Complete Variable Breakdown by Data Type

📝 STRING Variables (Text/Words)

Instance Variables (self.variables):

• `self.name` - Stores user's full name (e.g., "John Smith") • `self.color` - Stores chosen shoe color (e.g., "Red", "Blue") • `self.traction` - Stores traction type (e.g., "Turf", "Grass") • `self.support` - Stores support level (e.g., "High", "Mid", "Low") • `self.design` - Stores design choice (e.g., "Classic TA", "No design selected") • `self.discount_reason` - Stores why discount was given (e.g., "New customer welcome discount! 🎉")

Local String Variables:

• `choice` - User input as string (e.g., "1", "2", "yes", "no") • `size_input` - User's size input before conversion (e.g., "9.5") • `wants_design` - User's design preference input (e.g., "yes", "no") • `save_choice` - User's file save choice (e.g., "yes", "no") • `restart_choice` - User's restart choice (e.g., "yes", "no") • `timestamp` - Date/time string (e.g., "20250121_143052") • `safe_name` - Cleaned version of name for filename (e.g., "John_Smith") • `filename` - Full file path (e.g., "lacrosse_shoe_receipt_John_Smith_20250121.txt")

Constant String Lists:

• `self.COLORS` - List of color options ["Red", "Blue", "White", "Black"] • `self.TRACTION_TYPES` - List of traction options ["Turf", "Grass", "All-Terrain"] • `self.SUPPORT_LEVELS` - List of support options ["Low", "Mid", "High"] • `self.DESIGNS` - List of design options ["Classic TA", "Modern TA", "Bold TA", "Minimal TA"]

🔢 INTEGER Variables (Whole Numbers)

Local Integer Variables:

• `choice_num` - Converted user choice (e.g., 1, 2, 3, 4) • `i` - Loop counter in enumerate functions (1, 2, 3, 4)

Used in Functions:

• `enumerate(self.COLORS, 1)` - Creates integers 1, 2, 3, 4 for menu numbering • `enumerate(self.TRACTION_TYPES, 1)` - Creates integers 1, 2, 3 for menu • `enumerate(self.SUPPORT_LEVELS, 1)` - Creates integers 1, 2, 3 for menu • `enumerate(self.DESIGNS, 1)` - Creates integers 1, 2, 3, 4 for menu

Range Checking:

• `len(self.COLORS)` - Integer 4 (number of color options) • `len(self.TRACTION_TYPES)` - Integer 3 (number of traction options) • `len(self.SUPPORT_LEVELS)` - Integer 3 (number of support options) • `len(self.DESIGNS)` - Integer 4 (number of design options)

💰 FLOAT Variables (Decimal Numbers)

Instance Variables (self.variables):

• `self.size` - Shoe size with decimals (e.g., 9.5, 10.0, 11.5) • `self.base_cost` - Base shoe price (e.g., 100.0, 110.0, 120.0) • `self.discount` - Discount percentage (e.g., 5.0, 10.0, 15.0, 20.0) • `self.final_price` - Final calculated price (e.g., 85.50, 102.00)

Constant Float Variables:

• `self.MIN_SIZE = 5.0` - Minimum allowed shoe size • `self.MAX_SIZE = 15.0` - Maximum allowed shoe size

Local Float Variables:

• `base_price = 100.0` - Starting price for all shoes • `support_cost` - Additional cost for support (0.0, 10.0, or 20.0) • `discount_amount` - Dollar amount of discount (calculated from percentage)

Used in Calculations:

• Price calculations: `self.base_cost * (self.discount / 100)` • Support pricing: `20.0` for High, `10.0` for Mid, `0.0` for Low • Random discount values: `5, 10, 15, 20, 8, 12, 0` (stored as floats)

✅❌ BOOLEAN Variables (True/False)

Function Return Values:

• `get_user_name()` returns `True` (when name is valid) or continues loop • `restart_program()` returns `True` (restart) or `False` (exit)

Boolean Conditions (Used in if statements):

• `self.name != ""` - True if name is not empty • `self.name.replace(" ", "").isalpha()` - True if name contains only letters • `self.size >= self.MIN_SIZE and self.size <= self.MAX_SIZE` - True if size is valid • `1 <= choice_num <= len(self.COLORS)` - True if choice is in valid range • `wants_design in ['yes', 'y', 'true', '1']` - True if user wants design • `save_choice in ['yes', 'y', 'true', '1']` - True if user wants to save • `restart_choice in ['yes', 'y', 'true', '1']` - True if user wants to restart

Boolean Logic Expressions:

• `choice_num >= 1 and choice_num <= max_options` - Range validation • `self.support == "High"` - Exact string comparison (True/False) • `filename.endswith('.txt')` - File extension check (True/False)

📊 Summary Count

Data Type	Count	Examples
String	15+ variables	`self.name`, `self.color`, `choice`
Integer	8+ variables	`choice_num`, `i`, `len()` results
Float	10+ variables	`self.size`, `self.final_price`, `support_cost`
Boolean	10+ expressions	Function returns, condition checks

🎯 Key Observations

Most Used Data Types:

1. **Strings** - For user input, choices, and text storage
2. **Floats** - For all money calculations and shoe sizes
3. **Booleans** - For validation and program flow control
4. **Integers** - For menu choices and counting

Smart Design Choices:

• Used `float` for prices (handles cents: $10.50) • Used `float` for shoe sizes (handles half sizes: 9.5) • Used `string` for user input (flexible and safe) • Used `boolean` returns for clean function logic