

Kanam Academy Foundations

Standards Alignment Overview

Program Age Range: 12–15

Primary Language: Python

Standards Referenced: CSTA K-12 Computer Science Standards (Revised 2017)

Grade Band: 3A (Grades 6–8)

Document Version: 1.0

Publication Date: January 2026

1. Program Overview

Kanam Academy Foundations is a structured, project-based Python program designed for learners ages 12–15. The curriculum develops foundational computer science knowledge through direct instruction, guided coding practice, and cumulative project work.

Students progress from core programming constructs to structured automation concepts, culminating in a capstone project that requires explanation, refinement, and presentation of their work.

The program emphasizes:

- Computational thinking
 - Algorithm design
 - Structured program development
 - Data handling fundamentals
 - Systematic debugging and refinement
 - Clear communication of technical reasoning
-

2. Instructional Model

Kanam Academy Foundations follows a structured instructional framework:

- Direct, human-led instruction
- Guided coding practice
- Progressive weekly build outcomes
- Iterative testing and debugging
- Capstone presentation and explanation

Students are required to understand and explain their code rather than rely on automated generation. Projects are scaffolded to build independence while reinforcing core programming principles.

3. Program Learning Objectives

By completion of Kanam Academy Foundations, students will be able to:

- Design programs using variables, conditionals, and loops
 - Develop structured algorithms to solve defined problems
 - Store and process collections of data using lists
 - Decompose problems into logical steps
 - Systematically test and debug program behavior
 - Explain program logic, design decisions, and outcomes
-

4. CSTA Standards Alignment (Grade Band 3A)

This curriculum is designed in alignment with the Computer Science Teachers Association (CSTA) K-12 Computer Science Standards, Grade Band 3A (Grades 6–8).

The table below maps major curriculum components to relevant CSTA benchmarks.

Standards Mapping Table

Kanam Curriculum Component	Learning Focus	CSTA Code	CSTA Standard Description (Abbreviated)
Variables & Data Types	Represent and manipulate program data	3A-AP-14	Use data abstraction to manage complexity in programs
Conditionals	Implement branching logic	3A-AP-08	Model and implement algorithms using sequences, conditionals, and loops
Loops	Apply repetition to automate processes	3A-AP-08	Model and implement algorithms using sequences, conditionals, and loops
Program Design (Mini Projects)	Structured problem-solving and flow design	3A-AP-10	Use design processes to address computational problems
Lists & Data Tracker	Store and process structured data	3A-DA-11	Create programs that use data structures to store and process data
Debugging & Testing	Refine and validate program correctness	3A-AP-17	Systematically test and refine programs using a range of test cases
Rules-Based Automation Logic	Decision-tree logic and structured automation	3A-AP-08	Implement structured algorithms to automate decision processes
Capstone Project	Design, build, test, and present computational artifact	3A-IC-25	Test and refine computational artifacts to improve usability and reduce unintended bias

5. Strand Coverage Summary

Kanam Academy Foundations primarily addresses the following CSTA strands:

- Algorithms & Programming (Primary emphasis)
 - Data & Analysis (Supporting emphasis)
 - Impacts of Computing (Introductory exposure through project refinement and discussion)
-

6. Alignment Statement

This document provides a curriculum alignment overview referencing the CSTA K-12 Computer Science Standards (2017 Revision). This alignment has been prepared by Kanam Academy and has not undergone formal CSTA review or endorsement.

7. Citation

Computer Science Teachers Association (2017). *CSTA K-12 Computer Science Standards, Revised 2017*. Retrieved from <https://csteachers.org>

Kanam Academy

Standards Alignment Overview

Version 1.0 | January 2026