📅 WEEK 1 – Learn & Build the Core

Day	Focus	Tasks
Day 1	🚀 Python Basics + Setup	Learn `variables`, `print()`, `input()` → Set up your Python file in VS Code → Print "Welcome to Lacrosse Shoe Customizer!"
Day 2	🧱 Functions (Part 1)	Learn how to define and call a function → Create `choose_color()` and `choose_size()` functions
Day 3	🧠 Functions (Part 2) + Variables	Add `choose_traction()` and `choose_support()` → Store all choices in variables and return them
Day 4	🔁 Loops + Error Handling	Learn `while` loops & `try/except` → Let user retry input if they type something invalid
Day 5	🧮 Conditional Logic + Cost	Use `if/else` to assign cost or output based on features (e.g., "high ankle = +$20") → Add function to calculate total price

---

📅 WEEK 2 – Polish, Test, Expand

Day	Focus	Tasks
Day 6	🛠️ Combine + Build Main Menu	Build the main program loop: → Ask if user wants to start → Call all functions in order → Display full shoe summary
Day 7	🧪 **Troubleshooting & Bug Fixes**	Test all paths: → Try invalid input → See if prices and selections update properly → Fix bugs (I'll help!)
Day 8	💾 Save Results (Optional)	Learn how to save to a `.txt` file: → Output user's choices and total price to a receipt file
Day 9	🎨 Optional GUI (Tkinter) Start	Begin converting project to a GUI using `Tkinter` (if you're ready) → Replace `input()` with dropdowns or buttons

Day 10	🌟 Final Polishing & Presentation	Final cleanup, formatting, comments → Add your name/date to top of file → Prepare to demo it or share the code

Perfect! Here's how we'll update your **10-day plan and code structure** for the **Toddavery Lacrosse Shoe** project to include:

- 🔠 Multiple **TA initial design options**
- 🎲 A **random discount generator** with fun reasons
- 🛍️ Final price calculation with the discount applied

🗓️ Updated 10-Day Plan (New Features Integrated)

Day	Focus	Task Updates
Day 1	Python Setup	Print welcome, get user name ✅
Day 2	Functions Pt. 1	Add `choose_color()` and `choose_size()`
Day 3	Functions Pt. 2	Add `choose_traction()` and `choose_support()`

Day 4	TA Initial Design	Create `choose_design()` → user picks from multiple “TA” logos
Day 5	Conditional Logic + Cost	Add `calculate_cost()` + `calculate_discount()`
Day 6	Main Program Loop	Combine all inputs and display final summary
Day 7	Troubleshooting	Test logic, input errors, fix bugs
Day 8	Save to File	Add `save_to_file()` including TA design + discount reason
Day 9	Optional GUI (Tkinter)	Build dropdowns/buttons if time permits
Day 10	Final Polish	Clean up code, comments, format for demo

🎨 TA Initial Design Options

Add a new function to choose between **visual styles** of “TA”:

python
CopyEdit

```python
def choose_design():
    print("\n🖌️ Choose your TA design:")
    print("1. [T A]")
    print("2. ⌈T⌉⌈A⌉")
    print("3. (T★A)")
    print("4. TA™")
    print("5. TΛ - Aᴛ")

    choice = input("Enter the number of
your design: ")

    designs = {
        "1": "[T A]",
        "2": "⌈T⌉⌈A⌉",
        "3": "(T★A)",
        "4": "TA™",
        "5": "TΛ - Aᴛ"
    }

    return designs.get(choice, "TA")  #
Default to plain TA
```

🎲 Discount Generator with Fun Reasons

python
CopyEdit

```
import random

def calculate_discount():
    discounts = [
        (10, "🏆 Toddavery Loyalty
Bonus!"),
        (15, "🎉 New Customer
Special"),
        (5, "📦 Shipping Delay
Discount"),
        (20, "🔥 Limited Time Offer"),
        (8, "💡 Smart Choice Saver"),
    ]
    discount = random.choice(discounts)
    return discount  # Returns tuple:
(amount, reason)
```

🧮 Final Cost Function (With Discount)

Update your cost function to subtract the discount:

python
CopyEdit

```
def calculate_final_price(base_cost,
discount_amount):
    return base_cost - discount_amount
```

✅ Where to Use These in Your Code

Add these lines in your `main()` function (after calculating the base cost):

python
CopyEdit

```
design = choose_design()
discount_amount, discount_reason =
calculate_discount()
final_price =
calculate_final_price(cost,
discount_amount)

show_summary(name, color, size,
traction, support, cost,
discount_amount, discount_reason,
final_price)
save_to_file(name, color, size,
traction, support, design, cost,
discount_amount, discount_reason,
final_price)
```

📝 Updated Summary Function (with discount + design)

python

CopyEdit

```
def show_summary(name, color, size,
traction, support, base_cost,
discount_amt, discount_reason,
final_price):
    print("\n📝 Toddavery Shoe
Summary:")
    print(f"Customer: {name}")
    print(f"Color: {color}")
    print(f"Size: {size}")
    print(f"Traction: {traction}")
    print(f"Support: {support}")
    print(f"Original Cost:
${base_cost}")
    print(f"Discount: -${discount_amt}
→ {discount_reason}")
    print(f"💰 Final Price:
${final_price}")
```