# Lab Airplanes [Due before Sun up Monday 25/1/21]

Create an application like the sample one, you may use your art and add extra functionality if you wish but the basics must be covered first. You also need to follow the coding order and make commits when required at least (you can have more commits)

Your app should display two aircraft that are controlled by the left and right mouse button. Click down for the start of the velocity vector and up for the end point of the velocity vector thus enabling the user to input both direction and magnitude. The smaller airplane should travel twice as fast for the same input.

As always you should store your own information about the plane and use that to drive the sprites on screen not the other way around (ie. Sf::vector2f planelocation)

Create a new method to use when deciding if the two aircraft are close enough to collide (circle based collision)

float vector2fLenght(sf::Vector2f t\_vector);

accept the following assignment and submit you research document today.

<https://classroom.github.com/a/QUVMMjvA>

Things you will need to research

Sf::Sprite [ setorigin, setposition, settexturerectangle, setRotation, settexture ]

SF::Texture [loadfromfile, setrepeated]

SF::mouse [left,right]

Sf::Event::mousebutton [x,y, button]

Sf::event [mousebuttonpressed, mousebuttonreleased]

Order in which to complete this task.

1. ~~Accept assignment~~
2. ~~Git init~~
3. ~~Add gitignore~~
4. ~~Create draft research~~
5. ~~Commit and push~~
6. ~~Create template sfml game~~
7. ~~Commit and push~~
8. ~~Refine research doc~~
9. ~~Commit and push “research doc done” [Today before midnight]~~
10. ~~Add first plane and draw stationary on screen~~
11. ~~Commit and push~~
12. Move first plane at constant velocity
13. Stop plane at screen edge
14. Commit and push “moving plane”
15. Accept left mouse press and release and set velocity
16. Commit and push “flying plane”
17. Add second plane
18. Accept right mouse press and release and set 2nd plane velocity
19. Commit and push “two flying planes”
20. Detect collision
21. Commit and push “collisions”
22. Animate explosion
23. Commit and push “finished minimum”
24. Add something extra for 10%
25. Commit and push



