

# Python with AI

Week	Focus	Key Topics	Resources	Weekly Assignments & Milestone
1	Python Basics	Variables, data types, operators, conditionals, loops, functions, lists, strings, file I/O	freeCodeCamp Python (first 4 hours), <a href="https://python.org">Python.org</a> tutorial, Automate the Boring Stuff (Ch 1–6)	<ul style="list-style-type: none"><li>• Build BMI calculator, grade calculator, number guessing game</li><li>• Simple todo list CLI (add/delete/save to file)</li></ul> Milestone: Contact manager CLI app
2	Python Intermediate + OOP	Dictionaries, sets, comprehensions, modules, exceptions, classes/objects, inheritance	Automate the Boring Stuff (Ch 7–11), Real Python OOP articles	<ul style="list-style-type: none"><li>• Build a bank account class system</li><li>• Inventory manager with classes</li><li>• Simple text-based game (e.g., tic-tac-toe)</li></ul> Milestone: OOP-based student management system
3–4	DSA in Python	Arrays/strings, sorting/searching, linked lists, stacks/queues, recursion, trees (binary/BFS/DFS), hash maps, basic DP	GeeksforGeeks DSA Python, TakeUForward YouTube playlist, LeetCode Python	<ul style="list-style-type: none"><li>• Solve 80 Easy + 30 Medium LeetCode problems</li><li>• Implement stack-based</li></ul>

Week	Focus	Key Topics	Resources	Weekly Assignments & Milestone
				browser history, queue task scheduler • Build binary tree with traversal Milestone: Clothing inventory app using trees/hashmaps for fast size search
5	Frontend Basics	HTML/CSS, Bootstrap, JavaScript basics, React (components, props, state, hooks)	freeCodeCamp Responsive Web + React sections	• Static clothing catalog page • Interactive measurement form (validation + state) Milestone: React form for body measurements + skin color picker
6	Flask Backend	Flask setup, routes, templates, REST APIs, forms, SQLite/PostgreSQL	freeCodeCamp Flask tutorial, Flask official docs	• CRUD API for users & clothes • Store measurements in DB • Connect React form to Flask API Milestone: Full-stack measurement storage app

Week	Focus	Key Topics	Resources	Weekly Assignments & Milestone
7	Advanced Flask + Tools	Blueprints, JWT auth, file uploads, calling external Python scripts, Git, virtualenv	Miguel Grinberg Flask Mega-Tutorial (Ch 1–10)	<ul style="list-style-type: none"> <li>• Add user login/register (JWT)</li> <li>• Photo upload endpoint</li> <li>• Call mock SMPL script from Flask</li> </ul> Milestone: Authenticated app with photo upload
8	3D Pipeline Integration	OpenCV basics, subprocess for Blender/COLMAP, SMPL fitting, Three.js viewer setup	Previous POC guide, Three.js docs	<ul style="list-style-type: none"> <li>• Integrate manual measurement → SMPL mesh script</li> <li>• Basic Three.js viewer for static models</li> <li>• Flask endpoint to return processed model</li> </ul> Milestone: Manual measurement → 3D model viewer (static)
9	Core 3D Processing	COLMAP reconstruction, SMPLify-X, M3D-VTON basics, Blender Python scripting	Repos: COLMAP, SMPLify-X, M3D-VTON	<ul style="list-style-type: none"> <li>• Run photo → 3D reconstruction pipeline locally</li> <li>• Add photo-based path alongside manual</li> <li>• Basic clothing draping script</li> </ul> Milestone: Photo upload → 3D

Week	Focus	Key Topics	Resources	Weekly Assignments & Milestone
				body model in viewer
10	Virtual Try-On	Clothing swap logic, size scaling from measurements, mock brand catalog, skin color texturing	Three.js examples, Blender cloth sim basics	<ul style="list-style-type: none"> <li>• Clothing selector UI</li> <li>• Swap clothes on model (scale by measurements)</li> <li>• Apply skin tone material</li> </ul> Milestone: Working clothing try-on with 3–5 items
11	Mini Project (POC)	Full end-to-end POC: photos or manual → 3D model → try-on viewer	Combine all previous work	<ul style="list-style-type: none"> <li>• Polish UI/UX</li> <li>• Error handling</li> <li>• Test with 10 sample cases</li> </ul> Milestone: Complete local POC demo
12	Deployment & Optimization	Deployment (Heroku/Render/Vercel), basic system design (DB schema, API docs), performance tweaks	Flask deployment guides, React deploy	<ul style="list-style-type: none"> <li>• Deploy backend + frontend</li> <li>• Add mock brand API</li> <li>• Final polish + README</li> </ul> Milestone: Live deployed app + GitHub portfolio + demo video
13	Intro to Gen AI & PyTorch	ML basics, PyTorch tensors, neural networks,	PyTorch tutorials, Hugging Face	<ul style="list-style-type: none"> <li>• Build simple NN for classification</li> </ul>

Week	Focus	Key Topics	Resources	Weekly Assignments & Milestone
		pre-trained models, Hugging Face basics	course (free)	(e.g., MNIST) • Load/run pre-trained model from HF Milestone: Texture classifier script for clothes
14	Diffusion Models & Gen Tools	Diffusion basics (GANs intro, Stable Diffusion), Diffusers library, generative 3D (e.g., TripoSR, SIFU)	Diffusers docs, Hugging Face Diffusion course, SIFU repo	• Run Stable Diffusion for image gen • Integrate SIFU for clothed human recon Milestone: Generate custom clothing texture from text prompt
15	Advanced Gen AI for Try-On	VITON variants (StableVITON, IDM-VTON, CatVTON), DrapeNet for garment gen, collision handling	VITON repos (e.g., IDM-VTON), DrapeNet repo	• Clone/run IDM-VTON demo • Enhance try-on with generative draping (DrapeNet) Milestone: Integrate CatVTON for realistic 2D-to-3D try-on lift
16	Gen AI Project Integration	Fine-tuning basics, ethical AI, full app enhancement (e.g., gen custom clothes, AI textures)	Project repos + PyTorch fine-tune guides	• Add gen AI feature to main project (e.g., text-to-clothing swap) • Test/debug with

Week	Focus	Key Topics	Resources	Weekly Assignments & Milestone
				variations Milestone: Updated deployed app with Gen AI features + final portfolio

## 4-Month Training Plan for Freshers: Building Wearables E-Commerce with Virtual Try-On

This plan assumes 10 freshers with **zero programming knowledge**, split into **Team 1 (Virtual Try-On: 5 members)** and **Team 2 (E-Commerce Website: 5 members)**. The structure is **3 months of core tech stack training** (building foundational skills) + **1 month mini project** (hands-on application). Post-training, they develop the full product over 1 year, with cross-team integration in later stages.

### General Guidelines (for both teams):

- **Daily Routine:** 6-8 hours/day. 1-2 hours theory (videos/readings), 4-5 hours coding practice, 1 hour team discussions/reviews.
- **Resources:** Free/affordable (freeCodeCamp, GeeksforGeeks, Udemy on sale, official docs). Use VS Code/Replit for coding; GitHub for collaboration.
- **Evaluation:** Weekly quizzes, code reviews, and progress demos. Encourage pair programming.
- **Tech Stack:** Python-centric (Flask backend), React (frontend), shared basics. Team 1 emphasizes AI/3D/Gen AI; Team 2 focuses on web/e-commerce integrations.
- **Collaboration:** Weekly joint sessions for integration planning (e.g., APIs between try-on and e-commerce).

### Shared Phase (Weeks 1-4: Python Basics & DSA for All 10 Freshers)

Train together to build a common foundation.

- **Focus:** Programming intro, data handling, problem-solving.
- **Key Topics:** Variables, loops, functions, lists/dicts, OOP, file I/O, basic DSA (arrays, sorting/searching, stacks/queues, recursion).

- **Resources:** freeCodeCamp Python, Automate the Boring Stuff (Ch 1-11), GeeksforGeeks DSA Python, LeetCode easy problems.
- **Weekly Assignments:**
  - Week 1: Build calculators/games (e.g., BMI tool, guessing game).
  - Week 2: OOP projects (e.g., inventory class system).
  - Week 3-4: DSA exercises (solve 50 LeetCode easy; build queue-based shopping cart simulator).
- **Milestone:** Group project: Simple CLI e-commerce simulator (add items, calculate totals) using DSA.

## **Team 1: Virtual Try-On Learning Path (Weeks 5-12 Training + Weeks 13-16 Mini Project)**

Focus: Build the 360-degree avatar system with measurements/skin tone, Gen AI for textures/draping, multi-item try-on (clothes/shoes/accessories), fabric booklet integration (digital previews via Gen AI).

- **Weeks 5-6: Web Basics & 3D Foundations**
  - Topics: HTML/CSS/JS, React (components, state, hooks); OpenCV for image processing; Three.js for 3D rendering.
  - Resources: freeCodeCamp React, Three.js docs.
  - Assignments: Build measurement form UI; simple 3D viewer for static models.
  - Milestone: React app for user inputs (photos/measurements) + basic 3D render.
- **Weeks 7-8: Backend & 3D Processing**
  - Topics: Flask setup, APIs, file uploads; COLMAP/SMPL for reconstruction; Blender scripting for simulation.
  - Resources: Flask docs, POC repos (COLMAP, SMPLify-X).
  - Assignments: Photo upload endpoint; reconstruct 3D model from sample images.
  - Milestone: Flask backend serving 3D avatars from inputs.
- **Weeks 9-12: Gen AI & Advanced Try-On**
  - Topics: PyTorch basics, Diffusion models (Stable Diffusion/Diffusers); M3D-VTON/IDM-VTON for draping; Gen AI for fabric textures/skin tones; multi-garment layering.
  - Resources: Hugging Face courses, PyTorch tutorials, VITON repos.
  - Assignments: Generate textures from prompts; integrate try-on with size scaling; add fabric booklet preview (Gen AI sim + PDF gen).
  - Milestone: Full try-on demo with Gen AI enhancements (e.g., custom fabric visuals).
- **Weeks 13-16: Mini Project**
  - Build a prototype: User uploads photos/measurements → 360-degree avatar → try-on full outfit → Gen AI texture/fabric preview.

- Assignments: Weekly sprints (e.g., Week 13: Avatar gen; Week 14: Try-on layering; Week 15: Gen AI integration; Week 16: Testing/polish).
- Milestone: Demo app with 5-10 test outfits; integrate basic fabric booklet request (mock shipping).

## **Team 2: E-Commerce Website Learning Path (Weeks 5-12 Training + Weeks 13-16 Mini Project)**

Focus: Build the marketplace for wearables (clothes, shoes, ties, watches, glasses, caps), brand/product integrations, cart/checkout, user accounts, fabric booklet fulfillment.

### **• Weeks 5-6: Frontend & Basics**

- Topics: HTML/CSS/JS, Bootstrap, React (advanced: routing, forms, state management).
- Resources: freeCodeCamp React, [Roadmap.sh](https://roadmap.sh).
- Assignments: Product catalog UI; search/filter components for wearables.
- Milestone: React storefront with mock products.

### **• Weeks 7-8: Backend & Databases**

- Topics: Flask/Django setup, REST APIs, authentication (JWT), databases (PostgreSQL), CRUD for products/users.
- Resources: Django/Flask tutorials, Miguel Grinberg Flask series.
- Assignments: User registration/login; product API endpoints.
- Milestone: Backend for user accounts and catalog management.

### **• Weeks 9-12: E-Commerce Features & Integrations**

- Topics: Payment gateways (Stripe/Razorpay for India), logistics APIs (Shippo for shipping/booklets), brand integrations (mock APIs for multi-brands), cart/checkout logic.
- Resources: Stripe docs, Shippo API guides.
- Assignments: Implement cart with multi-item buy; fabric booklet request flow (address collection, mock fulfillment); brand product syncing.
- Milestone: Full e-commerce flow with integrations (e.g., add to cart → checkout → booklet request).

### **• Weeks 13-16: Mini Project**

- Build a prototype: Marketplace with 20+ wearables from 3-5 mock brands → search/browse → cart → checkout → booklet shipping integration.
- Assignments: Weekly sprints (e.g., Week 13: Catalog integration; Week 14: Cart/payment; Week 15: Booklet feature; Week 16: Testing/security).
- Milestone: Demo site with end-to-end purchase flow; prepare APIs for Team 1's try-on integration.



## Post-Training Development Phase (Months 5-16: Full Product Build)

- **Months 5-8:** Teams work independently (Team 1: Refine try-on accuracy/Gen AI; Team 2: Scale e-commerce with real brand APIs).
- **Months 9-12:** Integration sprints (e.g., embed try-on in product pages; link booklet requests to orders).
- **Months 13-16:** Testing, optimization, launch prep (user beta, bug fixes, scalability).
- **Oversight:** Assign mentors per team; bi-weekly cross-team syncs for alignment (e.g., API contracts).

This path ensures freshers gain skills progressively while contributing to the product. Adjust based on progress—focus on hands-on coding to build confidence! If needed, I can provide sample code or resources.