

# SFML

## Simple and Fast Multimedia Library

[www.sfml-dev.org](http://www.sfml-dev.org)

Carl Søyseth  
[caso13@student.bth.se](mailto:caso13@student.bth.se)

# Dagens agenda

Vad är SFML?

Vad kan man uppnå mha SFML?

Hur funkar SFML övergripligt?

Hur funkar ett spel?

Ett simpelt SFML program

Hur du bör gå tillväga med projektet

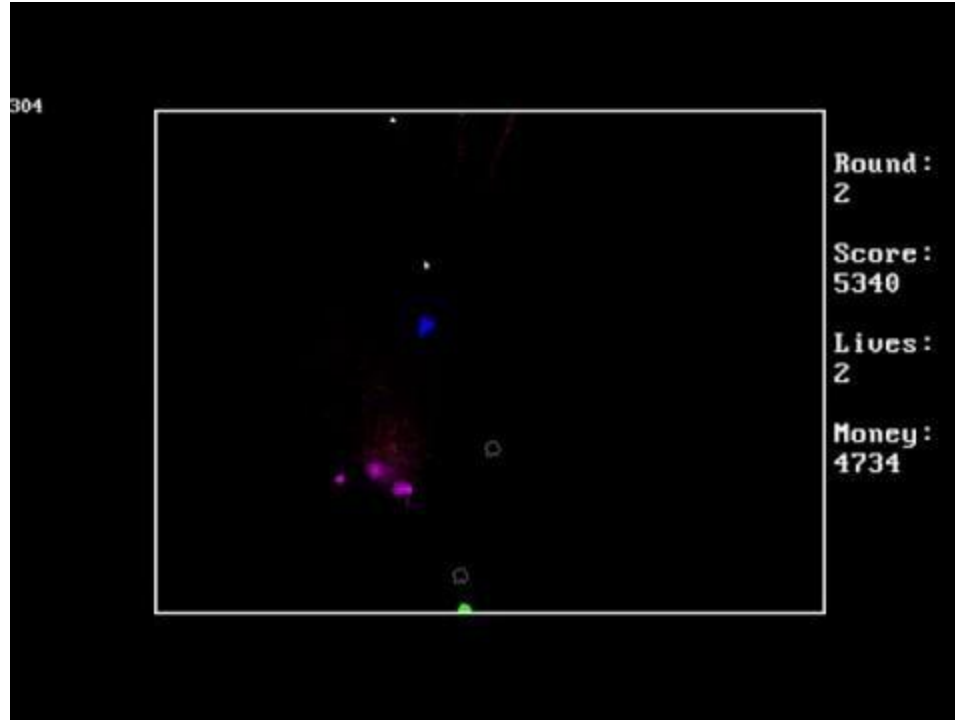
# SFML

- Ett bibliotek för att hantera simpel 2D rendering
- Använder OpenGL -> Funkar på Windows, Linux och MacOS
- Objektorienterat och skrivet i C++
- Modulär design

# SFML - moduler

System	Behövs alltid, samtliga av de övriga modulerna använder funktionalitet från system.
Window	Fönsterhantering och input. Hanterar skapandet och utritningen av ett fönster via OpenGL samt hanterar event för fönstret.
Graphics	2D-grafik, sprites, texturer, former, dvs allt som berör rendering.
Audio	Ljudhantering, används för att spela upp, spela in och manipulera ljud.
Network	Nätverk: TCP och UDP sockets, webbförfrågningar (HTML), filöverföring (FTP)

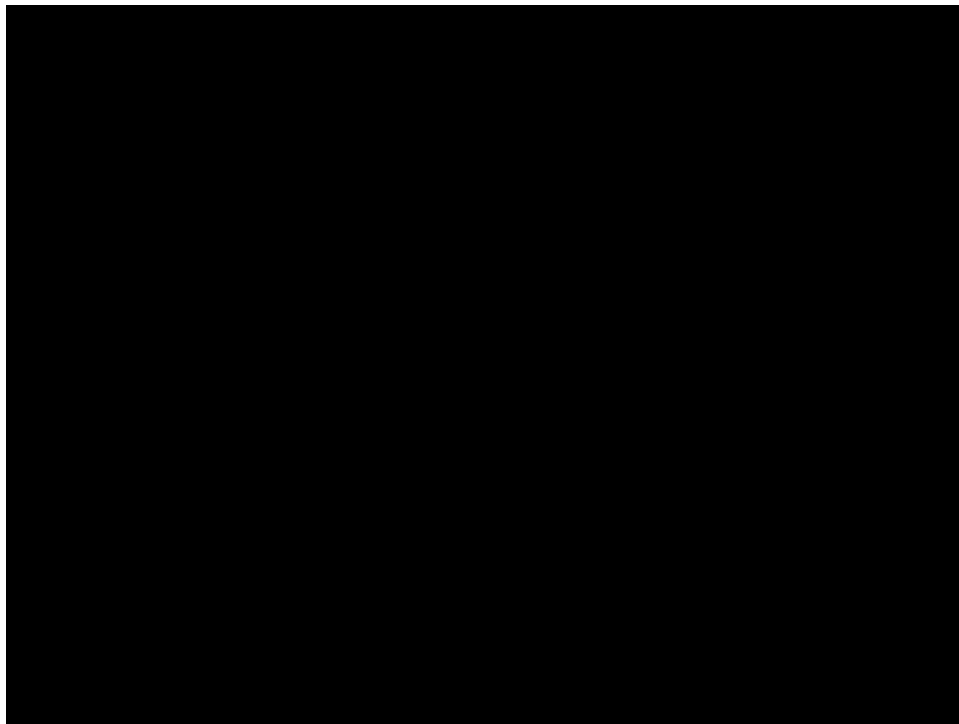
# Inspirerande exempel av **Fritjof Cavallin**



# Inspirerande exempel av **Fritjof Cavallin**



# Inspirerande exempel av **Fritjof Cavallin**



# Inspirerande exempel av Fritjof Cavallin





# Hur funkar spel?

- Tar någon form av input, utför beräkningar beroende på inputen och uppdaterar världen / spelet och illustrerar sedan ändringarna.
- Kallas för “spel-loopen”.
  - Kollar efter input
  - Uppdaterar världen
  - Ritar / renderar ändringar
  - Varje varv / iteration motsvarar en bild, även känt som FPS (frames per second).

# Hur funkar spel?

- Update()
  - Läsa in och tolka input
  - Applicera spellogiken på input
- Draw()
  - Den grafiska tolkningen av spellogiken
    - 1. Rensa fönstret (sätta alla pixlar till en specifik färg)
    - 2. Rita ut alla objekt
    - 3. Presentera resultatet

Rinse repeat

# Hur funkar spel?

- Förändringar måste vara relativt föregående bild / iteration
- Annars blir ändringar inkonsekventa och beror på bara uppdateringsfrekvensen (fps)
- `sf::Clock.restart()`
- Skicka float `dt` där det är relevant (när tex. Position ska uppdateras)

`Position.x = position.x * dt * speed.x;`

# Hur funkar spel - Ett applicerat exempel

- Snake
  - Vad händer under en frame?
    - Är Vänster / Höger / Bakåt / Framåt knapp nedtryckt?
    - Flytta alla delar i resp. Riktningt
    - Har huvudet kolliderat med någon bit?
    - Ska mat placeras?
    - Rita ut ormen och maten

# Former

- SFML har två fördefinierade former
  - Rektanglar
    - `sf::RectangleShape rect(sf::vector2(90., 60))`
    - Skapar en rektangel med storleken 90x60
  - Cirklar
    - `sf::CircleShape circ(100)` skapar en cirkel med radie 100
    - `sf::CircleShape circ(100, 3)` skapar en cirkel med radie med 100 Och 3 sidor (dvs triangel)

# Första programmet

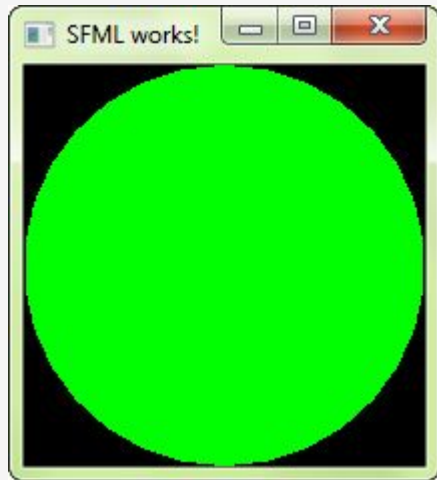
```
#include <SFML/Graphics.hpp>

int main()
{
    sf::RenderWindow window(sf::VideoMode(200, 200), "SFML works!");
    sf::CircleShape shape(100.f);
    shape.setFillColor(sf::Color::Green);

    while (window.isOpen())
    {
        sf::Event event;
        while (window.pollEvent(event))
        {
            if (event.type == sf::Event::Closed)
                window.close();
        }

        window.clear();
        window.draw(shape);
        window.display();
    }

    return 0;
}
```



<https://www.sfml-dev.org/tutorials/2.4/images/start-vc-app.png>

<https://www.sfml-dev.org/tutorials/2.4/start-vc.php>

# Det första programmet, forts

```
sf::RenderWindow window(sf::VideoMode(200, 200), "SFML works!");  
sf::CircleShape shape(100.f);  
shape.setFillColor(sf::Color::Green);
```

- `sf::RenderWindow`, tar två argument
  - `Sf::vector2f` - storlek på fönster,
  - `Std::string` - titel på fönstret
- `sf::CircleShape`
  - Tar en eller två argument - float, int
    - Flyttalet är radien
    - Heltalet är antal hörn
- `Shape.setFillColor`
  - `Sf::Color`

# Planera ditt projekt

- 2 hp, motsvarar ungefär 50 timmar
- Börja i god tid, helst idag.
- Lägg ribban rimligt högt, sätt realistiska mål.
- Gör något som intresserar dig.
- Planera väl, och ta reda på vad som kommer att krävas av dig.
- Maila vid bekymmer alt. besök programmeringsstugan, jag är där på tisdagar.
- Mycket eget ansvar, och det förväntas att ni själva läser på om hur SFML funkar
- Fördjupande föreläsning följer.



# Exempel på spel

- Ex. på rimliga projekt
  - Space invaders
  - Snake
  - Tetris
  - Simpel “Mario” klon
  - Pacman (lite utmanande)

# Intressanta länkar

<https://www.sfml-dev.org/tutorials/2.4/>

<https://www.sfml-dev.org/documentation/2.4.2/>

[https://www.sfml-dev.org/documentation/2.3/classsf\\_1\\_1Keyboard.php](https://www.sfml-dev.org/documentation/2.3/classsf_1_1Keyboard.php)

[https://www.sfml-dev.org/documentation/2.3/classsf\\_1\\_1Mouse.php](https://www.sfml-dev.org/documentation/2.3/classsf_1_1Mouse.php)

<https://github.com/>

<https://guides.github.com/activities/hello-world/>