

|  |
| --- |
| **from** datetime **import** datetime   **def** calculate**(**price**:** int**,** year**:** int**,** category**:** str**)** **->** float **or** str**:**  # Uses the datetime library to get current year for calculation  current\_year **=** datetime**.**now**().**year  # Conditional statement tests  **if** category **==** 'A'**:**  percentage **=** 0.25  **elif** category **==** 'B'**:**  percentage **=** 0.5  **else:**  percentage **=** 0.75   initial\_amount **=** percentage **\*** price  year\_difference **=** current\_year **-** year  # Depreciation amount is equal 1% of the price times by how many years old.  depreciation\_amount **=** year\_difference **\*** **(**0.01 **\*** price**)**   # Conditional statement tests if the depreciation is greater than 10%  **if** depreciation\_amount **>=** **(**0.1 **\*** price**):**  depreciation\_amount **=** 0.1 **\*** price  final\_amount **=** initial\_amount **-** depreciation\_amount  **return** category**,** initial\_amount**,** year\_difference**,** depreciation\_amount**,** final\_amount   **def** repeat\_process**(**stop**):**  # To loop dialog if an unexpected input is entered.  **while** **True:**  # Asking user if they wish to continue  answer **=** input**(**'Calculate for another vehicle (y/n)? '**)**  # Conditional statement checks user input, while loop breaks if the user input is 'y' or 'n'  **if** 'n' **in** answer**.**lower**():**  **print(**'Have a nice day!'**)**  stop **=** **True**  **break**  **elif** 'y' **in** answer**.**lower**():**  **print(**'-' **\*** 40**)**  **break**  **else:**  **print(**'Unexpected input. Please try again.'**)**  **return** stop   **if** \_\_name\_\_ **==** '\_\_main\_\_'**:**  # Initialising main loop  end **=** **False**  **while** **not** end**:**  **print(**'Utopia Vehicle Customs Calculator'**)**  **print(**'-' **\*** 40**)**   # Asking for user input for initial values.  market\_price **=** int**(**input**(**'Enter the market price: '**))**  engine\_capacity **=** int**(**input**(**'Enter the engine capacity: '**))**  manufacturer\_year **=** int**(**input**(**'Enter the manufacture year: '**))**   # Conditional statement checks the engine capacity for which category should be used.  **if** engine\_capacity **<=** 1600**:**  tier**,** initial**,** year\_diff**,** depreciation**,** final **=** calculate**(**market\_price**,** manufacturer\_year**,** 'A'**)**  **elif** 2000 **>** engine\_capacity **>** 1600**:**  tier**,** initial**,** year\_diff**,** depreciation**,** final **=** calculate**(**market\_price**,** manufacturer\_year**,** 'B'**)**  **elif** engine\_capacity **>=** 2000**:**  tier**,** initial**,** year\_diff**,** depreciation**,** final **=** calculate**(**market\_price**,** manufacturer\_year**,** 'C'**)**   # Displays information returned from the calculate function  **print(**f'Initial customs amount (Category {tier}): ${initial:,.2f}'**)**  **print(**f'Depreciation discount ({year\_diff} years): ${depreciation:,.2f}'**)**  **print(**f'Final customs amount: ${final:,.2f}'**)**   end **=** repeat\_process**(**end**)** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Type** | **Test Data** | **Reason it was selected** | **Expected output** | **Screenshot of actual output** |
| Normal | Market price: 21500 Engine Capacity: 2000 Manufacture year: 2006 Continue: n | Testing boundary condition for category C (engine capacity 2000 or higher) | No errors,  Category C,  15 years depreciation,  Initial amount 75% of market price: $16,125 |  |
| Normal | Market price: 21500 Engine Capacity: 1600  Manufacture year: 2020  Continue: n | Testing boundary condition for category A (engine capacity 1600 or lower) | No errors, Category A, 1 year depreciation, Initial amount 25% of market price: $5,375 |  |
| Normal | Market price: 21500 Engine Capacity: 1600  Manufacture year: 2020  Continue: n | Testing if prompt to calculate another works correctly | No errors,  Category B,  6 years depreciation,  Initial amount 50% of market price: $10,750  “Utopia Vehicle customs calculator “  “enter the market price:” |  |
| Abnormal | Market price:  One thousand | Testing if the market price prompt will take string input rather than integers | Value error |  |
| Abnormal | Market price:  21500  Engine Capacity:  1600  Manufacture year: 2019  Continue: yn | Testing if continue calculation prompt works with incorrect input | The program will exit with this input as the if statement only checks if the letter is present, not if other unexpected letters are in the string as well. |  |