

Customer Order Fulfillment Cell

Create a configurable robot program that builds three different customer pallet orders.

Customer AeroParts Distribution	Department Order Fulfillment	Difficulty Advanced	Time 2 class periods
---	--	-------------------------------	--------------------------------

Production Goal: Automate three customer-specific pallet configurations.

Background

AeroParts Distribution receives small orders with different quantities and arrangements. The current program must be rewritten for every order. Your team must create a flexible program that operators can configure using variables at the beginning of the file.

Learning Objectives

- Translate order requirements into pallet coordinates
- Use functions, loops, lists, or variables to create flexible code
- Verify several production configurations
- Document an operator-friendly workflow

Equipment

- VEX CTE Workcell
- 6-Axis Arm
- Cubes
- Three customer order cards
- Python project
- Engineering notebook

Student Instructions

- 1 Review the three customer order cards and draw each required pallet layout.
- 2 Identify which pickup and placement actions are common to all orders.
- 3 Create helper functions for pickup, safe travel, and placement.
- 4 Create operator variables that select order type or quantity.
- 5 Program Order A and verify every coordinate.
- 6 Run Orders B and C by changing only the approved configuration values.
- 7 Test each order twice and compare the physical pallet to the order card.
- 8 Add clear comments and an operator instruction block at the top of the program.

Engineering Requirements

- Correctly complete all three customer orders
- No code changes outside the configuration section between orders
- No manual product placement
- Begin and end at Home
- Include comments for operator inputs

Constraints & Safety

- Use only the provided parts and pallet area
- Maximum cycle time is five minutes per order
- The robot may carry one part at a time
- All motions must use safe clearance heights

Deliverables

- Three order-layout diagrams
- Configurable Python program
- Six-run order verification log
- Operator quick-start instructions
- Reflection

Reflection Questions

- 1 Which functions made the program easier to adapt?
- 2 What input error could an operator make?
- 3 How could input validation improve the system?
- 4 How would a barcode scanner change the workflow?

Engineering Support

Engineering Tip

Separate values that operators change from the motion logic that should remain protected.

Common Mistake

Hard-coding every order creates duplicate code and makes later maintenance difficult.

Stretch Goal

Use a list of coordinates so a new order can be added without changing the pickup/place functions.

Career Connection

Flexible manufacturing engineers design systems that switch products with minimal downtime or reprogramming.

Before You Submit

- Notebook evidence included
- Requirements checked
- Testing documented
- Reflection completed