

УНИВЕРСИТЕТ «СИНЕРГИЯ»

Актуализированная версия
утверждена на заседании Ученого совета
Университета «Синергия»
протокол № 11 от 28.11.2025 г.

Утверждаю
Ректор Университета «Синергия»
кандидат экономических наук, доцент
А. И. Васильев
01.12.2025 г

КОМПЛЕКТ РАБОЧИХ ПРОГРАММ

УЧЕБНЫХ ПРЕДМЕТОВ ДОПОЛНИТЕЛЬНОЙ ОБЩЕОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ «Разработка на PHP»

Наименование программы

г. Москва 2025

УНИВЕРСИТЕТ «СИНЕРГИЯ»

Актуализированная версия
утверждена на заседании Ученого совета
Университета «Синергия»
протокол № 11 от 28.11.2025 г.

Утверждаю
Ректор Университета «Синергия»
кандидат экономических наук, доцент
А. И. Васильев
01.12.2025 г

РАБОЧАЯ ПРОГРАММА УЧЕБНОГО ПРЕДМЕТА «Разработка на PHP»

(наименование дисциплины (курса, учебного предмета) по учебному плану)

Москва 2025

1. Цель изучения учебного предмета

1. Целью освоения учебного предмета является формирование навыков работы с кодом на языке PHP у обучающихся средствами проектных технологий в сфере серверной веб-разработки.

2. Планируемые результаты обучения по учебному предмету

Знать	<ul style="list-style-type: none">• основные понятия о профессии «Разработчик на PHP»;• основы синтаксиса PHP;• принципы работы веб-серверов.
Уметь	<ul style="list-style-type: none">• создавать коммерческие проекты в цифровой среде;• работать с синтаксисом PHP;• планировать собственную деятельность;• искать и выделять необходимую информацию, структурировать свои знания.
Владеть	<ul style="list-style-type: none">• основными инструментами разработчика;• техниками работы с веб-серверами и HTTP запросами;• навыками коммуникативных отношений внутри проектных групп и в коллективе.

3. Содержание учебного предмета

3.1. Распределение учебного времени, выделенного на контактную работу обучающихся с преподавателем, на самостоятельную работу обучающихся и учебные часы с использованием дистанционных образовательных технологий

Общая трудоемкость (объем) учебного предмета составляет 72 академических часа.

Продолжительность: 12 недель.

Недельная нагрузка: 6 часов в неделю.

Предварительный режим занятий: 3 дня в неделю по 2 часа.

№	Наименование тем учебного предмета	Общая трудоемкость, ч.	Всего, ч. Сумма	Контактная работа, ч			Учебные занятия с применением дистанционных технологий, ч	Самостоятельная работа, ч	Форма аттестации
				Лекции	Лабораторные работы	Практические и семинарские занятия			
1	2	3	4	5	6	7	8	9	10
1	Тема 1. Введение в PHP	18	0	0	0	0	9	9	-
2	Тема 2. Операторы и функции	18	0	0	0	0	9	9	-
3	Тема 3. Работа с объектами	18	0	0	0	0	9	9	-
4	Тема 4. Первое веб-приложение	17	0	0	0	0	8	9	-
	Промежуточная аттестация	1	0	0	0	0	0	0	Зачет
	Итого:	72	0	0	0	0	35	36	1

3.2. Содержание тем учебного предмета

Тема 1. Введение в PHP

- Знакомство с PHP. Основы синтаксиса и типы данных. Сравнения, константы и ветвления. Консольные скрипты, интерактивная консоль, настройки PHP, потоки ввода-вывода, аргументы запуска скриптов. Массивы, отслеживание потребления памяти. Циклы for-while и do-while.

Тема 2. Операторы и функции

- Операторы: арифметические, строковые, присваивания, комбинированные, сравнения, исполнения. Базовые функции работы со строками, введение в регулярные выражения. Синтаксис создания собственных функций, аргументы, возвращаемые значения. Строгая и не строгая типизация. Рекурсивные функции. Генераторы, анонимные функции, замыкания.

Тема 3. Работа с объектами

- Основы работы с объектами: манипуляции с датой и временем. Виды и обработка ошибок. Механизм исключений. Протокол HTTP запросов, отправка заголовков и ответа. Cookies и сессии.

Тема 4. Первое веб-приложение

- Разносим код приложения по разным файлам, функции работы с внешними файлами. HTML формы, обработка GET и POST запросов. Форматы данных: json, xml, yaml. Веб-сервера, виды хостингов. Разрешение зависимостей при помощи composer

4. Учебно-методическое и информационное обеспечение (курса, учебного предмета)

№ п/п	Вид и наименование литературы
Основная:	
1.	Шабашов, В. Я. Организация доступа к данным из PHP приложений для различных СУБД : учебное пособие по дисциплине «Web-программирование» : [16+] / В. Я. Шабашов. – Москва ; Берлин : Директ-Медиа, 2019. – 121 с. : ил., табл. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=499185 (дата обращения: 15.02.2023). – Библиогр.: с. 90. – ISBN 978-5-4475-9888-4. – DOI 10.23681/499185. – Текст : электронный.
2.	Маркин, А. В. Основы web-программирования на PHP : учебное пособие / А. В. Маркин, С. С. Шкарин. – Москва : Диалог-МИФИ, 2012. – 252 с. : табл., схем., ил. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=229742 (дата обращения: 15.02.2023). – Библиогр.: с. 238. – ISBN 978-5-86404-241-0. – Текст : электронный.
3.	Марухленко, А. Л. Разработка защищённых интерфейсов Web-приложений : учебное пособие : [16+] / А. Л. Марухленко, Л. О. Марухленко, М. А. Ефремов. – Москва ; Берлин : Директ-Медиа, 2021. – 175 с. : ил., табл. – Режим доступа: по подписке. – URL: https://biblioclub.ru/index.php?page=book&id=599050 (дата обращения: 15.02.2023). – Библиогр. в кн. – ISBN 978-5-4499-1676-1. – DOI 10.23681/599050. – Текст : электронный.
Дополнительная:	
5.	Мартин Р. Чистый код: создание, анализ и рефакторинг – М.: Питер, 2021 – 58 с.
6.	Локхарт Дж. Современный PHP. Новые возможности и передовой опыт. – М.: ДМК Пресс, 2022 – 38 с.
Информационное обеспечение	
7.	Гарант http://www.garant.ru
8.	Образовательная платформа lms.synergy.ru
9.	ЭБС Юрайт urait.ru

5. Учебно-материальная база, необходимая для осуществления образовательного процесса по (курсу, учебному предмету)

Материально-техническое обеспечение учебного предмета включает в себя:

- персональный компьютер/мобильное устройство (обучающийся обеспечивает самостоятельно) с любой операционной системой, позволяющей использовать браузеры и подключаться к сети «Интернет»;
- стандартный пакет офисных программ (обучающийся обеспечивает самостоятельно, полный список возможно уточнить у куратора программы);
- обеспечение доступа в электронную информационно-образовательную среду Университета - lms.synergy.ru;
- обеспечение доступа в электронную библиотеку urait.ru.

6. Методические рекомендации (указания, материалы) для преподавателей и обучающихся

В процессе освоения учебного предмета используются такие виды учебных занятий и работ как: учебные занятия с применением дистанционных технологий, самостоятельная работа.

Методические указания для обучающихся при обучении в виде занятий с применением ДОТ

В процессе занятий с ДОТ рекомендуется вести конспект, что позволит впоследствии вспомнить изученный учебный материал, выполнить самостоятельную работу.

Желательно оставить в рабочих конспектах поля, на которых делать пометки из рекомендованной литературы, дополняющие материал прослушанного занятия с применением ДОТ.

Занятия с применением ДОТ имеют логическое завершение, роль которого выполняет заключение. Также в завершении занятия с применением ДОТ преподаватель знакомит обучающихся с литературой (основной, дополнительной), с практическими заданиями для самостоятельной работы и даёт рекомендации по их выполнению. Полученную информацию целесообразно кратко и лаконично записывать.

Методические указания для преподавателей при проведении занятий с применением ДОТ

Занятия с применением ДОТ (далее занятие) – систематическое, последовательное изложение преподавателем учебного материала, записанное на видеокамеру.

Содержание тем занятий должно соответствовать цели учебного предмета и способствовать достижению планируемых результатов обучения по нему.

Объяснение нового материала рекомендуется начать с актуализации ранее изученных обучающимися тем, а также раскрытия цели и задач занятия. Затем перейти к его основной части, в которой формируются планируемые результаты обучения (знания) по учебному предмету.

Завершается занятие выводами по изученной теме. В этой части необходимо подчеркнуть, что обучающийся узнал нового на основе учебных элементов занятия. Также преподавателю необходимо ознакомить обучающихся с литературой (основной, дополнительной), с практическими заданиями и дать рекомендации по их выполнению.

Методические указания для обучающихся по выполнению самостоятельной работы

По мере прослушивания занятий в видеозаписи, слушатель последовательно выполняет самостоятельную работу, в которую входят: изучение рекомендованной (обязательной и дополнительной) литературы, выполнение заданий.

Прежде чем приступать к выполнению самостоятельной работы, слушателю необходимо подготовить оформленные в процессе занятий записи конспектов.

В ходе выполнения самостоятельной работы необходимо следовать рекомендациям, полученным от преподавателя в конце занятия, использовать материал занятий и конспекты, оформленные по ним, рекомендованную литературу.

Методические указания для преподавателей по организации самостоятельной работы

Для организации самостоятельной работы обучающихся преподаватель задания, направленные на формирование и закрепление полученных знаний по учебному предмету. Задания с указаниями размещаются на платформе электронного обучения lms.synergy.ru.

Перечень заданий для самостоятельной работы

Тема 1.

Задание 1.

Напишите PHP-скрипт, который объявляет переменную \$name и присваивает ей ваше имя. Затем выведите на экран приветствие, используя эту переменную.

Задание 2.

Создайте консольный PHP-скрипт, который принимает два числовых аргумента при запуске и выводит результат их суммы.

Задание 3.

Напишите программу на PHP, создающую ассоциативный массив, представляющий информацию о книгах в библиотеке (название книги, автор, год издания). Выведите этот массив на экран. Затем рассчитайте и выведите среднюю длину названий книг в массиве.

Задание 4.

Напишите PHP-скрипт, используя цикл for, который выводит на экран таблицу умножения от 1 до 10 для выбранного числа. Позвольте пользователю ввести число и затем выведите соответствующую таблицу умножения.

Задание 5.

Напишите PHP-скрипт, который принимает от пользователя два числа и сравнивает их. Выведите сообщение о том, какое из чисел больше, меньше или равно другому числу. Используйте константы для определения типа сравнения (больше, меньше, равно).

Задание 6.

Напишите PHP-скрипт, который запрашивает у пользователя его возраст и выводит сообщение, указывающее на то, является ли он совершеннолетним или нет.

Тема 2.

Задание 1.

Напишите PHP-скрипт, который использует все четыре вида операторов (арифметический, строковый, присваивания, комбинированный) для работы с двумя переменными: \$a = 5 и \$b = 3. Выведите результаты операций на экран.

Задание 2.

Напишите PHP-скрипт, который принимает на вход строку, а затем использует функцию `strlen()` для определения длины строки. Выведите результат на экран.

Задание 3.

Напишите функцию PHP с именем `greet()`, которая принимает один аргумент - имя пользователя, и возвращает строку приветствия. Затем вызовите эту функцию, передав ваше имя, и выведите результат на экран.

Задание 4.

Напишите PHP-скрипт, который складывает две переменные: `$a = 5` (integer) и `$b = "3"` (string). Обратите внимание на типы переменных и на то, как ведут себя операции сложения в зависимости от типов. Выведите результат на экран.

Задание 5.

Напишите PHP-скрипт, который использует анонимную функцию для вычисления квадрата числа. Присвойте анонимную функцию переменной `$square` и выведите результат вычисления квадрата числа 5, используя эту переменную.

Тема 3.

Задание 1.

Напишите PHP-скрипт, который использует объект `DateTime` для вывода текущей даты и времени в формате "ГГГГ-ММ-ДД ЧЧ:ММ:СС".

Задание 2.

Напишите PHP-скрипт, который делит два числа, предварительно проверив, что делитель не равен нулю. Используйте блок `try-catch` для обработки исключения при попытке деления на ноль.

Задание 3.

Напишите PHP-скрипт, который отправляет HTTP-заголовок с информацией о типе содержимого (Content-Type) как "text/html". Затем выведите простую HTML-страницу с сообщением "Привет, мир!".

Задание 4.

Создайте PHP-скрипт, который устанавливает куку с именем `"username"` и значением `"JohnDoe"` на текущую дату плюс 7 дней. Затем выведите сообщение на экране, приветствуя пользователя по его имени из куки.

Задание 5.

Напишите PHP-скрипт, в котором происходит деление на ноль. Обработайте возможное исключение (ошибку) с использованием конструкции `try-catch` и выведите сообщение об ошибке на экран.

Тема 4.

Задание 1.

Создайте простое веб-приложение на PHP, где код разделен на два файла: один содержит логику обработки запроса, а второй - код для отображения HTML страницы. Оба файла должны быть включены в основной файл приложения.

Задание 2.

Напишите PHP-скрипт, который читает содержимое файла "data.txt", добавляет к текущей дате и времени, а затем записывает результат обратно в тот же файл. Выведите на экран сообщение о успешной операции.

Задание 3.

Создайте HTML-форму с двумя полями для ввода: имя и возраст. Напишите PHP-скрипт, который принимает данные из формы методом POST и выводит сообщение на экране с приветствием пользователя и указанием его возраста.

Задание 4.

Создайте PHP-скрипт, который формирует массив данных и затем сохраняет его в трех разных файлах: JSON, XML и YAML. Выведите на экран сообщение о успешном сохранении в каждом формате.

Задание 5.

Опишите основные различия между веб-серверами Apache и Nginx. Укажите, в каких сценариях использования каждый из них может быть предпочтительным.

Задание 6.

Напишите краткое описание виртуального хостинга и выделенного сервера. Укажите, для каких типов веб-приложений каждый из них может быть предпочтителен, и приведите примеры сценариев использования.

Критерии оценивания

Оценка	Критерий оценивания:
5 (отлично)	Присутствуют и полностью раскрыты все пункты в рамках задания.
4 (хорошо)	Присутствуют все пункты в рамках задания, но раскрыты не в полной мере.
3 (удовлетворительно)	Присутствует половина пунктов в рамках задания.
2 (неудовлетворительно)	Отсутствует более половины пунктов в рамках задания.

7. Оценочные материалы для промежуточной аттестации обучающихся по (курсу, учебному предмету)

7.1. Промежуточная аттестация

Форма проведения промежуточной аттестации – зачет.

Порядок проведения промежуточной аттестации: проводится дистанционно, предполагает выполнение обучающимся одного практического задания, направленного на оценку сформированности планируемых результатов обучения (умений и практического опыта). Обучающийся получает задания в личном кабинете, выполняет его самостоятельно и сдает результаты на проверку. По результатам проверки обучающемуся выставляется оценка.

Практическое задание, выносимое на промежуточную аттестацию:

Простой калькулятор: Создайте простой веб-калькулятор на PHP. Калькулятор должен поддерживать базовые операции (сложение, вычитание, умножение, деление) и отображать результат на экране.

Добавьте также возможность сохранения истории вычислений.

Форма регистрации пользователя: Разработайте форму регистрации пользователя. Пользователь должен вводить свое имя, электронную почту и пароль. При успешной регистрации данные пользователя сохраняются в базе данных. Реализуйте также простую страницу входа.

Счетчик посещений: Создайте счетчик посещений для веб-страницы на PHP. Счетчик должен увеличиваться при каждом посещении страницы и отображаться пользователю. Информация о посещениях может быть сохранена в текстовом файле или базе данных.

Промежуточная аттестация проводится дистанционно. Ко времени на начала зачета обучающийся получает документ с перечнем заданий. Срок выполнения - 1 час. До окончания обозначенного времени обучающемуся необходимо отправить документ с выполненным заданием на почту куратору, который осуществляет проверку.

Критерии оценивания промежуточной аттестации

Оценка	Критерий оценивания:
5 (отлично)	Присутствует решение всех заданий. Задания раскрыты в полном объеме.
4 (хорошо)	Присутствует решение 2 заданий. Задания раскрыты в полном объеме.
3 (удовлетворительно)	Присутствует решение 1 задания. Задание раскрыто в полном объеме.
2 (неудовлетворительно)	Отсутствует решение заданий.