

Документ подписан простой электронной подписью  
 Информация о владельце:  
 ФИО: Косенко Сергей Михайлович  
 Должность: ректор  
 Дата подписания: 22.06.2026 12:41:43  
 Уникальный идентификатор документа:  
 e3a68f3eaa1e62674b54f4998099d3d6bfdcf836

## Тестовое задание для диагностического тестирования по дисциплине

### Инструментальные средства информационных систем, 6 семестр

Код, направление подготовки	09.03.02 Информационные системы и технологии
Направленность (профиль)	Информационные системы и технологии
Форма обучения	Очная
Кафедра разработчик	ИВТ
Выпускающая кафедра	ИВТ

Проверяемая компетенция	№	Задание	Варианты ответов	Тип сложности вопроса
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	1	Сколько систем контроля версий существует?	<ol style="list-style-type: none"> <li>1) Только Git</li> <li>2) Git и SVN</li> <li>3) Git и CVS</li> <li>4) Много</li> </ol>	низкий
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	2	Что такое GitHub?	<ol style="list-style-type: none"> <li>1) Программа для работы с Git</li> <li>2) Драйвер для Git</li> <li>3) Веб-сервис для хостинга IT-проектов и их совместной разработки, основанный на Git</li> <li>4) UI для работы с локальной версией Git</li> </ol>	низкий
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	3	Что такое репозиторий Git?	<ol style="list-style-type: none"> <li>1) Любая директория/папка в моей ОС</li> <li>2) Любая папка, находящаяся внутри Git</li> <li>3) Репозиторий Git представляет собой каталог файловой системы, в котором находятся файлы конфигурации репозитория, файлы журналов, хранящие операции, выполняемые над репозиторием, индекс, описывающий расположение файлов, и хранилище, содержащее собственно файлы</li> <li>4) Папка <code>.git/</code> и все входящие в нее</li> </ol>	низкий
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	4	Что делает команда <code>git status</code> ?	<ol style="list-style-type: none"> <li>1.Показывает состояние проекта: кол-во <code>untracked</code>, <code>deleted</code>, <code>new</code> и прочих файлов, количество коммитов, на которое отличается локальная версия репозитория от удаленного и так далее</li> <li>2.Показывает имя и email нашего пользователя, а также является ли он авторизованным в системе GitHub или нет</li> <li>3.Показывает место, занимаемое репозиторием на жестком диске и кол-во выделенного под репозиторий месте</li> <li>4.Такой команды нет, есть только команда</li> </ol>	низкий

			git show	
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	5	Что делает команда git add?	<ol style="list-style-type: none"> <li>1) Создает файл с указанным именем и сразу добавляет его в Git</li> <li>2) Добавляет локальный файл в удаленный репозиторий так, чтобы другие участники проекта могли его видеть</li> <li>3) Это алиас/синоним команды git commit</li> <li>4) Начинает отслеживать указанный файл или файлы</li> </ol>	низкий
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	6	Что означает статус файла untracked в выводе команды git status?	<ol style="list-style-type: none"> <li>1) Что система Git не отслеживает этот файл</li> <li>2) Что файл был удален из Git</li> <li>3) Что файл находится вне репозитория Git</li> <li>4) Что файл добавлен в .gitignore</li> </ol>	средний
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	7	Что означает статус файла new в выводе команды git status?	<ol style="list-style-type: none"> <li>1) Что файл только что был создан и еще не отслеживается системой Git</li> <li>2) Что файл только начал отслеживаться Git и пока не имеет истории</li> <li>3) Что файл удаляли из Git и потом восстановили командой git return</li> <li>4) Такого статуса нет, есть только статус deleted file</li> </ol>	средний
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	8	Что означает статус файла modified в выводе команды git status?	<ol style="list-style-type: none"> <li>1) Что файл имеет историю в системе Git и был изменен относительно его последнего состояния</li> <li>2) Такого статуса нет, есть только статусы new и deleted</li> <li>3) Этот статус виден только командой git ignore и означает, что файл перестал отслеживаться системой Git</li> <li>4) Статус означает, что файл добавлен в коммит</li> </ol>	средний
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	9	Что такое коммит?	<ol style="list-style-type: none"> <li>1) Это единица состояния проекта в Git</li> <li>2) Это результат вывода команды git diff</li> <li>3) Это обобщающее название одного из статусов файла в выводе git status: untracked, new, deleted или modified</li> <li>4) Это слово ничего не означает, его ввели только для того, чтобы путать новичков</li> </ol>	средний
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	10	Как сделать коммит?	<ol style="list-style-type: none"> <li>1) Всего лишь набрать команду git commit в любой момент времени</li> <li>2) Сделать изменения в файлах и перечислить их после git commit. Например так: git commit a.file, b.file</li> <li>3) Сделать изменения, собрать эти изменения командой "git add" или "git commit -a" и указать коммит-сообщение после ключа "-m"</li> <li>4) Нельзя сделать коммит, ведь такого понятия не существует</li> </ol>	средний
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2	11	Что такое ветка в репозитории Git?	<ol style="list-style-type: none"> <li>1) Это то же самое, что и коммит</li> <li>2) Это минимум два коммита с одинаковым коммит-сообщением</li> <li>3) Это разные пути развития проекта, по сути разные последовательности коммитов</li> </ol>	средний

ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3			4) Это механизм изменения конкретного файла	
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	12	Чем отличается master и origin master?	<ol style="list-style-type: none"> <li>1) Это просто два разных названия одной ветки</li> <li>2) master принадлежит локальному репозиторию, а origin master - удаленному</li> <li>3) Это две разные ветки локального репозитория</li> <li>4) Ветки origin master не существует</li> </ol>	средний
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	13	Чем отличаются команды "git push" и "git pull"?	<ol style="list-style-type: none"> <li>1) Это алиасы</li> <li>2) Команды "git pull" не существует, а команда "git push" нужна, чтобы выложить изменения в удаленный репозиторий</li> <li>3) Команды "git push" не существует, а команда "git pull" нужна, чтобы стянуть изменения из удаленного репозитория</li> <li>4) команда "git pull" нужна, чтобы стянуть изменения из удаленного репозитория, а команда "git push" нужна, чтобы выложить изменения в удаленный репозиторий</li> </ol>	средний
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	14	Что делает команда git log?	<ol style="list-style-type: none"> <li>1) Пишет указанный после файл в лог</li> <li>2) Такой команды нет, есть только команда git look</li> <li>3) Показывает историю коммитов</li> <li>4) Удаляет файл из репозитория</li> </ol>	средний
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	15	Что делает команда git show?	<ol style="list-style-type: none"> <li>1) Показывает изменения, сделанные в указанном коммите</li> <li>2) Показывает содержимое файла</li> <li>3) Показывает состояние проекта</li> <li>4) Показывает время</li> </ol>	средний
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	16	Как узнать, кто автор строки в файле, используя систему Git?	<ol style="list-style-type: none"> <li>1) Команда git show --author</li> <li>2) Команда git commit --author</li> <li>3) Команда git blame</li> <li>4) Команда git status</li> </ol>	высокий
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	17	Как узнать, какие изменения мы сделали локально относительно последнего состояния нашего репозитория?	<ol style="list-style-type: none"> <li>1) Команда git show</li> <li>2) Команда git diff</li> <li>3) Команда git izmeneniya</li> <li>4) Команда git commit</li> </ol>	высокий

ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	18	Как отменить действие команды "git add" на файл?	<ol style="list-style-type: none"> <li>1) Команда git abort</li> <li>2) Команда git stash</li> <li>3) Команда git not-add</li> <li>4) Команда git reset</li> </ol>	высокий
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	19	Как решить конфликт в Git?	<ol style="list-style-type: none"> <li>1) Руками поправить изменения там, где Git не смог это сделать автоматически и затем собрать все в коммит и запустить</li> <li>2) Никак, придется создавать репозиторий заново</li> <li>3) Выполнить команду git commit merge please</li> <li>4) Удалить файл, для которого Git не знает, как смирить изменения</li> </ol>	высокий
ОПК-2.2 ОПК-2.3 ОПК-3.1 ОПК-3.2 ОПК-3.3 ОПК-6.1 ОПК-6.2 ОПК-6.3 ОПК-7.1 ОПК-7.2 ОПК-7.3	20	Как привести измененный файл в начальное состояние (до изменения)?	<ol style="list-style-type: none"> <li>1) Команда git abort path/to/file</li> <li>2) Команда git checkout path/to/file</li> <li>3) Команда git pull path/to/file</li> <li>4) Команда git commit path/to/file</li> </ol>	высокий