Министерство науки и высшего образования Российской Федерации

Муромский институт (филиал)

федерального государственного бюджетного образовательного учреждения высшего образования

«Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»

(МИ ВлГУ)

Кафедра *ИС*

«УТВЕРЖДАЮ»
Заместитель директора по УР
_____Д.Е. Андрианов
19.05.2026

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Интернет-технологии

Направление подготовки *09.03.02 Информационные системы и технологии*

Профиль подготовки *Информационные системы и технологии*

Семестр	Трудоемкость, час./зач. ед.	Лекции, час.	Практические занятия, час.	Лабораторные работы, час.	Консультация, час.	Контроль, час.	Всего (контактная работа), час.	СРС, час.	Форма промежуточного контроля (экз., зач., зач. с оц.)
7	**180 / 5**	**30**		**24**	**5**	**0,35**	**59,35**	**94**	**Экз.(26,65)**
Итого	**180 / 5**	**30**		**24**	**5**	**0,35**	**59,35**	**94**	**26,65**

Муром, 2026 г.

1. Цель освоения дисциплины

Целями освоения дисциплины Интернет-технологии являются: знакомство студентов с основами программирования для Internet и знакомство с базовыми технологиями работы в глобальных сетях.

Основные задачи изучения дисциплины состоят в том, чтобы ознакомить студентов с основными понятиями программирования с использованием современных информационных технологий; а также, приобретение практических навыков работы в Internet. Формирование у студентов способностей к разработке объектов профессиональной деятельности в различных сферах с использованием Web-технологий и современных инструментальных средств построения информационных систем управления контентом. Привитие навыков работы в коллективе разработчиков будущим специалистам.

2. Место дисциплины в структуре ОПОП ВО

В рамках данной дисциплины осуществляется закрепление приобретенных компетенций и подводится итог образовательной деятельности по ним. Данная дисциплина базируется на предметах: технологии программирования, информатика, инфокоммуникационные системы и сети, информационные технологии и управление данными. Для полноценного освоения данной дисциплины студент должен знать основные средства разработки информационных технологий, уметь проектировать базовые и прикладные информационные технологии, владеть способами инсталляции, отладки программных и настройки технических средств ввода информационных систем в эксплуатацию.

3. Планируемые результаты обучения по дисциплине

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине, в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции	Результаты обучения по дисциплине	
ОПК-3 Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности;	ОПК-3.2 Применяет знания приемов безопасной работы в сети Интернет при поиске информации, связанной с профессиональной деятельностью	Знать современные информационные технологии и программные средства, в том числе отечественного производства, при решении задач профессиональной деятельности (ОПК-3.2)	Перечень вопросов,Отчет,Тест,Отчеты,Тест, Отчеты
ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе	ОПК-2.2 Применяет современные информационные технологии и программные средства, в том числе отечественного производства, при	Уметь выбирать современные информационные технологии и программные средства, в том числе отечественного производства, при	Перечень вопросов,Отчет,Тест,Отчеты,Тест, Отчеты

отечественного производства, и использовать их при решении задач профессиональной деятельности;	решении задач профессиональной деятельности	решении задач профессиональной деятельности (ОПК-2.2) Иметь навыки применения современных информационных технологий и программных средств, в том числе отечественного производства, при решении задач профессиональной деятельности (ОПК-2.2)	

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 5 зачетных единиц, 180 часов.

4.1. Форма обучения: очная

Уровень базового образования: среднее общее.
Срок обучения 4г.

4.1.1. Структура дисциплины

№ п\п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником							Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация	Контроль		
1	Введение в Интернет-технологии	7	2							10	Устный опрос
2	Особенности разработки web-проектов.	7	2							10	Устный опрос
3	Создание web-страниц с использованием HTML	7	4							8	Устный опрос
4	JavaScript и объектная модель документа	7	2							8	Устный опрос
5	Общее понятие динамического HTML (DHTML)	7	2		4					5	Отчет по лабораторной работе
6	Введение в расширяемый язык разметки (XML)	7	2							8	Промежуточная контрольная работа
7	Структура XML-документа	7	4							5	Устный опрос
8	Стилевые таблицы XSL	7	4							8	Устный опрос
9	Создание динамических web-ресурсов. CGI	7	4		8					16	Отчет по лабораторным работам
10	Реализация CGI	7	4		12					16	Итоговая контрольная работа. Отчет по лабораторным работам

Всего за семестр	180	30		24			5	0,35	94	Экз.(26,65)
Итого	180	30		24			5	0,35	94	26,65

4.1.2. Содержание дисциплины

4.1.2.1. Перечень лекций

Семестр 7

Раздел 1. Введение в Интернет-технологии

Лекция 1.

Введение в Интернет-технологии (2 часа).

Раздел 2. Особенности разработки web-проектов.

Лекция 2.

Особенности разработки web-проектов (2 часа).

Раздел 3. Создание web-страниц с использованием HTML

Лекция 3.

Создание web-страниц с использованием HTML (2 часа).

Лекция 4.

Создание HTML документов: текст, форма, фрейм, изображение, стили (2 часа).

Раздел 4. JavaScript и объектная модель документа

Лекция 5.

JavaScript и объектная модель (2 часа).

Раздел 5. Общее понятие динамического HTML (DHTML)

Лекция 6.

Общее понятие динамического HTML (DHTML) (2 часа).

Раздел 6. Введение в расширяемый язык разметки (XML)

Лекция 7.

Введение в расширяемый язык разметки (XML) (2 часа).

Раздел 7. Структура XML-документа

Лекция 8.

Иерархическая структура XML-документа (2 часа).

Лекция 9.

Типы анализаторов (2 часа).

Раздел 8. Стилевые таблицы XSL

Лекция 10.

Стилевые таблицы XSL (2 часа).

Лекция 11.

Инструкции xsl. Элементы xsl (2 часа).

Раздел 9. Создание динамических web-ресурсов. CGI

Лекция 12.

Создание динамических web-ресурсов. CGI (2 часа).

Лекция 13.

Динамический web-ресурс. Взаимодействие клиента с WWW-сервером. Структура HTTP-запросов и ответов (2 часа).

Раздел 10. Реализация CGI

Лекция 14.

Реализация CGI.Переменные окружения. Командная строка (2 часа).

Лекция 15.

Реализация CGI. Стандартный ввод. Стандартный вывод. Реализация кода обработки данных. Вывод результатов работы программы. Права доступа (2 часа).

4.1.2.2. Перечень практических занятий

Не планируется.

4.1.2.3. Перечень лабораторных работ

Семестр 7

Раздел 5. Общее понятие динамического HTML (DHTML)

Лабораторная 1.

Статический веб-сайт. Основы HTML (4 часа).

Раздел 9. Создание динамических web-ресурсов. CGI

Лабораторная 2.

Статический веб-сайт (4 часа).

Лабораторная 3.

Стили оформления (4 часа).

Раздел 10. Реализация CGI

Лабораторная 4.

Динамический веб-сайт (4 часа).

Лабораторная 5.

Интерактивный вебсайт (4 часа).

Лабораторная 6.

Первичная настройка Apache. Миграции и посев данных (4 часа).

4.1.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы

Перечень тем, вынесенных на самостоятельное изучение:

1. Историческое развитие web-технологий.
2. Перспективы развития web-технологий.
3. Новые управляющие элементы HTML 5.
4. Перспективы применения HTML 5 при разработке информационных и медийных web-ресурсов.
5. ECMAScript – практика применения.
6. Синтаксис JavaScript.
7. Внедрение технологии AJAX в web-ресурс.
8. Создание интерактивных средств с использованием ActionScript.
9. Практика использования 3D визуализации в web.
10. Практика использования XML как средства обмена информацией.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

4.1.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР

Не планируется.

4.1.2.6. Примерный перечень тем курсовых работ (проектов)

Не планируется.

4.2 Форма обучения: заочная

Уровень базового образования: среднее профессиональное.
Срок обучения 3г 6м.

Семестр	Трудоемкость, час./ зач. ед.	Лекции, час.	Практические занятия, час.	Лабораторные работы, час.	Консультация, час.	Контроль,час.	Всего (контактная работа), час.	СРС, час.	Форма промежуточного контроля (экз., зач., зач. с оц.)
6	**180 / 5**	**8**		**8**	**4**	**0,6**	**20,6**	**150,75**	**Экз.(8,65)**
Итого	**180 / 5**	**8**		**8**	**4**	**0,6**	**20,6**	**150,75**	**8,65**

4.2.1. Структура дисциплины

№ п\п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником							Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация	Контроль		
1	Введение в Интернет-технологии	6	2							15	Устный опрос
2	Особенности разработки web-проектов.	6	2							15	Устный опрос
3	Создание web-страниц с использованием HTML	6	4		8					15	Отчет по лабораторным работам
4	JavaScript и объектная модель документа	6								15	Устный опрос
5	Общее понятие динамического HTML (DHTML)	6								15	Устный опрос
6	Введение в расширяемый язык разметки (XML)	6								15	Промежуточная контрольная работа
7	Структура XML-	6								15	Устный опрос

	документа										
8	Стилевые таблицы XSL	6								15	Устный опрос
9	Создание динамических web-ресурсов. CGI	6								15	Устный опрос
10	Реализация CGI	6								15,75	Итоговая контрольная работа.
Всего за семестр		180	8		8	+		4	0,6	150,75	Экз.(8,65)
Итого		180	8		8			4	0,6	150,75	8,65

4.2.2. Содержание дисциплины

4.2.2.1. Перечень лекций

Семестр 6

Раздел 1. Введение в Интернет-технологии

Лекция 1.

Введение в Интернет-технологии (2 часа).

Раздел 2. Особенности разработки web-проектов.

Лекция 2.

Особенности разработки web-проектов (2 часа).

Раздел 3. Создание web-страниц с использованием HTML

Лекция 3.

Создание web-страниц с использованием HTML (2 часа).

Лекция 4.

Создание HTML документов: текст, форма, фрейм, изображение, стили (2 часа).

4.2.2.2. Перечень практических занятий

Не планируется.

4.2.2.3. Перечень лабораторных работ

Семестр 6

Раздел 1. Создание web-страниц с использованием HTML

Лабораторная 1.

Статический веб-сайт. Основы HTML (4 часа).

Лабораторная 2.

Стили оформления (4 часа).

4.2.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы

Перечень тем, вынесенных на самостоятельное изучение:

1. Историческое развитие web-технологий.
2. Перспективы развития web-технологий.
3. Новые управляющие элементы HTML 5.
4. Перспективы применения HTML 5 при разработке информационных и медийных web-ресурсов.
5. ECMAScript – практика применения.
6. Синтаксис JavaScript.
7. Внедрение технологии AJAX в web-ресурс.
8. Создание интерактивных средств с использованием ActionScript.
9. Практика использования 3D визуализации в web.
10. Практика использования XML как средства обмена информацией.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

4.2.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР

1. Обзор сервисов в глобальной сети Интернет.
2. История развития сети Интернет.
3. Реализация HTTP протокола в среде Delphi.
4. Обзор Интернет технологий для программирования Web.
5. Язык HTML.
6. Технология Java.
7. Особенности реализации технологии CGI.
8. Особенности реализации технологии IS API.
9. Технология ASP в Интернет.
10. Реализация почтовых клиентов в сети Интернет.
11. Технология XML.
12. Использование Flash технологии для разработки сайтов.
13. Особенности построения виртуальных миров в Интернет.
14. Организация поисковых систем в Интернет.
15. Обзор современных браузеров для работы в сети Интернет.
16. Использование каскадных таблиц стилей при построении сайтов.
17. Технология JavaScript.
18. Использование языка PHP для организации сайтов в сети Интернет.
19. Операторы языка JavaScript.
20. Серверные вставки (SSI).
21. Структура XML-документа.
22. Взаимодействие CGI и БД.
23. Безопасность в PHP.
24. Разработка сайтов при помощи языка Perl.
25. Защита БД от несанкционированного доступа в Интернет.
26. Стилевые таблицы XSL.
27. Общее понятие динамического HTML (DHTML).
28. Общие принципы создания Web-узла.
29. Создание таблиц в HTML.
30. HTML-формы.
31. Фреймы в HTML.
32. История развития HTML .
33. Создание форм в HTML.

4.2.2.6. Примерный перечень тем курсовых работ (проектов)

Не планируется.

5. Образовательные технологии

В процессе изучения дисциплины Интернет-технологии применяется контактная технология преподавания (за исключением самостоятельно изучаемых студентами вопросов). При проведении лабораторных работ применяется имитационный или симуляционный подход, когда преподавателем разбирается на конкретном примере проблемная ситуация, все шаги решения задачи студентам демонстрируются при помощи мультимедийной техники. Затем студенты самостоятельно решают аналогичные задания.

Во время выполнения лабораторных работ формируются творческие коллективы из 3-5 студентов, разрабатывающих интернет-сайт по одной тематике, тем самым формируется способность обучающихся к работе в малых творческих коллективах.

Удельный вес занятий, проводимых в интерактивных формах – 70%.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

Фонды оценочных материалов (средств) приведены в приложении.

7. Учебно-методическое и информационное обеспечение дисциплины.

7.1. Основная учебно-методическая литература по дисциплине

1. Маркарян Л.В. Инструментальные средства Internet-технологий : лаб. прак-тикум / Л.В. Маркарян. – М. : Изд. Дом НИТУ «МИСиС», 2018. – 92 с. - https://www.iprbookshop.ru/98167.html

2. Крахоткина Е. В. Технологии разработки Internet-приложений: учебное пособие. – Ставрополь: Изд-во СКФУ, 2016. – 124 с. - https://www.iprbookshop.ru/66043.html

3. Воронцов Ю.А., Козинец А.В. WEB-программирование распределённых информационных систем: Учебное пособие / МТУСИ. –М., 2017. –66 с. - https://www.iprbookshop.ru/92420.html

4. Ефромеев, Н. М. Основы web-программирования : учебное пособие / Н. М. Ефромеев, Е. В. Ефромеева. — Саратов : Вузовское образование, 2019. — 128 c. — ISBN 978-5-4487-0529-8. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/86300.html (дата обращения: 17.12.2022). — Режим доступа: для авторизир. пользователей. - DOI: https://doi.org/10.23682/86300 - https://www.iprbookshop.ru/86300.html

7.2. Дополнительная учебно-методическая литература по дисциплине

1. Торопова О.А. Основы web-программирования. Технологии HTML, DHTML: учеб. пособие /О.А. Торопова, И.Ф. Сытник. Саратов: Сарат. гос. техн. ун-т, 2012, 106 с. - https://www.iprbookshop.ru/76493.html

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

В образовательном процессе используются информационные технологии, реализованные на основе информационно-образовательного портала института (www.mivlgu.ru/iop), и инфокоммуникационной сети института:

- предоставление учебно-методических материалов в электронном виде;
- взаимодействие участников образовательного процесса через локальную сеть института и Интернет;
- предоставление сведений о результатах учебной деятельности в электронном личном кабинете обучающегося.

Информационные справочные системы:

http://php.net/manual/ru/ - PHP

http://htmlbook.ru/ - HTML, CSS

http://jquery.com/ - jQuery

http://jquery.page2page.ru – русскоязычная wiki о jQuery

http://javascript.ru/ - JavaScript

Программное обеспечение:

LibreOffice (Mozilla Public License v2.0)

Mozilla Firefox (MPL)

РЕД ОС (Соглашение №140/05-21У от 18.05.2021 года о сотрудничестве в области науки, развития инновационной деятельности)

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

iprbookshop.ru
php.net
htmlbook.ru
jquery.com
jquery.page2page.ru – русскоязычная wiki о jQuery
javascript.ru
mivlgu.ru/iop

8. Материально-техническое обеспечение дисциплины

Лаборатория распределенных систем

12 персональных компьютеров; проектор Nec V300X; экран настенный Lumien Master Picture

9. Методические указания по освоению дисциплины

Для успешного освоения теоретического материала обучающийся: знакомится со списком рекомендуемой основной и дополнительной литературы; уточняет у преподавателя, каким дополнительным пособиям следует отдать предпочтение; ведет конспект лекций и прорабатывает лекционный материал, пользуясь как конспектом, так и учебными пособиями.

До выполнения лабораторных работ обучающийся изучает соответствующий раздел теории. Перед занятием студент знакомится с описанием заданий для выполнения работы, внимательно изучает содержание и порядок проведения лабораторной работы. Лабораторная работа проводятся в компьютерном классе. Обучающиеся выполняют индивидуальную задачу компьютерного моделирования в соответствии с заданием на лабораторную работу. Полученные результаты исследований сводятся в отчет и защищаются по традиционной методике в классе на следующем лабораторном занятии. Необходимый теоретический материал, индивидуальное задание, шаги выполнения лабораторной работы и требование к отчету приведены в методических указаниях, размещенных на информационно-образовательном портале института.

Используя основную и дополнительную литературу студент должен самостоятельно изучить разделы дисциплины для самостоятельного изучения.

Форма заключительного контроля при промежуточной аттестации – экзамен. Для проведения промежуточной аттестации по дисциплине разработаны фонд оценочных средств и балльно-рейтинговая система оценки учебной деятельности студентов. Оценка по дисциплине выставляется в информационной системе и носит интегрированный характер, учитывающий результаты оценивания участия студентов в аудиторных занятиях, качества и своевременности выполнения заданий в ходе изучения дисциплины и промежуточной аттестации.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению *09.03.02 Информационные системы и технологии* и профилю подготовки *Информационные системы и технологии*
Рабочую программу составил *д.т.н., доцент, заведующий кафедрой Андрианов Д.Е.*_____

Программа рассмотрена и одобрена на заседании кафедры *ИС*

протокол № 21 от 05.05.2026 года.
Заведующий кафедрой *ИС* _____*Андрианов Д.Е.*
(Подпись)

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии факультета

протокол № 9 от 12.05.2026 года.
Председатель комиссии ФИТР _____ *Кутарова Е.И.*
(Подпись) (Ф.И.О.)

Приложение

Фонд оценочных материалов (средств) по дисциплине
Интернет-технологии

1. Оценочные материалы для проведения текущего контроля успеваемости по дисциплине

Блок 1 (знать).

1. Динамическими Web-страницами принято называть страницы, которые
   * перед отправкой клиенту проходят обработку на сервере.
   перед отправкой проходят обработку на машине клиента.
   перед отправкой клиенту проходят обработку в Интернет.
   формируются по жестко заданным правилам.

2. Какой стандарт был взят за основу языка HTML?
   SSML
   SGNL
   * SGML
   GSM

3. Что обеспечивает полную платформенную независимость языка HTML?
   Операционная система Unix.
   * Способ создания гипертекста.
   Операционная система MS-DOS.
   Кодировка текста.

4. Чем отличается стартовый тэг от завершающего тэга?
   Наличием прямого слеша.
   Наличием обратного слеша.
   *Отсутствием прямого слеша.
   Отсутствием обратного слеша.

5. Из каких частей состоит HTML-документ?
   * Заголовочной и тела документа.
   Заголовочной, тела документа и пролога.
   Верхнего меню, левого меню, основного поля и «подвала».
   Зона логотипа, верхнего меню, левого меню, основного поля и «подвала».

6. Каким образом можно вставить комментарий в HTML документ?
   <-- Это комментарий -->
   * <!-- Это комментарий -->
   <// Это комментарий -->
   // Это комментарий

7. Часть документа, которая отображается как текстовая и графическая (смысловая) информация называется:
   Заголовочная часть.
   Фейм.
   * Тело документа
   Страница Web-документа.

8. К какому типу языков программирования относится HTML?
   Транслятор
   Компилятор

* Интерпретатор
Эмулятор

9. Какой тэг используется для обозначения заголовка?
9.1. * Тэг <H1>, <H2> и т.д.
9.2. Тэг <Hx>
9.3. Тэг <H>
9.4. Тэг <Hi>

10. Сколько уровней заголовков поддерживают большинство браузеров?
7
5
4
* 6

Блок 2 (уметь).

1. Внутри предварительно сформатированного тэгом переформатирования текста разрешается использовать:
Символ возврата каретки, символ пробела, пропорциональный текст.
Символ возврата каретки, символ пробела, непропорциональный текст.
* Символ перевода строки, символ табуляции, непропорциональный текст.
Символ возврата каретки, символ табуляции, пропорциональный текст.

2. При помощи каких параметров можно задать вид пронумерованного списка?
2.1. Type, Align
2.2. Start, Align
2.3. * Type, Start
2.4. Start, Href

3. При помощи каких параметров можно задать вид элемента списка?
3.1. * Type, Value
2.2. Start, Align
2.3. Type, Start
2.4. Start, Href

4. Что в языке HTML используется для представления гипертекстовых ссылок и ссылок на сетевые сервисы внутри документа?
4.1. Href
4.2. * URL
4.3. SQL
4.4. XML

5. Выберите из предложенных вариантов строку HTML документа создающую маркер указывающий на раздел.
5.1. <A HREF="#named_anchor"> Текст </A>
5.2. * <A NAME="named_anchor"> Текст</A>
5.3. <A HREF="named_anchor"> Текст </A>
5.4. <A NAME="#named_anchor"> Текст</A>

6. В какой части HTML документа обычно подключается описание элементов JavaScript?
6.1. * В заголовочной
6.2. В любой части

6.3. В теле документа
6.4. Во внешнем файле.

7. Почему в языке JavaScript отсутствует необходимость инициализации переменных?
7.1. Тип определять не нужно
7.2. Используется один тип данных
7.3. * Тип данных определяется автоматически
7.4. Типы как таковые отсутствуют

8. Какое количество типов данных распознает JavaScript?
8.1. 3
8.2. 4
8.3. * 5
8.4. 6

9. Какие логические структуры допустимы в языке JavaScript?
9.1. if, for
9.2. is, else
9.3. if, case
9.4. * if, switch

10. Какова особенность вычисления логических операторов в JavaScript?
10.1. * возвращают значение одного из операндов
10.2. возвращают логическое выражение
10.3. возвращают числовое значение
10.4. не вычисляются

Блок 3 (владеть).

1. На какие три группы можно разделить элементы языка HTML?
* Структура документа, эффекты форматирования, управление программными средствами.
Структура файла, динамические эффекты форматирования, управление программными средствами.
Описание дескрипторов, динамические эффекты форматирования, управление программными системами.
Структура дескрипторов, динамические эффекты форматирования, управление программными средствами.

2. Выберете тэги, которые используются для отображения списка определений.
2.1. <OL>, <LI>, <UL>
2.2. <DT>, <DD>, <UL>
2.3. <DD>, <LI>, <DT>
2.4. * <DL>, <DT>, <DD>

3. Какие основные Internet-технологии Вам известны?
3.1. XML, HTML, CGI, IS API.
3.2. * HTML, CGI, IS API, ASP
3.3. XML, CGI, IS API, ASP.NET
3.4. XML, CGI, IS API, PHP

4. Приложение, которое выполняется на Web-сервере для создания динамического содержимого сайта на основе скриптовых технологий:

4.1. PHP
4.2. IS API
4.3. * CGI
4.4. XML

5. Приложение, которое загружается один раз и выполняется по мере поступления запроса выполняется по технологии:
5.1. PHP
5.2. * IS API
5.3. CGI
5.4. XML

6. Набор информационных материалов различного типа, определенным образом структурированных и доступных пользователю по его уникальному адресу это:
6.1. Web-сайт
6.2. WWW-сервер
6.3. Web-проект
6.4. * Web-ресурс

7. Что из перечисленного не относится к этапу планирования в разработке Web-проектов?
7.1. Постановка задачи
7.2. Определение потенциальной аудитории ресурса.
7.3. * Перечень средств разработки.
7.4. Конкурентный анализ

8. Что из перечисленного не относится к этапу проектирования в разработке Web-проектов?
8.1. * Постановка задачи
8.2. Перечень средств разработки.
8.3. Размещение ресурса.
8.4. Дальнейшая информационная поддержка.

9. В каком наборе разработчиков сайтов допущена ошибка?
9.1. Промоутер, художник-дизайнер, Web-мастер.
9.2. Промоутер, художник-дизайнер, Web-мастер, Web-программист.
9.3. * Промоутер, художник-дизайнер, системный программист, Web-мастер.
9.4. Промоутер, администратор сервера, Web-мастер, Web-программист.

10. Платформа для создания, развертывания и запуска Web-сервисов и приложений это:
10.1. ASP
10.2. ASP .NET
10.3. * Microsoft .NET Framework
10.4. ADO.NET

Общее распределение баллов текущего контроля по видам учебных работ для студентов

Рейтинг-контроль 1	Устный опрос 10 вопросов, 5 практических заданий, 1 лабораторная работа.	до 15 баллов
Рейтинг-контроль 2	Устный опрос 10 вопросов, 5 практических заданий, 2	до 15 баллов

	лабораторных работы.	
Рейтинг-контроль 3	Устный опрос 10 вопросов, 2 лабораторных работы.	до 15 баллов
Посещение занятий студентом	Посещение занятий	до 5 баллов
Дополнительные баллы (бонусы)	Активность студента	до 5 баллов
Выполнение семестрового плана самостоятельной работы	Самостоятельная работа	до 5 баллов

2. Промежуточная аттестация по дисциплине

Перечень вопросов к экзамену / зачету / зачету с оценкой.
Перечень практических задач / заданий к экзамену / зачету / зачету с оценкой (при наличии)

ОПК-2, ОПК-3
Блок 1 (знать)
Какой будет результат выполнения программного кода

```
$var1 = "World";
$hello = "var1";
echo $$hello;
```

•
var1
•
World
•
hello
•
hello World

Укажите правильный селектор для всех элементов с классом «tab»

```
<ul>
   <li class="feedtab news active">
      <a href="/news/">
         <b class="tab word">Новости</b>
      </a>
   </li>
   <li class="feedtab notifications">
      <a href="/notifications/">
         <b class="tab word">Ответы</b>
      </a>
   </li>
   <li class="feedtab comments">
      <a href="/comments/">
         <b class="tab word">Комментарии</b>
      </a>
   </li>
</ul>
```

•
.tab .word
•
b .tab.word

•
.feedtab .tab.word
•
.feedtab > .tab .word

Сколько записей будет иметь объект jQuery после выполнения следующей функции

```
$(".feedtab.active").parents().find(".tab")
```

HTML

```
<ul>
   <li class="feedtab news active">
      <a href="/news/">
         <b class="tab word">Новости</b>
      </a>
   </li>
   <li class="feedtab notifications">
      <a href="/notifications/">
         <b class="tab word">Ответы</b>
      </a>
   </li>
   <li class="feedtab comments">
      <a href="/comments/">
         <b class="tab word">Комментарии</b>
      </a>
   </li>
</ul>
```

•
0
•
1
•
2
•
3

Что произойдёт после нажатия на кнопку Find

```
$(function() {
   function Form(selector) {
      var self = this;
      this.selector = selector;
      this.field;

      this.setField = function(field) {
         this.field = field;
         return this;
      }

      this.onSubmit = function(){
         if (self.field) {
            alert(self.field.attr('name'));
         }
      };
```

```
        this.selector.submit(function() {
            self.onSubmit.call(self);
            return false;
        })
    }

    var search = new Form($("form"));
    search.setField(search.selector.find(">input"));
})
```

HTML

```
<form>
    <input type="hidden" name="journal" />

    <fieldset>
        <input type="text" name="query" placeholder="Search" />
    </fieldset>

    <fieldset>
        <button type="submit">
            <span>Find</span>
        </button>
    </fieldset>
</form>
```

- Ничего
- Появится сообщение с текстом «journal»
- Появится сообщение с текстом «query»
- Появится сообщение с текстом «journal, query»

Какой будет результат выполнения программного кода

```
abstract class Test {
    public $a;
    public $b;
    public $result;

    public function __construct($a, $b) {
        $this->a = $a;
        $this->b = $b;
    }

    abstract function calc();

    final public function getResult() {
        return $this->result;
    }
}
class Test1 extends Test {
    public function calc() {
```

```
        $this->result = $this->a + $this->b;
    }

}
class Test2 extends Test1 {
    public function calc() {
        parent::calc();
        $this->result *= 2;
    }
}

$obj = new Test1(2,3);
$obj->calc();
echo $obj->getResult();
```

•
Ошибка выполнения
•
5
•
10
•
0

Выберите правильное определение функции внутри основного кода:

•
```
var func = function();
```
•
```
function test(){}
```
•
```
func: function(){};
```
•
```
function(){}
```

Какие стили будут добавлены к элементу input после компиляции LESS файла

```
.padding(@value: 0) {
    padding: ~"@{value}px";
}

.calc(@a; @b; @c) {
    .padding(@a + @b);
}

.calc(@a; @b; @c) when (@c = "+") {
    .padding(@a + @b);
}

.calc(@a; @b; @c) when (@c = "-") {
    .padding(@a - @b);
}

.calc(@a; @b; @c) when (@c = "/") {
    .padding(@a / @b);
}

.calc(@a; @b; @c) when (@c = "*") {
    .padding(@a * @b);
```

```
}

input {
   .calc(20, 5, "/");
}
```

•
padding: 20px;
•
padding: 4px;
•
padding: 25px;
•
padding: 0px;

Выберите правильный формат инициализации переменных

•
var test = 2;
•
test := 2;
•
$test = 2;
•
var $test := 2;

Какие стили будут добавлены к элементу input после компиляции LESS файла

```
.padding(@vert: 0px; @hor: 0px) {
   padding: @vert @hor;
}

.calc(@a; @b: 5) {
   .padding(~"@{a}px"; ~"@{b}px");
}

input {
   .calc(10);
}
```

•
.calc(10)
•
padding: 10
•
padding: 10px 0px
•
padding: 10px 5px

Блок 2, уметь

Вопрос 10

Какой будет результат выполнения программного кода

```
class TestParent {
   public $result;

   public static function calc($a, $b) {
      return $a + $b;
   }

   public function getResult() {
```

```
        return $this->result;
    }
}
class Test extends TestParent {
    public function __construct($a, $b) {
        $this->result = parent::calc($a, $b);
    }

    public static function calc($a, $b) {
        $this->result = $a * $b;
        return $this->result;
    }
}

$obj = new Test(2,3);
echo $obj->getResult();
```

•

Ошибка выполнения

•

5

•

23

•

6

Выберите результат, который вернёт функция, после выполнения следующего кода

```
function Test(param1, param2) {
    this.result = param1 + param2;

    this.getResult = function() {
        return this.result;
    }
}

var test1 = new Test(10, 2);
test1.getResult()
```

•

NaN

•

102

•

12

•

0

Выберите корректный альтернативный селектор, описывающий строку с классом «.level1»

```
.block.pricelist .table > .header .column {
  font-weight: bold;
}
.block.pricelist .table > .content .row.level1 .column:first-child {
  padding-left: 15px;
}
.block.pricelist .table > .content .row.level2 .column:first-child {
  padding-left: 30px;
```

}

- .block .table.content .level1
- .block .table > .row.level1
- .block > .content > .level1
- .block .table .row.level1

Выберите правильный вариант подключения интерфейса к классу

- class Test include iTest
- class Test implements iTest()
- class Test implements iTest
- class Test interface iTest

Выберите правильный вариант подключения нескольких интерфейсов к классу

- class Test include iTest, iTest2
- class Test implements iTest() implements iTest2
- class Test implements iTest, iTest2
- class Test interface iTest and iTest2

Выберите корректный альтернативный селектор, описывающий только колонку внутри заголовка таблицы

```
.block.pricelist .table > .header .column {
  font-weight: bold;
}
.block.pricelist .table > .content .row.level1 .column:first-child {
  padding-left: 15px;
}
.block.pricelist .table > .content .row.level2 .column:first-child {
  padding-left: 30px;
}
```

- .table > .column
- .block.table .column
- .table .column
- .block .table .header .column

Какой будет результат выполнения программного кода

```
class TestParent {
   public $result;

   public static function calc($a, $b) {
      return $a + $b;
   }
```

```
    public function getResult() {
        return $this->result * 2;
    }
}
class Test extends TestParent {
    public function __construct($a, $b) {
        $this->result = self::calc($a, $b);
    }

    public static function calc($a, $b) {
        return $a * $b;
    }

    public function getResult() {
        return $this->result;
    }
}

$obj = new Test(2,3);
echo $obj->getResult();
```

- Ошибка выполнения
- 5
- 12
- 6

Выберите результат, который вернёт функция, после выполнения следующего кода

```
function Test(param1, param2) {
    this.result = param1 + param2;

    this.getResult = function() {
        return this.result;
    }
}

var test1 = new Test(10);
test1.getResult()
```

- NaN
- 10
- undefined
- 0

Выберите корректный альтернативный селектор, описывающий все колонки внутри таблицы

```
.block.pricelist .table > .header .column {
  font-weight: bold;
}
```

```
.block.pricelist .table > .content .row.level1 .column:first-child {
  padding-left: 15px;
}
.block.pricelist .table > .content .row.level2 .column:first-child {
  padding-left: 30px;
}
```

•
.table .column
•
.header .column
•
.table.column
•
.table > .column

Выберите правильный вариант объявления публичного метода класса

•
public function test() {}
•
class function test() {}
•
method test() {}
•
public method test() {}

Выберите результат, который вернёт функция, после выполнения следующего кода

```
function Test(param1, param2) {
   this.result = param1 + param2;

   this.getResult = function() {
      return this.result;
   }
}

var test1 = new Test("test", 2);
test1.getResult()
```

•
NaN
•
2
•
undefined
•
test2

Какой будет результат выполнения программного кода

```
class TestParent {
   public $result;

   public static function calc($a, $b) {
      return $a + $b;
   }

   public function getResult() {
      return $this->result * 2;
```

```
    }
}
class Test extends TestParent {
    public function __construct($a, $b) {
        $this->result = parent::calc($a, $b);
    }

    public static function calc($a, $b) {
        return $a * $b;
    }

    public function getResult() {
        return $this->result;
    }
}

$obj = new Test(2,3);
echo $obj->getResult();
```

- Ошибка выполнения
- 5
- 10
- 6

Выберите результат, который вернёт функция, после выполнения следующего кода

```
function Test(param1, param2) {
    this.result = param1 + param2;

    this.getResult = function() {
        return this.result;
    }
}

var test1 = new Test(2, "test");
test1.getResult()
```

- NaN
- 2
- undefined
- 2test

Выберите правильный вариант объявления абстрактного класса

- abstract public function test();
- abstract public function test() {}
- abstract method test();
-

abstract method test;

Выберите правильный вариант доступа к не статичным переменным класса из не статичной функции

- $this->$param1;
- $this::$param1;
- $this->param1;
- $this::param1;

Выберите результат, который вернёт функция, после выполнения следующего кода

```
function Test(param1, param2) {
   this.result = param1 + param2;

   this.getResult = function() {
      return this.result;
   }
}

var test1 = new Test("test", "test");
test1.getResult()
```

- NaN
- 0
- testtest
- undefined

Выберите результат, который вернёт функция, после выполнения следующего кода

```
function Test(param1, param2) {
   this.result = param1 + param2;

   this.getResult = function() {
      return this.result;
   }
}

var test1 = new Test("test", [1,2]);
test1.getResult()
```

- NaN
- 0
- test1,2
- undefined

Выберите правильный вариант доступа к статичным переменным класса из статичной функции

-

$this::$param1;
•
$this::param1;
•
self::$param1;
•
self::param1;
Выберите результат, который вернёт функция, после выполнения следующего кода

```
function Test(param1, param2) {
   this.result = param1 + param2;

   this.getResult = function() {
      return this.result;
   }
}

var test1 = new Test([1,2], [3,4]);
test1.getResult()
```

•
NaN
•
[1,2,3,4]
•
[]
•
1,23,4
Выберите правильный вариант вызова функции от родительского объекта
•
parent::test();
•
self::test();
•
child::test();
•
$this->test();
Выберите правильный вариант вызова не статичной, публичной функции от созданного объекта
•
$obj->test();
•
$obj::test();
•
$obj->$test();
•
$obj::$test();
Выберите результат, который вернёт функция, после выполнения следующего кода

```
function Test(param1, param2) {
   this.result = param1 + param2;

   this.getResult = function() {
      return this.result;
   }
```

```
}

var test1 = new Test({abc: 1}, {cde: 2});
test1.getResult()
```

- NaN
- {abc:1, cde:2}
- {}
- [object Object][object Object]

Выберите правильный вариант вызова статичной, публичной функции от созданного объекта

- $obj->test();
- $obj::test();
- $obj->$test();
- $obj::$test();

Выберите результат, который вернёт функция, после выполнения следующего кода

```
function Test(param1, param2) {
   this.result = param1 + param2;

   this.getResult = function() {
      return this.result;
   }
}

var test1 = new Test(2, "2");
test1.getResult()
```

- NaN
- 2
- 22
- 4

Сколько записей будет иметь объект jQuery после выполнения следующей функции

```
$("ul a:lt(2)")
```

HTML

```
<ul>
   <li class="feedtab news active">
      <a href="/news/">
         <b class="tab word">Новости</b>
      </a>
   </li>
```

```
    <li class="feedtab notifications">
        <a href="/notifications/">
            <b class="tab word">Ответы</b>
        </a>
    </li>
    <li class="feedtab comments">
        <a href="/comments/">
            <b class="tab word">Комментарии</b>
        </a>
    </li>
</ul>
```

•
0
•
1
•
2
•
3

Что произойдёт после нажатия на кнопку Find

```
$(function() {
    function Form(selector) {
        var self = this;
        this.selector = selector;
        this.field;

        this.setField = function(field) {
            this.field = field;
            return this;
        }

        this.onSubmit = function(){
            if (self.field) {
                alert(self.field.attr('name'));
            }
        };

        this.selector.submit(function() {
            self.onSubmit.call(self);
            return false;
        })
    }

    var search = new Form($("form"));
    search.setField(search.selector.find(">input"));
    search.onSubmit = function() {

    }
})
```

HTML

```
<form>
```

```
    <input type="hidden" name="journal" />

    <fieldset>
        <input type="text" name="query" placeholder="Search" />
    </fieldset>

    <fieldset>
        <button type="submit">
            <span>Find</span>
        </button>
    </fieldset>
</form>
```
•
Ничего
•
Появится сообщение с текстом «journal»
•
Появится сообщение с текстом «query»
•
Появится сообщение с текстом «journal, query»

Укажите правильный селектор для активного элемента

```
<ul>
    <li class="feedtab news active">
        <a href="/news/">
            <b class="tab word">Новости</b>
        </a>
    </li>
    <li class="feedtab notifications">
        <a href="/notifications/">
            <b class="tab word">Ответы</b>
        </a>
    </li>
    <li class="feedtab comments">
        <a href="/comments/">
            <b class="tab word">Комментарии</b>
        </a>
    </li>
</ul>
```
•
.ul li.news.active
•
ul > .active
•
li .feedtab.active
•
.li.active

Какой будет результат выполнения программного кода

```
$var1 = "hello";
$hello = "var1";
$hello_world = "Hello world";
echo ${$$hello."_world"};
```
•
var1

•
World
•
hello
•
hello World

Блок 2 (уметь)
Какие стили будут добавлены к элементу input после компиляции LESS файла
```
.padding(@vert: 0px; @hor: 0px) {
   padding: @vert @hor;
}

.calc(@a; @b: 5) {
   .padding(~"@{a}px"; 10px);
}

input {
   .calc(10; 15);
}
```
•
padding: 10px 15px
•
padding: 10 15
•
padding: 10px 10px
•
padding: 10px 5px
Какие стили будут добавлены к элементу input после компиляции LESS файла
```
.padding(@value: 0) {
   padding: ~"@{value}px";
}

.calc(@a; @b; @c) {
   .padding(@a + @b);
}

.calc(@a; @b; @c) when (@c = "+") {
   .padding(@a + @b);
}

.calc(@a; @b; @c) when (@c = "-") {
   .padding(@a - @b);
}

.calc(@a; @b; @c) when (@c = "/") {
   .padding(@a / @b);
}

.calc(@a; @b; @c) when (@c = "*") {
   .padding(@a * @b);
}

input {
```

```
    .calc(20, 5, "%");
}
```

•
padding: 25px;
•
padding: 4px;
•
Стили не будут добавлены
•
padding: 0px;

Выберите неправильный вариант оформления цикла FOR

•
for(var $i = 1; $i < 10; $i++);
•
for($j = 1; $j < 10; $j++);
•
for(;;);
•
for($i = 1, $j = 1;;$i+=5);

Выберите неправильное определение цикла FOR

•
for(var i in values);
•
for(var i=0;;i++);
•
for(var i in key=>value);
•
for(var i=0, j=0; i<10, j<10; i++, j++);

Какой будет результат выполнения программного кода

```
abstract class Test {
    public $a;
    public $b;
    public $result;

    public function __construct($a, $b) {
        $this->a = $a;
        $this->b = $b;
    }

    abstract function calc();

    final public function getResult() {
        return $this->result;
    }
}
class Test1 extends Test {
    public function calc() {
        $this->result = $this->a + $this->b;
    }

}
class Test2 extends Test1 {
    public function calc() {
```

```
        parent::calc();
        $this->result *= 2;
    }
}

$obj = new Test2(2,3);
$obj->calc();
echo $obj->getResult();
```
•
Ошибка выполнения
•
5
•
10
•
0

Выберите не правильное определение цикла WHILE
•
while(true);
•
while(i < 20);
•
while(i in [1,2,3]);
•
while;

Какие стили будут добавлены к элементу input после компиляции LESS файла
```
.padding(@vert: 0px; @hor: 0px) {
    padding: @vert @hor;
}

.calc(@a; @b: 5) {
    .padding(~"`@{a}+@{b}`px");
}

input {
    .calc(10; 15);
}
```
•
padding: 10px 15px
•
padding: 10px 5px
•
padding: 15px 0px
•
padding: 25px 0px

Какой будет результат выполнения программного кода
```
function test($a, $b) {
    return $a + $b;
}
$func = "test";
$args = array(2, 3);
echo call_user_func_array($func, $args);
```
•

Ошибка выполнения
•
5
•
23
•
test23
Укажите правильный селектор для ссылки с адресом /news/

```
<ul>
   <li class="feedtab news active">
      <a href="/news/">
         <b class="tab word">Новости</b>
      </a>
   </li>
   <li class="feedtab notifications">
      <a href="/notifications/">
         <b class="tab word">Ответы</b>
      </a>
   </li>
   <li class="feedtab comments">
      <a href="/comments/">
         <b class="tab word">Комментарии</b>
      </a>
   </li>
</ul>
```

•
.feedtab[href=’/news/’]
•
li a [href=’/news/’]
•
ul a[href=’/news/’]
•
ul > [href=’/news/’]
Какие стили будут добавлены к элементу input после компиляции LESS файла

```
.padding(@value: 0) {
   padding: ~"@{value}px";
}

.calc(@a; @b; @c) when (@c = "+") {
   .padding(@a + @b);
}

.calc(@a; @b; @c) when (@c = "-") {
   .padding(@a - @b);
}

.calc(@a; @b; @c) when (@c = "/") {
   .padding(@a / @b);
}

.calc(@a; @b; @c) when (@c = "*") {
   .padding(@a * @b);
}
```

```
.calc(@a; @b; @c) {
   .padding(@a + @b);
}

input {
   .calc(20, 5, "-");
}
```

•
padding: 15px;
•
padding: 25px;
•
padding: 20px;
•
padding: 0px;

Выберите неправильный вариант оформления цикла WHILE

•
while();
•
while(1);
•
while($i < 10);
•
while($i < test());

Сколько записей будет иметь объект jQuery после выполнения следующей функции

```
$("li.news a[href='/comments/'] b").parents()
```

HTML

```
<ul>
   <li class="feedtab news active">
      <a href="/news/">
         <b class="tab word">Новости</b>
      </a>
   </li>
   <li class="feedtab notifications">
      <a href="/notifications/">
         <b class="tab word">Ответы</b>
      </a>
   </li>
   <li class="feedtab comments">
      <a href="/comments/">
         <b class="tab word">Комментарии</b>
      </a>
   </li>
</ul>
```

•
0
•
1
•
2
•

3
Что произойдёт после нажатия на кнопку Find

```
$(function() {
    function Form(selector) {
        var self = this;
        this.selector = selector;
        this.field;

        this.setField = function(field) {
            this.field = field;
            return this;
        }

        this.onSubmit = function(){
            if (self.field) {
                alert(self.field.attr('name'));
            }
        };

        this.selector.submit(function() {
            self.onSubmit.call(self);
            return false;
        })
    }

    var search = new Form($("form"));
    search.setField(search.selector.find(">input"));
    search.onSubmit = function() {
        if (this.field) {
            alert(this.field.attr('type'));
        }
    }})
```

HTML

```
<form>
    <input type="hidden" name="journal" />

    <fieldset>
        <input type="text" name="query" placeholder="Search" />
    </fieldset>

    <fieldset>
        <button type="submit">
            <span>Find</span>
        </button>
    </fieldset>
</form>
```

- Ничего
- Появится сообщение с текстом «journal»
-

Появится сообщение с текстом «query»
•
Появится сообщение с текстом «hidden»
Какой будет результат выполнения программного кода

```
abstract class Test {
   public $a;
   public $b;
   public $result;

   public function __construct($a, $b) {
      $this->a = $a;
      $this->b = $b;
   }

   abstract function calc();

   final public function getResult() {
      return $this->result;
   }
}
class Test1 extends Test {
   public function calc() {
      $this->result = $this->a + $this->b;
   }

}
class Test2 extends Test1 {
   public function calc() {
      parent::calc();
      $this->result *= 2;
   }
}
class Test3 extends Test {

}

$obj = new Test3(2,3);
$obj->calc();
echo $obj->getResult();
```

•
Ошибка выполнения
•
5
•
10
•
0
Какой будет результат выполнения программного кода

```
function test($a, $b) {
   return $a + $b;
}
$func = "test";
```

```
$args = array(2, 3);
echo call_user_func_array($$func, $args);
```
•
Ошибка выполнения
•
5
•
23
•
test23
Какие классы будут доступны в HTML после компиляции LESS файла
```
.percent (@value) when (@value >= 0) {
   .w@{value} {
      width: ~"{@value}%";
   }
   .h@{value} {
      height: ~"{@value}%";
   }
   .percent(@value - 5);
}
.percent (0) {}
.percent (100);
```
•
классы «w» и «h»
•
классы «w0», «w100», «h0», «h100»
•
Классы не будут доступны в HTML
•
Классы от «w0» до «w100» с интервалом 5, и классы от «h0» до «h100» с интервалом 5
Что будет показано на экране после выполнения следующего кода
```
var type = "test3";
switch(type) {
   case "test1":
      alert("test1");
   case "test2":
      alert("test2");
      break;
   case "test3":
      alert("test3");
   default:
      alert("default");
      break;
}
```
•
test3
•
test1, test2, test3
•
test3, default
•
default
Какой будет результат выполнения программного кода

```
abstract class Test {
   public $a;
   public $b;
   public $result;

   public function __construct($a, $b) {
      $this->a = $a;
      $this->b = $b;
   }

   abstract function calc();

   final public function getResult() {
      return $this->result;
   }
}
class Test1 extends Test {
   public function calc() {
      $this->result = $this->a + $this->b;
   }

}
class Test2 extends Test1 {
   public function calc() {
      parent::calc();
      $this->result *= 2;
   }

   public function getResult() {
      return $this->result;
   }
}

$obj = new Test2(2,3);
$obj->calc();
echo $obj->getResult();
```

•
Ошибка выполнения
•
5
•
10
•
0

Сколько записей будет иметь объект jQuery после выполнения следующей функции

```
$(".word:lt(2)").parents("li").find("b")
```

HTML

```
<ul>
   <li class="feedtab news active">
      <a href="/news/">
         <b class="tab word">Новости</b>
```

```
        </a>
    </li>
    <li class="feedtab notifications">
        <a href="/notifications/">
            <b class="tab word">Ответы</b>
        </a>
    </li>
    <li class="feedtab comments">
        <a href="/comments/">
            <b class="tab word">Комментарии</b>
        </a>
    </li>
</ul>
```

- 0
- 1
- 2
- 3

Выберите неправильный вариант оформления цикла FOREACH

- `foreach(array(1,2,3) as $value);`
- `foreach(array(1,2,3));`
- `foreach(array(1,2,3) as $value => $value);`
- `foreach(array(1,2,3) as &$value);`

Какие стили будут добавлены к элементу input после компиляции LESS файла

```
@defaultFontSize: 14px;

.padding(@value: 0px) {
    padding: @value;
}

.margin(@value: 0px) {
    margin: @value;
}

.font-size(@value: @defaultFontSize) {
    font-size: @value;
}

.search(@value: 0px) {
    .padding(@value);
    .margin(@value / 2);
    .font-size(@defaultFontSize * 1.5);
}

input {
    .search(10px);
}
```

•
.search(10px)
•
padding: 0px
margin: 0px
font-size: 14px;
•
padding: 10px
margin: 10px
font-size: 21px;
•
padding: 10px
margin: 5px
font-size: 21px;
Что произойдёт после нажатия на кнопку Find

```
$(function() {
    function Form(selector) {
        var self = this;
        this.selector = selector;
        this.field;

        this.setField = function(field) {
            this.field = field;
            return this;
        }

        this.onSubmit = function(){
            if (self.field) {
                alert(self.field.attr('name'));
            }
        };

        this.selector.submit(function() {
            self.onSubmit.call(self);
            return false;
        })
    }

    var search = new Form($("form"));
    search.setField(search.selector.find("input:last"));
    search.onSubmit = function() {
        if (this.field) {
            alert(this.field.attr('type'));
        }
    }})
```

HTML

```
<form>
    <input type="hidden" name="journal" />

    <fieldset>
        <input type="text" name="query" placeholder="Search" />
```

```
    </fieldset>

    <fieldset>
      <button type="submit">
        <span>Find</span>
      </button>
    </fieldset>
</form>
```

- Ничего
- Появится сообщение с текстом «text»
- Появится сообщение с текстом «query»
- Появится сообщение с текстом «hidden»

Выберите верное утверждение

```
ul {
  li {
    &.active {
      a {
        color: red;
        text-decoration: none;
      }
    }
    a {
      text-transform: uppercase;
    }
  }
}
```

- Все ссылки внутри списка будут красного цвета и не подчёркнуты
- Все ссылки внутри списка будут в верхнем регистре и не подчёркнуты
- Все ссылки внутри активных элементов списка будут красного цвета и не подчёркнуты
- Все ссылки внутри списка будут красного цвета, а внутри активных элементов также не подчёркнуты

Блок 3 (владеть)

Можно ли вызвать «.w100» после объявления классов

```
.percent (@value) when (@value >= 0) {
  .w@{value} {
    width: ~"{@value}%";
  }
  .h@{value} {
    height: ~"{@value}%";
  }
  .percent(@value - 5);
}
.percent (0) {}
.percent (100);
```

.w100;
•
Да.
•
Нет. Вызвать можно только внутри селектора.
•
Нет. Созданные классы доступны только в HTML.
•
Нет. Такой класс не был создан.
Что произойдёт после нажатия на кнопку Find

```
$(function() {
   function Form(selector) {
      var self = this;
      this.selector = selector;
      this.field;

      this.setField = function(field) {
         this.field = field;
         return this;
      }

      this.onSubmit = function(){
         if (self.field) {
            alert(self.field.attr('name'));
         }
      };

      this.selector.submit(function() {
         self.onSubmit.call(self);
         return false;
      })
   }

   var search = new Form($("form"));
   search.setField(search.selector.find("input:last"));
   search.onSubmit = function() {

   }})
```

HTML

```
<form>
   <input type="hidden" name="journal" />

   <fieldset>
      <input type="text" name="query" placeholder="Search" />
   </fieldset>

   <fieldset>
      <button type="submit">
         <span>Find</span>
      </button>
   </fieldset>
```

```
</form>
```

- Ничего
- Появится сообщение с текстом «text»
- Появится сообщение с текстом «query»
- Появится сообщение с текстом «hidden»

Выберите правильное объяснение конструкции &.active

```
ul {
   li {
      &.active {
         a {
            color: red;
            text-decoration: none;
         }
      }
      a {
         text-transform: uppercase;
      }
   }
}
```

- Все элементы «li» имеющие класс «active»
- Все элементы «a» имеющие класс «active»
- Чётные элементы, имеющие класс «active»
- Все элементы «ul» имеющие класс «active»

Сколько записей будет иметь объект jQuery после выполнения следующей функции

```
$(".word").parents('ul')
```

HTML

```
<ul>
   <li class="feedtab news active">
      <a href="/news/">
         <b class="tab word">Новости</b>
      </a>
   </li>
   <li class="feedtab notifications">
      <a href="/notifications/">
         <b class="tab word">Ответы</b>
      </a>
   </li>
   <li class="feedtab comments">
      <a href="/comments/">
         <b class="tab word">Комментарии</b>
      </a>
   </li>
</ul>
```

•
0
•
1
•
2
•
3
Выберите правильный вариант оформления функции
•
function test(param1, param2);
•
function test(param1, param2){};
•
function test($param1, $param2 = array()){};
•
function test($param1 = defaultValue, $param2 = "2"){};
Вопрос 5
Что будет показано на экране после выполнения следующего кода

```
var type = "test";
switch(type) {
   case "test1":
      alert("test1");
   case "test2":
      alert("test2");
      break;
   case "test3":
      alert("test3");
   default:
      alert("default");
      break;
}
```

•
test1
•
test1, test2, test3
•
test2
•
default

Какой будет результат выполнения программного кода

```
abstract class Test {
   public $a;
   public $b;
   public $result;

   public function __construct($a, $b) {
      $this->a = $a;
      $this->b = $b;
   }

   abstract function calc();
```

```
    final public function getResult() {
        return $this->result;
    }
}
class Test1 extends Test {
    public function calc() {
        $this->result = $this->a + $this->b;
    }

}
class Test2 extends Test1 {
    public function calc() {
        parent::calc();
        $this->result *= 2;
    }
}

class Test3 extends Test2 {
    public function __construct($a, $b) {
        $this->a = $a * 2;
        $this->b = $b * 2;
    }
}

$obj = new Test3(2,3);
$obj->calc();
echo $obj->getResult();
```

- Ошибка выполнения
- 5
- 10
- 20

Какой будет результат выполнения программного кода

```
function test($a, $b) {
    return $a + $b;
}
$func = "test";
$args = array("2", "3");
echo call_user_func_array($func, $args);
```

- Ошибка выполнения
- 5
- 23
- test23

Вопрос 5

Какие стили будут добавлены к элементу input после компиляции LESS файла

```
@defaultFontSize: 14px;

.padding(@value: 0px) {
   padding: @value;
   .margin(@value / 2);
}

.margin(@value: 0px) {
   margin: @value;
   .font-size(@defaultFontSize * 1.5);
}

.font-size(@value: @defaultFontSize) {
   font-size: @value;
}

.search(@value: 0px) {
   .padding(@value);
}

input {
   .search(20px);
}
```

•
.search(20px)
•
padding: 20px;
margin: 10px;
font-size: 21px;
•
padding: 0px
•
padding: 20px

Что будет показано на экране после выполнения следующего кода

```
var type = "test1";
switch(type) {
   case "test1":
      alert("test1");
   case "test2":
      alert("test2");
      break;
   case "test3":
      alert("test3");
   default:
      alert("default");
      break;
}
```

•
test1
•
test1, test2, test3
•
test1, test2

•
default
Какой будет результат выполнения программного кода

```
function test($a, $b) {
   return $a . $b;
}
$func = "test";
$args = array(2, 3);
echo call_user_func_array($func, $args);
```

•
Ошибка выполнения
•
5
•
23
•
test23
Сколько записей будет иметь объект jQuery после выполнения следующей функции

```
$(".word").parents("li:not(.active)")
```

HTML

```
<ul>
   <li class="feedtab news active">
      <a href="/news/">
         <b class="tab word">Новости</b>
      </a>
   </li>
   <li class="feedtab notifications">
      <a href="/notifications/">
         <b class="tab word">Ответы</b>
      </a>
   </li>
   <li class="feedtab comments">
      <a href="/comments/">
         <b class="tab word">Комментарии</b>
      </a>
   </li>
</ul>
```

•
0
•
1
•
2
•
3
Выберите правильный вариант оформления пустого класса
•
class Test() {};
•
class Test {};

•
class Test();
•
class Test;
Можно ли вызвать класс «w97» в HTML
Тест
LESS:
```
.percent (@value) when (@value >= 0) {
   .w@{value} {
      width: ~"{@value}%";
   }
   .h@{value} {
      height: ~"{@value}%";
   }
   .percent(@value - 5);
}
.percent (0) {}
.percent (100);
```
•
Да.
•
Нет. Созданные классы доступны только в CSS.
•
Нет. Созданный класс не применим к тегу DIV.
•
Нет. Такой класс не был создан.
Что произойдёт после нажатия на кнопку Find

```
$(function() {
   function Form(selector) {
      var self = this;
      this.selector = selector;
      this.field;

      this.setField = function(field) {
         this.field = field;
         return this;
      }

      this.onSubmit = function(){
         if (self.field) {
            alert(self.field.attr('name'));
         }
      };

      this.selector.submit(function() {
         self.onSubmit.call(self);
         return false;
      })
   }

   var search = new Form($("form"));
   search.setField(search.selector.find("input:last"));
})
```

HTML

```
<form>
   <input type="hidden" name="journal" />

   <fieldset>
      <input type="text" name="query" placeholder="Search" />
   </fieldset>

   <fieldset>
      <button type="submit">
         <span>Find</span>
      </button>
   </fieldset>
</form>
```

- Ничего
- Появится сообщение с текстом «text»
- Появится сообщение с текстом «query»
- Появится сообщение с текстом «hidden»

Выберите корректный для CSS селектор, описывающий ссылку в активном элементе

```
ul {
   li {
      &.active {
         a {
            color: red;
            text-decoration: none;
         }
      }
      a {
         text-transform: uppercase;
      }
   }
}
```

- ul.active a
- ul li &.active a
- li.active a
- li > a.active

Какие стили будут добавлены к элементу input после компиляции LESS файла

```
@defaultFontSize: 14px;

.padding(@value: 0px) {
   padding: @value;
}

.margin(@value: 0px) {
```

```
    margin: @value;
    .font-size(@defaultFontSize * 1.5);
}

.font-size(@value: @defaultFontSize) {
    font-size: @value;
}

.search(@value: 0px) {
    .padding(@value);
}

input {
    .search();
}
```

•

.search()

•

padding: 0px;
margin: 0px;
font-size: 14px;

•

padding: 0px;
margin: 0px;
font-size: 21px;

•

padding: 0px;

Какой будет результат выполнения программного кода

```
abstract class Test {
    public $a;
    public $b;
    public $result;

    public function __construct($a, $b) {
        $this->a = $a;
        $this->b = $b;
    }

    abstract function calc();

    final public function getResult() {
        return $this->result;
    }
}
class Test1 extends Test {
    public function calc() {
        $this->result = $this->a + $this->b;
    }

}
class Test2 extends Test1 {
    public function calc() {
        parent::calc();
```

```
        $this->result *= 2;
    }
}

class Test3 extends Test1 {
    public function __construct($a, $b) {
        $this->a = $a * 2;
        $this->b = $b * 2;
    }
}

$obj = new Test3(2,3);
$obj->calc();
echo $obj->getResult();
```

•
Ошибка выполнения
•
5
•
10
•
20

Выберите корректное описание для селектора «form fieldset input»

```
<form>
    <input type="hidden" name="journal" />

    <fieldset>
        <input type="text" name="query" placeholder="Search" />
    </fieldset>

    <fieldset>
        <button type="submit">
            <span>Find</span>
        </button>
    </fieldset>
</form>
```

•
Указывает на все теги input внутри формы
•
Указывает на все теги fieldset внутри формы
•
Указывает на тег input внутри тега fieldset
•
Указывает на теги input и fieldset

Выберите правильный вариант оформления наследования класса

•
class Test() extends Test2, Test3 {};
•
class Test extends Test2 extends Test3 {};
•
class Test extends Test2 {};
•
class Test() parents Test2,Test3;

Какой будет результат выполнения программного кода

```
class TestParent {
   public $result;

   public static function calc($a, $b) {
      return $a + $b;
   }

   public function getResult() {
      return $this->result;
   }
}
class Test extends TestParent {
   public function __construct($a, $b) {
      $this->result = self::calc($a, $b);
   }

   public static function calc($a, $b) {
      return $a * $b;
   }
}

$obj = new Test(2,3);
echo $obj->getResult();
```

- Ошибка выполнения
- 5
- 23
- 6

Выберите результат, который вернёт функция, после выполнения следующего кода

```
var func = function(param) {
   param = param || 0;
   param *= 2;
   return param;
}

func();
```

- 0
- 2
- undefined
- NaN

Выберите правильный вариант оформления пустого абстрактного класса

- abstr class Test() {};
- abstract class Test {};

•
abstr class Test();
•
abstract class Test;
Какой будет результат выполнения программного кода

```
class TestParent {
   public $result;

   public static function calc($a, $b) {
      return $a + $b;
   }

   public function getResult() {
      return $this->result;
   }
}
class Test extends TestParent {
   public function __construct($a, $b) {
      $this->result = parent::calc($a, $b);
   }

   public static function calc($a, $b) {
      return $a * $b;
   }
}

$obj = new Test(2,3);
echo $obj->getResult();
```

•
Ошибка выполнения
•
5
•
23
•
6
Выберите корректное описание для селектора «form [type=’hidden’]»

```
<form>
   <input type="hidden" name="journal" />

   <fieldset>
      <input type="text" name="query" placeholder="Search" />
   </fieldset>

   <fieldset>
      <button type="submit">
         <span>Find</span>
      </button>
   </fieldset>
</form>
```

•
Указывает на все теги input внутри формы
•

Указывает на первый тег input
•
Указывает на тег input внутри тега fieldset
•
Указывает на тег button внутри fieldset
Выберите результат, который вернёт функция, после выполнения следующего кода

```
var func = function(param) {
   param = param || 0;
   param *= 2;
   return param;
}

func(2);
```

•
0
•
4
•
undefined
•
NaN
Выберите результат, который вернёт функция, после выполнения следующего кода

```
var func = function(param) {
   param = param || 0;
   param *= 2;
   return param;
}

func("test");
```

•
0
•
test
•
undefined
•
NaN
Выберите корректное описание для селектора «form fieldset:first-child»

```
<form>
   <input type="hidden" name="journal" />

   <fieldset>
      <input type="text" name="query" placeholder="Search" />
   </fieldset>

   <fieldset>
      <button type="submit">
         <span>Find</span>
      </button>
   </fieldset>
</form>
```

•

Указывает на первый элемент внутри формы
•
Указывает на первый элемент fieldset внутри формы
•
Указывает на первый элемент внутри тега fieldset
•
Указывает на последний элемент fieldset внутри формы
Какой будет результат выполнения программного кода

```
class TestParent {
   public $result;

   public static function calc($a, $b) {
      return $a + $b;
   }

   public function getResult() {
      return $this->result;
   }
}
class Test extends TestParent {
   public function __construct($a, $b) {
      $this->result = self::calc($a, $b);
   }

   public static function calc($a, $b) {
      $this->result = $a * $b;
      return $this->result;
   }
}

$obj = new Test(2,3);
echo $obj->getResult();
```

•
Ошибка выполнения
•
5
•
23
•
6
Выберите правильный вариант оформления интерфейса
•
interface iTest {};
•
public interface iTest {};
•
interface iTest();
•
class interface iTest;

Методические материалы, характеризующие процедуры оценивания

На основе типовых заданий из раздела 6.3. программным комплексом информационно-образовательного портала МИ ВлГУ формируются в автоматическом режиме тестовые задания для студентов: три вопроса из блока 1, три вопроса из блока 2 и четыре вопроса из блока 3. Программный комплекс формирует индивидуальные задания для каждого зарегистрированного в системе студента и устанавливает время прохождения тестирования. Результатом тестирования является процент правильных ответов, с учетом индивидуального семестрового рейтинга студента формируется экзаменационная оценка.

Максимальная сумма баллов, набираемая студентом по дисциплине равна 100.

Оценка в баллах	Оценка по шкале	Обоснование	***Уровень сформированности компетенций***
Более 80	«Отлично»	Содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному	***Высокий уровень***
66-80	«Хорошо»	Содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками	***Продвинутый уровень***
50-65	«Удовлетворительно»	Содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки	***Пороговый уровень***
Менее 50	«Неудовлетворительно»	Содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки	***Компетенции не сформированы***

3. Задания в тестовой форме по дисциплине

Примеры заданий:

// question: 0 name: Switch category to $course$/top/По умолчанию для IntTex-2
$CATEGORY: $course$/top/По умолчанию для IntTex-2

// question: 0 name: Switch category to $course$/top/По умолчанию для IntTex-2/PHP
$CATEGORY: $course$/top/По умолчанию для IntTex-2/PHP

// question: 0 name: Switch category to $course$/top/По умолчанию для IntTex-2/PHP/PHP средне
$CATEGORY: $course$/top/По умолчанию для IntTex-2/PHP/PHP средне

// question: 36926 name: Вопрос 1
::Вопрос 1::[html] Какой будет результат выполнения программного кода \n \n $var1 \= "World"; \n $hello \= "var1"; \n echo $$hello; {
~ var1
= World
~ hello
~ hello World
}

// question: 36925 name: Вопрос 10
::Вопрос 10::[html] Какой будет результат выполнения программного кода \n \n class TestParent \{ \n public $result; \n \n public static function calc($a, $b) \{ \n return $a + $b; \n \} \n \n public function getResult() \{ \n return $this->result; \n \} \n \} \n class Test extends TestParent \{ \n public function __construct($a, $b) \{ \n $this->result \= parent\:\:calc($a, $b); \n \} \n \n public static function calc($a, $b) \{ \n $this->result \= $a * $b; \n return $this->result; \n \} \n \} \n \n $obj \= new Test(2,3); \n echo $obj->getResult(); {
~ Ошибка выполнения
= 5
~ 23
~ 6
}

// question: 36924 name: Вопрос 11
::Вопрос 11::[html] Какой будет результат выполнения программного кода \n \n class TestParent \{ \n public $result; \n \n public static function calc($a, $b) \{ \n return $a + $b; \n \} \n \n public function getResult() \{ \n return $this->result * 2; \n \} \n \} \n class Test extends TestParent \{ \n public function __construct($a, $b) \{ \n $this->result \= self\:\:calc($a, $b); \n \} \n \n public static function calc($a, $b) \{ \n return $a * $b; \n \} \n \n public function getResult() \{ \n return $this->result; \n \} \n \} \n \n $obj \= new Test(2,3); \n echo $obj->getResult(); {
~ Ошибка выполнения

~ 5
~ 12
= 6
}

// question: 36923 name: Вопрос 12
::Вопрос 12::[html] Какой будет результат выполнения программного кода \n \n class TestParent \{ \n public $result; \n \n public static function calc($a, $b) \{ \n return $a + $b; \n \} \n \n public function getResult() \{ \n return $this->result * 2; \n \} \n \} \n class Test extends TestParent \{ \n public function __construct($a, $b) \{ \n $this->result \= parent\:\:calc($a, $b); \n \} \n \n public static function calc($a, $b) \{ \n return $a * $b; \n \} \n \n public function getResult() \{ \n return $this->result; \n \} \n \} \n \n $obj \= new Test(2,3); \n echo $obj->getResult(); {
~ Ошибка выполнения
= 5
~ 10
~ 6
}

// question: 36922 name: Вопрос 2
::Вопрос 2::[html] Какой будет результат выполнения программного кода \n \n $var1 \= "hello"; \n $hello \= "var1"; \n $hello_world \= "Hello world"; \n echo $\{$$hello."_world"\}; {
~ var1
~ World
~ hello
= hello World
}

// question: 36921 name: Вопрос 3
::Вопрос 3::[html] Какой будет результат выполнения программного кода \n \n function test($a, $b) \{ \n return $a + $b; \n \} \n $func \= "test"; \n $args \= array(2, 3); \n echo call_user_func_array($func, $args); {
~ Ошибка выполнения
= 5
~ 23
~ test23
}

// question: 36920 name: Вопрос 4
::Вопрос 4::[html] Какой будет результат выполнения программного кода \n \n function test($a, $b) \{ \n return $a + $b; \n \} \n $func \= "test"; \n $args \= array(2, 3);</p>\n <p>echo call_user_func_array($$func, $args); {
= Ошибка выполнения
~ 5
~ 23
~ test23

```
}

// question: 36919  name: Вопрос 5
::Вопрос 5::[html]<p>Какой будет результат выполнения программного кода</p>\n <p> </p>\n <p>function test($a, $b) \{</p>\n <p>    return $a + $b;</p>\n <p>\}</p>\n <p>$func \= "test";</p>\n <p>$args \= array("2", "3");</p>\n <p>echo call_user_func_array($func, $args);</p>{
    ~<p>Ошибка выполнения</p>
    =<p>5</p>
    ~<p>23</p>
    ~<p>test23</p>
}

// question: 36918  name: Вопрос 6
::Вопрос 6::[html]<p>Какой будет результат выполнения программного кода</p>\n <p> </p>\n <p>function test($a, $b) \{</p>\n <p>    return $a . $b;</p>\n <p>\}</p>\n <p>$func \= "test";</p>\n <p>$args \= array(2, 3);</p>\n <p>echo call_user_func_array($func, $args);</p>{
    ~<p>Ошибка выполнения</p>
    ~<p>5</p>
    =<p>23</p>
    ~<p>test23</p>
}

// question: 36917  name: Вопрос 7
::Вопрос 7::[html]<p>Какой будет результат выполнения программного кода</p>\n <p> </p>\n <p>class TestParent \{</p>\n <p>    public $result;</p>\n <p> </p>\n <p>    public static function calc($a, $b) \{</p>\n <p>        return $a + $b;</p>\n <p>    \}</p>\n <p> </p>\n <p>    public function getResult() \{</p>\n <p>        return $this->result;</p>\n <p>    \}</p>\n <p>\}</p>\n <p>class Test extends TestParent \{</p>\n <p>    public function __construct($a, $b) \{</p>\n <p>        $this->result \= self\:\:calc($a, $b);  </p>\n <p>    \}</p>\n <p> </p>\n <p>    public static function calc($a, $b) \{</p>\n <p>        return $a * $b;</p>\n <p>    \}</p>\n <p>\}</p>\n <p> </p>\n <p>$obj \= new Test(2,3);</p>\n <p>echo $obj->getResult();</p>{
    ~<p>Ошибка выполнения</p>
    ~<p>5</p>
    ~<p>23</p>
    =<p>6</p>
}

// question: 36916  name: Вопрос 8
::Вопрос 8::[html]<p>Какой будет результат выполнения программного кода</p>\n <p> </p>\n <p>class TestParent \{</p>\n <p>    public $result;</p>\n <p> </p>\n <p>    public static function calc($a, $b) \{</p>\n <p>        return $a + $b;</p>\n <p>    \}</p>\n <p> </p>\n <p>    public function getResult() \{</p>\n <p>        return $this->result;</p>\n <p>    \}</p>\n <p>\}</p>\n <p>class Test extends TestParent \{</p>\n <p>    public function __construct($a, $b) \{</p>\n <p>        $this->result \= parent\:\:calc($a, $b);  </p>\n <p>    \}</p>\n <p> </p>\n <p>    public static function calc($a, $b) \{</p>\n <p>        return $a * $b;</p>\n <p>    \}</p>\n <p>\}</p>\n <p> </p>\n <p>$obj \= new Test(2,3);</p>\n <p>echo $obj->getResult();</p>{
    ~<p>Ошибка выполнения</p>
    =<p>5</p>
    ~<p>23</p>
    ~<p>6</p>
```

}

// question: 36915 name: Вопрос 9

::Вопрос 9::[html] Какой будет результат выполнения программного кода \n \n class TestParent \{ \n public $result; \n \n public static function calc($a, $b) \{ \n return $a + $b; \n \} \n \n public function getResult() \{ \n return $this->result; \n \} \n \} \n class Test extends TestParent \{ \n public function __construct($a, $b) \{ \n $this->result \= self\:\:calc($a, $b); \n \} \n \n public static function calc($a, $b) \{ \n $this->result \= $a * $b; \n return $this->result; \n \} \n \} \n \n $obj \= new Test(2,3); \n echo $obj->getResult(); {

= Ошибка выполнения

~ 5

~ 23

~ 6

}

// question: 0 name: Switch category to $course$/top/По умолчанию для IntTex-2/LESS

$CATEGORY: $course$/top/По умолчанию для IntTex-2/LESS

// question: 0 name: Switch category to $course$/top/По умолчанию для IntTex-2/LESS/LESS легко

$CATEGORY: $course$/top/По умолчанию для IntTex-2/LESS/LESS легко

// question: 36941 name: Вопрос 1

::Вопрос 1::[html] Укажите правильный селектор для всех элементов с классом «tab» \n <ul> \n <li class\="feedtab news active"> \n <a href\="/news/"> \n <b class\="tab word">Новости \n \n \n <li class\="feedtab notifications"> \n <a href\="/notifications/"> \n <b class\="tab word">Ответы \n \n \n <li class\="feedtab comments"> \n <a href\="/comments/"> \n <b class\="tab word">Комментарии \n \n \n {

~ .tab .word

~ b .tab.word

= .feedtab .tab.word

~ .feedtab > .tab .word

}

// question: 36940 name: Вопрос 10

::Вопрос 10::[html] Выберите корректный альтернативный селектор, описывающий строку с классом «.level1» \n .block.pricelist .table > .header .column \{ \n font-weight\: bold; \n \} \n .block.pricelist .table > .content .row.level1 .column\:first-child \{ \n padding-left\: 15px; \n \} \n .block.pricelist .table > .content .row.level2 .column\:first-child \{ \n padding-left\: 30px; \n \} {

~ .block .table.content .level1

~ .block .table > .row.level1

~ .block > .content > .level1

= .block .table .row.level1
}

// question: 36939 name: Вопрос 11
::Вопрос 11::[html] Выберите корректный альтернативный селектор, описывающий только колонку внутри заголовка таблицы \n .block.pricelist .table > .header .column \{ \n font-weight\: bold; \n \} \n .block.pricelist .table > .content .row.level1 .column\:first-child \{ \n padding-left\: 15px; \n \} \n .block.pricelist .table > .content .row.level2 .column\:first-child \{ \n padding-left\: 30px; \n \} {
~ .table > .column
~ .block.table .column
~ .table .column
= .block .table .header .column
}

// question: 36938 name: Вопрос 12
::Вопрос 12::[html] Выберите корректный альтернативный селектор, описывающий все колонки внутри таблицы \n .block.pricelist .table > .header .column \{ \n font-weight\: bold; \n \} \n .block.pricelist .table > .content .row.level1 .column\:first-child \{ \n padding-left\: 15px; \n \} \n .block.pricelist .table > .content .row.level2 .column\:first-child \{ \n padding-left\: 30px; \n \} {
= .table .column
~ .header .column
~ .table.column
~ .table > .column
}

// question: 36937 name: Вопрос 2
::Вопрос 2::[html] Укажите правильный селектор для активного элемента \n <ul> \n <li class\="feedtab news active"> \n <a href\="/news/"> \n <b class\="tab word">Новости \n \n \n <li class\="feedtab notifications"> \n <a href\="/notifications/"> \n <b class\="tab word">Ответы \n \n \n <li class\="feedtab comments"> \n <a href\="/comments/"> \n <b class\="tab word">Комментарии \n \n \n {
~ .ul li.news.active
= ul > .active
~ li .feedtab.active
~ .li.active
}

// question: 36936 name: Вопрос 3
::Вопрос 3::[html] Укажите правильный селектор для ссылки с адресом /news/ \n <ul> \n <li class\="feedtab news active"> \n <a href\="/news/"> \n <b class\="tab word">Новости \n \n \n <li class\="feedtab notifications"> \n <a href\="/notifications/"> \n <b class\="tab word">Ответы \n \n \n <li class\="feedtab comments"> \n <a href\="/comments/"> \n <b class\="tab word">Комментарии \n \n \n {
~ .feedtab[href\=’/news/’]

~ li a [href\=’/news/’]
= ul a[href\=’/news/’]
~ ul > [href\=’/news/’]
}

// question: 36935 name: Вопрос 4
::Вопрос 4::[html] Выберите верное утверждение \n ul \{ \n li \{ \n &.active \{ \n a \{ \n color\: red; \n text-decoration\: none; \n \} \n \} \n a \{ \n text-transform\: uppercase; \n \} \n \} \n \} {
~ Все ссылки внутри списка будут красного цвета и не подчёркнуты
~ Все ссылки внутри списка будут в верхнем регистре и не подчёркнуты
= Все ссылки внутри активных элементов списка будут красного цвета и не подчёркнуты
~ Все ссылки внутри списка будут красного цвета, а внутри активных элементов также не подчёркнуты
}

// question: 36934 name: Вопрос 5
::Вопрос 5::[html] Выберите правильное объяснение конструкции &.active \n ul \{ \n li \{ \n &.active \{ \n a \{ \n color\: red; \n text-decoration\: none; \n \} \n \} \n a \{ \n text-transform\: uppercase; \n \} \n \} \n \} {
= Все элементы «li» имеющие класс «active»
~ Все элементы «a» имеющие класс «active»
~ Чётные элементы, имеющие класс «active»
~ Все элементы «ul» имеющие класс «active»
}

// question: 36933 name: Вопрос 6
::Вопрос 6::[html] Выберите корректный для CSS селектор, описывающий ссылку в активном элементе \n ul \{ \n li \{ \n &.active \{ \n a \{ \n color\: red; \n text-decoration\: none; \n \} \n \} \n a \{ \n text-transform\: uppercase; \n \} \n \} \n \} {
~ ul.active a
~ ul li &.active a
= li.active a
~ li > a.active
}

// question: 36932 name: Вопрос 7
::Вопрос 7::[html] Выберите корректное описание для селектора «form fieldset input» \n <form> \n <input type\="hidden" name\="journal" /> \n \n <fieldset> \n <input type\="text" name\="query" placeholder\="Search" /> \n </fieldset> \n \n <fieldset> \n <button type\="submit"> \n Find \n </button> \n </fieldset> \n </form> {

~ Указывает на все теги input внутри формы
~ Указывает на все теги fieldset внутри формы
= Указывает на тег input внутри тега fieldset
~ Указывает на теги input и fieldset
}

// question: 36931 name: Вопрос 8
::Вопрос 8::[html] Выберите корректное описание для селектора «form [type\='hidden']» \n <form> \n <input type\="hidden" name\="journal" /> \n \n <fieldset> \n <input type\="text" name\="query" placeholder\="Search" /> \n </fieldset> \n \n <fieldset> \n <button type\="submit"> \n Find \n </button> \n </fieldset> \n </form> {
~ Указывает на все теги input внутри формы
= Указывает на первый тег input
~ Указывает на тег input внутри тега fieldset
~ Указывает на тег button внутри fieldset
}

// question: 36930 name: Вопрос 9
::Вопрос 9::[html] Выберите корректное описание для селектора «form fieldset\:first-child» \n <form> \n <input type\="hidden" name\="journal" /> \n \n <fieldset> \n <input type\="text" name\="query" placeholder\="Search" /> \n </fieldset> \n \n <fieldset> \n <button type\="submit"> \n Find \n </button> \n </fieldset> \n </form> {
~ Указывает на первый элемент внутри формы
= Указывает на первый элемент fieldset внутри формы
~ Указывает на первый элемент внутри тега fieldset
~ Указывает на последний элемент fieldset внутри формы
}

// question: 0 name: Switch category to $course$/top/По умолчанию для IntTex-2/JS
$CATEGORY: $course$/top/По умолчанию для IntTex-2/JS

// question: 0 name: Switch category to $course$/top/По умолчанию для IntTex-2/JS/JS средне
$CATEGORY: $course$/top/По умолчанию для IntTex-2/JS/JS средне

// question: 36949 name: Вопрос 1
::Вопрос 1::[html] Сколько записей будет иметь объект jQuery после выполнения следующей функции \n \n $(".feedtab.active").parents().find(".tab") \n \n HTML \n \n <ul> \n <li class\="feedtab news active"> \n <a href\="/news/"> \n <b class\="tab word">Новости \n \n \n <li class\="feedtab notifications"> \n <a href\="/notifications/"> \n <b class\="tab word">Ответы \n \n \n <li class\="feedtab comments"> \n <a href\="/comments/"> \n <b class\="tab word">Комментарии \n \n \n {

~ 0
~ 1
~ 2
= 3
}

// question: 36948 name: Вопрос 2
::Вопрос 2::[html] Сколько записей будет иметь объект jQuery после выполнения следующей функции \n \n $("ul a\:lt(2)") \n \n HTML \n \n <ul> \n <li class\="feedtab news active"> \n <a href\="/news/"> \n <b class\="tab word">Новости \n \n \n <li class\="feedtab notifications"> \n <a href\="/notifications/"> \n <b class\="tab word">Ответы \n \n \n <li class\="feedtab comments"> \n <a href\="/comments/"> \n <b class\="tab word">Комментарии \n \n \n {
~ 0
~ 1
= 2
~ 3
}

// question: 36947 name: Вопрос 3
::Вопрос 3::[html] Сколько записей будет иметь объект jQuery после выполнения следующей функции \n \n $("li.news a[href\='/comments/'] b").parents() \n \n HTML \n \n <ul> \n <li class\="feedtab news active"> \n <a href\="/news/"> \n <b class\="tab word">Новости \n \n \n <li class\="feedtab notifications"> \n <a href\="/notifications/"> \n <b class\="tab word">Ответы \n \n \n <li class\="feedtab comments"> \n <a href\="/comments/"> \n <b class\="tab word">Комментарии \n \n \n {
= 0
~ 1
~ 2
~ 3
}

// question: 36946 name: Вопрос 4
::Вопрос 4::[html] Сколько записей будет иметь объект jQuery после выполнения следующей функции \n \n $(".word\:lt(2)").parents("li").find("b") \n \n HTML \n \n <ul> \n <li class\="feedtab news active"> \n <a href\="/news/"> \n <b class\="tab word">Новости \n \n \n <li class\="feedtab notifications"> \n <a href\="/notifications/"> \n <b class\="tab word">Ответы \n \n \n <li class\="feedtab comments"> \n <a href\="/comments/"> \n <b class\="tab word">Комментарии \n \n \n {
~ 0
~ 1
= 2

~ 3
}

// question: 36945 name: Вопрос 5
::Вопрос 5::[html] Сколько записей будет иметь объект jQuery после выполнения следующей функции \n \n $(".word").parents('ul') \n \n HTML \n \n <ul> \n <li class\="feedtab news active"> \n <a href\="/news/"> \n <b class\="tab word">Новости \n \n \n <li class\="feedtab notifications"> \n <a href\="/notifications/"> \n <b class\="tab word">Ответы \n \n \n <li class\="feedtab comments"> \n <a href\="/comments/"> \n <b class\="tab word">Комментарии \n \n \n {
~ 0
= 1
~ 2
~ 3
}

// question: 36944 name: Вопрос 6
::Вопрос 6::[html] Сколько записей будет иметь объект jQuery после выполнения следующей функции \n \n $(".word").parents("li\:not(.active)") \n \n HTML \n \n <ul> \n <li class\="feedtab news active"> \n <a href\="/news/"> \n <b class\="tab word">Новости \n \n \n <li class\="feedtab notifications"> \n <a href\="/notifications/"> \n <b class\="tab word">Ответы \n \n \n <li class\="feedtab comments"> \n <a href\="/comments/"> \n <b class\="tab word">Комментарии \n \n \n {
~ 0
~ 1
= 2
~ 3
}

// question: 0 name: Switch category to $course$/top/По умолчанию для IntTex-2/JS/JS сложно
$CATEGORY: $course$/top/По умолчанию для IntTex-2/JS/JS сложно

// question: 36957 name: Вопрос 1
::Вопрос 1::[html] Что произойдёт после нажатия на кнопку Find \n \n $(function() \{</p>\n <p> function Form(selector) \{</p>\n <p> var self \= this;</p>\n <p> this.selector \= selector;</p>\n <p> this.field;</p>\n <p> </p>\n <p> this.setField \= function(field) \{</p>\n <p> this.field \= field;</p>\n <p> return this;</p>\n <p> \}</p>\n <p> </p>\n <p> this.onSubmit \= function()\{</p>\n <p> if (self.field) \{</p>\n <p> alert(self.field.attr('name'));</p>\n <p> \}</p>\n <p> \};</p>\n <p> </p>\n <p> this.selector.submit(function() \{</p>\n <p> self.onSubmit.call(self);</p>\n <p> return false;</p>\n <p> \})</p>\n <p> \} </p>\n <p> </p>\n <p> var search \= new Form($("form")); \n search.setField(search.selector.find(">input")); \n \}) \n \n HTML \n

 \n <form> \n   <input type\="hidden" name\="journal" /> \n   \n   <fieldset> \n   <input type\="text" name\="query" placeholder\="Search" /> \n   </fieldset> \n \n   <fieldset> \n   <button type\="submit"> \n   Find \n   </button> \n   </fieldset> \n </form> {
~ Ничего
= Появится сообщение с текстом «journal»
~ Появится сообщение с текстом «query»
~ Появится сообщение с текстом «journal, query»
}

// question: 36956 name: Вопрос 2
::Вопрос 2::[html] Что произойдёт после нажатия на кнопку Find \n \n $(function() \{</p>\n <p> function Form(selector) \{</p>\n <p> var self \= this;</p>\n <p> this.selector \= selector;</p>\n <p> this.field;</p>\n <p> </p>\n <p> this.setField \= function(field) \{</p>\n <p> this.field \= field;</p>\n <p> return this;</p>\n <p> \}</p>\n <p> </p>\n <p> this.onSubmit \= function()\{</p>\n <p> if (self.field) \{</p>\n <p> alert(self.field.attr('name'));</p>\n <p> \}</p>\n <p> \};</p>\n <p> </p>\n <p> this.selector.submit(function() \{</p>\n <p> self.onSubmit.call(self);</p>\n <p> return false;</p>\n <p> \})</p>\n <p> \} </p>\n <p> </p>\n <p> var search \= new Form($("form")); \n   search.setField(search.selector.find(">input")); \n   search.onSubmit \= function() \{ \n   \n   \} \n \}) \n \n HTML \n   \n <form> \n   <input type\="hidden" name\="journal" /> \n   \n   <fieldset> \n   <input type\="text" name\="query" placeholder\="Search" /> \n   </fieldset> \n \n   <fieldset> \n   <button type\="submit"> \n   Find \n   </button> \n   </fieldset> \n </form> {
= Ничего
~ Появится сообщение с текстом «journal»
~ Появится сообщение с текстом «query»
~ Появится сообщение с текстом «journal, query»
}

// question: 36955 name: Вопрос 3
::Вопрос 3::[html] Что произойдёт после нажатия на кнопку Find \n \n $(function() \{</p>\n <p> function Form(selector) \{</p>\n <p> var self \= this;</p>\n <p> this.selector \= selector;</p>\n <p> this.field;</p>\n <p> </p>\n <p> this.setField \= function(field) \{</p>\n <p> this.field \= field;</p>\n <p> return this;</p>\n <p> \}</p>\n <p> </p>\n <p> this.onSubmit \= function()\{</p>\n <p> if (self.field) \{</p>\n <p> alert(self.field.attr('name'));</p>\n <p> \}</p>\n <p> \};</p>\n <p> </p>\n <p> this.selector.submit(function() \{</p>\n <p> self.onSubmit.call(self);</p>\n <p> return false;</p>\n <p> \})</p>\n <p> \} </p>\n <p> </p>\n <p> var search \= new Form($("form")); \n   search.setField(search.selector.find(">input")); \n   search.onSubmit \= function() \{ \n   if (this.field) \{ \n   alert(this.field.attr('type')); \n   \} \n   \}\}) \n \n HTML \n   \n <form> \n   <input type\="hidden" name\="journal" /> \n \n   <fieldset> \n   <input type\="text" name\="query" placeholder\="Search" /> \n   </fieldset> \n \n   <fieldset> \n   <button type\="submit"> \n   Find \n   </button> \n   </fieldset> \n </form> {
~ Ничего

~ Появится сообщение с текстом «journal»
~ Появится сообщение с текстом «query»
= Появится сообщение с текстом «hidden»
}

// question: 36954 name: Вопрос 4
::Вопрос 4::[html] Что произойдёт после нажатия на кнопку Find \n \n $(function() \{</p>\n <p> function Form(selector) \{</p>\n <p> var self \= this;</p>\n <p> this.selector \= selector;</p>\n <p> this.field;</p>\n <p> </p>\n <p> this.setField \= function(field) \{</p>\n <p> this.field \= field;</p>\n <p> return this;</p>\n <p> \}</p>\n <p> </p>\n <p> this.onSubmit \= function()\{</p>\n <p> if (self.field) \{</p>\n <p> alert(self.field.attr('name'));</p>\n <p> \}</p>\n <p> \};</p>\n <p> </p>\n <p> this.selector.submit(function() \{</p>\n <p> self.onSubmit.call(self);</p>\n <p> return false;</p>\n <p> \})</p>\n <p> \} </p>\n <p> </p>\n <p> var search \= new Form($("form")); \n search.setField(search.selector.find("input\:last")); \n search.onSubmit \= function() \{ \n if (this.field) \{ \n alert(this.field.attr('type')); \n \} \n \}\}) \n \n HTML \n \n <form> \n <input type\="hidden" name\="journal" /> \n \n <fieldset> \n <input type\="text" name\="query" placeholder\="Search" /> \n </fieldset> \n \n <fieldset> \n <button type\="submit"> \n Find \n </button> \n </fieldset> \n </form> {
~ Ничего
= Появится сообщение с текстом «text»
~ Появится сообщение с текстом «query»
~ Появится сообщение с текстом «hidden»
}

// question: 36953 name: Вопрос 5
::Вопрос 5::[html] Что произойдёт после нажатия на кнопку Find \n \n $(function() \{</p>\n <p> function Form(selector) \{</p>\n <p> var self \= this;</p>\n <p> this.selector \= selector;</p>\n <p> this.field;</p>\n <p> </p>\n <p> this.setField \= function(field) \{</p>\n <p> this.field \= field;</p>\n <p> return this;</p>\n <p> \}</p>\n <p> </p>\n <p> this.onSubmit \= function()\{</p>\n <p> if (self.field) \{</p>\n <p> alert(self.field.attr('name'));</p>\n <p> \}</p>\n <p> \};</p>\n <p> </p>\n <p> this.selector.submit(function() \{</p>\n <p> self.onSubmit.call(self);</p>\n <p> return false;</p>\n <p> \})</p>\n <p> \} </p>\n <p> </p>\n <p> var search \= new Form($("form")); \n search.setField(search.selector.find("input\:last")); \n search.onSubmit \= function() \{ \n \n \}\}) \n \n HTML \n \n <form> \n <input type\="hidden" name\="journal" /> \n \n <fieldset> \n <input type\="text" name\="query" placeholder\="Search" /> \n </fieldset> \n \n <fieldset> \n <button type\="submit"> \n Find \n </button> \n </fieldset> \n </form> {
= Ничего
~ Появится сообщение с текстом «text»
~ Появится сообщение с текстом «query»
~ Появится сообщение с текстом «hidden»
}

// question: 36952 name: Вопрос 6

::Вопрос 6::[html] Что произойдёт после нажатия на кнопку Find \n \n $(function() \{</p>\n <p> function Form(selector) \{</p>\n <p> var self \= this;</p>\n <p> this.selector \= selector;</p>\n <p> this.field;</p>\n <p> </p>\n <p> this.setField \= function(field) \{</p>\n <p> this.field \= field;</p>\n <p> return this;</p>\n <p> \}</p>\n <p> </p>\n <p> this.onSubmit \= function()\{</p>\n <p> if (self.field) \{</p>\n <p> alert(self.field.attr('name'));</p>\n <p> \}</p>\n <p> \};</p>\n <p> </p>\n <p> this.selector.submit(function() \{</p>\n <p> self.onSubmit.call(self);</p>\n <p> return false;</p>\n <p> \})</p>\n <p> \} </p>\n <p> </p>\n <p> var search \= new Form($("form")); \n search.setField(search.selector.find("input\:last")); \n \}) \n \n HTML \n \n <form> \n <input type\="hidden" name\="journal" /> \n \n <fieldset> \n <input type\="text" name\="query" placeholder\="Search" /> \n </fieldset> \n \n <fieldset> \n <button type\="submit"> \n Find \n </button> \n </fieldset> \n </form> {

~ Ничего

~ Появится сообщение с текстом «text»

= Появится сообщение с текстом «query»

~ Появится сообщение с текстом «hidden»

}

// question: 0 name: Switch category to $course$/top/По умолчанию для IntTex-2/PHP/PHP сложно

$CATEGORY: $course$/top/По умолчанию для IntTex-2/PHP/PHP сложно

// question: 36964 name: Вопрос 1

::Вопрос 1::[html] Какой будет результат выполнения программного кода \n \n abstract class Test \{ \n public $a; \n public $b; \n public $result; \n \n public function __construct($a, $b) \{ \n $this->a \= $a; \n $this->b \= $b; \n \} \n \n abstract function calc(); \n \n final public function getResult() \{ \n return $this->result; \n \} \n \} \n class Test1 extends Test \{ \n public function calc() \{ \n $this->result \= $this->a + $this->b; \n \} \n \n \} \n class Test2 extends Test1 \{ \n public function calc() \{ \n parent\:\:calc(); \n $this->result *\= 2; \n \} \n \} \n \n $obj \= new Test1(2,3); \n $obj->calc(); \n echo $obj->getResult(); {

~ Ошибка выполнения

= 5

~ 10

~ 0

}

// question: 36963 name: Вопрос 2

::Вопрос 2::[html] Какой будет результат выполнения программного кода \n \n abstract class Test \{ \n public $a; \n public $b; \n public $result; \n \n public function __construct($a, $b) \{ \n $this->a \= $a; \n $this->b \= $b; \n \} \n \n abstract function calc(); \n \n final public function getResult() \{ \n return $this->result; \n \} \n \} \n class Test1 extends Test \{ \n public

function calc() \{ \n $this->result \= $this->a + $this->b; \n   \} \n \n \} \n class Test2 extends Test1 \{ \n   public function calc() \{ \n   parent\:\:calc(); \n $this->result *\= 2; \n   \} \n \} \n \n $obj \= new Test2(2,3); \n $obj->calc(); \n echo $obj->getResult(); {
~ Ошибка выполнения
~ 5
= 10
~ 0
}

// question: 36962 name: Вопрос 3
::Вопрос 3::[html] Какой будет результат выполнения программного кода \n \n abstract class Test \{ \n   public $a; \n   public $b; \n   public $result; \n   \n   public function __construct($a, $b) \{ \n $this->a \= $a; \n $this->b \= $b; \n   \} \n   \n   abstract function calc(); \n   \n   final public function getResult() \{ \n   return $this->result; \n   \} \n \} \n class Test1 extends Test \{ \n   public function calc() \{ \n $this->result \= $this->a + $this->b; \n   \} \n \n \} \n class Test2 extends Test1 \{ \n   public function calc() \{ \n   parent\:\:calc(); \n $this->result *\= 2; \n   \} \n \} \n class Test3 extends Test \{ \n \n \} \n \n $obj \= new Test3(2,3); \n $obj->calc(); \n echo $obj->getResult(); {
= Ошибка выполнения
~ 5
~ 10
~ 0
}

// question: 36961 name: Вопрос 4
::Вопрос 4::[html] Какой будет результат выполнения программного кода \n \n abstract class Test \{ \n   public $a; \n   public $b; \n   public $result; \n   \n   public function __construct($a, $b) \{ \n $this->a \= $a; \n $this->b \= $b; \n   \} \n   \n   abstract function calc(); \n   \n   final public function getResult() \{ \n   return $this->result; \n   \} \n \} \n class Test1 extends Test \{ \n   public function calc() \{ \n $this->result \= $this->a + $this->b; \n   \} \n \n \} \n class Test2 extends Test1 \{ \n   public function calc() \{ \n   parent\:\:calc(); \n $this->result *\= 2; \n   \} \n \n   public function getResult() \{ \n   return $this->result; \n   \} \n \} \n \n $obj \= new Test2(2,3); \n $obj->calc(); \n echo $obj->getResult(); {
= Ошибка выполнения
~ 5
~ 10
~ 0
}

// question: 36960 name: Вопрос 5
::Вопрос 5::[html] Какой будет результат выполнения программного кода \n \n abstract class Test \{ \n   public $a; \n   public $b; \n

public $result; \n \n public function __construct($a, $b) \{ \n $this->a \= $a; \n $this->b \= $b; \n \} \n \n abstract function calc(); \n \n final public function getResult() \{ \n return $this->result; \n \} \n \} \n class Test1 extends Test \{ \n public function calc() \{ \n $this->result \= $this->a + $this->b; \n \} \n \n \} \n class Test2 extends Test1 \{ \n public function calc() \{ \n parent\:\:calc(); \n $this->result *\= 2; \n \} \n \} \n \n class Test3 extends Test2 \{ \n public function __construct($a, $b) \{ \n $this->a \= $a * 2; \n $this->b \= $b * 2; \n \} \n \} \n \n $obj \= new Test3(2,3); \n $obj->calc(); \n echo $obj->getResult(); {
~ Ошибка выполнения
~ 5
~ 10
= 20
}

// question: 36959 name: Вопрос 6
::Вопрос 6::[html] Какой будет результат выполнения программного кода \n \n abstract class Test \{ \n public $a; \n public $b; \n public $result; \n \n public function __construct($a, $b) \{ \n $this->a \= $a; \n $this->b \= $b; \n \} \n \n abstract function calc(); \n \n final public function getResult() \{ \n return $this->result; \n \} \n \} \n class Test1 extends Test \{ \n public function calc() \{ \n $this->result \= $this->a + $this->b; \n \} \n \n \} \n class Test2 extends Test1 \{ \n public function calc() \{ \n parent\:\:calc(); \n $this->result *\= 2; \n \} \n \} \n \n class Test3 extends Test1 \{ \n public function __construct($a, $b) \{ \n $this->a \= $a * 2; \n $this->b \= $b * 2; \n \} \n \} \n \n $obj \= new Test3(2,3); \n $obj->calc(); \n echo $obj->getResult(); {
~ Ошибка выполнения
~ 5
= 10
~ 20
}

// question: 0 name: Switch category to $course$/top/По умолчанию для IntTex-2/JS/JS легко
$CATEGORY: $course$/top/По умолчанию для IntTex-2/JS/JS легко

// question: 36986 name: Вопрос 1
::Вопрос 1::[html] Выберите правильное определение функции внутри основного кода\: {
~ var func \= function();
= function test()\{\}
~ func\: function()\{\};
~ function()\{\}
}

// question: 36985 name: Вопрос 10

::Вопрос 10::[html] Выберите результат, который вернёт функция, после выполнения следующего кода \n \n function Test(param1, param2) \{ \n this.result \= param1 + param2; \n \n this.getResult \= function() \{ \n return this.result; \n \} \n \} \n \n var test1 \= new Test(10, 2); \n test1.getResult() {

~ NaN

~ 102

= 12

~ 0

}

// question: 36984 name: Вопрос 11

::Вопрос 11::[html] Выберите результат, который вернёт функция, после выполнения следующего кода \n \n function Test(param1, param2) \{ \n this.result \= param1 + param2; \n \n this.getResult \= function() \{ \n return this.result; \n \} \n \} \n \n var test1 \= new Test(10); \n test1.getResult() {

= NaN

~ 10

~ undefined

~ 0

}

// question: 36983 name: Вопрос 12

::Вопрос 12::[html] Выберите результат, который вернёт функция, после выполнения следующего кода \n \n function Test(param1, param2) \{ \n this.result \= param1 + param2; \n \n this.getResult \= function() \{ \n return this.result; \n \} \n \} \n \n var test1 \= new Test("test", 2); \n test1.getResult() {

~ NaN

~ 2

~ undefined

= test2

}

// question: 36982 name: Вопрос 13

::Вопрос 13::[html] Выберите результат, который вернёт функция, после выполнения следующего кода \n \n function Test(param1, param2) \{ \n this.result \= param1 + param2; \n \n this.getResult \= function() \{ \n return this.result; \n \} \n \} \n \n var test1 \= new Test(2, "test"); \n test1.getResult() {

~ NaN

~ 2

~ undefined

= 2test

}

// question: 36981 name: Вопрос 14

::Вопрос 14::[html] Выберите результат, который вернёт функция, после выполнения следующего кода \n \n function Test(param1, param2) \{ \n   this.result \= param1 + param2; \n   \n   this.getResult \= function() \{ \n   return this.result; \n   \} \n \} \n \n var test1 \= new Test("test", "test"); \n test1.getResult() {
~ NaN
~ 0
= testtest
~ undefined
}

// question: 36980 name: Вопрос 15
::Вопрос 15::[html] Выберите результат, который вернёт функция, после выполнения следующего кода \n \n function Test(param1, param2) \{ \n   this.result \= param1 + param2; \n   \n   this.getResult \= function() \{ \n   return this.result; \n   \} \n \} \n \n var test1 \= new Test("test", [1,2]); \n test1.getResult() {
~ NaN
~ 0
= test1,2
~ undefined
}

// question: 36979 name: Вопрос 16
::Вопрос 16::[html] Выберите результат, который вернёт функция, после выполнения следующего кода \n \n function Test(param1, param2) \{ \n   this.result \= param1 + param2; \n   \n   this.getResult \= function() \{ \n   return this.result; \n   \} \n \} \n \n var test1 \= new Test([1,2], [3,4]); \n test1.getResult() {
~ NaN
~ [1,2,3,4]
~ []
= 1,23,4
}

// question: 36978 name: Вопрос 17
::Вопрос 17::[html] Выберите результат, который вернёт функция, после выполнения следующего кода \n \n function Test(param1, param2) \{ \n   this.result \= param1 + param2; \n   \n   this.getResult \= function() \{ \n   return this.result; \n   \} \n \} \n \n var test1 \= new Test(\{abc\: 1\}, \{cde\: 2\}); \n test1.getResult() {
~ NaN
~ \{abc\:1, cde\:2\}
~ \{\}
= [object Object][object Object]
}

// question: 36977 name: Вопрос 18
::Вопрос 18::[html] Выберите результат, который вернёт функция, после выполнения следующего кода \n \n function Test(param1, param2) \{ \n   this.result

\= param1 + param2; \n   \n   this.getResult \= function() \{ \n   return this.result; \n   \} \n \} \n \n var test1 \= new Test(2, "2"); \n test1.getResult() {
~ NaN
~ 2
= 22
~ 4
}

// question: 36976 name: Вопрос 2
::Вопрос 2::[html] Выберите неправильное определение цикла FOR {
~ for(var i in values);
~ for(var i\=0;;i++);
= for(var i in key\=>value);
~ for(var i\=0, j\=0; i<10, j<10; i++, j++);
}

// question: 36975 name: Вопрос 3
::Вопрос 3::[html] Выберите не правильное определение цикла WHILE {
~ while(true);
~ while(i < 20);
~ while(i in [1,2,3]);
= while;
}

// question: 36974 name: Вопрос 4
::Вопрос 4::[html] Что будет показано на экране после выполнения следующего кода \n var type \= "test3"; \n switch(type) \{ \n   case "test1"\: \n   alert("test1"); \n   case "test2"\: \n   alert("test2"); \n   break; \n   case "test3"\: \n   alert("test3"); \n   default\: \n   alert("default"); \n   break; \n \} {
~ test3
~ test1, test2, test3
= test3, default
~ default
}

// question: 36973 name: Вопрос 5
::Вопрос 5::[html] Что будет показано на экране после выполнения следующего кода \n var type \= "test"; \n switch(type) \{ \n   case "test1"\: \n   alert("test1"); \n   case "test2"\: \n   alert("test2"); \n   break; \n   case "test3"\: \n   alert("test3"); \n   default\: \n   alert("default"); \n   break; \n \} {
~ test1
~ test1, test2, test3
~ test2
= default
}

// question: 36972 name: Вопрос 6

::Вопрос 6::[html] Что будет показано на экране после выполнения следующего кода \n \n var type \= "test1"; \n switch(type) \{ \n case "test1"\: \n alert("test1"); \n case "test2"\: \n alert("test2"); \n break; \n case "test3"\: \n alert("test3"); \n default\: \n alert("default"); \n break; \n \} {

~ test1

~ test1, test2, test3

= test1, test2

~ default

}

// question: 36971 name: Вопрос 7

::Вопрос 7::[html] Выберите результат, который вернёт функция, после выполнения следующего кода \n \n var func \= function(param) \{ \n param \= param || 0; \n param *\= 2; \n return param; \n \} \n \n func(); {

= 0

~ 2

~ undefined

~ NaN

}

// question: 36970 name: Вопрос 8

::Вопрос 8::[html] Выберите результат, который вернёт функция, после выполнения следующего кода \n \n var func \= function(param) \{ \n param \= param || 0; \n param *\= 2; \n return param; \n \} \n \n func(2); {

~ 0

= 4

~ undefined

~ NaN

}

// question: 36969 name: Вопрос 9

::Вопрос 9::[html] Выберите результат, который вернёт функция, после выполнения следующего кода \n \n var func \= function(param) \{ \n param \= param || 0; \n param *\= 2; \n return param; \n \} \n \n func("test"); {

~ 0

~ test

~ undefined

= NaN

}

// question: 0 name: Switch category to $course$/top/По умолчанию для IntTex-2/LESS/LESS сложно

$CATEGORY: $course$/top/По умолчанию для IntTex-2/LESS/LESS сложно

// question: 36993 name: Вопрос 1

::Вопрос 1::[html] Какие стили будут добавлены к элементу input после компиляции LESS файла \n .padding(@value\: 0) \{ \n padding\: \~"@\{value\}px"; \n \} \n \n .calc(@a; @b; @c) \{ \n .padding(@a + @b); \n \} \n \n .calc(@a; @b; @c) when (@c \= "+") \{ \n .padding(@a + @b); \n \} \n \n .calc(@a; @b; @c) when (@c \= "-") \{ \n .padding(@a - @b); \n \} \n \n .calc(@a; @b; @c) when (@c \= "/") \{ \n .padding(@a / @b); \n \} \n \n .calc(@a; @b; @c) when (@c \= "*") \{ \n .padding(@a * @b); \n \} \n \n input \{ \n .calc(20, 5, "/"); \n \} {

~ padding\: 20px;

= padding\: 4px;

~ padding\: 25px;

~ padding\: 0px;

}

// question: 36992 name: Вопрос 2

::Вопрос 2::[html] Какие стили будут добавлены к элементу input после компиляции LESS файла \n .padding(@value\: 0) \{ \n padding\: \~"@\{value\}px"; \n \} \n \n .calc(@a; @b; @c) \{ \n .padding(@a + @b); \n \} \n \n .calc(@a; @b; @c) when (@c \= "+") \{ \n .padding(@a + @b); \n \} \n \n .calc(@a; @b; @c) when (@c \= "-") \{ \n .padding(@a - @b); \n \} \n \n .calc(@a; @b; @c) when (@c \= "/") \{ \n .padding(@a / @b); \n \} \n \n .calc(@a; @b; @c) when (@c \= "*") \{ \n .padding(@a * @b); \n \} \n \n input \{ \n .calc(20, 5, "%"); \n \} {

= padding\: 25px;

~ padding\: 4px;

~ Стили не будут добавлены

~ padding\: 0px;

}

// question: 36991 name: Вопрос 3

::Вопрос 3::[html] Какие стили будут добавлены к элементу input после компиляции LESS файла \n .padding(@value\: 0) \{ \n padding\: \~"@\{value\}px"; \n \} \n \n .calc(@a; @b; @c) when (@c \= "+") \{ \n .padding(@a + @b); \n \} \n \n .calc(@a; @b; @c) when (@c \= "-") \{ \n .padding(@a - @b); \n \} \n \n .calc(@a; @b; @c) when (@c \= "/") \{ \n .padding(@a / @b); \n \} \n \n .calc(@a; @b; @c) when (@c \= "*") \{ \n .padding(@a * @b); \n \} \n \n .calc(@a; @b; @c) \{ \n .padding(@a + @b); \n \} \n \n input \{ \n .calc(20, 5, "-"); \n \} {

= padding\: 15px;

~ padding\: 25px;

~ padding\: 20px;

~ padding\: 0px;

}

// question: 36990 name: Вопрос 4

::Вопрос 4::[html] Какие классы будут доступны в HTML после компиляции LESS файла \n .percent (@value) when (@value >\= 0) \{ \n .w@\{value\} \{ \n

 width\: \~"\{@value\}%"; \n \} \n .h@\{value\} \{ \n height\: \~"\{@value\}%"; \n \} \n .percent(@value - 5); \n \} \n .percent (0) \{\} \n .percent (100); {
~ классы «w» и «h»
~ классы «w0», «w100», «h0», «h100»
~ Классы не будут доступны в HTML
= Классы от «w0» до «w100» с интервалом 5, и классы от «h0» до «h100» с интервалом 5
}

// question: 36989 name: Вопрос 5
::Вопрос 5::[html] Можно ли вызвать «.w100» после объявления классов \n .percent (@value) when (@value >\= 0) \{ \n .w@\{value\} \{ \n width\: \~"\{@value\}%"; \n \} \n .h@\{value\} \{ \n height\: \~"\{@value\}%"; \n \} \n .percent(@value - 5); \n \} \n .percent (0) \{\} \n .percent (100); \n \n .w100; {
~ Да.
~ Нет. Вызвать можно только внутри селектора.
= Нет. Создан

Полный перечень тестовых заданий с указанием правильных ответов, размещен в банке вопросов на информационно-образовательном портале института по ссылке https://www.mivlgu.ru/iop/question/edit.php?courseid=485&deleteall=1&category=2687%2C17118&qbshowtext=0&recurse=0&recurse=1&showhidden=0

Оценка рассчитывается как процент правильно выполненных тестовых заданий из их общего числа.