

Документ подписан простой электронной подписью  
 Информация о владельце:  
 ФИО: Косенок Сергей Михайлович  
 Должность: ректор  
 Дата подписания: 11.06.2026 11:46:29  
 Уникальный программный ключ:  
 e3a68f3eaa1e62674b54f4998099d3d6bfdcf836

## Тестовое задание для диагностического тестирования по дисциплине:

# Инструменты и технологии web-разработки

Код, направление подготовки	09.04.04 ПРОГРАММНАЯ ИНЖЕНЕРИЯ
Направленность (профиль)	Разработка и интеграция информационных систем и сервисов
Форма обучения	Очная
Кафедра-разработчик	Автоматики и компьютерных систем
Выпускающая кафедра	Автоматики и компьютерных систем

Проверяемая компетенция	Задание	Варианты ответов	Тип сложности вопроса
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	1. С помощью какой функции PHP можно прочитать файл?	1. file_get_contents 2. print_r 3. array_merge 4. require	Низкий
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	2. Что будет в переменной \$result после выполнения кода \$result = require 1.php, если в 1.php написан код <?php return 7 % 4;?	1. 2 2. 4 3. 7 4. 3	Низкий
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	3. Что будет в переменной \$x после выполнения кода \$x = 2; \$x -= 2;?	1. 2 2. 0 3. -2 4. 4	Низкий
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	4. Как получить данные POST-запроса?	1. через функцию getPostData() 2. через константу POST 3. через переменную \$_POST 4. через функцию request()	Низкий
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	5. Что будет в переменной \$result после выполнения кода \$i = 5; \$result = \$i++;?	1. 4 2. 7 3. 5 4. 6	Низкий
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	6. Как нужно создавать новый массив?	1. \$a = new Array(); 2. \$a = null; 3. \$a = []; 4. \$a = array();	Средний
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	7. С помощью какой функции можно удалить Cookie?	1. getcookie 2. deletecookie 3. setcookie 4. readcookie	Средний
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	8. Какая из этих конструкций не относится к циклам?	1. isset 2. foreach 3. for 4. While	Средний
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	9. Как присвоить переменной значение?	5. \$x === 5 6. \$x => 5 7. \$x == 5 8. \$x = 5	Средний
ОПК-2.1, ОПК-2.2,	10. С помощью какой функции можно выделить слово из предложения?	1. strpos() 2. substr()	Средний

ПК-6.1, ПК-6.2		3. substr() 4. strchr()	
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	11. Когда вызывается деструктор?	1. при создании объекта 2. при создании класса 3. при удалении класса 4. при удалении объекта	Средний
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	12. Для чего нужен оператор self	1. для получения названия объекта 2. для доступа к статическим свойствам и методам 3. для того же, что и \$this 4. реализации методов класса	Средний
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	13. Интерфейсы нужны для того, чтобы	1. классы правильно наследовались 2. определять реализацию методов 3. определять какие методы будут в классе, реализующем интерфейс 4. для инкапсуляции	Средний
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	14. Абстрактный класс в PHP это	1. класс без свойств 2. класс без методов 3. класс, экземпляры которого нельзя создать 4. нету такого понятия	Средний
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	15. Константа класса определяется при помощи ключевого слова	1. constant 2. define 3. static 4. const	Средний
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	16. final class означает, что	1. он не может быть унаследован 2. может быть создан лишь 1 его экземпляр 3. он не является абстрактным 4. не содержит констант	Высокий
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	17. Для чего предназначено ключевое слово throw?	1. для генерации ошибки 2. для прекращения работы скрипта 3. для бросания исключения 4. для удаления класса	Высокий
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	18. Переменная \$this нужна для	1. все того же что и остальные переменные 2. обращения к свойствам и методам объекта 3. обращения к классу 4. обращения к объекту внутри класса	Высокий
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	19. Что означает protected	1. свойства и методы доступны в экземплярах самого класса и в экземплярах наследников 2. свойства и методы доступны только в экземплярах наследника 3. обратное к public 4. то же самое что private	Высокий
ОПК-2.1, ОПК-2.2, ПК-6.1, ПК-6.2	20. Какое ключевое слово позволяет обращаться к родительскому классу?	1. protected 2. parent 3. self 4. this	Высокий