sf::VideoMode Class

- VideoMode defines video mode (width, height, bits per pixel).
- A video mode is defines by a width and a height (in pixels) and a depth(in bits per pixel).
- video mode are used to setup windows (sf::Window or sf::RenderWindow) at creation time.

The main usage of video mode is for fullscreen mode: you must use one of the video modes thar are allowed by the OS (which are defined by what the monitor and graphics card support), otherwise your window creation will just fail.

```
sf::VideoMode::VideoMode (unsigned int modeWidth, unsigned int modeHeight, unsigned int modeBitsPerPixel = 32)
```

sf::VideoMode class some utility functions

isValid( )

Tell whether or not the video mode is valid. The validity of video modes is only relevant when using fullscreen windows; otherwise any video mode can be used with no restriction.

Returns- True if the video mode is valid for fullscreen mode, else false.

getFullscreenModes( )

sf::VideoMode provides a static function for retrieving the list of all the video modes supports by the system.

When creating a fullscreen window, the video mode is restricted to be compatible with what the graphics driver and monitor support. This function returns the complete list of all video modes that can be used in fullscreen mode. The returned array is sorted from best to worst, so that the first element will always give the best mode (higher width, height and bits-per-pixel).

```
int main(int argCount, char** argVector) {

    std::vector<sf::VideoMode> videoModes;
    videoModes = sf::VideoMode::getFullscreenModes();
    sf::VideoMode videoMode;

for(unsigned i = 0; i < videoModes.size(); i++) {

    if(videoModes[i].isValid()) // checking for valid modes
        std::cout << "Vanlid VideoMode: " << i << " - "
                                          << videoModes[i].width
                                          << "x" << videoModes[i].height
                                          << " b" << videoModes[i].bitsPerPixel
                                          << std::endl;
    else
        std::cout << "Invalid VideoMode: " << i << " - "
                                          << videoModes[i].width
                                          << "x" << videoModes[i].height
                                          << " b" << videoModes[i].bitsPerPixel
                                          << std::endl;

}

}
```

getDesktopMode( )

Returns- Current desktop video mode.

```
// Display the list of all the video modes available for fullscreen
std::vector<sf::VideoMode> modes = sf::VideoMode::getFullscreenModes();
for (std::size_t i = 0; i < modes.size(); ++i)
{
    sf::VideoMode mode = modes[i];
    std::cout << "Mode #" << i << ": "
              << mode.width << "x" << mode.height << " - "
              << mode.bitsPerPixel << " bpp" << std::endl;
}
// Create a window with the same pixel depth as the desktop
sf::VideoMode desktop = sf::VideoMode::getDesktopMode();
window.create(sf::VideoMode(1024, 768, desktop.bitsPerPixel), "SFML window");
```

To know more about sf::VideoMode class then check out the below linl :)

sf::VideoMode Class Reference (SFML / Learn / 2.5.1 Documentation)

VideoMode defines a video mode (width, height, bpp) A video mode is defined by a width and a height (in pixels) and a depth (in bits per pixel). Video modes are used to setup windows (sf::Window) at creation time.

https://www.sfml-dev.org/documentation/2.5.1/classsf_1_1VideoMode.php#ad5e04c044b0925523c75ecb173d2129a