



**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение
высшего образования

«ИРКУТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ФГБОУ ВО «ИГУ»

Факультет бизнес-коммуникаций и информатики

Кафедра естественнонаучных дисциплин

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине Б1.В.ДВ.03.01 Шаблоны проектирования

направление подготовки 09.03.03 Прикладная информатика

направленность (профиль) Прикладная информатика (разработка программного обеспечения)

Одобен
УМК факультета бизнес-коммуникаций
и информатики

Разработан в соответствии с ФГОС ВО

с учетом требований проф. стандарта

Председатель УМК

М.Г. Синчурина

ФИО, должность, ученая степень, звание

подпись, печать

Разработчики:

(подпись)

преподаватель

(занимаемая должность)

А.С. Воловиков

(инициалы, фамилия)



(подпись)

доцент

(занимаемая должность)

А.Г. Балахчи

(инициалы, фамилия)

Цель фонда оценочных средств. Оценочные средства предназначены для контроля и оценки образовательных достижений обучающихся, освоивших программу учебной дисциплины «Б1.В.ДВ.03.01 Шаблоны проектирования». Перечень видов оценочных средств соответствует рабочей программе дисциплины.

Фонд оценочных средств включает контрольные материалы для проведения текущего контроля (в следующих формах: практическое задание, тест) и промежуточной аттестации в форме вопросов и заданий к зачету с оценкой.

Структура и содержание заданий – задания разработаны в соответствии с рабочей программой дисциплины «Б1.В.ДВ.03.01 Шаблоны проектирования».

1. Паспорт фонда оценочных средств

Компетенция	Индикаторы компетенций	Результаты обучения
ПК-2 Способность разрабатывать, внедрять и адаптировать прикладное программное обеспечение	ПК-2.1	Знать технологии разработки программного обеспечения: методы, средства, процедуры и инструменты
	ПК-2.2	Уметь внедрять и адаптировать прикладное программное обеспечение
	ПК-2.3	Владеть навыками решения задач реализации и модификации ПО: планирования и оценки проекта по разработке ПО; анализа системных и программных требований; проектирования алгоритмов, структур данных и программных структур; кодирования с использованием различных языков программирования и разметки; рефакторинга ПО; тестирования и отладки программного кода; сопровождения
ПК-3 Способность осуществлять тестирование компонентов информационных систем и по заданным сценариям	ПК-3.1	Знать основные методы тестирования компонентов информационных систем
	ПК-3.2	Уметь пользоваться специальным программным обеспечением для автоматизированного тестирования (при необходимости) на уровне запуска готовых тестов
	ПК-3.3	Владеть навыками разработки тестовых сценариев компонентов информационных систем, проведения тестирования, исследования и анализа результатов

Компетенция	Индикаторы компетенций	Результаты обучения
ПК-4 Способность создавать (модифицировать) и сопровождать информационные системы и сервисы по видам обеспечения	ПК-4.1	Знать основные виды информационных систем в области прикладного программного обеспечения, основные технологии проектирования по видам обеспечения
	ПК-4.2	Уметь описывать структуру ИС на базе DFD и SADT диаграмм, осуществлять эксплуатацию и сопровождение информационных систем и сервисов по видам обеспечения
	ПК-4.3	Владеть методами проектирования ИС и сервисов в соответствии с прикладной задачей по видам обеспечения

2. Показатели и критерии оценивания компетенций, шкалы оценивания

2.1. Показатели и критерии оценивания компетенций

№ п/п	Раздел, тема	Код индикатора компетенции	Наименование ОС	
			ТК	ПА
1	Клиент серверная	ПК-2.1, ПК-2.2, ПК-2.3	Пз	Нет!
2	Многоуровневая	ПК-2.1	Пз	Нет!
3	Модель - представление - контроллер (MVC)	ПК-2.1, ПК-2.2, ПК-2.3	Пз	Нет!
4	Сервис - ориентированная (SOA)	ПК-2.1, ПК-2.2, ПК-2.3	Пз	Нет!
5	Общие данные	ПК-2.1, ПК-2.2, ПК-2.3	Тест, Пз	Тест
6	Издатель - подписчик	ПК-2.1	Пз	Нет!
7	Событийно - ориентированная (EDA)	ПК-2.1, ПК-2.2, ПК-2.3	Пз	Нет!
8	Разделение ответственности (CQRS)	ПК-2.1, ПК-2.2, ПК-2.3	Тест, Пз	Тест
9	Фабрика (Factory)	ПК-2.1, ПК-2.2, ПК-2.3	Пз	Нет!
10	Фабричный метод (Factory Method)	ПК-2.1, ПК-2.2, ПК-2.3	Пз	Нет!
11	Одиночка (Singleton)	ПК-2.1, ПК-2.2, ПК-2.3	Пз	Нет!
12	Мост (Bridge)	ПК-2.1	Пз	Нет!
13	Декоратор (Decorator)	ПК-2.1, ПК-2.2, ПК-2.3	Тест, Пз	Тест
14	Адаптер (Adapter)	ПК-2.1, ПК-2.2, ПК-2.3	Тест, Пз	Тест
15	Заместитель (Proxy)	ПК-2.1, ПК-2.2, ПК-2.3	Тест, Пз	Тест
16	Наблюдатель (Observer)	ПК-2.1	Пз	Нет!

2.2. Критерии оценивания результатов обучения для текущего контроля успеваемости и промежуточной аттестации

Оценочное средство	Критерии оценивания	Шкала оценивания
Практическое задание	Задание выполнено верно. Выбран оптимальный путь решения. Присутствует развернутое описание алгоритма решения	Отлично
	Задание выполнено верно. Допущены негрубые логические ошибки при описании алгоритма решения. Отсутствуют пояснения к решению задания	Хорошо
	Ход решения задания верный, но допущены ошибки приведшие к неправильному ответу	Удовлетворительно
	В работе получен неверный ответ, связанный с грубыми ошибками допущенными в ходе решения, либо решение отсутствует полностью	Неудовлетворительно
Тест	Студентом даны правильные ответы на 91-100% заданий	Отлично
	Студентом даны правильные ответы на 81-90% заданий	Хорошо
	Студентом даны правильные ответы на 71-80% заданий	Удовлетворительно
	Студентом даны правильные ответы менее чем на 70% заданий	Неудовлетворительно

2.3. Оценочные средства для текущего контроля (примеры)

2.3.1. Материалы для компьютерного тестирования обучающихся

Общие критерии оценивания

Процент правильных ответов	Оценка
91% – 100%	5 (отлично)
81% – 90%	4 (хорошо)
71% – 80%	3 (удовлетворительно)
Менее 70%	2 (неудовлетворительно)

Соответствие вопросов теста индикаторам формируемых и оцениваемых компетенций

№ вопроса в тесте	Код индикатора компетенции
1	ПК-2.1, ПК-2.2, ПК-2.3
2	ПК-2.1, ПК-2.2, ПК-2.3
3	ПК-2.1, ПК-2.2, ПК-2.3
4	ПК-2.1, ПК-2.2, ПК-2.3
5	ПК-2.1, ПК-2.2, ПК-2.3

Ключ ответов

№ вопроса в тесте	Номер ответа (или ответ, или соответствие)
1	a
2	b
3	a
4	a
5	a

Перечень тестовых вопросов

№ 1. Задание с единичным выбором. Выберите один правильный ответ.

Что такое шаблон проектирования Заместитель (Proxy)?

a. Что такое шаблон проектирования Заместитель (Proxy)?

№ 2. Задание с единичным выбором. Выберите один правильный ответ.

Можно ли использовать несколько Декораторов для одного объекта?

a. Нет

b. Да

№ 3. Задание с единичным выбором. Выберите один правильный ответ.

Что такое CQRS?

a. Это архитектурный шаблон проектирования с разделением ответственности за команды и запросы

b. Это высокоскоростная система передачи данных

№ 4. Задание с единичным выбором. Выберите один правильный ответ.

В чем заключается шаблон проектирования Адаптер?

a. В чем заключается шаблон проектирования Адаптер?

№ 5. Задание с единичным выбором. Выберите один правильный ответ.

Для чего используется шаблон проектирования "Общие данные"?

a. Данный шаблон является простым вариантом интеграции различных система на основе взаимодействия базы данных

b. Это распространенный вариант объединения различных систем

2.3.2. Практические задания для оценки компетенции «ПК-2.1»

№ 1. Разработка программного обеспечения.

Конвертация валют

Разработка программного обеспечения с использованием архитектурного шаблона проектирования "Клиент-Сервер"

Стек: Python, PostgreSQL

№ 2. Разработка программного обеспечения.

Конвертация валют

Стек: Python, PostgreSQL

№ 3. Разработка программного обеспечения.

Конвертация валют

Разработка программного обеспечения с использованием архитектурного шаблона проектирования "MVC"

Стек: Python, PostgreSQL

№ 4. Разработка программного обеспечения.

Конвертация валют

Разработка программного обеспечения с использованием архитектурного шаблона проектирования "SOA"

Стек: Python, PostgreSQL

№ 5. Разработка программного обеспечения.

Интеграция

Разработка программного обеспечения с использованием архитектурного шаблона "Общие данные"

Стек: Python, PostgreSQL

№ 6. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения на основе архитектурного шаблона "Издатель - подписчик"

Стек: Python, PostgreSQL

№ 7. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения на основе архитектурного шаблона "EDA"

Стек: Python, PostgreSQL

№ 8. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения на основе архитектурного шаблона "CQRS"

Стек: Python, PostgreSQL

№ 9. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения с использованием шаблона "Фабрика"

Стек: Python, PostgreSQL

№ 10. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения с использованием шаблона проектирования "Фабричный метод"

Стек: Python, PostgreSQL

№ 11. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения с использованием шаблона "Singleton"

Стек: Python, PostgreSQL

№ 12. Разработка программного обеспечения.

Публикации рецептов

Стек: Python, PostgreSQL

№ 13. Разработка программного обеспечения.

Публикации рецептов

Стек: Python, PostgreSQL

№ 14. Разработка программного обеспечения.

№ 15. Разработка программного обеспечения.

№ 16. Разработка программного обеспечения.

Публикации рецептов

Стек: Python, PostgreSQL

2.3.3. Практические задания для оценки компетенции «ПК-2.2»

№ 17. Разработка программного обеспечения.

Конвертация валют

Разработка программного обеспечения с использованием архитектурного шаблона проектирования "Клиент-Сервер"

Стек: Python, PostgreSQL

№ 18. Разработка программного обеспечения.

Конвертация валют

Стек: Python, PostgreSQL

№ 19. Разработка программного обеспечения.

Конвертация валют

Разработка программного обеспечения с использованием архитектурного шаблона проектирования "MVC"

Стек: Python, PostgreSQL

№ 20. Разработка программного обеспечения.

Конвертация валют

Разработка программного обеспечения с использованием архитектурного шаблона проектирования "SOA"

Стек: Python, PostgreSQL

№ 21. Разработка программного обеспечения.

Интеграция

Разработка программного обеспечения с использованием архитектурного шаблона "Общие данные"

Стек: Python, PostgreSQL

№ 22. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения на основе архитектурного шаблона "Издатель - подписчик"

Стек: Python, PostgreSQL

№ 23. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения на основе архитектурного шаблона "EDA"

Стек: Python, PostgreSQL

№ 24. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения на основе архитектурного шаблона "CQRS"

Стек: Python, PostgreSQL

№ 25. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения с использованием шаблона "Фабрика"

Стек: Python, PostgreSQL

№ 26. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения с использованием шаблона проектирования "Фабричный метод"

Стек: Python, PostgreSQL

№ 27. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения с использованием шаблона "Singleton"

Стек: Python, PostgreSQL

№ 28. Разработка программного обеспечения.

Публикации рецептов

Стек: Python, PostgreSQL

№ 29. Разработка программного обеспечения.

Публикации рецептов

Стек: Python, PostgreSQL

№ 30. Разработка программного обеспечения.

№ 31. Разработка программного обеспечения.

2.3.4. Практические задания для оценки компетенции «ПК-2.3»

№ 32. Разработка программного обеспечения.

Конвертация валют

Разработка программного обеспечения с использованием архитектурного шаблона проектирования "Клиент-Сервер"

Стек: Python, PostgreSQL

№ 33. Разработка программного обеспечения.

Конвертация валют

Стек: Python, PostgreSQL

№ 34. Разработка программного обеспечения.

Конвертация валют

Разработка программного обеспечения с использованием архитектурного шаблона проектирования "MVC"

Стек: Python, PostgreSQL

№ 35. Разработка программного обеспечения.

Конвертация валют

Разработка программного обеспечения с использованием архитектурного шаблона проектирования "SOA"

Стек: Python, PostgreSQL

№ 36. Разработка программного обеспечения.

Интеграция

Разработка программного обеспечения с использованием архитектурного шаблона "Общие данные"

Стек: Python, PostgreSQL

№ 37. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения на основе архитектурного шаблона "Издатель - подписчик"

Стек: Python, PostgreSQL

№ 38. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения на основе архитектурного шаблона "EDA"

Стек: Python, PostgreSQL

№ 39. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения на основе архитектурного шаблона "CQRS"

Стек: Python, PostgreSQL

№ 40. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения с использованием шаблона "Фабрика"

Стек: Python, PostgreSQL

№ 41. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения с использованием шаблона проектирования "Фабричный метод"

Стек: Python, PostgreSQL

№ 42. Разработка программного обеспечения.

Публикации рецептов

Разработка программного обеспечения с использованием шаблона "Singleton"

Стек: Python, PostgreSQL

№ 43. Разработка программного обеспечения.

Публикации рецептов

Стек: Python, PostgreSQL

№ 44. Разработка программного обеспечения.

Публикации рецептов

Стек: Python, PostgreSQL

№ 45. Разработка программного обеспечения.

№ 46. Разработка программного обеспечения.

3. Промежуточная аттестация

3.1. Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и опыта деятельности

Зачет является заключительным этапом процесса формирования компетенций обучающегося при изучении дисциплины и имеет целью проверку и оценку знаний обучающегося по теории, и применению полученных знаний, умений и навыков при решении практических задач.

Зачет проводится по расписанию, сформированному учебно-методическим управлением, в сроки, предусмотренные календарным учебным графиком. Зачет принимается преподавателем, ведущим лекционные занятия.

Зачет проводится только при предъявлении обучающимся зачетной книжки и при условии выполнения всех контрольных мероприятий, предусмотренных учебным планом и рабочей программой дисциплины. Обучающимся на зачете представляется право выбрать один из билетов. Время подготовки к ответу составляет 30 минут. По истечении установленного времени обучающийся должен ответить на вопросы билета. Результаты зачета оцениваются по четырехбалльной системе и заносятся в зачетно-экзаменационную ведомость и зачетную книжку. В зачетную книжку заносятся только положительные оценки. Подписанный преподавателем экземпляр ведомости сдается не позднее следующего дня в деканат.

В случае неявки обучающегося на зачет в зачетно-экзаменационную ведомость делается отметка «не явка». Обучающиеся, не прошедшие промежуточную аттестацию по дисциплине, должны ликвидировать академическую задолженность в установленном локальными нормативными актами порядке.

3.2. Вопросы к зачету с оценкой

№	Вопрос	Код компетенции
1.	Что такое шаблон проектирования Клиент-Сервер и как он работает?	ПК-2.1, ПК-2.2, ПК-2.3
2.	Что такое процессорное разделение нагрузки и как он работает в архитектуре Клиент-Сервер?	ПК-2.1, ПК-2.2, ПК-2.3
3.	Какие могут быть риски и ограничения в создании приложения на основе архитектуры Клиент-Сервер?	ПК-2.1, ПК-2.2, ПК-2.3
4.	Что такое многоуровневый шаблон проектирования?	ПК-2.1, ПК-2.2, ПК-2.3
5.	Какие основные компоненты входят в многоуровневую архитектуру?	ПК-2.1, ПК-2.2, ПК-2.3
6.	Какие преимущества имеет многоуровневый подход в разработке программного обеспечения?	ПК-2.1, ПК-2.2, ПК-2.3
7.	Что такое архитектура MVC и какие ее основные компоненты?	ПК-2.1, ПК-2.2, ПК-2.3

№	Вопрос	Код компетенции
8.	Какие преимущества может предоставить использование архитектуры MVC?	ПК-2.1, ПК-2.2, ПК-2.3
9.	Как взаимодействуют между собой компоненты архитектуры MVC?	ПК-2.1, ПК-2.2, ПК-2.3
10.	Что такое SOA (архитектура сервисно-ориентированного приложения)?	ПК-2.1, ПК-2.2, ПК-2.3
11.	Какие преимущества имеют SOA приложения по сравнению с традиционными монолитными приложениями?	ПК-2.1, ПК-2.2, ПК-2.3
12.	Какие компоненты входят в SOA архитектуру приложений?	ПК-2.1, ПК-2.2, ПК-2.3
13.	Какова роль веб-сервисов в SOA архитектуре приложений?	ПК-2.1, ПК-2.2, ПК-2.3
14.	Что такое шаблон проектирования "Общие данные"?	ПК-2.1, ПК-2.2, ПК-2.3
15.	С какой целью применяется шаблон "Общие данные"?	ПК-2.1, ПК-2.2, ПК-2.3
16.	Что такое архитектура "Издатель - подписчик"?	ПК-2.1, ПК-2.2, ПК-2.3
17.	Какие принципы лежат в основе архитектуры "Издатель - подписчик"?	ПК-2.1, ПК-2.2, ПК-2.3
18.	Какая роль играют подписчики архитектуре "Издатель - подписчик"?	ПК-2.1, ПК-2.2, ПК-2.3
19.	Какие преимущества имеет эта архитектура "Издатель - подписчик" по сравнению с другими подходами?	ПК-2.1, ПК-2.2, ПК-2.3
20.	Что такое Событийно-ориентированная архитектура (COA) и как она отличается от других подходов в архитектуре ПО?	ПК-2.1, ПК-2.2, ПК-2.3
21.	Какие основные компоненты входят в архитектуру COA?	ПК-2.1, ПК-2.2, ПК-2.3
22.	Какие типы сообщений могут использоваться в архитектуре COA?	ПК-2.1, ПК-2.2, ПК-2.3
23.	Как происходит мониторинг и отладка приложений, созданных на основе архитектуры COA?	ПК-2.1, ПК-2.2, ПК-2.3
24.	Что такое CQRS и как она используется в архитектуре приложений?	ПК-2.1, ПК-2.2, ПК-2.3
25.	Какую проблему решает CQRS в разработке программного обеспечения?	ПК-2.1, ПК-2.2, ПК-2.3
26.	Как CQRS взаимодействует с другими архитектурными паттернами и практиками, такими как DDD и Event Sourcing?	ПК-2.1, ПК-2.2, ПК-2.3
27.	Какие рекомендации можно дать для использования CQRS в реальных проектах?	ПК-2.1, ПК-2.2, ПК-2.3
28.	Что такое шаблон проектирования Фабрика?	ПК-2.1, ПК-2.2, ПК-2.3
29.	Какова цель использования данного шаблона?	ПК-2.1, ПК-2.2, ПК-2.3
30.	Можно ли определить свои собственные компоненты?	ПК-2.1, ПК-2.2, ПК-2.3
31.	Какие преимущества в использовании этого шаблона проектирования?	ПК-2.1, ПК-2.2, ПК-2.3
32.	Что такое шаблон проектирования Фабричный метод?	ПК-2.1, ПК-2.2, ПК-2.3

№	Вопрос	Код компетенции
33.	В каких случаях следует использовать Фабричный метод?	ПК-2.1, ПК-2.2, ПК-2.3
34.	Как работает Фабричный метод в контексте наследования?	ПК-2.1, ПК-2.2, ПК-2.3
35.	Какие альтернативные подходы существуют для создания объектов?	ПК-2.1, ПК-2.2, ПК-2.3
36.	Что такое шаблон проектирования Одиночка?	ПК-2.1, ПК-2.2, ПК-2.3
37.	Какие проблемы решает шаблон проектирования Одиночка?	ПК-2.1, ПК-2.2, ПК-2.3
38.	Какая разница между Одиночкой и Глобальной переменной?	ПК-2.1, ПК-2.2, ПК-2.3
39.	Что такое шаблон проектирования Мост?	ПК-2.1, ПК-2.2, ПК-2.3
40.	Какие проблемы решает шаблон Мост?	ПК-2.1, ПК-2.2, ПК-2.3
41.	Как устранить жесткую связь между абстракцией и реализацией?	ПК-2.1, ПК-2.2, ПК-2.3
42.	Какая задача может быть решена с помощью шаблона Мост?	ПК-2.1, ПК-2.2, ПК-2.3
43.	Что такое шаблон проектирования Декоратор?	ПК-2.1, ПК-2.2, ПК-2.3
44.	Какие проблемы решает шаблон Декоратор?	ПК-2.1, ПК-2.2, ПК-2.3
45.	Можно ли использовать несколько Декораторов для одного объекта?	ПК-2.1, ПК-2.2, ПК-2.3
46.	В чем заключается шаблон проектирования Адаптер?	ПК-2.1, ПК-2.2, ПК-2.3
47.	Какой основной принцип лежит в основе Адаптера?	ПК-2.1, ПК-2.2, ПК-2.3
48.	В каких случаях необходимо применять Адаптер?	ПК-2.1, ПК-2.2, ПК-2.3
49.	Что такое шаблон проектирования Заместитель (Proxy)?	ПК-2.1, ПК-2.2, ПК-2.3
50.	Какие задачи решает шаблон проектирования Заместитель (Proxy)?	ПК-2.1, ПК-2.2, ПК-2.3
51.	Как работает Удаленный заместитель (Remote Proxy)?	ПК-2.1, ПК-2.2, ПК-2.3
52.	Как работает Защищающий заместитель (Protective Proxy)?	ПК-2.1, ПК-2.2, ПК-2.3
53.	Какие преимущества имеет использование шаблона проектирования Заместитель (Proxy)?	ПК-2.1, ПК-2.2, ПК-2.3
54.	Что такое шаблон проектирования Наблюдатель?	ПК-2.1, ПК-2.2, ПК-2.3
55.	Какие возможности предоставляет шаблон Наблюдатель для модификации приложения?	ПК-2.1, ПК-2.2, ПК-2.3
56.	Какие преимущества и недостатки имеет использование шаблона Наблюдатель?	ПК-2.1, ПК-2.2, ПК-2.3

3.3. Тематика курсовых работ

По данной дисциплине выполнение курсовых проектов (работ) не предусматривается.

3.4. Материалы для компьютерного тестирования обучающихся

Общие критерии оценивания

Процент правильных ответов	Оценка
91% – 100%	5 (отлично)
81% – 90%	4 (хорошо)
71% – 80%	3 (удовлетворительно)

Процент правильных ответов	Оценка
Менее 70%	2 (неудовлетворительно)

Соответствие вопросов теста индикаторам формируемых и оцениваемых компетенций

№ вопроса в тесте	Код индикатора компетенции
1	ПК-2.1, ПК-2.2, ПК-2.3
2	ПК-2.1, ПК-2.2, ПК-2.3
3	ПК-2.1, ПК-2.2, ПК-2.3
4	ПК-2.1, ПК-2.2, ПК-2.3
5	ПК-2.1, ПК-2.2, ПК-2.3

Ключ ответов

№ вопроса в тесте	Номер ответа (или ответ, или соответствие)
1	a
2	b
3	a
4	a
5	a

Перечень тестовых вопросов

№ 1. Задание с единичным выбором. Выберите один правильный ответ.

Что такое шаблон проектирования Заместитель (Proxy)?

a. Что такое шаблон проектирования Заместитель (Proxy)?

№ 2. Задание с единичным выбором. Выберите один правильный ответ.

Можно ли использовать несколько Декораторов для одного объекта?

a. Нет

b. Да

№ 3. Задание с единичным выбором. Выберите один правильный ответ.

Что такое CQRS?

a. Это архитектурный шаблон проектирования с разделением ответственности за команды и запросы

b. Это высокоскоростная система передачи данных

№ 4. Задание с единичным выбором. Выберите один правильный ответ.

В чем заключается шаблон проектирования Адаптер?

a. В чем заключается шаблон проектирования Адаптер?

№ 5. Задание с единичным выбором. Выберите один правильный ответ.

Для чего используется шаблон проектирования "Общие данные"?

a. Данный шаблон является простым вариантом интеграции различных система на основе взаимодействия базы данных

b. Это распространенный вариант объединения различных систем