бюджетное профессиональное образовательное учреждение Вологодской области

«Череповецкий лесомеханический техникум им. В.П. Чкалова»

Специальность **09.02.07** «Информационные системы и программирование»

**ОТЧЕТ ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ**

**ПП по ПМ.02 Осуществление интеграции программных модулей**

Выполнил студент 2 курса группы ИС-\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

место практики\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Период прохождения:

с «\_\_\_» \_\_\_\_\_\_\_ 2024 г.

по «\_\_\_» \_\_\_\_\_\_\_ 2024 г.

Руководитель практики от

предприятия

должность\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

подпись\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Руководитель практики от

техникума: Материкова А.А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Оценка:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2024 года

г. Череповец

2024

Содержание

Введение

Место прохождения производственной практики – ООО “Малленом Системс”

Сроки прохождения производственного практики – с 10.11.24 по 23.11.24

Цель производственной практики — улучшить подготовку студента в выбранной области, познакомить его с тонкостями профессии на реальном производстве, дать возможность приобрести и закрепить практические навыки, а также развить профессиональные качества и лучше понять свою специальность.

Задачи производственной практики:

1. Укрепить теоретические знания, полученные во время учёбы;
2. Получить практический опыт по специальности;
3. Узнать, как работают и устроены профильные предприятия изнутри;
4. Понять, какие обязанности будет выполнять специалист на рабочем месте;
5. Изучить стандарты основных направлений производственной деятельности.

**Задание 1 (11.11.2024)**

Общее описание предприятия

Структура организации

Генеральный директор – Анна Эдуардовна Живиця

Учредители – 10 физических лиц

Число сотрудников – свыше 100 человек

Внутренняя организация работы и охрана труда на предприятии

График работы

С понедельника по пятницу с 9:00 до 18:00

Охрана труда

На предприятии созданы все необходимые условия для безопасной работы сотрудников. Регулярно проводятся обучения по охране труда, применяются современные средства индивидуальной защиты, осуществляется соблюдение правил пожарной безопасности.

Должностные инструкции IT-специалистов компании

Должностные инструкции IT-специалистов компании «Маленное Системс» определяют их обязанности, права и ответственность. В зависимости от должности, специалисты занимаются разработкой программного обеспечения, его тестированием, технической поддержкой, администрированием серверов и баз данных, а также другими задачами.

Интеграция программных модулей

Формулирование требований к программным модулям на основе анализа проектной и технической документации для оценки взаимодействия компонентов.

Выполнение интеграции модулей в программное обеспечение.

Проведение отладки программного модуля с использованием специализированных инструментов.

Разработка тестовых наборов и сценариев для программного обеспечения.

Проверка компонентов программного обеспечения на соответствие стандартам кодирования.

**Задание 2 (12.11.2024)**

1. Создание GIT репозитория:

- Создан GIT-репозиторий для производственной практики с необходимой структурой.

Структура репозитория:

- Отчет

- Задания

- Документы

Ссылка на репозиторий:

Иванов Артём Вячеславович, ссылка на репозиторий - https://github.com/Wensday1337/Ivanov

2. Добавление отчета и документов на практику:

- В репозиторий добавлены:

- Отчет о выполненной практике, включающий подробное описание первого задания.

- Документы, связанные с практикой.

**Задание 3 (13.11.2024)**

Введение

В рамках производственной практики были разработаны два модуля для обработки изображений и взаимодействия с пользователем. Заказчиком выступила компания ООО «Малленом Системс». Данный отчет содержит описание выполненных заданий, а также основные моменты, связанные с проектированием, реализацией и тестированием модулей.

Цели и задачи

- Разработать модуль обработки изображений, который будет поддерживать изменения размера и поворот изображений.

- Создать модуль взаимодействия с пользователем, который позволит собирать данные о входных изображениях и передавать их для обработки.

Разработка модулей

1. Модуль обработки и работы с изображениями

Модуль был назван image\_processing.py и включает в себя следующие функции:

1. Изменение размера изображения:

- Функция принимает путь к входному изображению, путь для сохранения измененного изображения, а также новые размеры (ширину и высоту).

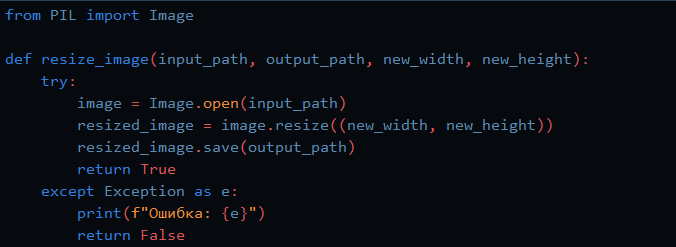
- Используется библиотека Pillow для выполнения данной операции.

2. Поворот изображения:

- Функция принимает путь к входному изображению, путь для сохранения повёрнутого изображения и угол поворота.

- Также реализована обработка ошибок для случаев, когда файл не найден или имеет неподдерживаемый формат.

Пример реализации функции изменения размера:



Модуль взаимодействия с пользователем

Модуль был назван user\_interface.py и обеспечивает сбор пользовательского ввода и взаимодействие с модулем обработки изображений.

Основные функции модуля:

1. Получение пользовательского ввода:

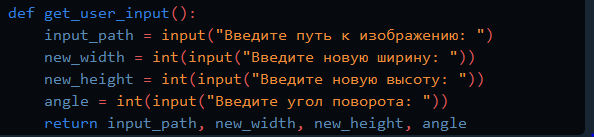
- Модуль запрашивает у пользователя путь к изображению, а также новые размеры и угол поворота.

- Реализована проверка корректности введенных данных, необходимо отображать ошибки, если данные введены неправильно.

2. Вызов функций обработки изображений:

- После получения данных от пользователя модуль вызывает соответствующие функции обработки изображений и выводит сообщения об успешной операции или ошибках при выполнении.

Пример реализации функции для получения ввода от пользователя:



Инструкция по запуску:

1. Установите библиотеку Pillow:

Убедитесь, что у вас установлена библиотека Pillow:

pip install Pillow

2. Создайте файлы:

Сохраните первый модуль как image\_processing.py, а второй как user\_interface.py.

3. Запустите пользовательский интерфейс:

В командной строке выполните:

python user\_interface.py

4. Ввод данных:

Следуйте инструкциям на экране, чтобы ввести путь к изображению, новые размеры и угол поворота.

Заключение

В результате работы были успешно разработаны и протестированы два модуля: для обработки изображений и взаимодействия с пользователем. Оба модуля продемонстрировали правильность работы в различных сценариях и соответствовали требованиям технического задания.

**Задание 4 (18.11.2024)**

1. PEP 8 — это стиль кодирования для языка Python, который помогает поддерживать единообразие и читаемость кода. Ниже приводятся основные рекомендации PEP 8:

Отступы:

- Используйте 4 пробела на уровень отступа. Не используйте табуляцию.

Максимальная длина строки:

- Строки кода не должны превышать 79 символов. Для комментариев и документации — 72 символа.

Пропуски:

- Разделяйте функции и классы с двумя пустыми строками.

- Разделяйте методы внутри классов одной пустой строкой.

- Используйте один пробел перед и после операторов (например, a = b + c).

Импортируемые модули:

- Импортируйте стандартные библиотеки, сторонние библиотеки и локальные модули в следующем порядке, каждая группа должна быть разделена пустой строкой.

Именование:

- Используйте snake\_case для имен переменных и функций.

- Используйте CamelCase для имен классов.

- Константы должны именоваться UPPER\_SNAKE\_CASE.

Комментирование:

- Комментарии должны быть ясными и объяснять "почему" код делает что-то, а не "что" он делает.

- Однострочные комментарии должны начинаться с заглавной буквы и быть отделены от кода двумя пробелами.

- Многострочные комментарии должны быть отделены пустыми строками.

Строки документации (docstrings):

- Используйте строки документации для описания функций и классов. Они должны быть в тройных кавычках и пояснять назначение функции, ее параметры и возвращаемые значения.

Сравнение с None:

- Для сравнения с None используйте is и is not, а не == или != (например, if obj is None:).

Использование пробелов в выражениях:

- Не добавляйте пробелы перед запятыми, точками с запятой или двоеточиями (например, func(a, b) вместо func(a , b)).

Тернарные операторы:

- Используйте явные конструкции if-else для многострочных условий. Тенарные операторы должны использоваться аккуратно.

2. Инспектирование модуля image\_processing.py по стандарту PEP 8:

1. Отступы: В функции отсутствует правильное форматирование отступов (4 пробела). Следует исправить отступы для комментариев и кода.

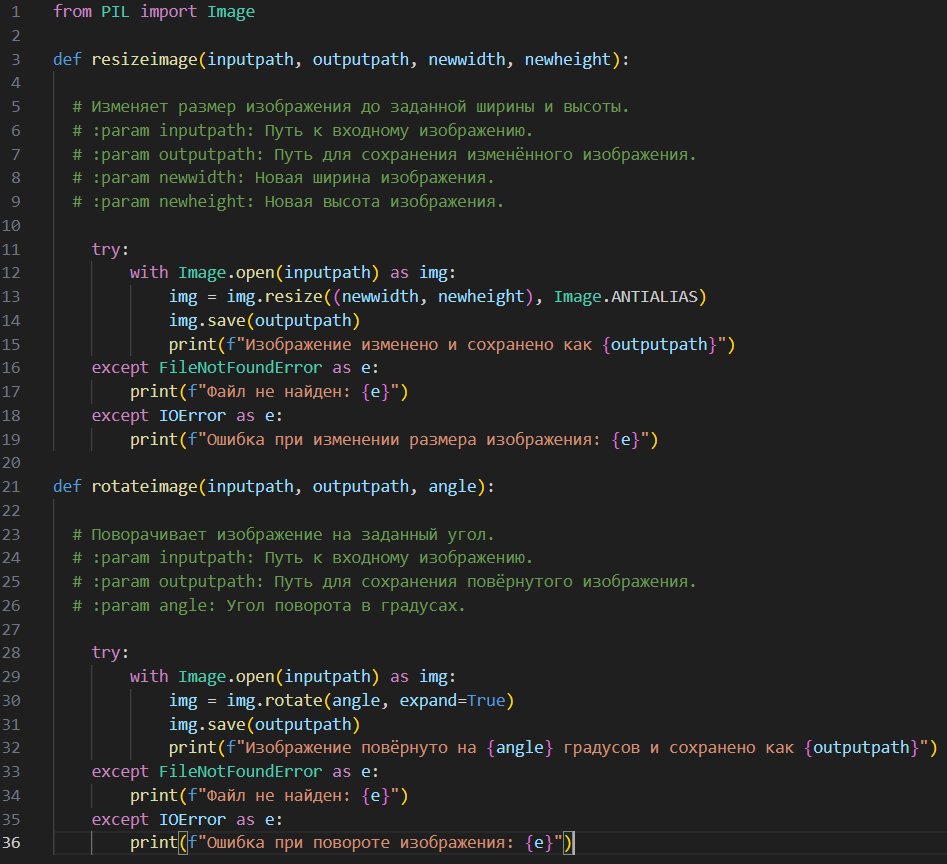
2. Пробелы: Необходимо добавлять два пробела перед комментариями для лучшей читаемости.

3. Кодовые строки: Рекомендуется использовать длинные строки комментариев и кода с учетом максимальной длины строки в 79 символов.

4. Импорт: Импортируйте модули в первую очередь перед определением функций.

5. Исключения: Рекомендуется использовать более специфичные исключения вместо общего `Exception`.

Исправленный код:



Инспектирование модуля user\_interface.py по стандарту PEP 8:

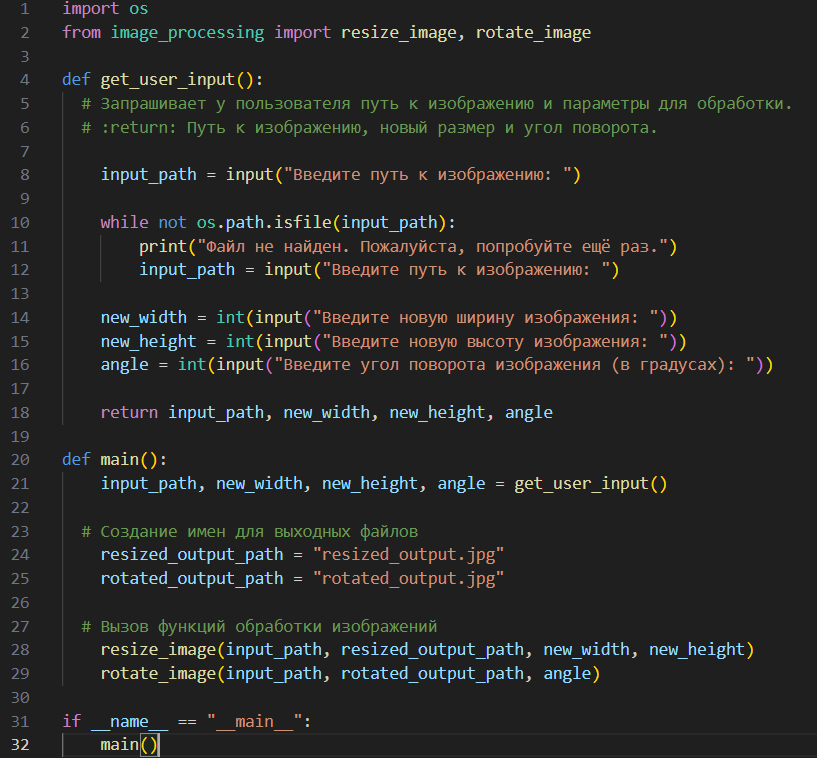
1. Отступы: Обратите внимание на правильное использование отступов (4 пробела).

2. Комментарии: Для более ясного выделения комментариев используйте пробелы.

3. Импорт: Импортируйте модули в первой части файла.

4. Проверка имени: Убедитесь, что используете двойные подчеркивания в if \_\_name\_\_ == "\_\_main\_\_":.

Исправленный код:



Общие рекомендации:

- Старайтесь поддерживать однородность в стиле кода во всех модулях.

- Убедитесь, что все функции и классы имеют соответствующие комментарии и документацию.

- Не забывайте тестировать код после внесения изменений.

3. Интеграция модулей друг с другом предполагает создание связей и взаимодействия между ними для обеспечения корректной работы всей системы. В моём коде коде интеграцию создает импортирование модулей:



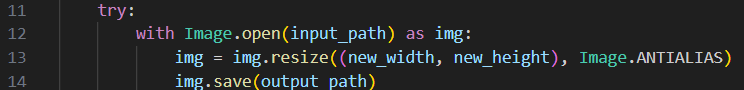
А также взаимодействие через функции: Когда пользователь вводит данные (путь к изображению, размеры и угол поворота), вы передаёте эти данные в функции resize\_image и rotate\_image.:

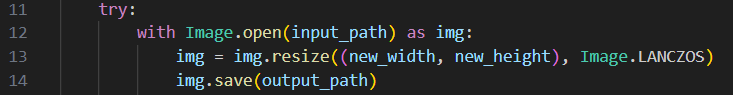


4. При проведении откладки VScode выдавал ошибки:



Это исправляется благодаря изменению строк, так как новая версия PIL не поддерживает ‘ANTIALIAS’:





А также в этой строке “resized\_output.jpg” и “rotated\_output.jpg” заменил формат изображения на “png”



5. Проведение тестирования программных модулей

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Наименование  теста | Тестовые  данные | Ожидаемый  результат | Фактический  результат | Результат  тестирования | Комментарий |
| Test1 | Ширина: 900  Длина: 300  Угол: 90 | Создание 2 изображений, где одно повернуто на 90 градусов, а второе изменило свое разрешение | Создание 2 изображений, где одно повернуто на 90 градусов, а второе изменило свое разрешение | Код работает отлично, без ошибок, но принимает только .png | - |
| Test2 | Ширина: 1920  Длина: 200  Угол: 1 | Создание 2 изображений, где одно повернуто на 1 градус, а второе изменило свое разрешение | Создание 2 изображений, где одно повернуто на 1 градус, а второе изменило свое разрешение |