

Chino Valley Unified School District

High School Course Description

A. CONTACTS	
1. School/District Information:	School/District: Chino Valley Unified School District Street Address: 5130 Riverside Dr., Chino, CA 91710 Phone: (909) 628 - 1201 Website: www.chino.k12.ca.us
2. Course Contact:	District Contact: Office of Secondary Curriculum and Instruction Position/Title: Director of Secondary Curriculum and Instruction Site: District Office Phone: (909)628-1201 X1630
B. COVER PAGE - COURSE ID	
1. Course Title:	Introduction to Python Coding
2. Transcript Title/Abbreviation:	Intro to Python
3. Transcript Course Code/Number:	5E81
4. Seeking Honors Distinction:	No
5. Subject Area/Category:	Meets UC/CSU "g" Elective: Mathematics - Computer Science
6. Grade Level(s):	9-12
7. Unit Value:	5 units Semester Credit
8. Course Previously Approved by UC:	Yes
9. Classified as a Career Technical Education Course:	No
10. Modeled after an UC-approved course:	Yes
11. Repeatable for Credit:	No
12. Date of Board Approval:	April 21, 2022
13. Brief Course Description:	The goal of this course is to enable students to write in the Python programming language as they complete several programming projects. These projects increase in difficulty and length as the course progresses, including writing game programs and Object-Oriented programs. Students study how computers deal with data and instructions, the internal components of a computer, and computer logic. This course is designed for students who have a desire for a career in computer programming or engineering.
14. Prerequisites:	None
15. Context for Course:	As the world becomes more technologically advanced, current trends indicate an increased need for individuals who can program and develop computer applications. Introduction to Python Honors enables students to write in the Python programming language, giving them the skills needed to be competitive leaders both in school and later in the work force. Students study how computers deal with data and instructions, the internal components of a computer, and computer logic. This course is designed for students who have a desire for a career in computer programming or engineering. This course serves as a foundations course in the Biomedical Science and Technology Academy at Chino High School.
16. History of Course Development:	This course is designed for students who have a desire to learn more about a career in computer programming or engineering. This course aligns with the Computer Science Teachers Association (CSTA) standards. The course has been developed to serve as a foundation for Biomedical Science and Technology (BST) course of study.
17. Textbooks:	Not applicable
18. Supplemental Instructional Materials:	Edhesive: Intro to Computer Science Author: Edhesive's Curriculum Team Publisher: Edhesive Site: https://edhesive.com/courses/apcs_introduction

Chino Valley Unified School District

High School Course Description

	CodeHS Python Course Author: CodeHS Curriculum Team Publisher: CodeHS Site: https://docs.google.com/document/d/16J34mC3bv7KgMurRQWcvUJiiGrLeT6MzroduxN_mq9w/edit Learn Python Author: Data Camp Design Team Publisher: Data Camp Site: https://www.learnpython.org
--	---

C. COURSE CONTENT

1. Course Purpose:

This course enables students to write in the Python programming language as they complete several programming projects. These projects increase in rigor and length as the course progresses, building essential skills in critical thinking, problem solving, and perseverance. Students study how computers deal with data and instructions, the internal components of a computer, and computer logic. This course is designed for students who have a desire to learn more about a career in computer programming or engineering. This course is aligned with the Computer Science Teachers Association (CSTA) standards.

2. Course Outline:

Unit 1: Beginning in Computer Science

In this unit, students are oriented to the basics of computer hardware and software. Students explore basic programs which give them a foundation of variables, how the user inputs information, and how the programmer outputs information. These basic programs demonstrate critical thinking, connecting, and applying concepts. Students are also oriented to binary and how information is sent around the world. Students are also exposed to potential careers in Computer Science.

- Lesson 1: Welcome
- Lesson 2: What is Computer Science?
- Lesson 3: Using Python – Installing and online IDE
- Lesson 4: First Program
- Lesson 5: Hardware Basics
- Lesson 6: Output
- Lesson 7: Input
- Lesson 8: Data Types and Variables
- Lesson 9: Analog vs. Digital
- Lesson 10: Bits and Pieces – Understanding Binary
- Lesson 11: Career Connection – Who Uses Computer Science?

Unit 2: Number Calculations and Data

Students learn and develop algorithms to transform variables using some basic arithmetic, algebra, and modulus. Students learn about abstraction by creating functions to carry out tasks. These are problem solving basics which are essential to Computer Science. Students learn about built in randomization functions. Students are exposed to a branch of Computer Science called Big Data. Reading and transforming a data set to produce new valuable information.

- Lesson 1: Computer History
- Lesson 2: Basic Calculations
- Lesson 3: Modular Division
- Lesson 4: Built-in Functions
- Lesson 5: Random Numbers
- Lesson 6: Color Code
- Lesson 7: Big Data
- Lesson 8: Working with a Real Data Set
- Lesson 9: Career Connection – Data Scientists

Chino Valley Unified School District

High School Course Description

Unit 3: Making Decisions

Students learn, solve, compute Boolean variables and their importance. Conditional statements such as if, if-else, if-else-if statements are taught as well as nested conditional statements. Students empathize to problem solve these values. In addition, students solve problems related to order, and determine which questions to ask with design of more complex algorithms. Students create and apply algorithms such as max min to solve problems. Finally, students observe career applications in practice.

- Lesson 1: Max and Min
- Lesson 2: Simple If's
- Lesson 3: Booleans
- Lesson 4: If – Else
- Lesson 5: Else – If
- Lesson 6: Defining Algorithms
- Lesson 7: Algorithm Challenge
- Lesson 8: History – Turing Machines
- Lesson 9: Career Connection – Computer Science in Medicine

Unit 4: Repetition and Loops

Students learn the principle of iteration which is one of the cornerstones of why computers help us in everyday use. Computers can repeat and iterate functions repeatedly very fast to simulate and solve problems. Students learn about loops and build programs that iterate many times based upon some Boolean condition. Students connect previous knowledge and new concepts to create, predict and fix more complex algorithms. This previous knowledge will transition into critically thinking about and designing a playable game. Students learn about another branch of computer science called Cybersecurity and more potential careers.

- Lesson 1: Loops
- Lesson 2: Count Variables
- Lesson 3: Two Ways to End a Loop
- Lesson 4: Data Revisited
- Lesson 5: Connection – Games with a Purpose (GWAP)
- Lesson 6: Making a Game
- Lesson 7: Internet - Cybersecurity
- Lesson 8: Career Connection - Game Developers

Unit 5: Graphics

Students learn and apply elementary graphics basics such as lines, coordinates, shapes, and color. Students connect to create new shapes, using the tools and functions within the language to build functions and see patterns within drawings. Students transition to creating basic animations from these drawings. This new knowledge bridges the gap between how animated movies are produced and careers in computer science in the entertainment industry.

- Lesson 1: Color Revisited
- Lesson 2: X & Y coordinates
- Lesson 3: Lines
- Lesson 4: Draw a House
- Lesson 5: Circles
- Lesson 6: Emoticons
- Lesson 7: Kaleidoscope
- Lesson 8: Animation
- Lesson 9: History of Movies
- Lesson 10: Career Connection – Computer Science in Entertainment

Chino Valley Unified School District

High School Course Description

Unit 6: For Loops

Students learn additional iteration techniques, such as FOR loops. Students learn and think about how to apply different tools in different situations and why. Students learn about counters and variables which change with iterations through loops. Students build new algorithms to solve simulations modeling behavior. Simulations are fundamental in Computer Science to test and learn the fundamentals of artificial intelligence.

- Lesson 1: Review – Looping
- Lesson 2: Range Function
- Lesson 3: For Loops
- Lesson 4: Counting by Other Than 1
- Lesson 5: Summing
- Lesson 6: Review Algorithms and Tracing
- Lesson 7: Modeling and Simulation
- Lesson 8: Introduction to EarSketch
- Lesson 9: Career Connection – Dance and Music

Unit 7: Text and String Processing

Students learn about more built-in and useful functions dealing with Strings, and characters. Students learn how to input and output text files which will make use of vast amounts of data to be processed within a program. Students learn to problem solve processing Strings to make programs run without error and easier to use. Students will investigate more applications of computer science in the fields of Compression, Cryptography, Cybersecurity, and how as well as how computer science is related to the Design of the Internet.

- Lesson 1: ASCII and Character Functions
- Lesson 2: Processing Strings
- Lesson 3: Text Files – Input
- Lesson 4: Text Files – Output
- Lesson 5: Processing Strings in Files
- Lesson 6: External Resource: Text Compression
- Lesson 7: Journey to Cryptography
- Lesson 8: Career Connection – Cybersecurity

Unit 8: Functions

Students dive deep into how to create powerful functions and how to use Abstraction. We will also learn how to make functions more abstract and powerful by using parameters and return values. Creating power functions is fundamental in Computer Science to break programs down into smaller parts and then use these smaller working parts to build powerful larger programs. We'll also learn troubleshooting strategies to fix our code as it gets more complicated. Analyzing our own code can be challenging without the proper tools. Knowing which tools to use and why is higher-level thinking.

- Lesson 1: What are Functions?
- Lesson 2: Creating Functions
- Lesson 3: Parameters
- Lesson 4: Chris Bosh on Functions
- Lesson 5: Functions – Returning values
- Lesson 6: Example – Using Several Functions
- Lesson 7: Tracing Code
- Lesson 8: Functions in EarSketch
- Lesson 9: Career Connection – Social Justice

Chino Valley Unified School District

High School Course Description

3. Key Assignments:

Unit 1: Beginning in Computer Science

Assignments

Students complete practice programs demonstrating understanding of how to manipulate variables within input and output of information. Students take short quizzes focusing on vocabulary and analyzing code. Students bring their skills together for a unit end project called Silly Sentences asking the user for information and then outputting a transformation of the information via variables. Finally, they will take a test on the material.

Assignment: Silly Sentences

- More ASCII Art
- More Input Practice Problems
- What Is a Computer – Project

Unit 2: Number Calculations and Data

Assignments

Students will complete practice programs demonstrating understanding of how to manipulate variables using the +, -, *, /, division modulus, factorial and PEMDAS. Students will take short quizzes focusing on vocabulary, analyzing code, and predicting output. Students will bring their skills together for a unit end project called Room Area asking the user for information, problem solving to calculate and output area. Finally, they will take a test for this unit.

Assignment: Room Area

- Modular Division Practice Problems
- Practice Calculations

Unit 3: Making Decisions

Assignments

Students complete practice programs demonstrating understanding of Boolean variables and conditional statements. Students will take short quizzes focusing on vocabulary, analyzing code, and predicting output. Students bring their skills together for a unit end project called Chatbox where through a series of complex conditional statements, students will try to have a basic AI conversation with the user based upon what the user inputs. Empathy is key in understanding how code must be written to account for all types of user input to keep the program running.

Assignment: Chatbot

- Basic If-Then Statement Practice Exercises

Unit 4: Repetition and Loops

Assignments

Students complete practice programs demonstrating an understanding of how to use and manipulate loops. Students take short quizzes focusing on vocabulary, analyzing code, and predicting the output. Students bring their skills together for a unit end project called Student Schedule where students draw a student schedule by using a while loop. Students ask the user for their first and last names, and then a list of their classes and room numbers. Students will take summative test for this unit.

Assignment: Student Schedule

- Practice Loops Worksheet

Unit 5: Graphics

Assignments

Students complete practice programs demonstrating an understanding of all sorts of drawings of shapes. Students take short quizzes focusing on vocabulary, analyzing code, and predicting the output. Students bring their skills together for a unit end project called Animation where students create a drawing and then animate it. There is a summative test for this unit.

Assignment: Animation

- Practice Line Drawings

Chino Valley Unified School District

High School Course Description

Unit 6: For Loops

Assignments

Students complete practice programs demonstrating an understanding of when and how to use FOR loops. Students take short quizzes focusing on vocabulary, analyzing code, and predicting output dealing with FOR loops. Students bring their skills together for a unit end project called Shapes where students will create shapes using loops which have repetitive patterns seeing the power of how to use parameters. Students take a summative test for this unit.

Assignment: Shapes

- Practice Loops Worksheet

Unit 7: Text and String Processing

Assignments

Students complete practice programs demonstrating an understanding of reading and writing text files. In addition, students process code related to Strings and Characters. Students take short quizzes focusing on vocabulary, analyzing code, and predicting output dealing with basic data files. Students bring their skills together for a unit end project called Random Joke Generator, where students read in a database of jokes and then write a program that gives the user jokes based upon their desires. Finally, students take a test for this unit.

Assignment: Random Joke Generator

Unit 8: Functions

Assignments

Students complete practice programs writing small and specific smaller functions with and without parameters. In addition, students will write functions which will return numerical, String, and Boolean values. Students take short quizzes focusing on vocabulary, analyzing code, and predicting output dealing with how functions behave regarding global and local variables. Students will bring their skills together for a unit end project called Calendar where students will make a basic calendar. Finally, students take a test for this unit.

Assignment: Calendar

- Subprograms Worksheet

4. Instructional Methods and/or Strategies:

APB (Activity, Project, and Problem-based) Instructional Design providing students with unique opportunities to work collaboratively, identify problems, apply what they know, persevere through challenges, find unique solutions, and lead their own learning.

5. Assessment Including Methods and/or Tools:

The evaluation of student progress and evaluation will be based on the following criteria outlined in Board Policy:

- Assessments: 60-75% of the final grade
- Assignments and class discussions: 25-40% of the final grade