P1 Asm to Python

Converting Legacy Code to Python: A Step-by-Step Guide

Introduction

In this document, we will outline the process of converting legacy code to Python using Box AI. This guide is intended to be a helpful resource for others who may need to undertake a similar task.

Legacy Code Overview

The legacy code in question is used to calculate the final discount price of a product. The original code is written in a proprietary language and requires conversion to Python for integration with our modern software stack.

Step 1: Understand the Legacy Code

Before converting the legacy code, it is essential to understand its functionality and logic. Review the code carefully, and take note of the following:
- Input parameters
- Calculations and formulas used
- Conditional statements and loops
- Output values

Step 2: Identify Equivalent Python Libraries and Functions

Once we understand the legacy code, we need to identify equivalent Python libraries and functions that can replicate its functionality. In this case, we will use Python's built-in `math` library for calculations.

Step 3: Convert Legacy Code to Python

Using the knowledge gained from the previous steps, we can now convert the legacy code to Python. We will break down the code into smaller sections and translate each part into Python.

Legacy Code

```
1   // Calculate discount price
2   discount_price = original_price - (original_price * discount_percentage / 100)
```

Python Equivalent

```
1   import math
2
3   def calculate_discount_price(original_price, discount_percentage):
4       discount_price = original_price - (original_price * discount_percentage / 100)
5       return discount_price
```

Step 4: Test and Validate the Python Code

After converting the legacy code to Python, it is crucial to test and validate the new code to ensure it produces the same results as the original code. We will write unit tests to verify the functionality of the `calculate_discount_price` function.

Test Code

```
1   import unittest
```