.	Проработка теоретического материала. Решение индивидуальных задач.	20.12	Сдача индивидуальных задач

В освоении дисциплины инвалидами и лицами с ограниченными возможностями здоровья большое значение имеет индивидуальная учебная работа (консультации) – дополнительное разъяснение учебного материала.

Индивидуальные консультации по предмету являются важным фактором, способствующим индивидуализации обучения и установлению воспитательного контакта между преподавателем и обучающимся инвалидом или лицом с ограниченными возможностями здоровья.

8. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине.

8.1 Перечень информационных технологий

- Проверка домашних заданий и консультирование посредством электронной почты.
- Использование электронных презентаций при проведении лекционных занятий.

8.2 Перечень необходимого программного обеспечения

- Интерпретатор языка Python
- Среда разработки IDLE

8.3 Перечень информационных справочных систем:

- Электронная библиотечная система eLIBRARY.RU (<http://www.elibrary.ru/>)

9. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

№	Вид работ	Материально-техническое обеспечение дисциплины и оснащенность
1.	Лекционные занятия	Лекционная аудитория, оснащенная презентационной техникой (проектор, экран, компьютер/ноутбук) и соответствующим программным обеспечением (ПО), доска
2.	Лабораторные занятия	Лаборатория, укомплектованная техническими средствами обучения – компьютерами с соответствующим программным обеспечением, маркерная доска.
3.	Групповые (индивидуальные) консультации	Аудитория, укомплектованная маркерной доской и оснащенная компьютером.
4.	Текущий контроль, промежуточная аттестация	Аудитория, укомплектованная техническими средствами обучения – компьютерами с соответствующим программным обеспечением
5.	Самостоятельная работа	Кабинет для самостоятельной работы, оснащенный компьютерной техникой с возможностью подключения к сети «Интернет», программой экранного увеличения и обеспеченный доступом в электронную информационно-образовательную среду университета.