



#platformgames #coding8 #scratch

QUESTION

Depth of Knowledge Chart

How can I...

...CREATE my own original (or heavily inspired) platform game.

*Commenting your code is like cleaning your bathroom -
you never want to do it, but it really does create a more
pleasant experience for you and your guests.*

- Ryan Campbell

TASK

[S] Utilise **Griffpatch's Platformer Tutorial** to create a "platform engine" that will drive the game mechanics of a future platform game.

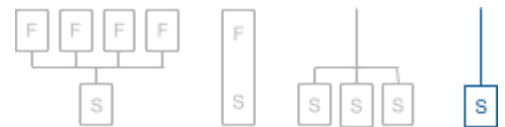
NOTES:

- ❑ It is recommended that you follow the tutorial carefully, ensure you read what is written on the screen and do NOT just blindly copy the code, much of the code added throughout the tutorial is appended/altering previous lessons.
- ❑ To fully complete the "platform engine" you will need to follow the whole tutorial. You may find it helpful to download the sprite at right.
- ❑ It is **EXPECTED** that you add comments to the code. Right-Click on the block and select "add comments".
- ❑ Comments should be meaningful in that they explain to a reader (and more importantly yourself) what each piece of the code does.
- ❑ Do **NOT** move on to making your own game until you are confident that you understand how the platform engine code works.
- ❑ **Can you alter the code to ensure it follows the class coding standards?** (inc: Essential Blocks, waits, naming conventions, etc...)

[S]

GAME DESIGN DOCUMENT**ASSESSMENT NOTES**

- ❑ The focus on this task is good programming style.
- ❑ Student's will be expected to meaningfully comment in their code.

**INVESTIGATE**

[READ]

[WATCH] **Top 10 Side-Scrolling Games**[WATCH] **Top 10 Best Ever Platforms**

[THEORY] Naming/Identifier conventions

[THEORY] Software development lifecycle

[THEORY] Machine instruction cycle

RESOURCES[CODE] **Griffpatch Platformer Tutorial**[CODE] **Griffpatch Scrolling and Sensing**[SPRITES] **Griffpatch's Cat Costumes (sprite)**

[TEMPLATE]

I'M STUCK

[HINT]

[CHECK] **Assessment Notes****VERSION 2.0**

[THINK]

[MODIFY]

[EXTEND]

ACADEMIC STANDARDS & BENCHMARKS

[assessment notes](#) | [learning standards](#)

- ☐ 1.1 Students create original works or responsibly repurpose or remix digital resources in new creations.
- ☐ 2.1 Students communicate information and maintain a digital portfolio of their work.
- ☐ 2.2 Students will provide adequate instructions within the scope of the user experience.
In addition Coding 8 students will be expected to include internal code documentation/comments to explain how the components of the algorithm function.
- ☐ 3.1 Students know and use a deliberate design process for generating ideas, testing concepts and creating programming solutions.
- ☐ 3.2 Students develop, test and refine code as part of a cyclical design process (software development lifecycle)
- ☐ 4.1 Students will use a development environment to select and combine control structures while writing modular code, using conventional casing and naming practices, to keep code consistent and readable.
- ☐ 4.2 Students will select and apply boolean expressions with the appropriate control structures to write logically correct programs.
- ☐ 4.3 Students can effectively declare and initialise the appropriate primitive data types and constants.
- ☐ 4.4 Students understand and use technology systems.



This work is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 3.0 Unported License](#).
Based on a work at [aselley.com](#)
One Page Learning [\[template here\]](#)