Advanced Placement

Computer Science Principles

Syllabus

Instructor

Teacher:

School:

Course Overview

CS Matters in Maryland (http://csmatters.org/) is an NSF-funded CS10K project focusing on the new AP Computer Science Principles (CSP) course. The objectives of the course are consistent with those defined within the AP Computer Science Principles Curriculum Framework by the College Board. This course will be taught using the curriculum developed by CS Matters.

The CS Matters CSP curriculum was developed with one key goal: to provide all students the opportunity to learn computer science within a rigorous and engaging framework. To reach, retain, and teach traditionally underrepresented groups, the curriculum is designed to foster welcoming learning environments that are respectful of the diverse strengths of all students. The theme of the CS Matters course is *data* -- where it comes from, how it is collected and made available, how it can be analyzed and visualized, and the impact of “big data” on society.

Unit 1, *Your Virtual World*, informs and involves students in the many ways in which computing shapes their environment. Students study scalable problem solving by participating in citizen science, contributing their thoughts and recording their reactions in daily journals, investigating innovations of particular importance to them, and collaborating with partners and groups. Core lessons from Unit 1 include Impact of Innovation, A Problem Solving Process that Scales, Societal Impact, and Privacy in the Age of Big Data.

Unit 2, *Developing Programming Tools*, introduces students to software development using the Python programming language. The unit begins by focusing on the motivation for programming and then teaches the fundamentals of procedural programming, including data storage and retrieval, sequence, selection, iteration, and functions. This unit plays a pivotal role that allows subsequent units to challenge students to implement their own code to investigate their virtual world.

Unit 3, *Information and the Internet*, continues the emphasis on impact while examining the Internet, its core technologies, and its design. The unit explores how the design and technologies of the Internet affect innovation. Students take on the roles of Internet technologies by acting out the parts these technologies play. The first and third units together equip students to complete the College Board's *Explore* Performance Task.

In the Explore Performance Task (EPT), students choose and explore a computing innovation. The EPT requires students to select and investigate a computational innovation that: has or has had the potential to have significant beneficial and harmful effects on our society, economy, or culture, consumes, produces, and/or transforms data and raises at least one data storage concern, data privacy concern, or data security concern. Students will have 8 hours (480 minutes) of class time to complete the EPT.

Unit 4, *Data Acquisition*, focuses on data, modeling, and simulations, while introducing fundamental concepts of probability and statistics. Students address the potential and limits of modeling by developing and testing hypotheses. Using computational thinking and the programming skills they learned in the prior unit, students build and test a model that leverages the power of computing to increase the accuracy of its results.

Unit 5, *Data Manipulation*, orients students to the conceptual foundations and core strategies for managing big data. Students investigate several data manipulation strategies, focusing on common algorithms and methods of evaluating them. The study of algorithms leads to small individual programming projects that acquaint students with the College Board's *Create* performance task.

Unit 6, *Data Visualization*, serves as a bridge between the introduction to computing and the development of more substantial programming artifacts. This unit includes several options for teachers to strengthen their students' creative programming abilities. The first lessons use EarSketch to engage students in computation through collaborative music composition. Other lessons use Bokeh from Continuum Analytics to equip students to create their own data visualizations.

In the Create Performance Task (CPT), students bring ideas to life through software development. The CPT requires students to design and iteratively develop a program. The College Board encourages but does not require students to collaborate with a partner, and once students begin the development of the CPT, they may receive help only from the collaborative partner. While not requiring it, this lesson supports a collaborative effort. Even if collaboration is used, each student must independently complete a significant level of planning, designing, and developing the program. Students will have 12 hours (720 minutes) of class time to complete the CPT.

Instruction in the Six Computational Practices and Seven Big Ideas is integrated throughout the course. The following table indicates the number of times each learning objective is addressed in each unit. Each Big Idea is taught in at least three units. Most

Big Ideas are taught in every unit. The table below indicates the number of times lesson objectives associated with each big idea are taught in each unit. These are in addition to the instructional values of the Performance Tasks.

Summary of Big Ideas Taught in Each Unit							
	Big Ideas						
Unit	1	2	3	4	5	6	7
1	2	2	1	3	0	0	9
2	11	1	1	3	31	0	2
3	11	11	3	7	1	20	8
4	6	9	20	2	9	0	0
5	3	6	7	9	0	1	0
6	27	20	4	6	33	4	4

Six Computational Thinking Practices are used by students in each unit of the course.

P1: Connecting Computing
P2: Creating Computational Artifacts
P3: Abstracting
P4: Analyzing Problems and Artifacts
P5: Communicating
P6: Collaborating

Seven Big Ideas guide student activities throughout the course.

BI1: Creativity
BI2: Abstraction
BI3: Data and Information
BI4: Algorithms
BI5: Programming
BI6: The Internet
BI7: Global Impact

Sample Assignment Associations			
Sample Assignments	Page Numbers	Computational Practices	Big Ideas
1	5	1, 4 and 6	1, 6 and 7
2	7	2, 4 and 6	1, 4 and 5
3	11	1, 2, 3, 4 and 5	2 and 6
4	14	1, 3 and 5	2, 3 and 7
5	17	1, 4 and 5	1 and 3
6	20	2, 3, 5 and 6	1, 2, 4 and 5

Course Content by Unit

1. Your Virtual World

The main idea of Unit 1 is to explore the effect of data on students' lives. Students will discover what is known about them online, explore the issue of data privacy, and the rights to access or change that data. After finding ways that the world of information is changing, students will learn how data is stored using binary codes, hardware and files.

Unit 1 Lessons:

1 - 1. Into the Darkness: A World Without Digital Communication
1 - 2. Into the Light: How Computers and the Internet Enhance Innovation
1 - 3. Exploring Innovations
1 - 4. It's Just Bits
1 - 5. How Innovation Affects Our Lives
1 - 6. A Problem Solving Process that Scales
1 - 7. Unit 1 Assessment
1 - 8. Practice Performance Task

Unit 1 Learning Objectives:

Creativity	**1.1.1**	**Apply a creative development process when creating computational artifacts.**
Creativity	**1.2.1**	**Create a computational artifact for creative expression.**
Creativity	**1.2.2**	**Create a computational artifact using computing tools and techniques to solve a problem.**
Creativity	**1.2.3**	**Create a new computational artifact by combining or modifying existing artifacts.**
Creativity	**1.2.5**	**Analyze the correctness, usability, functionality, and suitability of computational artifacts.**
Creativity	**1.3.1**	**Use computing tools and techniques for creative expression.**
Abstraction	**2.1.1**	**Describe the variety of abstractions used to represent data.**
Abstraction	**2.1.2**	**Explain how binary sequences are used to represent digital data.**
Abstraction	**2.2.3**	**Identify multiple levels of abstractions used when writing programs.**
Data	**3.1.3**	**Explain the insight and knowledge gained from digitally processed data by using appropriate visualizations, notation and precise language.**
Data	**3.3.1**	**Analyze how data representation, storage, security, and transmission of data involve computational manipulation of information.**
Internet	**6.1.1**	**Explain the abstractions in the Internet and how it functions.**
Internet	**6.2.1**	**Explain characteristics of the Internet and the systems built on it.**
Internet	**6.2.2**	**Explain how the characteristics of the Internet influence the systems built on it.**

Impact	**7.1.1**	**Explain how computing innovations affect communication, interaction and cognition.**
Impact	**7.1.2**	**Explain how people participate in a problem solving process that scales.**
Impact	**7.2.1**	**Explain how computing has impacted innovations in other fields.**
Impact	**7.3.1**	**Analyze the beneficial and harmful effects of computing.**
Impact	**7.4.1**	**Explain the connections between computing and economic, social and cultural contexts.**

Computational Thinking Practices used in Unit 1: P1, P2, P4, P5 & P6

Sample Assignment 1

Students work in pairs to examine Wikipedia article, “Ten Commandments of Computer Ethics” (http://en.wikipedia.org/wiki/Ten_Commandments_of_Computer_Ethics) and identify and share the two commandments they think are the most commonly violated.

Pairs discuss how the commandments can be violated using cloud technology or social media.

Pairs prepare to share with the class how two rules are commonly violated. Identify the commandments commonly violated and the impact or consequence of each violation.

As a class, produce a revised version of the commandments.

Associated Computational Practices: P1, P4 and P6

Associated Big Ideas: BI1, BI6, BI7

Learning Objectives and Associated Computational Practices:

LO 1.2.4 [P6]

LO 6.3.1 [P1]

LO 7.1.1, [P4]

LO 7.3.1 [P4]

LO 7.4.1 [P1]

2. Developing Programs

The main idea of Unit 2 is to create solutions with code, debug, and verify results. Programming uses creative expression to solve problems correctly using abstraction, both individually and collaboratively. Students implement algorithms using math and logic, and evaluate programs for correctness.

The programming unit takes the formality of algorithm expression one step further by having students write programs in Python. In the programming unit, the levels of abstraction increase as students learn how a programming language can be used to control the machine on which the program is running. The topics of the programming unit are input/output, calculations with numbers, branching statements and Boolean logic, iteration, procedural abstraction, and processing data in a list.

We plan to use Python 3.x with the PyCharm IDE and the textbook *How to Think Like a Computer Scientist,* hosted by Runestone Interactive: (http://interactivepython.org/runestone/static/CCPS_Python/index.html)

Unit 2 Lessons:

2 - 1. Programming: Introduction and Motivation
2 - 2. Using Python and PyCharm
2 - 3. Algorithms: Basics
2 - 4. Algorithms: Pseudocode
2 - 5. Reading Python Code and Debugging
2 - 6. Types and Evaluation
2 - 7. Creating and Assigning Variables
2 - 8. Comparison, Logical Operators, and Conditional Execution
2 - 9. Nested and Chained Conditional Statements
2 - 10. Iteration: For Loops
2 - 11. Iteration: While Loops
2 - 12. Functions: Parameters and Return Values
2 - 13. Algorithms: Layers of Abstraction
2 - 14. Functions: Scope and Abstraction
2 - 15. Strings: Traversing, Slicing, and Parsing
2 - 16. Lists: Creation, Traversal, Insertion, and Removal
2 - 17. Unit Assessment

Unit 2 Learning Objectives:

Creativity	**1.1.1**	**Apply a creative development process when creating computational artifacts.**
Creativity	**1.2.1**	**Create a computational artifact for creative expression.**
Creativity	**1.2.2**	**Create a computational artifact using computing tools and techniques to solve a problem.**
Creativity	**1.2.3**	**Create a new computational artifact by combining or modifying existing artifacts.**

Creativity	**1.2.4**	**Collaborate in the creation of computational artifacts.**
Creativity	**1.2.5**	**Analyze the correctness, usability, functionality, and suitability of computational artifacts.**
Creativity	**1.3.1**	**Use computing tools and techniques for creative expression.**
Abstraction	**2.1.2**	**Explain how binary sequences are used to represent digital data.**
Abstraction	**2.2.3**	**Identify multiple levels of abstractions used when writing programs.**
Abstraction	**2.3.1**	**Use models and simulations to represent phenomena.**
Abstraction	**2.3.2**	**Use models and simulations to formulate, refine and test hypothesis.**
Data	**3.1.1**	**Use computers to process information, find patterns and test hypothesis about digitally processed information to gain insight and knowledge.**
Data	**3.1.2**	**Collaborate when processing information to gain insight and knowledge.**
Data	**3.1.3**	**Explain the insight and knowledge gained from digitally processed data by using appropriate visualizations, notation and precise language.**
Data	**3.2.1**	**Extract information from data to discover and explain connections, patterns or trends.**
Data	**3.2.2**	**Use large data sets to explore and discover information and knowledge.**
Data	**3.3.1**	**Analyze how data representation, storage, security, and transmission of data involve computational manipulation of information.**
Algorithms	**4.1.1**	**Develop an algorithm for implementation in a program.**
Algorithms	**4.1.2**	**Express an algorithm in a language.**
Programming	**5.1.2**	**Develop a correct program to solve problems.**
Programming	**5.1.3**	**Collaborate to develop a program.**
Programming	**5.2.1**	**Explain how programs implement algorithms.**
Programming	**5.3.1**	**Use abstraction to manage complexity in programs.**
Programming	**5.4.1**	**Evaluate the correctness of a program.**
Programming	**5.5.1**	**Employ appropriate mathematical and logical concepts in programming.**

Computational Thinking Practices used in Unit 2: P1, P2, P3, P4, P5 and P6

Sample Assignment 2

LinkyListy Role Play:

Create a human “list of students” by starting with an empty list the front of the room and then modifying the list by following the following program.

Designate a section of the board or a poster to act as the console/printer for output and a volunteer student to act as the printer driver.

Designate an area at the front of the room for the computer memory (and future students).

The names should be changed to match those of students in the class. Also designate three students to play the role of fish and one as a dog.

As students are called by the program, have each student come up to stand in front of the room where memory resides. Have each student point to the student that follows them to form a list as they come to the front.

```
students = []
students.append('Joe')
students.append('Pat')
students.append('Alea')
students.append('Marta')
print(students)
#add additional commands to append or insert  4 or 5 more
students
print(students)
print(len(students))
print(students[3])
students.reverse()
print(students)
print('David' in students)
students.sort()
print(students)
more = ['Tom', 'Laverne']
students = students + more
print(students)
pets = ['fish'*3,'dog']
del students[1]
del students[2:4]
students.insert(0, 'Jennifer')
students = students + pets
print(students)
print(students.index('Marta'))
```

Associated Computational Practices: P2, P3, P4, P6

Associated Big Ideas: BI1, BI4, BI5

Learning Objectives and Associated Computational Practices:

LO 1.2.4 [P6]

LO 4.1.1 [P2]

LO 5.1. [P6]

LO 5.3.1 [P3]

LO 5.4.1 [P4]

3. Information and the Internet

The main idea of Unit 3 is to explore the Internet to prepare students to do research on a current Internet related topic in detail on their own. A variety of algorithms from different areas of interest are investigated. Students will learn how information is transmitted online and how search engines find and organize the data using complex algorithms. Studying cybersecurity and cryptography allows students to understand more about the issues concerning the privacy of data online which leads to a discussion on the ethics and social issues affecting the increased use of online data.

Unit 3 Lessons:

3 - 1. The Internet: Basics of Information Transmission
3 - 2. The Internet: Present and Future
3 - 3. How the Internet Works: Routing
Optional 3 - 4. How the Internet Works: Domain Name System
Optional 3 - 5. How the Internet Works: DNS Activity
3 - 6. Search Engines: Finding Information
3 - 7. Search Engines: Page Rank and Retrieval
3 - 8. Basic Statistics with Excel
3 - 9. Practice for Explore Performance Task
3 - 10. Cybersecurity: Attacks, Protection, and Impact
3 - 11. Cryptography: Symmetric Encryption

Unit 3 Learning Objectives:

Creativity	**1.1.1**	**Apply a creative development process when creating computational artifacts.**
Creativity	**1.2.1**	**Create a computational artifact for creative expression.**
Creativity	**1.2.2**	**Create a computational artifact using computing tools and techniques to solve a problem.**
Creativity	**1.2.3**	**Create a new computational artifact by combining or modifying existing artifacts.**
Creativity	**1.2.5**	**Analyze the correctness, usability, functionality, and suitability of computational artifacts.**
Creativity	**1.3.1**	**Use computing tools and techniques for creative expression.**
Abstraction	**2.1.1**	**Describe the variety of abstractions used to represent data.**
Abstraction	**2.1.2**	**Explain how binary sequences are used to represent digital data.**
Data	**3.3.1**	**Analyze how data representation, storage, security, and transmission of data involve computational manipulation of information.**
Algorithms	**4.1.1**	**Develop an algorithm for implementation in a program.**
Algorithms	**4.2.2**	**Explain the difference between solvable and unsolvable problems in computer science.**
Algorithms	**4.2.3**	**Explain the existence of undecidable problems in computer science.**
Programming	**5.1.1**	**Develop a program for creative expression to satisfy personal curiosity or to create new knowledge.**

Internet	**6.1.1**	**Explain the abstractions in the Internet and how it functions.**
Internet	**6.2.1**	**Explain characteristics of the Internet and the systems built on it.**
Internet	**6.2.2**	**Explain how the characteristics of the Internet influence the systems built on it.**
Internet	**6.3.1**	**Identify existing cybersecurity concerns and potential options that address these issues with the Internet and the systems built on it.**
Impact	**7.1.1**	**Explain how computing innovations affect communication, interaction and cognition.**
Impact	**7.3.1**	**Analyze the beneficial and harmful effects of computing.**
Impact	**7.4.1**	**Explain the connections between computing and economic, social and cultural contexts.**

Computational Thinking Practices used in Unit 3: P1, P2, P3, P4, P5 and P6

Sample Assignment 3

In this lesson, students will expand their knowledge of how the Domain Name System (DNS) works by acting as a class to simulate the use of DNS to retrieve web pages.

Once the simulation is functioning students enhance its efficiency through the use of caching.

Poison the DNS cache by adding false DNS replies (DNS poisoning).

Students discuss with their groups how DNS works and how it supports Internet growth. Then they explain in their journals how:

- DNS works.
- Caching is both a benefit and a security risk.
- DNS supports Internet growth.

Associated Computational Practices: P1, P2, P3, P4, P5

Associated Big Ideas: BI2, BI6

Learning Objectives and Associated Computational Practices:

LO 2.3.1 [P3]

LO 4.1.1 [P2]

LO 6.1,1 [P3]

LO 6.2.1 [P5]

LO 6.2.2 [P4]

LO 6.3.1 [P1]

Performance Task: **Explore**. Research an innovation that uses the Internet.

Students will have 8 hours (480 minutes) of class time to complete this performance task.

4. Data Acquisition

The main idea of Unit 4 is to enable students to find data online and analyze it using a spreadsheet and Python code. To prepare for the performance task, students will follow the pattern of developing a hypothesis and then digging into data to find the answer. File input and output will allow students to use large sets of data, find patterns, look for metadata (data about data) and create visualizations of the data. Students will also investigate what is a reasonable solution and how the new data revolution can drive discovery and decision making.

Unit 4 Lessons:

4 - 1. Data Acquisition and Analysis
4 - 2. What are Models and Simulations?
4 - 3. Using Data and Simulations
4 - 4. File Input and Output using Python
4 - 5. Data Collection, Analysis, and Simulation
Optional 4 - 6. Hypothesis Testing with Simulations in NetLogo
4 - 7. Unit 4 Assessment

Unit 4 Learning Objectives:

Creativity	**1.1.1**	**Apply a creative development process when creating computational artifacts.**
Creativity	**1.2.1**	**Create a computational artifact for creative expression.**
Creativity	**1.2.2**	**Create a computational artifact using computing tools and techniques to solve a problem.**
Creativity	**1.2.3**	**Create a new computational artifact by combining or modifying existing artifacts.**
Creativity	**1.2.4**	**Collaborate in the creation of computational artifacts.**
Creativity	**1.2.5**	**Analyze the correctness, usability, functionality, and suitability of computational artifacts.**
Creativity	**1.3.1**	**Use computing tools and techniques for creative expression.**
Abstraction	**2.2.3**	**Identify multiple levels of abstractions used when writing programs.**
Abstraction	**2.3.1**	**Use models and simulations to represent phenomena.**
Abstraction	**2.3.2**	**Use models and simulations to formulate, refine and test hypothesis.**
Data	**3.1.1**	**Use computers to process information, find patterns and test hypothesis about digitally processed information to gain insight and knowledge.**
Data	**3.1.2**	**Collaborate when processing information to gain insight and knowledge.**
Data	**3.1.3**	**Explain the insight and knowledge gained from digitally processed data by using appropriate visualizations, notation and precise language.**
Data	**3.2.1**	**Extract information from data to discover and explain connections, patterns or trends.**
Data	**3.2.2**	**Use large data sets to explore and discover information and knowledge.**
Data	**3.3.1**	**Analyze how data representation, storage, security, and transmission of data**

		involve computational manipulation of information.
Algorithms	4.2.1	Explain the difference between algorithms that run in a reasonable time and those that do not run in a reasonable time.
Programming	5.1.2	Develop a correct program to solve problems.
Internet	6.3.1	Identify existing cybersecurity concerns and potential options that address these issues with the Internet and the systems built on it.
Impact	7.4.1	Explain the connections between computing and economic, social and cultural contexts.

Sample Assignment: 4

Part 1: View the following videos.

Bill Nye and a scaled model of the solar system (4:17)
https://www.youtube.com/watch?v=97Ob0xR0Ut8

Computer Generated Model of a Solar System (2:41)
https://www.youtube.com/watch?v=8z5mwAlxBYc

Part 2: Each student creates a journal entry responding to these two questions:

What was the main idea presented by each video?

What aspect(s) of the models helped make that point?

Part 3: Students discuss each of the following with elbow partners then groups.

How do the models in these videos depend on computing?

Consider the strengths and weaknesses of each model. What understanding can be better drawn from the first model and what understanding can better be drawn from the second?

What questions could be answered using these two models?

Part 4: From each group students share at least one response to each prompt.

Associated Computational Practices: P1, P3 and P5

Associated Big Ideas: BI2, BI3, BI7

Learning Objectives and Associated Computational Practices:

LO 2.3.1 [P3]

LO 2.3.2 [P3]

LO 3.1.3 [P5]

LO 7.4.1 [P1]

5. Data Manipulation

The main idea of Unit 5 is to create a synthesis of algorithms, programming, and data. Python lists will be used to work with a large set of data using searching, sorting, and other ways to manipulate data. Models and simulations will be used to represent real-life situations. Algorithms will be compared for their effectiveness as well as their readability.

Unit 5 Lessons:

5 - 1. Manipulating Large Data Sets
5 - 2. Searching
5 - 3. Sorting
5 - 4. Comparing Algorithms
5 - 5. Advanced Algorithms
5 - 6. Create Performance Task Partial Practice

Unit 5 Learning Objectives:

Creativity	**1.1.1**	**Apply a creative development process when creating computational artifacts.**
Creativity	**1.2.1**	**Create a computational artifact for creative expression.**
Creativity	**1.2.2**	**Create a computational artifact using computing tools and techniques to solve a problem.**
Creativity	**1.2.3**	**Create a new computational artifact by combining or modifying existing artifacts.**
Creativity	**1.2.4**	**Collaborate in the creation of computational artifacts.**
Creativity	**1.2.5**	**Analyze the correctness, usability, functionality, and suitability of computational artifacts.**
Creativity	**1.3.1**	**Use computing tools and techniques for creative expression.**
Abstraction	**2.1.1**	**Describe the variety of abstractions used to represent data.**
Abstraction	**2.2.1**	**Develop an abstraction when writing a program or creating other computational artifacts.**
Abstraction	**2.2.2**	**Use multiple levels of abstraction to write programs.**
Abstraction	**2.2.3**	**Identify multiple levels of abstractions used when writing programs.**
Abstraction	**2.3.1**	**Use models and simulations to represent phenomena.**
Abstraction	**2.3.2**	**Use models and simulations to formulate, refine and test hypothesis.**
Data	**3.1.2**	**Collaborate when processing information to gain insight and knowledge.**
Data	**3.1.3**	**Explain the insight and knowledge gained from digitally processed data by using appropriate visualizations, notation and precise language.**
Data	**3.2.1**	**Extract information from data to discover and explain connections, patterns or trends.**
Data	**3.2.2**	**Use large data sets to explore and discover information and knowledge.**
Algorithms	**4.1.1**	**Develop an algorithm for implementation in a program.**
Algorithms	**4.1.2**	**Express an algorithm in a language.**

Algorithms	4.2.1	Explain the difference between algorithms that run in a reasonable time and those that do not run in a reasonable time.
Algorithms	4.2.2	Explain the difference between solvable and unsolvable problems in computer science.
Algorithms	4.2.3	Explain the existence of undecidable problems in computer science.
Algorithms	4.2.4	Evaluate algorithms analytically and empirically for efficiency, correctness and clarity.
Programming	5.1.1	Develop a program for creative expression to satisfy personal curiosity or to create new knowledge.
Programming	5.1.2	Develop a correct program to solve problems.
Programming	5.1.3	Collaborate to develop a program.
Programming	5.2.1	Explain how programs implement algorithms.
Programming	5.3.1	Use abstraction to manage complexity in programs.
Programming	5.4.1	Evaluate the correctness of a program.
Programming	5.5.1	Employ appropriate mathematical and logical concepts in programming.
Internet	6.1.1	Explain the abstractions in the Internet and how it functions.
Impact	7.1.2	Explain how people participate in a problem solving process that scales.

Computational Thinking Practices used in Unit 5: P1, P2, P3, P4, P5 and P6

Sample Assignment 5

Read The Rise of Big Data in chunks: An Introduction to “Big Data” (20 mins) Reading can be found at: http://www.foreignaffairs.com/articles/139104/kenneth-neil-cukier-and-viktor-mayer-schoenberger/the-rise-of-big-data

Break students into groups or pairs and jigsaw the seven units of the reading. Each group is to summarize their section in a tweet sized comment (not more than 140 characters). Share tweets with the class.

Explain to students that big data is impacting every area of life. By using more data and processing power we can make better decisions. As an illustration, show a clip from the movie Moneyball: (3 mins) https://www.youtube.com/watch?v=rMObWsKaIls

After students watch, they create a journal entries explaining at least two ways data was used to better manage the baseball team. Partners discuss journal entries.. Share at least one observation with table groups and then share at least one observation from each group with the class.

Associated Computational Practices: P1, P4 and P5

Associated Big Ideas: BI1, BI3

Learning Objectives and Associated Computational Practices:

LO 1.2.5 [P4]

LO 3.1.1 [P4]

LO 3.1.3 [P5]

LO 3.2.1 [P1]

6. Data Visualization

The main idea of Unit 6 is to explore ways to create and understand data through data visualization. Students will also prepare for the final performance task by doing a group project.

Unit 6 Lessons:

At least one of these three lessons will be completed. Each lesson is scheduled for multiple days.

6 - 1. EarSketch
6 - 2. Data Visualization with Python and Bokeh
6 - 3. Dataquest

Unit 6 Learning Objectives:

Creativity	**1.1.1**	**Apply a creative development process when creating computational artifacts.**
Creativity	**1.2.1**	**Create a computational artifact for creative expression.**
Creativity	**1.2.2**	**Create a computational artifact using computing tools and techniques to solve a problem.**
Creativity	**1.2.3**	**Create a new computational artifact by combining or modifying existing artifacts.**
Creativity	**1.2.4**	**Collaborate in the creation of computational artifacts.**
Creativity	**1.2.5**	**Analyze the correctness, usability, functionality, and suitability of computational artifacts.**
Creativity	**1.3.1**	**Use computing tools and techniques for creative expression.**
Abstraction	**2.2.1**	**Develop an abstraction when writing a program or creating other computational artifacts.**
Abstraction	**2.2.2**	**Use multiple levels of abstraction to write programs.**
Abstraction	**2.2.3**	**Identify multiple levels of abstractions used when writing programs.**
Data	**3.1.2**	**Collaborate when processing information to gain insight and knowledge.**
Data	**3.1.3**	**Explain the insight and knowledge gained from digitally processed data by using appropriate visualizations, notation and precise language.**
Data	**3.2.1**	**Extract information from data to discover and explain connections, patterns or trends.**
Algorithms	**4.1.1**	**Develop an algorithm for implementation in a program.**
Algorithms	**4.1.2**	**Express an algorithm in a language.**
Algorithms	**4.2.4**	**Evaluate algorithms analytically and empirically for efficiency, correctness and clarity.**
Programming	**5.1.1**	**Develop a program for creative expression to satisfy personal curiosity or to create new knowledge.**
Programming	**5.1.2**	**Develop a correct program to solve problems.**
Programming	**5.1.3**	**Collaborate to develop a program.**
Programming	**5.2.1**	**Explain how programs implement algorithms.**
Programming	**5.3.1**	**Use abstraction to manage complexity in programs.**

Programming	**5.4.1**	**Evaluate the correctness of a program.**
Programming	**5.5.1**	**Employ appropriate mathematical and logical concepts in programming.**
Internet	**6.1.1**	**Explain the abstractions in the Internet and how it functions.**

Computational Thinking Practices used in Unit 6: P1, P2, P3, P4, P5 and P6

Sample Assignment 6

Create a new EarSketch project. With your partner, develop an algorithm to meet the specifications below and to create music you like.

Length: 20 measures or longer Tempo: any
Structure

Your project should have an Intro, A and B sections, and an outro.
Use at least 3 audio clips in each section.
Have a repeated pattern such as ABAB.
Have an “Intro” section that is a unique beginning.
Have an “Outro” section that is unique to end the composition.
Use multiple effects including 4-parameter and the 7-parameter setEffect()

Individually develop separate programs to implement the algorithm.
Required Documentation:

Include your name, and a project description in the comments at the top of the file. Declare and use your own variables that uses audio clip constants as data abstractions for clips from the EarSketch Audio Loop browser.

Partners compare the two implementations and identify at least one important difference. Each individual should explain why they implemented the program as they did and compare their approach with their partner's.

Associated Computational Practices: P2, P3, P5 and P6

Associated Big Ideas: BI1, BI2, BI4, BI5

Learning Objectives and Associated Computational Practices:

LO 1.2.3 [P2]

LO 1.2.4 [P6]

LO 2.2.1 [P2]

LO 2.2.2 [P3]

LO 4.1.1 [P2]

LO 4.1.2 [P5]

LO 5.1.1 [P2]

LO 5.1.3 [P6]

Performance Task: **Create - Applications from Ideas.**

Students will have 12 hours (720 minutes) of class time to complete this performance task.