TOWNSHIP OF UNION PUBLIC SCHOOLS

Grade 8
Computer Literacy

Adopted February 15, 2022

Mission Statement

The mission of the Township of Union Public Schools is to build on the foundations of honesty, excellence, integrity, strong family, and community partnerships. We promote a supportive learning environment where every student is challenged, inspired, empowered, and respected as diverse learners. Through cultivation of students' intellectual curiosity, skills and knowledge, our students can achieve academically and socially, and contribute as responsible and productive citizens of our global community.

Philosophy Statement

The Township of Union Public School District, as a societal agency, reflects democratic ideals and concepts through its educational practices. It is the belief of the Board of Education that a primary function of the Township of Union Public School System is to formulate a learning climate conducive to the needs of all students in general, providing therein for individual differences. The school operates as a partner with the home and community.

Course Description

All students receive computer science and design thinking instruction from Kindergarten through grade 12. The study of these disciplines focuses on a deep understanding of concepts that enable students to think critically and systematically about leveraging technology to solve local and global issues. Authentic learning experiences that enable students to apply content knowledge, integrate concepts across disciplines, develop computational thinking skills, acquire and incorporate varied perspectives, and communicate with diverse audiences about the use and effects of computing prepares New Jersey students for college and careers.

The Township of Union Public Schools Grade 8 Computer Science Curriculum transitions students from thinking about computer science as a tool to solve their own problems towards considering the broader social impacts of computing. The following units explore the importance of using data to solve real world problems while examining the challenges and tradeoffs which accompany this process. These units explore some of the modern problems with machine learning, especially around bias and impact.

Course Description

All students receive computer science and design thinking instruction from Kindergarten through grade 12. The study of these disciplines focuses on a deep understanding of concepts that enable students to think critically and systematically about leveraging technology to solve local and global issues. Authentic learning experiences that enable students to apply content knowledge, integrate concepts across disciplines, develop computational thinking skills, acquire and incorporate varied perspectives, and communicate with diverse audiences about the use and effects of computing prepares New Jersey students for college and careers.

The Township of Union Public Schools Grade 8 Computer Science Curriculum transitions students from thinking about computer science as a tool to solve their own problems towards considering the broader social impacts of computing. The following units explore the importance of using data to solve real world problems while examining the challenges and tradeoffs which accompany this process. These units explore some of the modern problems with machine learning, especially around bias and impact.

Curriculum Units/Pacing Guide

Unit # / Title	Number of Days
Unit 1: The Design Process	~5 weeks
Unit 2: Data and Society	~7 weeks
Unit 3: Physical Computing	~3 weeks
Unit 4: AI and Machine Learning	~3 weeks

Unit Standards Overview

Overview	Standards	Unit Skills Focus	Content-Specific Practices (when applicable)
Unit 1: The Design Process The Design Process unit transitions students from thinking about computer science as a tool to solve their own problems towards considering the broader social impacts of computing. Through a series of design challenges, students are asked to consider and understand the needs of others while developing a solution to a problem. The second half of the unit consists of an iterative team project, during which students have the opportunity to identify a need that they care about, prototype solutions both on paper and in App Lab, and test their solutions with real users to get feedback and drive further iteration.	NJSLS – Computer Science and Design Thinking (See below)	By the end of the unit, students should see the design process as a form of problem solving that prioritizes the needs of a user. They should be able to identify user needs and assess how well different designs address them. In particular, they know how to develop paper and digital prototypes, gather and respond to feedback about a prototype, and consider ways different user interfaces do or do not affect the usability of their apps. Students should leave the unit with a basic understanding of other roles in software development, such as product management, marketing, design, and testing, and how to use what they have learned about computer science as a tool for social impact.	• Empathy • User • Usability • Prototype • User Interface • Bug • Feature • Decode • Encode • Binary System • ASCII • Bit • Pixel • Decrypt • Encrypt • Innovation • Analog • Digital • Array • For Loop • Parameter • Prototype • Circuit • Machine Learning • Features • Label • Model • Supervised Learning • Training • Unsupervised Learning • Categorical Data • Classification • Bias • Numerical Data • Ethics
Suggested Resources *Provide links to specific resources/activities*	Code.org Unit 4 Lesson Guide https://studio.code.org/s/csd4-2021?section_id=3522356 Code.org Unit 4 Resources https://studio.code.org/s/csd4-2021/resources Code.org Unit 4 Teacher Forum https://forum.code.org/tags/c/csd/236/csd-unit-4		
Unit 2: Data and Society The Data and Society unit is about the importance of using data to solve problems and it highlights how computers can help in this process. The first chapter explores different systems used to represent information in a computer and the challenges and tradeoffs posed by using them. In the second chapter, students learn how collections of	NJSLS – Computer Science and Design Thinking (See below)	By the end of the unit, students should have a broad understanding of the role of data and data representation in solving information problems. They should be able to explain the necessary components of any data representation scheme, as well as the particulars of binary and the common ways that various types of simple and complex data are represented in binary code.	

data are used to solve problems, and how computers help to automate the steps of this process. In the final project, students gather their own data and use it to develop an automated solution to a problem.		Students should also be able to design and implement a data-based solution to a given problem and determine how the different aspects of this problem solving process could be automated.
Suggested Resources *Provide links to specific resources/activities*	Code.org Unit 5 Lesson Guide https://studio.code.org/s/csd5-2021?section_id=3522356 Code.org Unit 5 Resources https://studio.code.org/s/csd5-2021/resources Code.org Teacher Forum https://forum.code.org/c/csd/236	
Unit 3: Physical Computing In the Physical Computing unit, students further develop their programming skills, while exploring more deeply the role of hardware platforms in computing. Harkening back to the Input/Storage/Processing/Output model for a computer, students look towards modern "smart" devices to understand the ways in which non-traditional computing platforms take input and provide output in ways that couldn't be done with the traditional keyboard, mouse, and monitor. Using App Lab and Adafruit's Circuit Playground, students develop programs that utilize the same hardware inputs and outputs that we see in many modern smart devices, and they get to see how a simple rough prototype can lead to a finished product. The unit concludes with a design challenge that asks students to use the Circuit Playground as the basis for an innovation of their own design.	NJSLS – Computer Science and Design Thinking (See below)	By the end of the unit, students should be able to design and build a physical computing device that integrates hardware inputs and outputs with software. This unit builds on the skills and understandings from the Interactive Animations and Games unit with more sophisticated programming constructs, such as arrays, for-loops, and parameters, as well as deepens students' understanding of the types of input and output that can be used in computing. Students should leave the unit feeling equipped to use physical computing to solve problems in fun and innovative ways.
Suggested Resources *Provide links to specific resources/activities*	Code.org Unit 6 Lesson Guide https://studio.code.org/s/csd6-2021?section_id=3522356 Code.org Unit 6 Resources	

	https://studio.code.org/s/csd6-2021/resources Code.org Unit 6 Teacher Forum https://forum.code.org/tags/c/csd/236/csd-unit-6 Code.org Circuit Playground https://code.org/circuitplayground		
Unit 4: AI and Machine Learning In this optional AI and Machine Learning unit, students learn how computers can find patterns in data to make decisions. Students use the Problem Solving Process for machine learning to define a problem, prepare their data, train a model, then test and evaluate their model for accuracy and potential bias. Students explore a variety of scenarios and datasets that lend themselves to machine learning. They also explore some of the modern problems with machine learning, especially around bias and impact.	NJSLS – Computer Science and Design Thinking (See below) AI4K12 National Guidelines 2021 (See below)	Students will use AI Lab to create a machine learning model to solve a problem, and use App Lab to create an app that uses their model. Students should leave the unit with an understanding of how machine learning models make decisions from data, and with the ability to create machine learning models from their own data to solve problems in their community. Students use AI Lab to train their machine learning models, then import their models into App Lab to further customize their apps. This unit assumes students have prior App Lab experience, so we recommend completing this unit after Unit 4, which focuses on designing for users and introduces App Lab.	
Suggested Resources *Provide links to specific resources/activities*	Code.org Unit 7 Lesson Guide https://studio.code.org/s/csd7-2021?section_id=3522356 Code.org Unit 7 Resources https://studio.code.org/s/csd7-2021/resources Code.org Teacher Forum https://forum.code.org/c/csd/236 App Lab Required Experiences https://docs.google.com/document/d/1SkMCN4Eid3-mGoRUQkurQGJqoOJ9PydfB7uGE0xZyFA/preview		

Curricular Units

Unit 1: The Design Process			
Content Standards	**Critical Knowledge & Skills** ("Unpacked" Standards)	**Content-Specific Practices** (when applicable)	**Standard Mastery Examples** *When possible, provide links to specific samples/ documents/ assignments/etc.*
8.1.8.CS.1	The study of human–computer interaction can improve the design of devices and extend the abilities of humans.	• Recommend improvements to computing devices in order to improve the ways users interact with the devices.	• Lesson 1: Designing With Empathy • Lesson 2: Understanding Your User • Lesson 5: User Interfaces • Lesson 13: Prototype Testing • Lesson 14: Design Mode in App Lab • Lesson 15: Build a Digital Prototype • Lesson 18: Testing the App • Lesson 19: Bugs and Features • Lesson 20: Updating Your Prototype • Lesson 21: Project - App Presentation
8.1.8.CS.2	Software and hardware determine a computing system's capability to store and process information. The design or selection of a computing system involves multiple considerations and potential trade-offs.	• Design a system that combines hardware and software components to process data.	• Lesson 3: User Centered Design Define and Prepare • Lesson 4: User Centered Design Try and Reflect
8.2.5.NT.2	Technology innovation and improvement may be influenced by a variety of factors. Engineers create and modify technologies to meet people's needs and wants; scientists ask questions about the natural world.	• Identify new technologies resulting from the demands, values, and interests of individuals, businesses, industries, and societies.	• Lesson 3: User Centered Design Define and Prepare • Lesson 4: User Centered Design Try and Reflect • Lesson 9: Designing Apps for Good • Lesson 10: Market Research • Lesson 21: Project - App Presentation

8.1.8.IC.2	Advancements in computing technology can change individuals' behaviors. Society is faced with trade-offs due to the increasing globalization and automation that computing brings.	• Describe issues of bias and accessibility in the design of existing technologies.	• Lesson 3: User Centered Design Define and Prepare • Lesson 4: User Centered Design Try and Reflect • Lesson 6: Feedback and Testing • Lesson 7: Identifying User Needs • Lesson 9: Designing Apps for Good • Lesson 10: Market Research • Lesson 21: Project - App Presentation
8.1.2.AP.1	Individuals develop and follow directions as part of daily life. A sequence of steps can be expressed as an algorithm that a computer can process.	• Model daily processes by creating and following algorithms to complete tasks.	• Lesson 5: User Interfaces • Lesson 6: Feedback and Testing • Lesson 7: Identifying User Needs • Lesson 8: Project - Paper Prototype • Lesson 11: Exploring UI Elements • Lesson 12: Build a Paper Prototype • Lesson 13: Prototype Testing
8.1.5.AP.6	Individuals develop programs using an iterative process involving design, implementation, testing, and review.	• Develop programs using an iterative process, implement the program design, and test the program to ensure it works as intended.	• Lesson 5: User Interfaces • Lesson 6: Feedback and Testing • Lesson 7: Identifying User Needs • Lesson 8: Project - Paper Prototype • Lesson 13: Prototype Testing • Lesson 18: Testing the App • Lesson 19: Bugs and Features • Lesson 20: Updating Your Prototype
8.2.5.ED.2	Engineering design is a systematic and creative process of communicating and collaborating to meet a design challenge. Often, several design solutions exist, each letter in some way than the others.	• Collaborate with peers to collect information, brainstorm to solve a problem, and evaluate all possible solutions to provide the best results with supporting sketches or models.	• Lesson 6: Feedback and Testing • Lesson 7: Identifying User Needs • Lesson 8: Project - Paper Prototype • Lesson 13: Prototype Testing • Lesson 14: Design Mode in App Lab • Lesson 15: Build a Digital Prototype • Lesson 16: Events in AppLab • Lesson 17: Linking Prototype Screens

			• Lesson 18: Testing the App • Lesson 19: Bugs and Features • Lesson 20: Updating Your Prototype
8.1.8.AP.4	Programs use procedures to organize code and hide implementation details. Procedures can be repurposed in new programs. Defining parameters for procedures can generalize behavior and increase reusability.	• Decompose problems and sub-problems into parts to facilitate the design, implementation, and review of programs.	• Lesson 11: Exploring UI Elements • Lesson 12: Build a Paper Prototype • Lesson 14: Design Mode in App Lab • Lesson 15: Build a Digital Prototype • Lesson 16: Events in AppLab • Lesson 17: Linking Prototype Screens • Lesson 19: Bugs and Features • Lesson 20: Updating Your Prototype
8.1.8.AP.9	Individuals design and test solutions to identify problems taking into consideration the diverse needs of the users and the community.	• Document programs in order to make them easier to follow, test, and debug.	• Lesson 11: Exploring UI Elements • Lesson 12: Build a Paper Prototype • Lesson 14: Design Mode in App Lab • Lesson 15: Build a Digital Prototype • Lesson 16: Events in AppLab • Lesson 17: Linking Prototype Screens • Lesson 19: Bugs and Features • Lesson 20: Updating Your Prototype • Lesson 21: Project - App Presentation
8.1.12.AP.7	Complex programs are developed, tested, and analyzed by teams drawing on the members' diverse strengths using a variety of resources, libraries, and tools.	• Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.	• Lesson 6: Feedback and Testing • Lesson 7: Identifying User Needs • Lesson 13: Prototype Testing • Lesson 14: Design Mode in App Lab • Lesson 15: Build a Digital Prototype • Lesson 16: Events in AppLab

			• Lesson 17: Linking Prototype Screens • Lesson 18: Testing the App • Lesson 19: Bugs and Features • Lesson 20: Updating Your Prototype • Lesson 21: Project - App Presentation
8.1.8.DA.1	People use digital devices and tools to automate the collection, use, and transformation of data. The manner in which data is collected and transformed is influenced by the type of digital device(s) available and the intended use of the data.	• Organize and transform data collected using computational tools to make it usable for a specific purpose.	• Lesson 13: Prototype Testing • Lesson 18: Testing the App
8.1.8.DA.5	Computer models can be used to simulate events, examine theories and inferences, or make predictions.	• Test, analyze, and refine computational models.	• Lesson 13: Prototype Testing • Lesson 19: Bugs and Features • Lesson 20: Updating Your Prototype • Lesson 21: Project - App Presentation
8.1.8.AP.7	Individuals design and test solutions to identify problems taking into consideration the diverse needs of the users and the community.	• Design programs, incorporating existing code, media, and libraries, and give attribution.	• Lesson 14: Design Mode in App Lab • Lesson 15: Build a Digital Prototype • Lesson 16: Events in AppLab • Lesson 17: Linking Prototype Screens • Lesson 19: Bugs and Features • Lesson 20: Updating Your Prototype •
8.2.8.ED.7	Engineering design requirements and specifications involve making trade-offs between competing requirements and desired design features.	• Design a product to address a real-world problem and document the iterative design process, including decisions made as a result of specific constraints and trade-offs (e.g., annotated sketches).	• Lesson 14: Design Mode in App Lab • Lesson 15: Build a Digital Prototype • Lesson 16: Events in AppLab • Lesson 17: Linking Prototype Screens • Lesson 18: Testing the App • Lesson 19: Bugs and Features

			• Lesson 20: Updating Your Prototype

Unit 1 Assessment Plan	
Formative Assessment *When possible, provide links to specific samples/ documents/ assignments/etc.*	**Summative Assessment** *When possible, provide links to specific samples/ documents/ assignments/etc.*
• Observations • Completion of Activity Guides in CSD Unit 4 Resources	• Lesson 8: Project - Paper Prototype • Lesson 21: Project - App Presentation • Completion of Activity Guides in CSD Unit 4 Resources

Unit 1 Suggested Modifications/Accommodations/Extension Activities		
English Language Learners (ELL) *When possible, provide links to specific samples/ documents/ assignments/etc.*	**Special Education / 504** *When possible, provide links to specific samples/ documents/ assignments/etc.*	**Gifted and Talented** *When possible, provide links to specific samples/ documents/ assignments/etc.*
CS Discoveries Approach to Differentiation "Learning to use resources is a key goal of the course, and given resources provide an opportunity for students to self-differentiate in how they interact with key course content. This may include proactive differentiation, such as printing out resources ahead of time for students. It can also include just-in-time differentiation, such as monitoring students as they reach the end of a project and referring them to additional resource" **Examples of Strategies and Practices that Support English Language Learners:** • Pre-teaching of vocabulary and concepts • Visual learning, including graphic organizers • Text to Speech • Think-pair-share • Cooperative learning groups • Teacher modeling • Pairing students with beginning English language skills with students who have more advanced English language skills	CS Discoveries Approach to Differentiation "In order to meet the needs of a wide variety of learners, CS Discoveries is designed with flexibility that allows teachers to differentiate their instruction at the class and student level." *Refer to students' IEP for specific modifications and accommodations **Examples of Strategies and Practices that Support Students with Disabilities:** • Use of visual and multisensory formats • Use of assisted technology • Use of prompts • Modification of content and student products • Testing accommodations • Authentic assessments	CS Discoveries Approach to Differentiation "Challenge levels are found after the assessment levels in many of the programming lessons. These levels include new code and challenges that go beyond the learning objectives of the lesson. Most also include a "Free Play" option that allows students to use the new skills they have learned in whatever way they choose."

• Documentation Resource on code.org		

Unit 1 Connections	
NJSLS - Technology *When possible, provide links to specific samples/ documents/ assignments/etc.* Refer to the NJ Technology Standards	**Career Readiness Practices** *When possible, provide links to specific samples/ documents/ assignments/etc.* Refer to the NJ Career Readiness Practices
Technology Standards: Technology standards are embedded throughout all curricular units: Standard 8.1 Computer Science • Computer Science, previously a strand entitled 'Computational Thinking: Programming' in standard 8.2 of the 2014 NJSLS-Technology, outlines a comprehensive set of concepts and skills, such as data and analysis, algorithms and programming, and computing systems. Standard 8.2 Design Thinking • This standard, previously standard 8.2 Technology Education of the 2014 NJSLS – Technology, outlines the technological design concepts and skills essential for technological and engineering literacy. The new framework design, detailed previously, includes Engineering Design, Ethics and Culture, and the Effects of Technology on the Natural world among the disciplinary concepts.	**Career Ready Practices and Standard 9.1, 9.2, and 9.3 Career Ready Practices:** • CRP2. Apply appropriate academic and technical skills. • CRP4. Communicate clearly and effectively and with reason. • CRP5. Consider the environmental, social and economic impacts of decisions. • CRP6. Demonstrate creativity and innovation. • CRP7. Employ valid and reliable research strategies. • CRP8. Utilize critical thinking to make sense of problems and persevere in solving them. • CRP9. Model integrity, ethical leadership and effective management. • CRP10. Plan education and career paths aligned to personal goals. • CRP11. Use technology to enhance productivity. • CRP12. Work productively in teams while using cultural global competence.
21st Century Skills *When possible, provide links to specific samples/ documents/ assignments/etc.* Refer to the 21st Century Life and Skills	**Interdisciplinary Connections** *When possible, provide links to specific ELA/Math/Sci/SS standards as well as samples/ documents/ assignments/etc.* Refer to the NJ Student Learning Standards
21st Century Themes • Career Awareness • Career Exploration **21st Century Skills** • Creativity and Innovation (E) • Critical Thinking and Problem Solving (T) (A) • Communication (E) • Collaboration (E) (T)	**Interdisciplinary connections are made across grades and content areas to model the integration of knowledge and skills in the real world.**

Unit 2: Data and Society			
Content Standards	**Critical Knowledge & Skills** ("Unpacked" Standards)	**Content-Specific Practices** (when applicable)	**Standard Mastery Examples** *When possible, provide links to specific samples/ documents/ assignments/etc.*
8.1.5.DA.3	Individuals can select, organize, and transform data into different visual representations and communicate insights gained from the data.	• Organize and present collected data visually to communicate insights gained from different views of the data.	• Lesson 1: Representation Matters • Lesson 2: Patterns and Representation • Lesson 3: ASCII and Binary Representation • Lesson 4: Representing Images • Lesson 5: Representing Numbers • Lesson 6: Combining Representations • Lesson 8: Create a Representation
8.1.5.NI.2	Distinguishing between public and private information is important for safe and secure online interactions. Information can be protected using various security measures (i.e., physical and digital).	• Describe physical and digital security measures for protecting sensitive personal information.	• Lesson 7: Keeping Data Secret
8.1.8.NI.2	Protocols, packets, and addressing are the key components for reliable delivery of information across networks.	• Model the role of protocols in transmitting data across networks and the Internet and how they enable secure and errorless communication.	• Lesson 7: Keeping Data Secret
8.1.2.AP.1	Individuals develop and follow directions as part of daily life. A sequence of steps can be expressed as an algorithm that a computer can process.	• Model daily processes by creating and following algorithms to complete tasks.	• Lesson 8: Create a Representation • Lesson 13: Automating Data Decisions • Lesson 16: Project - Make a Recommendation
8.1.8.AP.4	Programs use procedures to organize code and hide implementation details. Procedures can be repurposed in new programs. Defining parameters	• Decompose problems and sub-problems into parts to facilitate the design,	• Lesson 8: Create a Representation

	for procedures can generalize behavior and increase reusability.	implementation, and review of programs.	
8.1.8.DA.1	People use digital devices and tools to automate the collection, use, and transformation of data. The manner in which data is collected and transformed is influenced by the type of digital device(s) available and the intended use of the data.	• Organize and transform data collected using computational tools to make it usable for a specific purpose.	• Lesson 9: Problem Solving and Data • Lesson 10: Structuring Data • Lesson 12: Making Decisions with Data • Lesson 13: Automating Data Decisions • Lesson 16: Project - Make a Recommendation
8.1.12.AP.7	Complex programs are developed, tested, and analyzed by teams drawing on the members' diverse strengths using a variety of resources, libraries, and tools.	• Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.	• Lesson 13: Automating Data Decisions
8.2.5.NT.2	Technology innovation and improvement may be influenced by a variety of factors. Engineers create and modify technologies to meet people's needs and wants; scientists ask questions about the natural world.	• Identify new technologies resulting from the demands, values, and interests of individuals, businesses, industries, and societies.	• Lesson 14: Problem Solving with Big Data
8.1.5.NI.2	Distinguishing between public and private information is important for safe and secure online interactions. Information can be protected using various security measures (i.e., physical and digital).	• Describe physical and digital security measures for protecting sensitive personal information.	• Lesson 14: Problem Solving with Big Data • Lesson 16: Project - Make a Recommendation
8.2.5.ED.2	Engineering design is a systematic and creative process of communicating and collaborating to meet a design challenge. Often, several design solutions exist, each letter in some way than the others.	• Collaborate with peers to collect information, brainstorm to solve a problem, and evaluate all possible solutions to provide the best results with supporting sketches or models.	• Lesson 16: Project - Make a Recommendation
8.2.8.ED.7	Engineering design requirements and specifications involve making trade-offs between competing	• Design a product to address a real-world problem and document the iterative design process, including decisions made as a	• Lesson 16: Project - Make a Recommendation

	requirements and desired design features.	result of specific constraints and trade-offs (e.g., annotated sketches).	

Unit 2 Assessment Plan	
Formative Assessment *When possible, provide links to specific samples/ documents/ assignments/etc.*	**Summative Assessment** *When possible, provide links to specific samples/ documents/ assignments/etc.*
• Observations • Completion of Activity Guides in CSD Unit 5 Resources	• Lesson 8: Project - Create a Representation • Lesson 16: Project - Make a Recommendation • Completion of Activity Guides in CSD Unit 5 Resources

Unit 2 Suggested Modifications/Accommodations/Extension Activities		
English Language Learners (ELL) *When possible, provide links to specific samples/ documents/ assignments/etc.*	**Special Education / 504** *When possible, provide links to specific samples/ documents/ assignments/etc.*	**Gifted and Talented** *When possible, provide links to specific samples/ documents/ assignments/etc.*
CS Discoveries Approach to Differentiation "Learning to use resources is a key goal of the course, and given resources provide an opportunity for students to self-differentiate in how they interact with key course content. This may include proactive differentiation, such as printing out resources ahead of time for students. It can also include just-in-time differentiation, such as monitoring students as they reach the end of a project and referring them to additional resource" **Examples of Strategies and Practices that Support English Language Learners:** • Pre-teaching of vocabulary and concepts • Visual learning, including graphic organizers • Text to Speech • Think-pair-share • Cooperative learning groups • Teacher modeling	CS Discoveries Approach to Differentiation "In order to meet the needs of a wide variety of learners, CS Discoveries is designed with flexibility that allows teachers to differentiate their instruction at the class and student level." *Refer to students' IEP for specific modifications and accommodations **Examples of Strategies and Practices that Support Students with Disabilities:** • Use of visual and multisensory formats • Use of assisted technology • Use of prompts • Modification of content and student products • Testing accommodations • Authentic assessments	CS Discoveries Approach to Differentiation "Challenge levels are found after the assessment levels in many of the programming lessons. These levels include new code and challenges that go beyond the learning objectives of the lesson. Most also include a "Free Play" option that allows students to use the new skills they have learned in whatever way they choose."

• Pairing students with beginning English language skills with students who have more advanced English language skills • Documentation Resource on code.org		

Unit 2 Connections	
NJSLS - Technology *When possible, provide links to specific samples/ documents/ assignments/etc.* Refer to the NJ Technology Standards	**Career Readiness Practices** *When possible, provide links to specific samples/ documents/ assignments/etc.* Refer to the NJ Career Readiness Practices
Technology Standards: Technology standards are embedded throughout all curricular units: Standard 8.1 Computer Science • Computer Science, previously a strand entitled 'Computational Thinking: Programming' in standard 8.2 of the 2014 NJSLS-Technology, outlines a comprehensive set of concepts and skills, such as data and analysis, algorithms and programming, and computing systems. Standard 8.2 Design Thinking • This standard, previously standard 8.2 Technology Education of the 2014 NJSLS – Technology, outlines the technological design concepts and skills essential for technological and engineering literacy. The new framework design, detailed previously, includes Engineering Design, Ethics and Culture, and the Effects of Technology on the Natural world among the disciplinary concepts.	**Career Ready Practices and Standard 9.1, 9.2, and 9.3 Career Ready Practices:** • CRP2. Apply appropriate academic and technical skills. • CRP4. Communicate clearly and effectively and with reason. • CRP5. Consider the environmental, social and economic impacts of decisions. • CRP6. Demonstrate creativity and innovation. • CRP7. Employ valid and reliable research strategies. • CRP8. Utilize critical thinking to make sense of problems and persevere in solving them. • CRP9. Model integrity, ethical leadership and effective management. • CRP10. Plan education and career paths aligned to personal goals. • CRP11. Use technology to enhance productivity. • CRP12. Work productively in teams while using cultural global competence.
21st Century Skills *When possible, provide links to specific samples/ documents/ assignments/etc.* Refer to the 21st Century Life and Skills	**Interdisciplinary Connections** *When possible, provide links to specific ELA/Math/Sci/SS standards as well as samples/ documents/ assignments/etc.* Refer to the NJ Student Learning Standards
21st Century Themes • Career Awareness • Career Exploration **21st Century Skills** • Creativity and Innovation (E) • Critical Thinking and Problem Solving (T) (A) • Communication (E) • Collaboration (E) (T)	**Interdisciplinary connections are made across grades and content areas to model the integration of knowledge and skills in the real world.**

Unit 3: Physical Computing			
Content Standards	**Critical Knowledge & Skills** ("Unpacked" Standards)	**Content-Specific Practices** (when applicable)	**Standard Mastery Examples** *When possible, provide links to specific samples/ documents/ assignments/etc.*
8.2.5.NT.2	Technology innovation and improvement may be influenced by a variety of factors. Engineers create and modify technologies to meet people's needs and wants; scientists ask questions about the natural world.	• Identify new technologies resulting from the demands, values, and interests of individuals, businesses, industries, and societies.	• Lesson 1: Innovations in Computing
8.1.8.AP.2	Programmers create variables to store data values of different types and perform appropriate operations on their values.	• Create clearly named variables that represent different data types and perform operations on their values.	• Lesson 2: Designing Screens with Code • Lesson 3: The Circuit Playground • Lesson 5: Board Events • Lesson 7: Analog Input • Lesson 9: Project - Make a Game • Lesson 10: Arrays and Color LEDs • Lesson 12: Arrays and For Loops • Lesson 15: Circuits and Physical Prototypes • Lesson 16: Project - Prototype an Innovation •
8.1.8.AP.4	Programs use procedures to organize code and hide implementation details. Procedures can be repurposed in new programs. Defining parameters for procedures can generalize behavior and increase reusability.	• Decompose problems and sub-problems into parts to facilitate the design, implementation, and review of programs.	• Lesson 2: Designing Screens with Code • Lesson 3: The Circuit Playground • Lesson 5: Board Events • Lesson 7: Analog Input • Lesson 9: Project - Make a Game • Lesson 10: Arrays and Color LEDs • Lesson 12: Arrays and For Loops • Lesson 16: Project - Prototype an Innovation •

8.1.8.AP.7	Individuals design and test solutions to identify problems taking into consideration the diverse needs of the users and the community.	• Design programs, incorporating existing code, media, and libraries, and give attribution.	• Lesson 2: Designing Screens with Code • Lesson 3: The Circuit Playground • Lesson 5: Board Events • Lesson 7: Analog Input • Lesson 9: Project - Make a Game • Lesson 10: Arrays and Color LEDs • Lesson 12: Arrays and For Loops • Lesson 16: Project - Prototype an Innovation
8.1.8.AP.9	Individuals design and test solutions to identify problems taking into consideration the diverse needs of the users and the community.	• Document programs in order to make them easier to follow, test, and debug.	• Lesson 2: Designing Screens with Code • Lesson 3: The Circuit Playground • Lesson 5: Board Events • Lesson 7: Analog Input • Lesson 9: Project - Make a Game • Lesson 10: Arrays and Color LEDs • Lesson 12: Arrays and For Loops • Lesson 15: Circuits and Physical Prototypes • Lesson 16: Project - Prototype an Innovation
8.1.8.CS.2	Software and hardware determine a computing system's capability to store and process information. The design or selection of a computing system involves multiple considerations and potential trade-offs.	• Design a system that combines hardware and software components to process data.	• Lesson 3: The Circuit Playground • Lesson 5: Board Events • Lesson 7: Analog Input • Lesson 9: Project - Make a Game • Lesson 10: Arrays and Color LEDs • Lesson 12: Arrays and For Loops • Lesson 15: Circuits and Physical Prototypes • Lesson 16: Project - Prototype an Innovation
8.1.8.CS.4	Troubleshooting a problem is more effective when knowledge of the specific device along with a systematic process is used to identify the source of a problem.	• Systematically apply troubleshooting strategies to identify and resolve hardware and software problems in computing systems.	• Lesson 3: The Circuit Playground • Lesson 5: Board Events • Lesson 7: Analog Input • Lesson 8: The Program Design Process • Lesson 9: Project - Make a Game • Lesson 12: Arrays and For Loops

			• Lesson 15: Circuits and Physical Prototypes • Lesson 16: Project - Prototype an Innovation
8.1.2.AP.1	Individuals develop and follow directions as part of daily life. A sequence of steps can be expressed as an algorithm that a computer can process.	• Model daily processes by creating and following algorithms to complete tasks.	• Lesson 4: Input Unplugged • Lesson 9: Project - Make a Game • Lesson 15: Circuits and Physical Prototypes • Lesson 16: Project - Prototype an Innovation
8.1.8.AP.3	Control structures are selected and combined in programs to solve more complex problems.	• Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.	• Lesson 5: Board Events • Lesson 7: Analog Input • Lesson 9: Project - Make a Game • Lesson 12: Arrays and For Loops • Lesson 16: Project - Prototype an Innovation
8.1.5.AP.6	Individuals develop programs using an iterative process involving design, implementation, testing, and review.	• Develop programs using an iterative process, implement the program design, and test the program to ensure it works as intended.	• Lesson 5: Board Events • Lesson 7: Analog Input • Lesson 9: Project - Make a Game • Lesson 12: Arrays and For Loops • Lesson 16: Project - Prototype an Innovation
8.1.8.CS.1	The study of human–computer interaction can improve the design of devices and extend the abilities of humans.	• Recommend improvements to computing devices in order to improve the ways users interact with the devices.	• Lesson 5: Board Events • Lesson 7: Analog Input • Lesson 9: Project - Make a Game • Lesson 12: Arrays and For Loops • Lesson 16: Project - Prototype an Innovation •
8.1.8.AP.5	Programs use procedures to organize code and hide implementation details. Procedures can be repurposed in new programs. Defining parameters for procedures can generalize behavior and increase reusability.	• Create procedures with parameters to organize code and make it easier to reuse.	• Lesson 8: The Program Design Process • Lesson 12: Arrays and For Loops • Lesson 15: Circuits and Physical Prototypes
8.1.12.AP.4	Trade-offs related to implementation, readability, and program performance are considered when selecting and combining control structures.	• Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.	• Lesson 8: The Program Design Process

8.2.5.ED.2	Engineering design is a systematic and creative process of communicating and collaborating to meet a design challenge. Often, several design solutions exist, each better in some way than the others.	• Collaborate with peers to collect information, brainstorm to solve a problem, and evaluate all possible solutions to provide the best results with supporting sketches or models.	• Lesson 9: Project - Make a Game • Lesson 16: Project - Prototype an Innovation
8.2.8.ED.7	Engineering design requirements and specifications involve making trade-offs between competing requirements and desired design features.	• Design a product to address a real-world problem and document the iterative design process, including decisions made as a result of specific constraints and trade-offs (e.g., annotated sketches).	• Lesson 9: Project - Make a Game • Lesson 16: Project - Prototype an Innovation
8.1.8.IC.2	Advancements in computing technology can change individuals' behaviors. Society is faced with trade-offs due to the increasing globalization and automation that computing brings.	• Describe issues of bias and accessibility in the design of existing technologies.	• Lesson 15: Circuits and Physical Prototypes

Unit 3 Assessment Plan	
Formative Assessment *When possible, provide links to specific samples/ documents/ assignments/etc.*	**Summative Assessment** *When possible, provide links to specific samples/ documents/ assignments/etc.*
• Observations • Completion of Assigned Levels for Unit 6 • Completion of Activity Guides in CSD Unit 6 Resources	• Lesson 9: Project - Make a Game • Lesson 16: Project - Prototype an Innovation • Completion of Activity Guides in CSD Unit 6 Resources

Unit 3 Suggested Modifications/Accommodations/Extension Activities		
English Language Learners (ELL) *When possible, provide links to specific samples/ documents/ assignments/etc.*	**Special Education / 504** *When possible, provide links to specific samples/ documents/ assignments/etc.*	**Gifted and Talented** *When possible, provide links to specific samples/ documents/ assignments/etc.*
CS Discoveries Approach to Differentiation	CS Discoveries Approach to Differentiation "In order to meet the needs of a wide variety of learners, CS Discoveries is	CS Discoveries Approach to Differentiation "Challenge levels are found after the assessment levels in many of the

"Learning to use resources is a key goal of the course, and given resources provide an opportunity for students to self-differentiate in how they interact with key course content. This may include proactive differentiation, such as printing out resources ahead of time for students. It can also include just-in-time differentiation, such as monitoring students as they reach the end of a project and referring them to additional resource" **Examples of Strategies and Practices that Support English Language Learners:** • Pre-teaching of vocabulary and concepts • Visual learning, including graphic organizers • Text to Speech • Think-pair-share • Cooperative learning groups • Teacher modeling • Pairing students with beginning English language skills with students who have more advanced English language skills • Documentation Resource on code.org	designed with flexibility that allows teachers to differentiate their instruction at the class and student level." *Refer to students' IEP for specific modifications and accommodations **Examples of Strategies and Practices that Support Students with Disabilities:** • Use of visual and multisensory formats • Use of assisted technology • Use of prompts • Modification of content and student products • Testing accommodations • Authentic assessments	programming lessons. These levels include new code and challenges that go beyond the learning objectives of the lesson. Most also include a "Free Play" option that allows students to use the new skills they have learned in whatever way they choose."

Unit 3 Connections	
NJSLS - Technology *When possible, provide links to specific samples/ documents/ assignments/etc.* Refer to the NJ Technology Standards	**Career Readiness Practices** *When possible, provide links to specific samples/ documents/ assignments/etc.* Refer to the NJ Career Readiness Practices
Technology Standards: Technology standards are embedded throughout all curricular units: Standard 8.1 Computer Science • Computer Science, previously a strand entitled 'Computational Thinking: Programming' in standard 8.2 of the 2014 NJSLS-Technology, outlines a comprehensive set of concepts and skills, such as data and analysis, algorithms and programming, and computing systems. Standard 8.2 Design Thinking • This standard, previously standard 8.2 Technology Education of the 2014 NJSLS – Technology, outlines the technological design concepts and skills essential for technological and engineering	**Career Ready Practices and Standard 9.1, 9.2, and 9.3 Career Ready Practices:** • CRP2. Apply appropriate academic and technical skills. • CRP4. Communicate clearly and effectively and with reason. • CRP5. Consider the environmental, social and economic impacts of decisions. • CRP6. Demonstrate creativity and innovation. • CRP7. Employ valid and reliable research strategies. • CRP8. Utilize critical thinking to make sense of problems and persevere in solving them. • CRP9. Model integrity, ethical leadership and effective management. • CRP10. Plan education and career paths aligned to personal goals.

literacy. The new framework design, detailed previously, includes Engineering Design, Ethics and Culture, and the Effects of Technology on the Natural world among the disciplinary concepts.	• CRP11. Use technology to enhance productivity. • CRP12. Work productively in teams while using cultural global competence.
21st Century Skills *When possible, provide links to specific samples/ documents/ assignments/etc.* Refer to the 21st Century Life and Skills	**Interdisciplinary Connections** *When possible, provide links to specific ELA/Math/Sci/SS standards as well as samples/ documents/ assignments/etc.* Refer to the NJ Student Learning Standards
21st Century Themes • Career Awareness • Career Exploration **21st Century Skills** • Creativity and Innovation (E) • Critical Thinking and Problem Solving (T) (A) • Communication (E) • Collaboration (E) (T)	**Interdisciplinary connections are made across grades and content areas to model the integration of knowledge and skills in the real world.**

Unit 4: AI and Machine Learning			
NJSLS-CSDT 2020 Standards			
Content Standards	**Critical Knowledge & Skills** ("Unpacked" Standards)	**Content-Specific Practices** (when applicable)	**Standard Mastery Examples** *When possible, provide links to specific samples/ documents/ assignments/etc.*
8.1.5.AP.6	Individuals develop programs using an iterative process involving design, implementation, testing, and review.	• Develop programs using an iterative process, implement the program design, and test the program to ensure it works as intended.	• Lesson 1: Introduction to Machine Learning • Lesson 6: Introduction to AI Lab • Lesson 12: Numerical Data in AI Lab
8.1.8.IC.2	Advancements in computing technology can change individuals' behaviors. Society is faced with trade-offs due to the increasing globalization and automation that computing brings.	• Describe issues of bias and accessibility in the design of existing technologies.	• Lesson 1: Introduction to Machine Learning • Lesson 7: Importing Models in App Lab • Lesson 8: Model Cards • Lesson 14: AI Code of Ethics • Lesson 19: Troubleshooting Models •
8.2.5.NT.2	Technology innovation and improvement may be influenced by a variety of factors. Engineers create and modify technologies to meet people's needs and wants; scientists ask questions about the natural world.	• Identify new technologies resulting from the demands, values, and interests of individuals, businesses, industries, and societies.	• Lesson 3: Innovations in AI
8.1.8.DA.5	Computer models can be used to simulate events, examine theories and inferences, or make predictions.	• Test, analyze, and refine computational models.	• Lesson 4: Patterns in Data
8.1.12.DA.6	The accuracy of predictions or inferences made from a computer model is affected by the amount, quality, and diversity of data.	• Create computational models that represent the relationships among different elements of data collected from a phenomenon or process.	• Lesson 5: Classification Models • Lesson 6: Introduction to AI Lab • Lesson 9: Saving Models in AI Lab • Lesson 11: Numerical Models • Lesson 12: Numerical Data in AI Lab

			• Lesson 15: Mini-Project: Make a Machine Learning App • Lesson 18: Survey Data in AI Lab • Lesson 21: Design an AI App
8.1.2.DA.3	Data can be used to make predictions about the world.	• Identify and describe patterns in data visualizations.	• Lesson 6: Introduction to AI Lab • Lesson 9: Saving Models in AI Lab • Lesson 12: Numerical Data in AI Lab • Lesson 15: Mini-Project: Make a Machine Learning App • Lesson 21: Design an AI App
8.1.8.AP.4	Programs use procedures to organize code and hide implementation details. Procedures can be repurposed in new programs. Defining parameters for procedures can generalize behavior and increase reusability.	• Decompose problems and sub-problems into parts to facilitate the design, implementation, and review of programs.	• Lesson 7: Importing Models in App Lab • Lesson 10: Model Cards in App Lab • Lesson 13: Customizing Apps • Lesson 20: Creating an App • Lesson 21: Design an AI App
8.1.12.AP.4	Trade-offs related to implementation, readability, and program performance are considered when selecting and combining control structures.	• Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue.	• Lesson 7: Importing Models in App Lab
8.1.8.AP.9	Individuals design and test solutions to identify problems taking into consideration the diverse needs of the users and the community.	• Document programs in order to make them easier to follow, test, and debug.	• Lesson 9: Saving Models in AI Lab • Lesson 10: Model Cards in App Lab • Lesson 13: Customizing Apps • Lesson 15: Mini-Project: Make a Machine Learning App • Lesson 18: Survey Data in AI Lab • Lesson 21: Design an AI App
8.1.8.AP.2	Programmers create variables to store data values of different types and perform appropriate operations on their values.	• Create clearly named variables that represent different data types and perform operations on their values.	• Lesson 10: Model Cards in App Lab • Lesson 13: Customizing Apps • Lesson 20: Creating an App • Lesson 21: Design an AI App

8.1.8.AP.3	Control structures are selected and combined in programs to solve more complex problems.	• Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.	• Lesson 10: Model Cards in App Lab • Lesson 20: Creating an App • Lesson 21: Design an AI App
8.1.12.IC.1	The design and use of computing technologies and artifacts can positively or negatively affect equitable access to information and opportunities.	• Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.	• Lesson 14: AI Code of Ethics
8.2.5.ED.2	Engineering design is a systematic and creative process of communicating and collaborating to meet a design challenge. Often, several design solutions exist, each letter in some way than the others.	• Collaborate with peers to collect information, brainstorm to solve a problem, and evaluate all possible solutions to provide the best results with supporting sketches or models.	• Lesson 16: Issue Statements • Lesson 17: Survey Planning • Lesson 19: Troubleshooting Models
8.1.12.AP.7	Complex programs are developed, tested, and analyzed by teams drawing on the members' diverse strengths using a variety of resources, libraries, and tools.	• Collaboratively design and develop programs and artifacts for broad audiences by incorporating feedback from users.	• Lesson 17: Survey Planning • Lesson 18: Survey Data in AI Lab • Lesson 19: Troubleshooting Models • Lesson 21: Design an AI App
8.1.8.DA.1	People use digital devices and tools to automate the collection, use, and transformation of data. The manner in which data is collected and transformed is influenced by the type of digital device(s) available and the intended use of the data.	• Organize and transform data collected using computational tools to make it usable for a specific purpose.	• Lesson 18: Survey Data in AI Lab • Lesson 21: Design an AI App
AI4K12 National Guidelines 2021			
Content Standards	**Critical Knowledge & Skills** ("Unpacked" Standards)	**Content-Specific Practices** (when applicable)	**Standard Mastery Examples** *When possible, provide links to specific samples/ documents/ assignments/etc.*
3-A-ii.3-5	To give students a feel for the problem of learning to classify we must ask them to learn a class that's not intuitively obvious, e.g., learn "poisonous fish"by examining cartoon fish images labeled "poisonous" or	• Model how supervised learning identifies patterns in labeled data.	• Lesson 1: Introduction to Machine Learning • Lesson 5: Classification Models • Lesson 11: Numerical Models

	"not poisonous". They can then be asked to describe which features indicate a fish is poisonous, e.g., red fish with square heads. Using images as input simplifies the task because the features are intuitive, even though the classification rule should not be This extends the K-2 version by having students draw a decision tree instead of merely verbalizing their proposed rule. In addition, the task can be made richer in 3-5 by increasing the number of classes or by making the class definitions more complex. For example, a fish could be poisonous if it is either red with a square head or blue with a round head or purple with pointy spines and any shape head. Each node of the decision tree can test one feature value, e.g., color,so complex features require deeper trees.		
3-A-iv.3-5	Teachable Machine provides three classes by default and has a separate "Hold to Record" button for each class, so training examples are implicitly labeled based on which class they are recorded for. After training, the model is classifying webcam input in real time but receives no feedback.	• Demonstrate how training data are labeled when using a machine learning tool.	• Lesson 1: Introduction to Machine Learning
3-A-i.6-8	People are flexible learners who employ multiple strategies. Computers use specialized algorithms that require large amounts of data or many trials, and only solve narrowly defined problems. While humans can construct reasoners by explicitly programming them, for complex problems it is often more convenient to let the machine learning algorithm do the work.	• Contrast the unique characteristics of human learning with the ways machine learning systems operate.	• Lesson 2: Types of Machine Learning • Lesson 6: Introduction to AI Lab

3-A-i.9-12	Supervised learning is like being corrected by a coach. Unsupervised learning is like noticing that your store has three kinds of customers based on their distinctive purchasing patterns. Reinforcement learning is like trying different moves in a video game and seeing which yields the most points (greatest reward).	• Define supervised, unsupervised, and reinforcement learning algorithms, and give examples of human learning that are similar to each algorithm.	• Lesson 2: Types of Machine Learning
3-A-ii.K-2	To give students a feel for the problem of learning to classify we must ask them to learn a class that's not intuitively obvious, e.g., learn "poisonous fish" by examining cartoon fish images labeled "poisonous" or "not poisonous". They can then be asked to describe which features indicate a fish is poisonous, e.g., red fish with square heads. Using images as input simplifies the task because the features are intuitive, even though the classification rule should not be.	• Identify patterns in labeled data and determine the features that predict labels.	• Lesson 4: Patterns in Data
3-C-i.K-2	Classify food as healthy/unhealthy, or classify toys as safe or unsafe for babies. Optionally, they could build a decision tree using these features, but most important to understand is that they could feed their data to a machine learning algorithm to create the decision tree for them.	• Create a labeled dataset with explicit features to illustrate how computers can learn to classify things like foods, movies, or toys.	• Lesson 4: Patterns in Data • Lesson 17: Survey Planning
3-A-iii.6-8	Within a tabular dataset, each training example is a row in the table and is described by a set of feature values; the features are the columns of the table. Classification assigns each example to one of a discrete set of classes (e.g., cat or dog); prediction outputs a continuous value, such as predicting a person's height from their age. The learning algorithm is likely to be a decision	• Train and evaluate a classification or prediction model using machine learning on a tabular dataset	• Lesson 5: Classification Models • Lesson 6: Introduction to AI Lab • Lesson 9: Saving Models in AI Lab • Lesson 12: Numerical Data in AI Lab • Lesson 15: Mini-Project: Make a Machine Learning App • Lesson 18: Survey Data in AI Lab • Lesson 21: Design an AI App

	tree learner rather than a neural network		
3-C-i.3-5	Sites such as MachineLearningForKids will train decision tree classifiers based on data of this type.	• Create a labeled dataset with explicit features of several types and use a machine learning tool to train a classifier on this data.	• Lesson 6: Introduction to AI Lab • Lesson 9: Saving Models in AI Lab
3-C-i.6-8	At each node of the decision tree, the learning algorithm tries to pick a feature that will be most helpful in separating the remaining instances into different classes. Features that don't correlate strongly with any class will not be chosen.	• Create a dataset for training a decision tree classifier or predictor and explore the impact that different feature encodings have on the decision tree.	• Lesson 6: Introduction to AI Lab • Lesson 9: Saving Models in AI Lab • Lesson 15: Mini-Project: Make a Machine Learning App • Lesson 18: Survey Data in AI Lab • Lesson 21: Design an AI App
3-A-iii.3-5	Computers can learn to classify instances or predict values by being shown labeled examples. If the results on new inputs are unsatisfactory, additional training may be required to improve the accuracy	• Train a classification model using machine learning, and then examine the accuracy of the model on new inputs	• Lesson 8: Model Cards • Lesson 11: Numerical Models
3-C-iii.3-5	Machine learning algorithms require a representative collection of data in order to build an accurate model. Training datasets drawn from historical data may reflect pre-existing human and societal biases	• Examine features and labels of training data to detect potential sources of bias.	• Lesson 8: Model Cards • Lesson 19: Troubleshooting Models
3-C-iii.6-8	Bias can result if the model is asked to classify inputs that don't resemble the training data, or if the training data contains irrelevant correlations we don't want the classifier to rely on.	• Explain how the choice of training data shapes the behavior of the classifier, and how bias can be introduced if the training set is not properly balanced.	• Lesson 8: Model Cards • Lesson 15: Mini-Project: Make a Machine Learning App • Lesson 17: Survey Planning • Lesson 19: Troubleshooting Models • Lesson 21: Design an AI App

3-A-ii.9-12	In supervised learning the model is trained on a training set to produce the correct labels for labeled data. We evaluate the results by measuring the percent of items in a test set that are labeled correctly. In unsupervised learning, the model is trained to assign each input to a cluster of similar inputs. The clusters are determined by the learning algorithm since there are no labels attached to the training data. We evaluate the results by examining the clusters to see if they capture useful distinctions in the dataset.	• Use either a supervised or unsupervised learning algorithm to train a model on real world data, then evaluate the results.	• Lesson 15: Mini-Project: Make a Machine Learning App • Lesson 21: Design an AI App
3-A-iv.9-12	The steps are: deciding what problem you want to solve, figuring out where you will obtain the training data, choosing a feature set, figuring out how to label the data, running the learning algorithm, use of a cross-validation set to decide when training should stop, and using a test set to measure performance.	• Illustrate what happens during each of the steps required when using machine learning to construct a classifier or predictor.	• Lesson 16: Issue Statements • Lesson 17: Survey Planning • Lesson 18: Survey Data in AI Lab • Lesson 19: Troubleshooting Models • Lesson 21: Design an AI App
3-C-i.9-12	Humans decide which features to include in a dataset and how to encode them. This can have consequences for machine learning algorithms trained on these datasets.	• Compare two real world datasets in terms of the features they comprise and how those features are encoded.	• Lesson 8: Model Cards • Lesson 18: Survey Data in AI Lab

3-A-ii.9-12	In supervised learning the model is trained on a training set to produce the correct labels for labeled data. We evaluate the results by measuring the percent of items in a test set that are labeled correctly. In unsupervised learning, the model is trained to assign each input to a cluster of similar inputs. The clusters are determined by the learning algorithm since there are no labels attached to the training data. We evaluate the results by examining the clusters to see if they capture useful distinctions in the dataset.	• Use either a supervised or unsupervised learning algorithm to train a model on real world data, then evaluate the results.	• Lesson 21: Design an AI App

Unit 4 Assessment Plan	
Formative Assessment *When possible, provide links to specific samples/ documents/ assignments/etc.*	**Summative Assessment** *When possible, provide links to specific samples/ documents/ assignments/etc.*
• Observations • Completion of Assigned Levels for Unit 7 • Completion of Activity Guides in CSD Unit 7 Resources	• Lesson 15: Make a Machine Learning App (Project Guide) • Lesson 21: Design an AI App • Completion of Activity Guides in CSD Unit 7 Resources

Unit 4 Suggested Modifications/Accommodations/Extension Activities		
English Language Learners (ELL) *When possible, provide links to specific samples/ documents/ assignments/etc.*	**Special Education / 504** *When possible, provide links to specific samples/ documents/ assignments/etc.*	**Gifted and Talented** *When possible, provide links to specific samples/ documents/ assignments/etc.*
CS Discoveries Approach to Differentiation "Learning to use resources is a key goal of the course, and given resources provide an opportunity for students to self-differentiate in how they interact with key course content. This may include proactive differentiation, such as printing out resources ahead of time for students. It can also include just-in-time differentiation, such as monitoring students as	CS Discoveries Approach to Differentiation "In order to meet the needs of a wide variety of learners, CS Discoveries is designed with flexibility that allows teachers to differentiate their instruction at the class and student level." *Refer to students' IEP for specific modifications and accommodations	CS Discoveries Approach to Differentiation "Challenge levels are found after the assessment levels in many of the programming lessons. These levels include new code and challenges that go beyond the learning objectives of the lesson. Most also include a "Free Play" option that allows students to use the new

they reach the end of a project and referring them to additional resource" **Examples of Strategies and Practices that Support English Language Learners:** • Pre-teaching of vocabulary and concepts • Visual learning, including graphic organizers • Text to Speech • Think-pair-share • Cooperative learning groups • Teacher modeling • Pairing students with beginning English language skills with students who have more advanced English language skills • Documentation Resource on code.org	**Examples of Strategies and Practices that Support Students with Disabilities:** • Use of visual and multisensory formats • Use of assisted technology • Use of prompts • Modification of content and student products • Testing accommodations • Authentic assessments	skills they have learned in whatever way they choose."

Unit 4 Connections	
NJSLS - Technology *When possible, provide links to specific samples/ documents/ assignments/etc.* Refer to the NJ Technology Standards	**Career Readiness Practices** *When possible, provide links to specific samples/ documents/ assignments/etc.* Refer to the NJ Career Readiness Practices
Technology Standards: Technology standards are embedded throughout all curricular units: Standard 8.1 Computer Science • Computer Science, previously a strand entitled 'Computational Thinking: Programming' in standard 8.2 of the 2014 NJSLS-Technology, outlines a comprehensive set of concepts and skills, such as data and analysis, algorithms and programming, and computing systems. Standard 8.2 Design Thinking • This standard, previously standard 8.2 Technology Education of the 2014 NJSLS – Technology, outlines the technological design concepts and skills essential for technological and engineering literacy. The new framework design, detailed previously, includes Engineering Design, Ethics and Culture, and the Effects of Technology on the Natural world among the disciplinary concepts.	**Career Ready Practices and Standard 9.1, 9.2, and 9.3 Career Ready Practices:** • CRP2. Apply appropriate academic and technical skills. • CRP4. Communicate clearly and effectively and with reason. • CRP5. Consider the environmental, social and economic impacts of decisions. • CRP6. Demonstrate creativity and innovation. • CRP7. Employ valid and reliable research strategies. • CRP8. Utilize critical thinking to make sense of problems and persevere in solving them. • CRP9. Model integrity, ethical leadership and effective management. • CRP10. Plan education and career paths aligned to personal goals. • CRP11. Use technology to enhance productivity. • CRP12. Work productively in teams while using cultural global competence.
21st Century Skills *When possible, provide links to specific samples/ documents/ assignments/etc.*	**Interdisciplinary Connections** *When possible, provide links to specific ELA/Math/Sci/SS standards as well as samples/ documents/ assignments/etc.*

Refer to the 21st Century Life and Skills	Refer to the NJ Student Learning Standards
21st Century Themes • Career Awareness • Career Exploration **21st Century Skills** • Creativity and Innovation (E) • Critical Thinking and Problem Solving (T) (A) • Communication (E) • Collaboration (E) (T)	**Interdisciplinary connections are made across grades and content areas to model the integration of knowledge and skills in the real world.**