

Балаковский инженерно-технологический институт – филиал федерального государственного автономного образовательного учреждения высшего образования
«Национальный исследовательский ядерный университет «МИФИ»

Факультет атомной энергетики и технологий
Кафедра «Информационные системы и технологии»

РАБОЧАЯ ПРОГРАММА

по дисциплине «Технологии создания web-ресурсов»

Направления подготовки/специальность

«09.03.02 Информационные системы и технологии»

Основная профессиональная образовательная программа

«Информационные системы и технологии»

Квалификация выпускника

Бакалавр

Форма обучения

Очная

Балаково

Цель освоения дисциплины

Целями освоения дисциплины «Технологии создания web-ресурсов» в области обучения, воспитания, развития, соотнесенные с общими целями ООП ВО и требованиями профессиональных стандартов («06.016. Руководитель проектов в области информационных технологий», «06.015. Специалист по информационным системам») являются: формирование профессиональной ответственности в области проектирования, разработки, эксплуатации и сопровождения информационных систем и технологий, используемых для информационного обеспечения деятельности различных предприятий и организаций, в том числе организаций атомной отрасли; формализация и алгоритмизация поставленных задач; написание программного кода с использованием языков программирования, определения и манипулирования данными; проверка и отладка программного кода; проверка работоспособности программного обеспечения; анализ требований к программному обеспечению.

Место дисциплины в структуре ООПВО

Необходимыми условиями для освоения дисциплины являются знания, умения и практические навыки по предшествующим дисциплинам и практикам в соответствии с требованиями освоения компетенций:

Интерактивное программирование web-приложений / Создание интерактивных приложений для интернет

Управление данными

Знания, умения и практические навыки, полученные при освоении дисциплины, необходимы при изучении следующих дисциплин и прохождения практик:

Методы и средства проектирования информационных систем и технологий

Современные технологии интернет-программирования

Информационный менеджмент

Электронный документооборот / Системы электронного документооборота на предприятии

Производственная практика (проектно-технологическая)

Производственная практика (преддипломная)

Государственная итоговая аттестация

При освоении данной дисциплины студент сможет частично продемонстрировать следующие обобщенные трудовые функции: Разработка архитектуры ИС, Проектирование и дизайн ИС, Сбор информации для инициации проекта в соответствии с полученным заданием

Компетенции обучающегося, формируемые в результате освоения дисциплины

В процессе освоения данной дисциплины у студента формируются следующие компетенции:

профессиональные

Задачи профессиональной деятельности (ЗПД)	Объект или область знания	Код и наименование компетенции	Индикаторы достижения компетенции
проектирование базовых и прикладных информационных технологий	информационные процессы, технологии, системы и сети, их инструментальное (программное, техническое, организационное) обеспечение, способы и методы проектирования, отладки, производства и эксплуатации информационных технологий и систем в различных областях и сферах деятельности.	ПК-6.1 Способен разрабатывать и адаптировать прикладное программное обеспечение	З-ПК-6.1 Знать: принципы и методы разработки, внедрения и адаптации прикладного программного обеспечения У-ПК-6.1 Уметь: программировать приложения и создавать программные прототипы решения прикладных задач В-ПК-6.1 Владеть: инструментальными средствами проектирования и разработки прикладного программного обеспечения
выбор исходных данных для проектирования	информационные процессы, технологии, системы и сети, их инструментальное (программное, техническое, организационное) обеспечение, способы и методы проектирования, отладки, производства и эксплуатации информационных технологий и систем в различных областях и сферах деятельности	ПК-12 Способен проводить выбор исходных данных для проектирования с учетом требований заказчика	З-ПК-12 Знать: требования к разработке технического задания, его структуру и принципы составления У-ПК-12 Уметь: анализировать исходную документацию заказчика В-ПК-12 Владеть: методикой составления технического задания

Задачи воспитания, реализуемые в рамках освоения дисциплины

Направление/цели	Создание условий, обеспечивающих	Использование воспитательного потенциала учебной дисциплины	Вовлечение в разноплановую внеучебную деятельность
Профессиональное воспитание	формирование профессиональной ответственности в области проектирования, разработки, эксплуатации и сопровождения информационных систем и технологий, используемых	1. Использование для формирования чувства личной ответственности в области проектирования, разработки, эксплуатации и сопровождения информационных систем и технологий, используемых для информационного обеспечения деятельности различных предприятий и организаций, в том числе организаций атомной отрасли воспитательного потенциала блока профессиональных дисциплин: Управление данными; Интерактивное программирование web-	1. Организация и проведение научно-практических конференций, форумов, круглых столов, вебинаров по вопросам профессиональной деятельности 2. Участие в студенческих олимпиадах и конкурсах научных проектов, творческих мероприятиях, конкурсах

	для информационного обеспечения деятельности различных предприятий и организаций, в том числе организаций атомной отрасли (В26).	приложений; Создание интерактивных приложений для интернет; Технологии создания web-ресурсов; Основы web-программирования; Информационная теория управления 2. Развитие навыков творческого мышления путем содействия и поддержки участия студентов в научно-практических мероприятиях внутривузовского регионального и/или всероссийского уровня по информатике и информационным технологиям.	профессионального мастерства, в том числе по стандартам WorldSkills. 3. Участие в подготовке публикаций в периодических научных изданиях; 4. Участие в деятельности студенческого научного общества
--	--	--	---

Структура и содержание учебной дисциплины

Дисциплина преподается студентам на очной форме обучения в 5-ом семестре.

Общая трудоемкость дисциплины составляет 5 зачетных единицы, 180 ак. часа.

Календарный план

№ Р а з д е л а	№ Т е м ы	Наименование раздела (темы) дисциплины	Виды учебной деятельности (в часах)					Аттестация раздела (форма)	Макси- маль- ный балл за раздел
			Всего	Лекции	Лабораторные	Практические	СРС		
1	Основы программирования на языке PHP							КИ	25
	1	Программное обеспечение для работы с PHP. Основы PHP, создание и выполнение простейших скриптов.	24	2	-	2	20		
	2	Обработка данных HTML-форм средствами PHP. Работа с массивами и функциями в PHP. Работа со строками и регулярными выражениями в PHP.	28	2	-	6	20		
	3	Работа с файловой системой средствами PHP. Загрузка файлов и динамическое формирование изображений средствами PHP.	30	2	-	8	20		
2	Взаимодействие PHP и MySQL							КИ	25
	4	Использование СУБД MySQL и PHP. Управление сеансами пользователя и безопасность скриптов.	32	4/2	-	8	20		
	5	Объектно-ориентированное программирование на PHP. Установки и модификация системы управления содержимым на PHP/MySQL.	26	6/4	-	8	12		

Вид промежуточной аттестации	40					Э	50
Итого	180/16	16/6	-	32/10	92		100

КИ- контроль по итогам

* - занятия в интерактивной форме

Сокращенное наименование форм текущего контроля и аттестации разделов:

Обозначение	Полное наименование
КИ	Контроль по итогам
Э	Экзамен

Содержание лекционного курса

Темы лекции. Вопросы, отрабатываемые на лекции	Всего часов	Учебно-методическое обеспечение
1	2	3
<p>Лекция 1. Программное обеспечение для работы с PHP. Основы PHP, создание и выполнение простейших скриптов.</p> <ol style="list-style-type: none"> 1. Основные типовые операции по обработке программы на PHP. 2. Программное обеспечение. Установка сервера. 3. Синтаксис языка PHP. 4. Побитовые операторы PHP. 5. Приоритеты операторов PHP по возрастанию. 6. Циклы и условия в языке PHP. 	2	1-3
<p>Лекция 2. Обработка данных HTML-форм средствами PHP. Работа с массивами и функциями в PHP. Работа со строками и регулярными выражениями в PHP.</p> <ol style="list-style-type: none"> 1. Способы обмена данными. 2. Управляющие последовательности. 3. GET и POST методы. 4. Массивы и функции в PHP. 5. Строковая функция и переменные 6. Регулярные выражения в PHP. 	2	1-3
<p>Лекция 3. Работа с файловой системой средствами PHP. Загрузка файлов и динамическое формирование изображений средствами PHP.</p> <ol style="list-style-type: none"> 1. Функции для работы с файлами. 2. Режимы открытия файлов. 3. Функции для работы с папками. 4. Глобальный массив \$_FILES. 5. Загрузка файлов на сервер. 6. Формирование изображений средствами PHP. 	4	1-3
<p>Лекция 4. Использование СУБД MySQL и PHP. Управление сеансами пользователя и безопасность скриптов..</p> <ol style="list-style-type: none"> 1. Взаимодействие MySQL и PHP. 2. Запросы к базе данных. 3. Сеансы пользователей. 4. Безопасность скриптов. 	4	1-3

Лекция 5. Объектно-ориентированное программирование на PHP. Установки и модификация системы управления содержимым на PHP/MySQL. 1. Инкапсуляция, наследование, полиморфизм. 2. Функции для работы с классами и объектами. 3. Системы управления содержимым.	2	1-3
---	---	-----

Перечень практических занятий

Тема практического занятия. Вопросы, отрабатываемые на практическом занятии	Всего часов	Учебно-методическое обеспечение
1	2	3
Математические операции с использованием PHP	2	1-3
Функции пользователя в PHP	2	1-3
Циклы, массивы, функции в PHP	2	1-3
Работа с файлами в PHP	2	1-3
Создание базы данных MySQL	2	1-3
Простой вывод данных в PHP	2	1-3
Ввод и правка данных с помощью формы в PHP	4	1-3
Работа с заметками в PHP	4	1-3
Страница статистики inform.php	4	1-3
Реализация поиска по сайту	4	1-3
Передача файлов на сервер	4	1-3

Задания для самостоятельной работы студентов

Вопросы для самостоятельного изучения (задания)	Всего часов	Учебно-методическое обеспечение
1	2	3
Основы синтаксиса PHP. Правила объявления переменных. Имена переменных. Типы данных переменных. Операции над переменными. Константы. Выражения и операции. Условный оператор. Циклы. Оператор выбора switch-case.	20	1-3
Передача параметров PHP-программам. Интерфейс CGI. Методы GET и POST. Специальные переменные окружения CGI. Заголовки протокола HTTP. Коды ответов протокола HTTP. Создание формы и ее обработка в сценарии. Проверка параметров формы. Защита от спама с помощью CAPTCHA. Механизм сессий. Массивы и Cookies. Массивы и списки. Массив и список. Цикл foreach. Функции list() и array(). Функции для сортировки массивов sort(), asort(), ksort(). Добавление и удаление элементов массива. Функции автоматического заполнения массива. Сравнение массивов. Операции над массивом	20	1-3
Функции для работы со строками, файлами и каталогами в PHP. Генератор случайных чисел. Дата и время. Основные строковые функции. Функции преобразования строк и кодировок. Функции шифрования. Чтение и запись файла. Создание временных файлов.	20	1-3

Специальные функции для работы с файлами. Функции для работы с каталогами.		
Функции для работы с MySQL. Преимущества SQL. Создание таблиц. Добавление записей в таблицу. Обновление записей. Выборка и удаление записей.	20	1-3
Подключение к серверу MySQL. Передача запросов серверу. Работа с базой данных. Получение информации о результате запроса.	12	1-3

Образовательные технологии

При реализации учебного материала курса используются различные образовательные технологии, способствующие созданию атмосферы свободной и творческой дискуссии как между преподавателем и студентами, так и в студенческой группе. Целью при этом является выработка у студентов навыков и компетенций, позволяющих самостоятельно вести исследовательскую и научно-педагогическую работу.

Практическая подготовка при реализации учебной дисциплины организуется путем проведения практических занятий, предусматривающих участие обучающихся в выполнении отдельных элементов работ, связанных с будущей профессиональной деятельностью.

Практическая подготовка также включает в себя занятия лекционного типа, которые предусматривают передачу учебной информации обучающимся, необходимой для последующего выполнения работ, связанных с будущей профессиональной деятельностью.

Аудиторные занятия проводятся в виде лекций, практических занятий с использованием ПК и компьютерного проектора. Самостоятельная работа студентов проводится под руководством преподавателей, с оказанием консультаций и помощи при подготовке к практическим занятиям.

Фонд оценочных средств

Фонд оценочных средств по дисциплине обеспечивает проверку освоения планируемых результатов обучения (компетенций и их индикаторов) посредством мероприятий текущего, рубежного и промежуточного контроля по дисциплине.

Связь между формируемыми компетенциями и формами контроля их освоения представлена в следующей таблице:

№ п/п	Наименование контролируемых разделов (темы)	Код и наименование индикатора достижения компетенций	Наименование оценочного средства
Входной контроль			
1	Входной контроль		Вопросы входного контроля (письменно)
Аттестация разделов, текущий контроль успеваемости			
2	Реализация сценариев JavaScript	3-ПК-6.1, У-ПК-6.1, В-ПК-6.1, 3-ПК-12, У-ПК-12, В-ПК-12	Контроль по итогам в форме тестирования (письменно)
3	Реализация сценариев jQuery	3-ПК-6.1, У-ПК-6.1, В-ПК-6.1, 3-ПК-12, У-ПК-12, В-ПК-12	Контроль по итогам в форме тестирования (письменно)

Промежуточная аттестация			
4	Экзамен	3-ПК-6.1, У-ПК-6.1, В-ПК-6.1, 3-ПК-12, У-ПК-12, В-ПК-12	Экзаменационный тест (письменно)

Входной контроль предназначен для выявления пробелов в знаниях студентов и готовности их к получению новых знаний. Оценочные средства для входного контроля представляют собой вопросы, которые задаются студентам в письменной форме.

Перечень вопросов входного контроля

1. Основные этапы в развитии HTML.
2. Теговая модель и базовая структура HTML-документов.
3. Основные требования к заглавной части HTML.
4. Дерево элементов HTML. Родственные связи между элементами. Принципы наследования.
5. Основные элементы HTML для форматирования текста.
6. Дополнительные (вспомогательные) элементы HTML для форматирования текста.
7. Основные элементы HTML для вставки изображений и создания гиперссылок.
8. Основные элементы HTML для работы со списками.
9. Основные элементы HTML для работы с таблицами.
10. Блочные и строчные элементы HTML. Определения и основные особенности.
11. Универсальные элементы HTML. Назначение и принципы использования.
12. Атрибуты элементов HTML. Принципы наследования. Универсальные атрибуты.
13. Адресация в HTML. Варианты и примеры абсолютной и относительной адресации.
14. Каскадные таблицы стилей CSS. Предпосылки появления и история развития.
15. Основы синтаксиса CSS. Назначение и особенности использования.
16. Методы определения CSS. Встраивание, вложение и связывание.
17. Методы определения CSS. Принципы каскадирования и наследования стилей.
18. Единицы измерения в CSS. Перечень абсолютных и относительных единиц измерения.
19. Способы задания цвета в CSS. Цветовые таблицы (палитры). Принципы подбора цвета.
20. Шрифтовое оформление в CSS. Гарнитур. Семейство и тип шрифта. Понятие о «безопасных» шрифтах.
21. Шрифтовое оформление в CSS. Настройка типа, размера, начертания и модификации шрифта.
22. Собираемое шрифтовое оформление.
23. Оформление текста в CSS. Выравнивание, отступы и промежутки, трансформация, интервалы
24. и декорация.
25. Базовый синтаксис CSS. Селекторы тегов.
26. Базовый синтаксис CSS. Классы и идентификаторы.
27. Базовый синтаксис CSS. Контекстные, соседние и дочерние селекторы.
28. Базовый синтаксис CSS. Селекторы атрибутов.
29. Блочная модель CSS. Рамки, поля и отступы.
30. Блочная модель CSS. Позиционирование элементов.

31. Блочная модель CSS. Многослойность, выравнивание и обтекание.

Текущий контроль – это непрерывно осуществляемый мониторинг уровня усвоения знаний и формирования умений и навыков в течение семестра. Текущий контроль знаний, умений и навыков студентов осуществляется в ходе учебных (аудиторных) занятий, проводимых по расписанию. Формами текущего контроля выступают используются colloquium, выполнение практических контрольных заданий.

Перечень вопросов colloquium

1. Что такое short tags и как его использовать в PHP?
2. Как изолировать PHP код от HTML? Для чего это нужно?
3. Как использовать комментарии в коде PHP?
4. Что такое \$this и self в PHP?
5. В чем разница между одинарными и двойными кавычками в PHP?
6. Сколько в PHP типов данных? Какие это типы данных?
7. Какой тип типизации данных в PHP?
8. Что такое переменная и как ее использовать?
9. Что такое область видимости переменной?
10. Что такое переменные извне?
11. В чем разница между GET и POST?
12. Что такое константы? Как их использовать?
13. Что такое волшебные константы?
14. Что такое оператор?
15. Какие существуют операторы?
16. Что такое арифметические операторы?
17. Что такое логические операторы?
18. Что такое оператор присваивания?
19. Что такое побитовые операторы?
20. Что такое операторы сравнения?
21. Что такое оператор управления ошибками?
22. Что такое операторы исполнения?
23. Что такое операторы инкремента и декремента?
24. Что работает быстрее: преинкремент или постинкремент?
25. Что такое строковые операторы?
26. Что такое строковые операторы?
27. Обязательно ли использовать закрывающий тег ?>в конце файла?
28. В чем разница между require(), require_once(), include(), include_once()?
29. В чем отличие цикла while от do while?
30. Что такое goto и как он работает?
31. В чем различия между операторами break и continue?
32. Как прервать выполнение for, foreach, while, do-while или switch?
33. Как пропустить часть итерации цикла внутри for, foreach, while, do-while или switch?
34. Что такое declare?
35. Почему в declare нельзя использовать переменные и константы?
36. Что такое альтернативный синтаксис?
37. Есть ли разница между одинарными и двойными кавычками?
38. В чем различия между echo и print?
39. Дана строка "Hello world!". Как перевернуть строку?

Перечень практических заданий

1. Создайте переменную \$var и присвойте ей значение 'hello'. Обращаясь к отдельным символам этой строки выведите на экран символ 'h', символ 'e', символ 'o'.
2. Напишите скрипт, который считает количество секунд в часе.
3. Переделайте приведенный код так, чтобы в нем использовались операции +=, -=, *=, /=, ++, --.

Количество строк кода при этом не должно измениться. Код для переделки:

```
<?php
    $var = 1;
    $var = $var + 12;
    $var = $var - 14;
    $var = $var * 5;
    $var = $var / 7;
    $var = $var + 1;
    $var = $var - 1;
    echo $var;
?>
```

4. Создайте переменную \$a и присвойте ей значение 3. Выведите значение этой переменной на экран.
5. Создайте переменные \$a=10 и \$b=2. Выведите на экран их сумму, разность, произведение и частное (результат деления).
6. Создайте переменные \$c=15 и \$d=2. Просуммируйте их, а результат присвойте переменной \$result. Выведите на экран значение переменной \$result.
7. Создайте переменные \$a=10, \$b=2 и \$c=5. Выведите на экран их сумму.
8. Создайте переменные \$a=17 и \$b=10. Отнимите от \$a переменную \$b и результат присвойте переменной \$c. Затем создайте переменную \$d, присвойте ей значение 7. Сложите переменные \$c и \$d, а результат запишите в переменную \$result. Выведите на экран значение переменной \$result.
9. Создайте переменную \$text и присвойте ей значение 'Привет, Мир!'. Выведите значение этой переменной на экран.
10. Создайте переменные \$text1='Привет, ' и \$text2='Мир!'. С помощью этих переменных и операции сложения строк выведите на экран фразу 'Привет, Мир!'.
11. Создайте переменную \$name и присвойте ей ваше имя. Выведите на экран фразу 'Привет, %Имя%!'. Вместо %Имя% должно стоять ваше имя.
12. Создайте переменную \$age и присвойте ей ваш возраст. Выведите на экран 'Мне %Возраст% лет!'.
13. Создайте переменную \$text и присвойте ей значение 'abcde'. Обращаясь к отдельным символам этой строки выведите на экран символ 'a', символ 'c', символ 'e'.
14. Дана произвольная строка, например, 'abcde'. Поменяйте первую букву (то есть букву 'a') этой строки на '!'.
15. Создайте переменную \$num и присвойте ей значение '12345'. Найдите сумму цифр этого числа.

Аттестация раздела по дисциплине проводится в форме тестирования. Тест содержит от 10 вопросов. На выполнение задания отводится от 20 минут. Тест – это форма контроля, направленная на проверку уровня освоения контролируемого теоретического и практического материала по дидактическим единицам дисциплины (терминологический аппарат, основные методы).

Примерный перечень тестовых заданий:

Тестовые задания 1. (Т1)

1. Какими разделителями окружены скрипты php скрипта?

- a) `<? php ... ?>`
- b) `<php>... </php>`
- c) `</> php </>`

2. Используя синтаксис языка php, “ напишитеHello, World!”:

- a) `printf “Hello, World!”`
- b) `<p> Hello, World! </p>`
- c) `echo “Hello, World!”`

3. С какого символа начинаются все переменные в php?

- a) `<>`
- b) `$`
- c) `!`

4. К какому языку программирования синтаксически наиболее близок php?

- a) C и Perl
- b) .Net
- c) VBScript

5. Перечислите основные типы данных в php:

- a) Boolean, float, logic, text, subject, NULL
- b) String, boolean, text, massive, logic, NULL
- c) Boolean, integer, float, string, array, object, resource, NULL

6. В php используется ... типизация данных.

- a) Динамическая
- b) Синтаксическая
- c) Коммутационная

7. Как правильно включить файл “time.inc”?

- a) `<? php include "time.inc"; ?>`
- b) `</> php include ‘time.inc’ <>`
- c) `<!-- include file="time.inc"—>`

8. Укажите правильный способ создания функции в php:

- a) `create newFunction()`
- b) `new_function newFunction()`
- c) `function newFunction()`

9. Укажите неверно заданное имя:

- a) `$my-Var`
- b) `$myVar`
- c) `$my_Var`

10. Укажите верно заданный массив с использованием синтаксиса языка php:

- a) `$months = array["September", "October", "November"]`
- b) `$months = "September", "October", "November"`
- c) `$months = array("September", "October", «November")`

11. В php файлы cookie задаются следующим образом:

- a) `setcookie()`
- b) `makecookie()`
- c) `createcookie()`

12. Для написания комментариев в пределах работающего кода на php используется запись

вида:

- a) `/*...*/`

b) <!--...-->

c) <comment>...</comment>

13. Как правильно записать добавление 1 (единицы) к переменной \$count?

a) \$count+1

b) \$count++

c) \$count+=1

14. В php суперглобальная переменная, содержащая информацию о местоположении скриптов, путях, заголовках – это:

a) \$_SERVER

b) \$_GLOBALS

c) \$_GET

15. Укажите оператор, написанный на языке php, который делал бы файл "filename.txt" доступным для чтения.

a) fopen ("filename.txt", "r")

b) open ("filename.txt", "read")

c) fopen ("filename.txt", "r+")

Тестовые задания 2 (Т2)

1. Каждый php-оператор должен заканчиваться:

a) Двоеточием (:)

b) Точкой с запятой (;)

c) Знаки препинания в php не используются

2. Какой способ использования тегов рекомендован для языка программирования php?

a) <?

b) <php

c) <? php

3. Укажите код, результатом выполнения которого является вывод на экран имени ("переменная name").

a) ?php

```
$name++'Mike';
```

```
$age++'37';
```

```
echo $age;
```

```
?
```

b) //?php

```
?name = Mike;
```

```
?age = 37;
```

```
echo $name;
```

```
?//
```

c) <? php

```
$name = 'Mike';
```

```
$age = 37;
```

```
echo $name;
```

```
?>
```

4. Вызов функции из самой себя – это:

a) Рекурсия

b) Регрессия

c) Массив

5. Выберите верное утверждение, характеризующее множественное наследование в php.

а) Множественное наследование задается через подключение родственных классов. Для этого используется функция `_multi()`

б) В php не существует понятия «множественное наследование», т.е. у класса может быть лишь один родитель. Эмуляция множественного наследования задается функцией `_call ()` или трейтами

с) Организовать наследование позволяет метод `_constructor()`, создающий родственные экземпляры базового класса

6. Какая ошибка содержится в коде на картинке?

```
<?php
define("MSG", " Hi SoloLearners!", true);
echo msg;

// Outputs "Hi SoloLearners!"
?>
```

а) Отсутствует `$` перед `define`, логическое утверждение `true` не заключено в кавычки

б) Код написан верно

с) `MSG` и `msg` – это разные значения. Чтобы интерпретировать их как значение для одной и той же переменной, следует использовать одинаковый регистр

7. Имеются следующие варианты написания кода: (1) `<?php echo 'Выполнить документ';?>` и (2) `<?= 'Выполнить документ' ?>`. Какое из приведенных ниже утверждений верное?

а) Записи (1) и (2) представляют собой одинаковый код. Разница в том, что (1) – это стандартный синтаксис, (2) – краткий. В php приемлемы оба варианта

б) Запись (1) представляет собой код, который выведет на экран фразу ‘Выполнить документ’, (2) – это комментарий к коду, который никак не отобразится у конечных пользователей

с) Запись (1) – это программа, написанная на языке php, (2) – на языке Java. Результирующий вывод при их исполнении одинаковый

8. Какой тип данных следует использовать для выражения “-42”?

а) `boolean`

б) `string`

с) `int`

9. Для чего используется ключевое слово `global`?

а) Для получения доступа к глобальной переменной, которая находится внутри функции

б) Для подключения библиотек из глобального каталога

с) Для указания того, что переменная будет многократно использоваться

10. Расставьте слова (1) `echo` (2) “Robert” (3) `$name=` (4) `$name` в таком порядке, чтобы на экран выводилось имя, переданное переменной `name`.

а) 3, 2, 1, 4

б) 1, 3, 2, 4

с) 3, 1, 4, 2

11. Какой результат выведет следующий код: `$x = 'y'; $y = 'x'; echo $$x; ?`

а) `x+y`

б) `x*y`

с) `x`

12. Логический оператор ИЛИ на языке php выражается символом:

а) `|`

б) `||`

с) `&&`

13. В выражении “\$a=2; \$b=\$a++;” чему равно значение \$b?

- a) Невозможно определить по заданным условиям
- b) 3
- c) 2

14. Какие нововведения появились в php 5.6?

- a) Выражения в константах, нефиксированные параметры в функциях
- b) Короткая запись массивов, константы классов
- c) API хеширование паролей, трейты, замыкания

15. В чем разница между print и echo?

- a) print – это конструкция, echo – это функция
- b) В объеме задействованной оперативной памяти при выполнении кода. Обращаясь к функции print, интерпретатор задействует больше оперативной памяти, чем при обращении к функции echo
- c) print – это функция, echo - это конструкция

Критерии оценки тестовых заданий:

- 1. Полнота знаний теоретического контролируемого материала.
- 2. Количество правильных ответов.

Промежуточная аттестация осуществляется в форме экзамена. Экзамен проводится в форме тестирования. Тест содержит от 20 вопросов. Тест – это форма контроля, направленная на проверку уровня освоения контролируемого теоретического и практического материала по дидактическим единицам дисциплины (терминологический аппарат, основные методы).

Примерный перечень экзаменационных тестовых заданий:

Экзаменационное тестовое задание (Т1).

- 1. Где указан верный оператор вывода?
 - a) echo "";
 - b) echo "";
 - c) message "";
 - d) show "";
- 2. Какой символ указывает переменную?
 - a) &
 - b) %
 - c) \$
 - d) !
 - e) #
- 3. Чем константы отличаются от переменных?
 - a) Константы используются только в функциях
 - b) Переменные используются только в функциях
 - c) Константы нельзя переопределять
 - d) Отличий нет
- 4. Что такое оператор эквивалентности?
 - a) Он сравнивает переменные на соответствие, но не на их тип
 - b) Он сравнивает переменные на соответствие и на их тип
 - c) Такого оператора не существует
- 5. Какие типы переменных существуют в PHP?

- a) int, float, string, char
- b) int, float, string, bool
- c) int, float, string
- d) int, float, string, char, bool

6. Какие арифметические операции существуют в PHP?

- a) +, -, *, /, ^
- b) +, -, *, /, \%
- c) +, -, *, /
- d) +, -, *, /, %

7. Что делает код ниже? `UPDATE `db` SET `price` = '0';`?

- a) Устанавливает новое значение в таблице db для поля price
- b) Добавляет новое поле price со значением 0 в таблицу db
- c) Устанавливает значение 0 в таблице price
- d) Устанавливает новое значение для всей базы данных

8. Где верно создан комментарий в PHP?

- a) `/* /*`
- b) `/`
- c) `**`
- d) `#`

9. Для чего используется язык PHP?

- a) Для работы с базами данных
- b) Для работы на сервере
- c) Для работы со скриптами на стороне клиента
- d) Для работы на сервере и базами данных

10. Что такое MySQL?

- a) Это система для управления базой данных
- b) Это база данных, а также язык запросов к ней
- c) Это технология для работы PHP и базы данных
- d) Это язык программирования для работы с БД

11. Что делает код ниже? `SELECT `id` FROM `table``?

- a) Выбирает id из всех таблиц, которые начинаются на table
- b) Выбирает всё из таблицы table
- c) Выбирает id из таблицы table
- d) Выбирает id из всей базы данных
- e) Выбирает table из таблицы id

12. Что выведет код ниже?

```
$x = 1;
```

```
$y = 5;
```

```
$z = 2.1;
```

```
echo $z != $y && $x < $y;
```

- a) "false"
- b) 1
- c) "true"
- d) 0

e) Ошибку

13. Что значат цифры в описании типа данных колонки `DECIMAL(5,2)`

- a) Всего 5 числовых знаков, из них 2 на дробную часть
- b) Всего 7 числовых знаков, из них 2 на дробную часть

- c) Всего 7 числовых знаков, из них 5 на дробную часть
- d) Всего 7 числовых знаков, из них 0 на дробную часть

14. Какой SQL-оператор создаёт базу данных в MySQL?

- a) INSERT DATABASE dbname;
- b) USE dbname
- c) CREATE DATABASE dbname;
- d) ADD DATABASE dbname;

15. Выберите существующие виды создания таблиц

- a) CREATE TABLE tbl_name LIKE ...
- b) CREATE TABLE tbl_name AS SELECT ...
- c) CREATE TABLE tbl_name (create_definition,...)
- d) правильного ответа нет

16. Какая строковая функция отвечает за склейку строк?

- a) LOWER()
- b) CHAR()
- c) CONCAT()
- d) REPLACE()

17. Какие функции позволяют получить одновременно текущие дату/время?

- a) CURRENT_TIMESTAMP()
- b) NOW()
- c) CURTIME()
- d) TIME()

18. Какие математические функции в MySQL округляют числовые значения?

- a) FLOOR()
- b) CEIL()
- c) ROUND()
- d) POW()

19. Выберите варианты старта транзакции в MySQL

- a) START TRANSACTION
- b) BEGIN
- c) SETUP TRANSACTION
- d) START

20. Для чего не может использоваться PHP?

- a) для создания операционных систем
- b) для создания скриптов, выполняющихся в командной строке
- c) для создания клиентских GUI-приложений

Экзаменационное тестовое задание (Т2).

1. Как в PHP создать новую переменную с именем `animal` и строковым значением `'cat'`?

- a) `animal = 'cat';`
- b) `$animal = cat`
- c) `string animal = "cat";`
- d) `var animal = 'cat';`
- e) `$animal = 'cat'.`

2. Какой из вариантов ответа согласуется с PSR — официальными стандартами PHP?

- a) Открывающую фигурную скобку для классов и методов необходимо располагать на отдельной строке, недопустимо использовать короткий открывающий тег, классам необходимо давать имена в стиле StudlyCaps

b) Открывающую фигурную скобку для классов и методов необходимо располагать на той же строке, недопустимо использовать короткий открывающий тег, классам необходимо давать имена в стиле camelCase

c) Открывающую фигурную скобку для классов и методов необходимо располагать на той же строке, недопустимо использовать короткий открывающий тег, классам необходимо давать имена в стиле StudlyCaps

d) Ни один из этих ответов не является полностью правильным

3. Можно ли передавать в функцию значения, указывая их тип? `foo(String, Bool, Array, Int);`

a) Да, для всех типов, кроме String

b) Да, только для массивов

c) Да, можно для всех типов

d) Да, можно для всех типов

4. Каким будет результат выполнения следующего кода в PHP 5?

```
echo 0.1 + 0.2;
```

a) 0.30000000000000004

b) 0.3000004

c) 0.3

d) 3

5. Что делает этот код?

```
namespace HelloWorld;
```

a) Создает одно пространство имен: HelloWorld

b) Создает два пространства имен: HelloWorld и World

c) Создает два пространства имен: Hello и World

d) Создает два пространства имен: Hello и HelloWorld

6. Какая из приведенных ниже функций удаляет первый элемент из массива и возвращает его значение?

a) `array_compare()`

b) `array_shift()`

c) `array_flip()`

d) `array_pop()`

e) `array_remove_first()`

7. Чем отличается скрипт на языке PHP от скрипта на языке JavaScript?

a) ничем не отличаются

b) способом встраивания в HTML-код

c) скрипт на языке JavaScript обрабатывается клиентом, а PHP-скрипт – сервером

8. Владимиру потребовался оператор безусловного перехода в PHP. Как пишется такой оператор в PHP (версия PHP 5.2)?

a) `jmp`

b) Такого оператора в PHP версии ниже 5.3 не существует.

c) `goto`

d) `header()`

9. Какая разница между равенством и эквивалентностью?

a) Знак эквивалентности работает лишь для строк, а знак равенства применим к любым типам.

b) Знак равенства проверяет лишь значения операндов, а знак эквивалентности значения и их типы.

c) Никакой разницы нет.

d) Знак эквивалентности проверяет лишь значения операндов, а знак равенства значения и их типы.

10. Какая функция в PHP позволяет найти вхождение подстроки?

- a) strpos()
- b) strtok()
- c) str_repeat()
- d) substr()

11. Что будет, если запустить такой скрипт:

```
<?php
    if (null == 0) echo "true";
    else echo "false";
?>
```

- a) true
- b) Ошибка, поскольку null нельзя сравнивать с 0.
- c) Ошибка, поскольку null в PHP нет.
- d) false

12. Какого типа данных в PHP нет?

- a) Указатель.
- b) Resource.
- c) null.
- d) Объект.

13.

```
<?php
    for ($i = 0; $i < 5; $i++) {
        if ($i % 2 == 0) continue;
        echo $i;
    }
?>
```

Что будет после запуска кода?

- a) 24
- b) 13
- c) 024
- d) 013

14. Что выведет данный код:

```
<?php
    echo "1" + "2";
?>
```

- a) 3
- b) 32
- c) 12
- d) Ошибку синтаксиса.

15.

```
<?php
    function myfunc(&$a) {
        $a++;
    }
    $b = 5;
    myfunc($b);
    echo $b;
?>
```

Что выведет данный код?

- a) 6
- b) 56
- c) Ошибку из-за лишнего символа "&" перед именем переменной в функции.
- d) 5

16. Есть строка: "my string". Есть код:

```
<?php
if (strpos("mystring", "m") == false) echo "Символа m в строке mystring нет";
?>
```

a) Функция strpos() вернула 0, который в PHP равен false. Чтобы не было ошибки надо вместо знака равенства (==) использовать знак эквивалентности (===).

b) Здесь нет ошибок, так как strpos("mystring", "m") не равен false, и строки "Символа m в строке mystring нет" не появляется.

c) Нет фигурных скобок у блока операторов при срабатывании условия.

d) Функция strpos() неправильно использована. Сначала должна идти искомая строка, а уже потом та строка, в которой происходит поиск.

17. Какой из сценариев следит за нормальной работой сервера и в случае сбоя перезагружает его:

- a) mysqldebug
- b) mysql_safe
- c) safemysqld

18. Какой из операторов позволяет создавать пользователей MySQL и настроить их привилегии:

- a) SELECT USER
- b) NEW
- c) GRANT

19. Размер таблицы ограничен её:

- a) типом
- b) рамками
- c) границами

20. MySQL не позволяет корректно применять такие выражения:

- a) постоянные
- b) регулярные
- c) переменные

Шкалы оценки образовательных достижений

Баллы (итоговой рейтинговой оценки)	Оценка (балл за ответ на экзамене)	Требования к знаниям
--	---	----------------------

90-100	Отлично	Ответы на поставленные вопросы полные, четкие и развернутые. Решения задач логичны, доказательны и демонстрируют аналитические и творческие способности студента.
70-89	Хорошо	Даются полные ответы на поставленные вопросы. Показано умение выделять причинно-следственные связи. При решении задач допущены незначительные ошибки, исправленные с помощью «наводящих» вопросов преподавателя.
60-69	Удовлетворительно	Ответы на вопросы и решения поставленных задач недостаточно полные. Логика и последовательность в решении задач имеют нарушения. В ответах отсутствуют выводы.

Учебно-методическое и информационное обеспечение учебной дисциплины

Основная литература:

1. Диков, А. В. Клиентские технологии веб-программирования: JavaScript и DOM : учебное пособие / А. В. Диков. — Санкт-Петербург : Лань, 2020. — 124с. <https://e.lanbook.com/reader/book/126934/#1>

Дополнительная литература:

2. Лабораторный практикум по семантическим технологиям и веб-сервисам : учебное пособие / В. В. Климов, А. А. Чернышов, А. И. Баландина, А. Д. Косткина. — Москва : НИЯУ МИФИ, 2016. — 68 с. <https://e.lanbook.com/reader/book/119502/#1>

3. Семантический веб / Г. Антониоу, П. Грос, в. Ф. Хармелен, Р. Хоекстра ; перевод с английского Т. Шульга. — Москва : ДМК Пресс, 2016. — 240 с. <https://e.lanbook.com/reader/book/69963/#1>

Программное обеспечение и Интернет-ресурсы:

1. Электронно-библиотечная система «Лань»
2. Электронно-библиотечная система «Юрайт»
3. Российская национальная библиотека - <http://www.nlr.ru/>
4. Поисковая система - <http://www.rambler.ru>.
5. Поисковая система - <http://www.yandex.ru>.
6. Гарант - <http://base.garant.ru/>.
7. Интернет-Университет Информационных Технологий - <http://www.intuit.ru>

Для проведения практических занятий и выполнения самостоятельной работы используются учебные компьютерные классы с выходом в Интернет и лицензионным программным обеспечением.

Материально-техническое обеспечение учебной дисциплины

Лекционные занятия проводятся в аудиториях: учебная мебель, учебная доска, комплект мультимедийного оборудования

Практические занятия проводятся в компьютерных классах: учебная мебель, учебная доска, комплект мультимедийного оборудования, персональные компьютеры.

Для самостоятельной работы обучающихся имеется: читальный зал с выходом в сеть Интернет: Учебная мебель, комплект мультимедийного оборудования, персональные компьютеры, МФУ.

Учебно-методические рекомендации для студентов

1. Указания для прослушивания лекций

Перед началом занятий внимательно ознакомиться с учебным планом проведения лекций и

списком рекомендованной литературы.

Перед посещением очередной лекции освежить в памяти основные концепции пройденного ранее материала. Подготовить при необходимости вопросы преподавателю. Не надо опасаться, что вопросы могут быть простыми.

На лекции основное внимание следует уделять не формулам и математическим выкладкам, а содержанию изучаемых вопросов, определениям и постановкам задач.

В процессе изучения лекционного курса необходимо по возможности часто возвращаться к основным понятиям и методам решения задач (здесь возможен выборочный контроль знаний студентов).

Желательно использовать конспекты лекций, в которых используется принятая преподавателем система обозначений.

Для более подробного изучения курса следует работать с рекомендованными литературными источниками и вновь появляющимися источниками.

2. Указания для участия в практических занятиях

Перед посещением уяснить тему практического занятия и самостоятельно изучить теоретические вопросы.

В конце практического занятия при необходимости выяснить у преподавателя неясные вопросы.

Основные результаты выполнения работы необходимо распечатать.

3. Указания для выполнения самостоятельной работы

Получить у преподавателя задание и список рекомендованной литературы. Изучение теоретических вопросов следует проводить по возможности самостоятельно, но при затруднениях обращаться к преподавателю.

Подготовить письменный отчет о проделанной работе.

При выполнении заданий по усмотрению преподавателя работа может быть оценена без письменного отчета на основе ответов на контрольные вопросы, при условии активной самостоятельной работы.

Подготовить ответы на контрольные вопросы.

Методические рекомендации для преподавателей

1. Указания для проведения лекций

На первой вводной лекции сделать общий обзор содержания курса и отметить новые методы и подходы к решению задач, рассматриваемых в курсе, довести до студентов требования кафедры, ответить на вопросы.

При подготовке к лекционным занятиям необходимо продумать план его проведения, содержание вступительной, основной и заключительной части лекции, ознакомиться с новинками учебной и методической литературы, публикациями периодической печати по теме лекционного занятия. Уточнить план проведения семинарского занятия по теме лекции. Перед изложением текущего лекционного материала напомнить об основных итогах, достигнутых на предыдущих лекциях. С этой целью задать несколько вопросов аудитории и осуществить выборочный контроль знания студентов.

В ходе лекционного занятия преподаватель должен назвать тему, учебные вопросы, ознакомить студентов с перечнем основной и дополнительной литературы по теме занятия. Раскрывая содержание учебных вопросов, акцентировать внимание студентов на основных категориях, явлениях и процессах, особенностях их протекания. Раскрывать сущность и содержание различных точек зрения и научных подходов к объяснению тех или иных явлений и процессов.

Следует аргументированно обосновать собственную позицию по спорным теоретическим вопросам. Приводить примеры. Задавать по ходу изложения лекционного материала риторические вопросы и самому давать на них ответ. Это способствует активизации мыслительной деятельности студентов, повышению их внимания и интереса к материалу лекции, ее содержанию. Преподаватель должен руководить работой студентов по конспектированию лекционного материала, подчеркивать

необходимость отражения в конспектах основных положений изучаемой темы, особо выделяя, категориальный аппарат. В заключительной части лекции необходимо сформулировать общие выводы по теме, раскрывающие содержание всех вопросов, поставленных в лекции. Объявить план очередного практического занятия, дать краткие рекомендации по подготовке студентов к семинару. Определить место и время консультации студентам, пожелавшим выступить на занятии с докладами.

На последней лекции уделить время для обзора наиболее важных положений, рассмотренных в курсе.

2. Указания для проведения практических занятий

Четко обозначить тему занятия.

Обсудить основные понятия, связанные с темой.

В процессе решения задач вести дискуссию со студентами о правильности применения теоретических знаний.

Отмечать студентов, наиболее активно участвующих в решении задач и дискуссиях.

В конце занятия задать аудитории несколько контрольных вопросов.

3. Указания по контролю самостоятельной работы студентов

По усмотрению преподавателя задание на самостоятельную работу может быть индивидуальным или фронтальным.

При использовании индивидуальных заданий требовать от студента письменный отчет о проделанной работе.

При применении фронтальных заданий вести коллективные обсуждения со студентами основных теоретических положений.

С целью контроля качества выполнения самостоятельной работы требовать индивидуальные отчеты (допустимо вместо письменного отчета применять индивидуальные контрольные вопросы).

Программа составлена в соответствии с требованиями ОС НИЯУ МИФИ и учебным планом основной образовательной программы

Рабочую программу составил ст.пр. кафедры А.Г. Мотков

Рецензент: доцент И.А. Штырова

Программа одобрена на заседании УМКН «Информационные системы и технологии» от 4.07.2023 года, протокол №5

Председатель учебно-методической комиссии О.В. Виштак