Unit 1: Primitive Types (5 weeks)

This unit introduces students to the Java programming language and the use of classes, providing students with a firm foundation of concepts that will be leveraged and built upon in all future units. Students will focus on writing the main method and will start to call preexisting methods to produce output. Students will start to learn about three built-in data types and learn how to create variables, store values, and interact with those variables using basic operations. The ability to write expressions is essential to representing the variability of the real world in a program and will be used in all future units. Primitive data is one of two categories of variables covered in this course. The other category, reference data, will be covered in Unit 2.

Topics Covered	- Welcome to CSA - Why programming? Why Java? - Variables and Data Types - Expressions and Assignments Statements - Compound Assignment Operators - User Input - Casting and Ranges of Variables
08/12/24 – 09/13/24	

Unit 2: Using Objects (5 weeks)

In the first unit, students used primitive types to represent real-world data and determined how to use them in arithmetic expressions to solve problems. This unit introduces a new type of data: reference data. Reference data allows real-world objects to be represented in varying degrees specific to a programmer's purpose. This unit builds on students' ability to write expressions by introducing them to `Math` class methods to write expressions for generating random numbers and other more complex operations. In addition, strings and the existing methods within the `String` class are an important topic within this unit. Knowing how to declare variables or call methods on objects is necessary throughout the course but will be very important in Units 5 and 9 when teaching students how to write their own classes and about inheritance relationships.

Topics Covered	- Objects: Instances of Classes - Creating and Storing Objects (Instantiation) - Calling a Void Method - Calling a Void Method with Parameters - Calling a Non-void Method
09/13/24 – 10/18/24	

- String Objects: Concatenation, Literals, and More
- String Methods
- Wrapper Classes: Integer and Double

Using the Math class

Unit 3: Boolean Expressions and `if` Statements (4 weeks)

Algorithms are composed of three building blocks: sequencing, selection, and iteration. This unit focuses on selection, which is represented in a program by using conditional statements. Conditional statements give the program the ability to decide and respond appropriately and are a critical aspect of any nontrivial computer program. In addition to learning the syntax and proper use of conditional statements, students will build on the introduction of Boolean variables by writing Boolean expressions with relational and logical operators. The third building block of all algorithms is iteration, which you will cover in Unit 4. Selection and iteration work together to solve problems.

Topics Covered	• Boolean Expressions
	• `if` Statements and Control Flow
10/21/24 – 11/08/24	• `if-else` Statements `else`
	• `if` Statements
	Compound Boolean Expressions
	• Equivalent Boolean Expressions
	• Comparing Objects

Unit 4: Iteration (4 weeks)

This unit focuses on iteration using while and for loops. As you saw in Unit 3, Boolean expressions are useful when a program needs to perform different operations under different conditions. Boolean expressions are also one of the main components in iteration. This unit introduces several standard algorithms that use iteration. Knowledge of standard algorithms makes solving similar problems easier, as algorithms can be modified or combined to suit new situations. Iteration is used when traversing data structures such as arrays, ArrayLists, and 2D arrays. In addition, it is a necessary component of several standard algorithms, including searching and sorting, which will be covered in later units.

Topics Covered	• `while` loops `for`
	• loops
11/11/24 – 12/13/24	Developing Algorithms Using Strings
	• Nested Iteration
	• Informal Code Analysis
Associated Lab: Consumer Review Lab	

Unit 5: Writing Classes (6 weeks)

This unit will pull together information from all previous units to create new, user-defined reference data types in the form of classes. The ability to accurately model real-world entities in a computer program is a large part of what makes computer science so powerful. This unit focuses on identifying appropriate behaviors and attributes of real-world entities and organizing these into classes. Students will build on what they learn in this unit to represent relationships between classes through hierarchies, which appear in Unit 9. The creation of computer programs can have extensive impacts on societies, economies, and cultures. The legal and ethical concerns that come with programs and the responsibilities of programmers are also addressed in this unit.

Topics Covered 12/06/24 – 02/21/25	- Anatomy of a Class - Constructors - Documentation with Comments - Accessor Methods - Mutator Methods - Writing Methods - Static Variables and Methods - Scope and Access `this` Keyword - Ethical and Social Implications of Computing Systems

Unit 6: Array (5 weeks)

This unit focuses on data structures, which are used to represent collections of related data using a single variable rather than multiple variables. Using a data structure along with iterative statements with appropriate bounds will allow for similar treatment to be applied more easily to all values in the collection. Just as there are useful standard algorithms when dealing with primitive data, there are standard algorithms to use with data structures. In this unit, we apply standard algorithms to arrays; however, these same algorithms are used with ArrayLists and 2D arrays as well. Additional standard algorithms, such as standard searching and sorting algorithms, will be covered in the next unit.

Topics Covered 02/24/25 – 04/04/25	- Array Creation and Access - Traversing Arrays - Enhanced for Loop for Arrays - Developing Algorithms Using Arrays
Associated Lab: Magpie Lab	

Unit 7: `ArrayList` (4 weeks)

As students learned in Unit 6, data structures are helpful when storing multiple related data values. Arrays have a static size, which causes limitations related to the number of elements stored, and it can be challenging to reorder elements stored in arrays. The ArrayList object has a dynamic size, and the class contains methods for insertion and deletion of elements, making reordering and shifting items easier. Deciding which data structure to select becomes increasingly important as the size of the data set grows, such as when using a large real-world data set. In this unit, students will also learn about privacy concerns related to storing large amounts of personal data and about what can happen if such information is compromised.

Topics Covered	● Introduction to `ArrayList`
04/07/25 – 05/09/25	● `ArrayList` Methods
	● Traversing ArrayLists
	● Developing Algorithms Using ArrayLists
	● Searching
	● Sorting
	● Ethical Issues Around Data Collection
Associated Lab: Pokemon Simulation, Blackjack, Mad Libs, Data Lab ** This lab is not currently available on CodeHS. Visit the AP Classroom for options to complete this lab.	

Unit 8: Final Project (2-4 weeks)

Students apply what they learned towards a final project on a topic of their choice.

Topics Covered	● Allow students to think creatively about the applications of the concepts covered in the course
	● Scoping a project
	● Designing an application from scratch
	● Incremental development
	● Testing
	● Debugging