



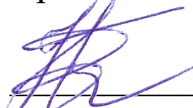
Министерство образования Иркутской области
Государственное бюджетное профессиональное
образовательное учреждение Иркутской области
«Иркутский авиационный техникум»

**Методические указания
по выполнению самостоятельной работы
по междисциплинарному курсу
МДК.02.04 Разработка прикладных приложений
специальности
09.02.01 Компьютерные системы и комплексы**

Иркутск, 2025

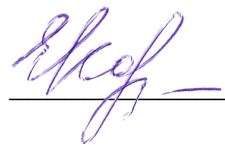
РАССМОТРЕНЫ

Председатель ЦК

 / Н.Р.
Огородникова /

УТВЕРЖДАЮ

Зам. директора

 Е.А. Коробкова

№	Разработчик ФИО
1	Хромовских Юрий Юрьевич
2	Александрова Алена Сергеевна

Пояснительная записка

МДК.02.04 Разработка прикладных приложений относится к ПМ.02 Проектирование управляющих программ компьютерных систем и комплексов. Самостоятельная работа является одним из видов учебно работы обучающегося без взаимодействия с преподавателем.

Основные цели самостоятельной работы:

- систематизация и закрепление теоретических знаний и практических умений обучающихся;
- углубление и расширение теоретических знаний, формирование умений использовать справочную документацию и дополнительную литературу;
- развитие познавательных способностей и активности обучающихся, творческой инициативы, самостоятельности, ответственности и организованности;
- формирование самостоятельного мышления;
- развитие исследовательских умений.

Рекомендации для обучающихся по выработке навыков самостоятельной работы:

- Слушать, записывать и запоминать лекцию.
- Внимательно читать план выполнения работы.
- Выбрать свой уровень подготовки задания.
- Обращать внимание на рекомендуемую литературу. Из перечня литературы выбирать ту, которая наиболее полно раскрывает вопрос задания.
- Учиться кратко излагать свои мысли. Использовать общие правила написания конспекта.
- Обращать внимание на достижение основной цели работы.

Тематический план

Раздел Тема	Тема занятия	Название работы	Количество часов
Раздел 1. Разработка прикладных приложений Тема 7. Коллекции и интерфейсы	Объектно-ориентированное программирование (ООП).	Объектно-ориентированное программирование (ООП).	2
Тема 27. Основы командной разработки	Принципы командной разработки. Использование системы контроля версий.	Принципы командной разработки. Использование системы контроля версий.	2

Самостоятельная работа №1

Название работы: Принципы командной разработки. Использование системы контроля версий..

Цель работы: Продемонстрировать понимание целей, задач и вариантов практического применения систем контроля версий при разработке проекта.

Уровень СРС: воспроизводящая.

Форма контроля: Проверка практической работы с применением ИКТ.

Количество часов на выполнение: 2 часа.

Задание:

Провести обоснование выбора системы контроля версий при командной разработке проекта, привести методику её использования по ролям разработчиков проекта.

Критерии оценки:

оценка «5» - Проведено сравнительное обоснование выбора системы контроля версий на основе методологии командной разработки, описан функционал и приведены алгоритмы её использования в зависимости от роли разработчика проекта, указаны принципы командного взаимодействия и разграничения прав при использовании СКВ всеми участниками процесса разработки, включая административный персонал, приведен конкретный пример применения СКВ для двух различных участников проекта.

оценка «4» - Проведено обоснование выбора системы контроля версий на основе методологии командной разработки, описан функционал и приведены алгоритмы её использования в зависимости от роли разработчика проекта, приведен конкретный пример применения СКВ участником проекта.

оценка «3» - Проведено обоснование выбора системы контроля версий, приведен алгоритмы её использования на примере применения СКВ участником проекта.

Самостоятельная работа №2

Название работы: Объектно-ориентированное программирование (ООП)..

Цель работы: Продемонстрировать понимание и умение применять основные принципы объектно-ориентированного программирования: инкапсуляция, наследование, полиморфизм и абстракция.

Уровень СРС: воспроизводящая.

Форма контроля: Проверка практической работы с применением ИКТ.

Количество часов на выполнение: 2 часа.

Задание:

Разработайте программу на выбранном языке программирования (например, Python, Java, C#), которая моделирует систему управления библиотекой.

Программа должна включать следующие классы и функциональности:

1. Класс *Book*:

Свойства: title, author, ISBN, available (доступность).

Методы: display_info() (выводит информацию о книге), check_availability() (проверяет, доступна ли книга).

2. Класс *Library*:

Свойства: books (список книг).

Методы: add_book(), remove_book(), find_book_by_title(), find_book_by_author().

3. Класс *User*:

Свойства: name, user_id.

Методы: borrow_book(book), return_book(book).

4. Класс *Admin* (наследует от *User*):

Дополнительные методы: add_book_to_library(book), remove_book_from_library(book).

Требования:

Используйте инкапсуляцию для защиты данных.

Примените наследование для класса Admin.

Используйте полиморфизм в методах, связанных с операциями над книгами.

Продемонстрируйте абстракцию в проектировании классов и их взаимодействии.

Критерии оценки:

оценка «5» - Реализованы основные классы и методы. Применены базовые принципы ООП (инкапсуляция, наследование). Программа работает без критических ошибок. Использован полиморфизм для улучшения функциональности. Программа демонстрирует хорошую структуру и читаемость кода. Продемонстрирована высокая степень абстракции в проектировании. Включены дополнительные функциональности, улучшающие пользовательский опыт (например, возможность резервирования книги, история взятых книг и т.д.). Программа включает обработку исключений и проверку входных данных.

оценка «4» - Реализованы основные классы и методы. Применены базовые принципы ООП (инкапсуляция, наследование). Программа работает без критических ошибок. Использован полиморфизм для улучшения функциональности. Программа демонстрирует хорошую структуру и читаемость кода.

оценка «3» - Реализованы основные классы и методы. Применены базовые принципы ООП (инкапсуляция, наследование). Программа работает без критических ошибок.