# Unit 2: Game Development 1 Copied from: Intro to Computer Science through Game Development & Design, Copied on: 02/21/22

Content Area: **21st Century Life and Careers**
Course(s): **Sample Course**
Time Period: **SeptOct**
Length: **10 days**
Status: **Published**

**Game Development 1**

## Department of Curriculum and Instruction

**Belleville Public Schools**

**Curriculum Guide**

# Introduction to Game Design and Development thru Computer Science

# Unit 2: Game Development

**Belleville Board of Education**

**102 Passaic Avenue**

**Belleville, NJ 07109**

**Prepared by:** Teacher, Corey Woodring

Dr. Richard Tomko, Ph.D., M.J., Superintendent of Schools

Ms. LucyAnn Demikoff, Director of Curriculum and Instruction K-12

Ms. Nicole Shanklin, Director of Elementary Education K-8, ESL Coordinator K-12

Mr. George Droste, Director of Secondary Education

Board Approved: September 23, 2019

## Unit Overview

This course is designed to introduce students to game design and development through classroom discussions and hands-on programming. Students will learn the specific commands needed to make a game while navigating through the game development levels. This will culminate in a final project in which each student designs and creates their own unique game and collects peer feedback for improvement.

*This guide is written with Python-language classrooms in mind, but can easily be adapted for JavaScript.*

## Enduring Understanding

- **Properties** are characteristics of objects. This could be any data related to an object, like the max health of the hero or the direction that the fire travels from the fire-spewer.

Have students identify the **properties** of different **objects** in the classroom. For example, have students list the **properties** of the desks or tables where they're sitting. Note that the **properties**they are identifying, like the size, shape, and color of the **object** are **values** that correspond to **keys**. Identify that each property will have this kind of **key**, **value** pair, with the **property key**being **size** and the **value** being **3 feet**.

## Essential Questions

**Why is it helpful to be able to change the properties of different objects?**

Sample Response:

> When developing my game, I can control the way characters look and how they are able to interact with each other.

**How is changing an object's properties different from using a method like `hero.moveRight()`?**

Sample Response:

> Changing an object's properties changes the data that is stored as part of an object. Using a method allows the object to do something, but doesn't necessarily change the data associated with the object.

## Exit Skills

What are the steps students should work through to create their games? How is is this an iterive process?

Explain the following steps that students should work through to create their games:

1. **Design.** This can be done on paper. Describe the idea, sketch the map, list the goals.
2. **Plan.** Take the design and write a flowchart or plain-English plan for how students will code the project in the game engine.
3. **Build.** Take the initial design, and build it in the game engine (in this case, use the level Tabula Rasa to build the game).
4. **Test.** As the students build their game, they should always be testing it by playing the game.
5. **Feedback.** The students should share their game link with friends and gather feedback about what makes the game fun or frustrating.
6. **Improvements.** Based on the feedback, the students go back to the building phase and make improvements to the game.

## New Jersey Student Learning Standards (NJSLS-S)

- **CSTA 1A-AP-15** Using correct terminology, describe steps taken and choices made during the iterative process of program development.
- **CSTA 3A-AP-23** Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs.

| | |
|---|---|
| TECH.8.2.12.E | Computational Thinking: Programming: Computational thinking builds and enhances problem solving, allowing students to move beyond using knowledge to creating knowledge. |
| TECH.8.2.12.E.1 | Demonstrate an understanding of the problem-solving capacity of computers in our world. |
| TECH.8.2.12.E.2 | Analyze the relationships between internal and external computer components. |
| TECH.8.2.12.E.3 | Use a programming language to solve problems or accomplish a task (e.g., robotic functions, website designs, applications, and games). |
| TECH.8.2.12.E.4 | Use appropriate terms in conversation (e.g., troubleshooting, peripherals, diagnostic software, GUI, abstraction, variables, data types and conditional statements). |
| TECH.8.2.12.E.CS1 | Computational thinking and computer programming as tools used in design and engineering. |

## Interdisciplinary Connections

| | |
|---|---|
| LA.RH.6-8.5 | Describe how a text presents information (e.g., sequentially, comparatively, causally). |
| LA.RH.6-8.7 | Integrate visual information (e.g., in charts, graphs, photographs, videos, or maps) with other information in print and digital texts. |
| SCI.HS-ETS1-2 | Design a solution to a complex real-world problem by breaking it down into smaller, more manageable problems that can be solved through engineering. |
| SOC.6.3.12.C | Economics, Innovation, and Technology |
| | A model can be very simple, such as writing total cost as a product of unit price and number bought, or using a geometric shape to describe a physical object like a coin. Even such simple models involve making choices. It is up to us whether to model a coin as a three-dimensional cylinder, or whether a two-dimensional disk works well enough for our purposes. Other situations—modeling a delivery route, a production schedule, or a comparison of loan amortizations—need more elaborate models that use other tools from the mathematical sciences. Real-world situations are not organized and labeled for analysis; formulating tractable models, representing such models, and analyzing them is |

appropriately a creative process. Like every such process, this depends on acquired expertise as well as creativity.

## Learning Objectives

- Apply the tools and concepts introduced throughout the Game Development 1 Course to a creative game.
- Explain in words the design decisions that led them to their specific program and game.

## Suggested Activities & Best Practices

**Guidelines for Suggested Activities:**

- Includes activities **appropriate & specific** to the development of the Unit;
- Is comprised of the variety of learning activities that will be referenced in lesson plans, constructed/developed and instructionally delivered in the classroom;
- Are authentic;
- Recognizes the learning styles of the students;
- Integrates problem- or project-based learning.

## Assessment Evidence - Checking for Understanding (CFU)

Unit test-summative assessment

Evaluation rubric-formative assessment

Multimedia reports-alternate assessment

Create a Multimedia poster-benchmark assessment

- Admit Tickets
- Anticipation Guide
- Common Benchmarks
- Compare & Contrast
- Create a Multimedia Poster
- DBQ's
- Define
- Describe
- Evaluate
- Evaluation rubrics
- Exit Tickets
- Explaining
- Fist- to-Five or Thumb-Ometer
- Illustration
- Journals
- KWL Chart
- Learning Center Activities
- Multimedia Reports
- Newspaper Headline
- Outline
- Question Stems
- Quickwrite
- Quizzes
- Red Light, Green Light
- Self- assessments
- Socratic Seminar
- Study Guide
- Surveys
- Teacher Observation Checklist
- Think, Pair, Share
- Think, Write, Pair, Share
- Top 10 List
- Unit review/Test prep
- Unit tests
- Web-Based Assessments
- Written Reports

## Primary Resources & Materials

CodeCombat, CodeMonkey, Game Salad, Scratch, Alice, Grasshopper

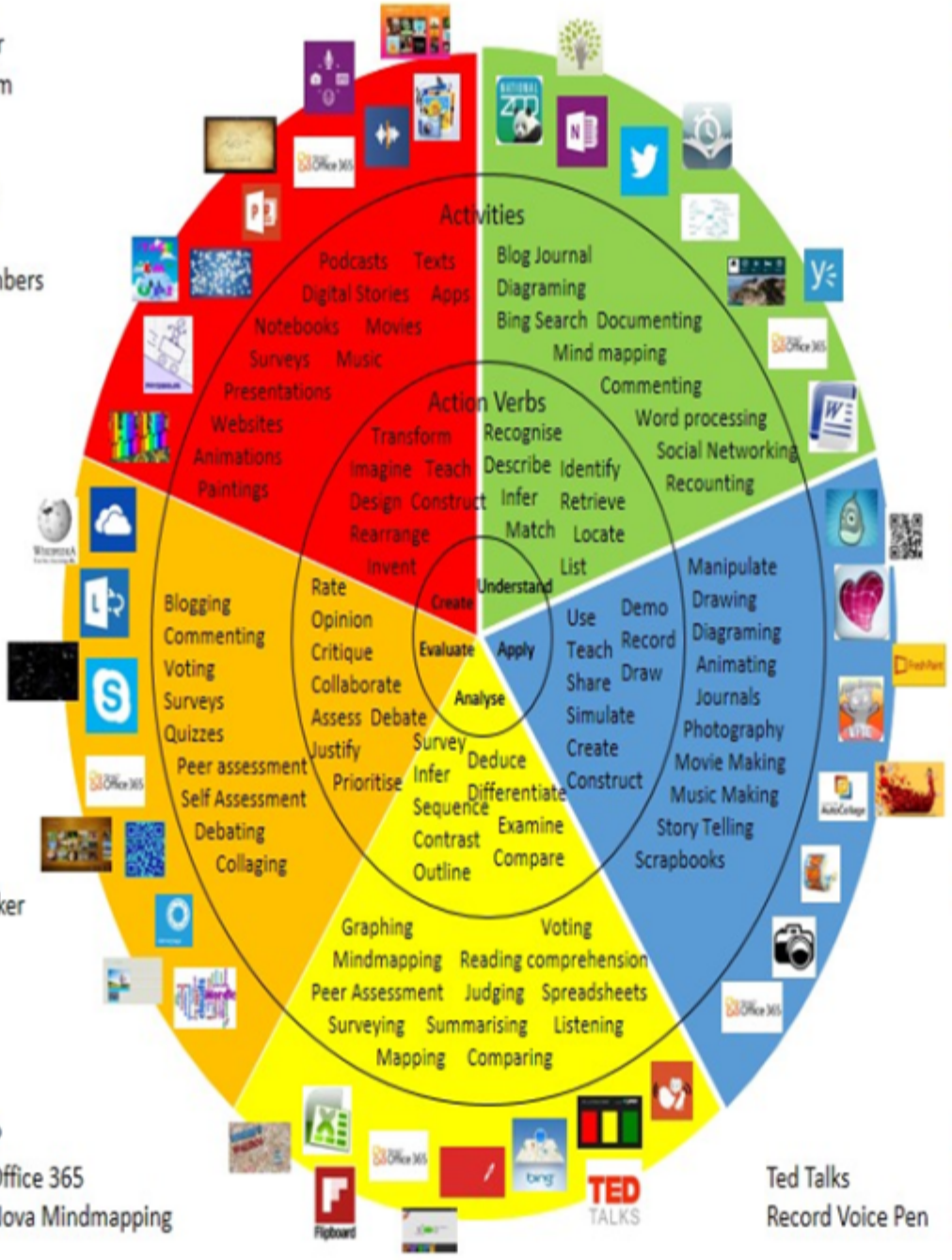
## Ancillary Resources

## Technology Infusion

# Win 8.1 Apps/Tools Pedagogy Wheel

Podcasts
Photostory 3
Kid Story Builder
Music Maker Jam
Paint A Story
Office 365
MS PowerPoint
Stack 'Em Up
NqSquared Numbers
Physamajig
Xylophone 8

Wikipedia
Skydrive
Lync
SkyMap
Skype
Office 365
Puzzle Touch
Easy QR
Memorylage
Life Moments
Word Cloud Maker




Where's Waldo?
MS Excel Office 365
Flipboard Nova Mindmapping

Ted Talks
Record Voice Pen

Originally taken from http://www.coetail.com/vzimmer/files/2013/02/iPadagogy-Wheel.001.jpg
And adapted for Windows 8.1 devices by Charlotte Beckhurst @CharBeckhurst

## Alignment to 21st Century Skills & Technology

Mastery and infusion of **21st Century Skills & Technology** and their Alignment to the core content areas is essential to student learning. The core content areas include:

- English Language Arts;
- Mathematics;
- Science and Scientific Inquiry (Next Generation);
- Social Studies, including American History, World History, Geography, Government and Civics, and Economics;
- World languages;
- Technology;
- Visual and Performing Arts.

## 21st Century Skills/Interdisciplinary Themes

- Communication and Collaboration
- Creativity and Innovation
- Critical thinking and Problem Solving
- ICT (Information, Communications and Technology) Literacy
- Information Literacy
- Life and Career Skills
- Media Literacy

## 21st Century Skills

- Civic Literacy

- Environmental Literacy
- Financial, Economic, Business and Entrepreneurial Literacy
- Global Awareness
- Health Literacy

## Differentiation

**Differentiations:**

- Small group instruction
- Small group assignments
- Extra time to complete assignments
- Pairing oral instruction with visuals
- Repeat directions
- Use manipulatives
- Center-based instruction
- Token economy
- Study guides
- Teacher reads assessments allowed
- Scheduled breaks
- Rephrase written directions
- Multisensory approaches
- Additional time
- Preview vocabulary
- Preview content & concepts
- Story guides
- Behavior management plan
- Highlight text
- Student(s) work with assigned partner
- Visual presentation
- Assistive technology
- Auditory presentations
- Large print edition
- Dictation to scribe
- Small group setting

**Hi-Prep Differentiations:**

- Alternative formative and summative assessments
- Choice boards
- Games and tournaments
- Group investigations
- Guided Reading
- Independent research and projects
- Interest groups
- Learning contracts
- Leveled rubrics
- Literature circles
- Multiple intelligence options
- Multiple texts
- Personal agendas

- Project-based learning
- Problem-based learning
- Stations/centers
- Think-Tac-Toes
- Tiered activities/assignments
- Tiered products
- Varying organizers for instructions

**Lo-Prep Differentiations**

- Choice of books or activities
- Cubing activities
- Exploration by interest
- Flexible grouping
- Goal setting with students
- Jigsaw
- Mini workshops to re-teach or extend skills
- Open-ended activities
- Think-Pair-Share
- Reading buddies
- Varied journal prompts
- Varied supplemental materials

## Special Education Learning (IEP's & 504's)

- printed copy of board work/notes provided
- additional time for skill mastery
- assistive technology
- behavior management plan
- Center-Based Instruction
- check work frequently for understanding
- computer or electronic device utilizes
- extended time on tests/ quizzes
- have student repeat directions to check for understanding
- highlighted text visual presentation
- modified assignment format
- modified test content

- modified test format
- modified test length
- multi-sensory presentation
- multiple test sessions
- preferential seating
- preview of content, concepts, and vocabulary
- Provide modifications as dictated in the student's IEP/504 plan
- reduced/shortened reading assignments
- Reduced/shortened written assignments
- secure attention before giving instruction/directions
- shortened assignments
- student working with an assigned partner
- teacher initiated weekly assignment sheet
- Use open book, study guides, test prototypes

## English Language Learning (ELL)

- teaching key aspects of a topic.  Eliminate nonessential information
- using videos, illustrations, pictures, and drawings to explain or clarif
- allowing products (projects, timelines, demonstrations, models, drawings, dioramas, poster boards, charts, graphs, slide shows, videos, etc.) to demonstrate student’s learning;
- allowing students to correct errors (looking for understanding)
- allowing the use of note cards or open-book during testing
- decreasing the amount of workpresented or required
- having peers take notes or providing a copy of the teacher’s notes
- modifying tests to reflect selected objectives
- providing study guides
- reducing or omitting lengthy outside reading assignments
- reducing the number of answer choices on a multiple choice test
- tutoring by peers
- using computer word processing spell check and grammar check features
- using true/false, matching, or fill in the blank tests in lieu of essay tests

## At Risk

- allowing students to correct errors (looking for understanding)
- teaching key aspects of a topic.  Eliminate nonessential information

- allowing products (projects, timelines, demonstrations, models, drawings, dioramas, poster boards, charts, graphs, slide shows, videos, etc.) to demonstrate student's learning
- allowing students to select from given choices
- allowing the use of note cards or open-book during testing
- collaborating (general education teacher and specialist) to modify vocabulary, omit or modify items to reflect objectives for the student, eliminate sections of the test, and determine how the grade will be determined prior to giving the test.
- decreasing the amount of workpresented or required
- having peers take notes or providing a copy of the teacher's notes
- marking students' correct and acceptable work, not the mistakes
- modifying tests to reflect selected objectives
- providing study guides
- reducing or omitting lengthy outside reading assignments
- reducing the number of answer choices on a multiple choice test
- tutoring by peers
- using authentic assessments with real-life problem-solving
- using true/false, matching, or fill in the blank tests in lieu of essay tests
- using videos, illustrations, pictures, and drawings to explain or clarify

## Talented and Gifted Learning (T&G)

- Above grade level placement option for qualified students
- Advanced problem-solving
- Allow students to work at a faster pace
- Cluster grouping
- Complete activities aligned with above grade level text using Benchmark results
- Create a blog or social media page about their unit
- Create a plan to solve an issue presented in the class or in a text
- Debate issues with research to support arguments
- Flexible skill grouping within a class or across grade level for rigor
- Higher order, critical & creative thinking skills, and discovery
- Multi-disciplinary unit and/or project
- Teacher-selected instructional strategies that are focused to provide challenge, engagement, and growth opportunities
- Utilize exploratory connections to higher-grade concepts
- Utilize project-based learning for greater depth of knowledge

## Sample Lesson

This lesson will focus on loops. There are different kinds of loops, like “for loops” and “until loops”, but first we will learn how to use a simple loop.

**Essential Questions**

1. What is a loop?

2. When do we use loops in our code?

3. What is a syntax of a loop that makes the monkey walk in the shape of a square?

4. How do we tell the difference between code that is inside the loop and code that is outside the loop?

**Objective**

In this lesson, students will: Define loops as a programming term Understand why using loops in programming is more efficient complete challenges 21-25

**Components**

step -x (stepping backwards) turn left/right times

Instructions: “x.times->” Terms: loop Feature: tab (indentation)

**Standards**

CCSS-Math Standards: PRACTICE.MP1 PRACTICE.MP3 PRACTICE.MP4 PRACTICE.MP5 CONTENT.3.MD.B.4 CONTENT.4.MD.C.5

CCSS-ELA Standards: Literacy.RST.6-8.3 Literacy.RST.6-8.4 Literacy.RST.9-10.3 Literacy.RST.9-10.4 Literacy.RST.9-10.7

**Enduring Understanding**

Programming is not only about writing the correct statements in the correct order; it is also about knowing how to write clear and short code. Imagine that we have to write a simple program to make the monkey climb up a high staircase of 100 steps and we can only use the function “stepUp” to make the monkey climb up one step at a time.

Ask your students: “Do you think that the programmer wrote a line of code for every stair-step?” Just imagine how LONG this code would be ! It would be 100 lines of code! So, instead of code that looks like this (X100): stepUp stepUp stepUp etc... Wouldn’t it be great to write something shorter? Ask your students to suggest a shorter way. How about something like this? stepUp 100 times Luckily, this is possible. Not exactly the way we just wrote it now, but quite similar. Code that is written in such a way is called a loop.

LESSON: Loops

Explain to your class that a “simple loop” is a sequence of instructions that repeats a specified number of times. There are also other kinds of loops (for loops, until loops) that last until a particular condition is met, but we will learn about those later on. Back to the staircase example…the way to write that in Coding Adventure would be: 100.times -> stepUp

The number 100 represents the number of times that we want the code inside the loop to run for.

Note the special syntax: the dot between the number and the word times, the space before the ->, and the indentation of the code inside the loop (stepUpis the code inside the loop). Make sure your students know how to use the “Tab” key on their keyboards in order to get indentation into the code. Another alternative is to press the spacebar four times.

Remember that you can click the “times” button at the bottom in order to get a loop into the code without having to worry about the syntax.

Let’s show another example to better clarify the use of a simple loop. Write the following code on the left-hand side of the board:

```
step 10

 turn left

 step 10

turn left

step 10

 turn left

 step 10

 turn left
```

Walk-through 3 mins. Open challenge 25 and click the reset button to reset the code. Go over the code with your students. Read the statements out loud, slowly and clearly. This walk-through is intended to show students how to read code correctly.

Walk your students through the process of identifying the pattern of bananas arranged in an Lshape and translate that into the sequence of statements: turn left step 5 turn right step 5 Then, fix the inside of the loop to match the sequence of statements and hit “run". This will not solve the challenge as the loop runs 3 times instead of 4. Ask your students how many times the L pattern repeats itself? The solution is the last hint needed in order to solve this challenge correctly, i.e. replacing the 3 by 4.

In the Loop Part 3: 5 Minutes Debriefing

Imagine you had to give instructions to somebody to find a place that is 5 blocks down the street. Do you say: walk a block, then walk another block, then another, then another, and then one more. No. You simply say:

walk 5 blocks down the street. That is because the same action has to be done more than once.

Remind your students that it is the same in coding. When there is a repeating pattern of things to do, then a loop is a good way to keep the program short and easy to understand. Just find the pattern, write it once and add the line of code that tells the computer how many times to repeat.