|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | openFrameworks 0.9 Cheat Sheet | | | | | Version 0.1  bart@hangaar.net |
|  | |  | |  | | |
| **Default Coordinate System**  The coordinate system used is (0,0) for top left, x increasing to right and y increasing downwards. The bottom right point can be found using:    ofGetWindowWidth(); // get window width  ofGetWindowHeight(); // get window height | | | **Variable Types**   |  |  | | --- | --- | | int | 1, 2, 10, 100, -5, ... | | float | 0.5, 1.976, 3.14, -3.1, ... | | bool | true or false | | string | “hello world”, ofToString(variable); | | ofPoint | 2D or 3D point (x,y,z) | | ofPolyline | Multiple line segment class | |  |  | | | **Keyboard & Mouse input**  ofGetMouseX(); //get x-pos of mouse  ofGetMouseY(); //get y-pos of mouse  if (ofGetMousePressed()) {…}  //do when mouse pressed;  if (ofGetKeyPressed('c')) {…}  //do when 'c' pressed; | |
| **Where to place your code**  void ofApp::setup() { // DO ONLY ONCE; }  void ofApp::update() { // NON-DRAWING CODE; }  void ofApp::draw() { // DRAWING CODE; } | | | **Framerate**  ofSetFrameRate(60);//limits framerate to 60 FPS  ofSetVerticalSync(true); //sync with screen  ofSetWindowTitle(ofToString(ofGetFrameRate())); | | **ofImage**  ofImage image = ofImage("image.jpg"); //load  image.draw(x, y); //draw image at x,y  image.draw(x, y, width, height); //scale | |
| **Basic Shapes**  ofDrawLine(x1, y1, x2, y2);  ofDrawCircle(x, y, radius);  ofDrawRectangle(x, y, width, height);  ofDrawTriangle(x1, y1, x2, y2, x3, y3);  ofDrawEllipse(x, y, width, height); | | | **Text**  ofSetWindowTitle("Your app name");  ofDrawBitmapString("yo!", x, y);  ofDrawBitmapStringHighlight("yo!", x, y);  ofToString(variable); // converts to string  ofLog() << "text"; // prints to console; | | **Loops**  Repeat the same code multiple times by incrementing a variable  for (int i = 0; i < 100; i += 1) {  //code to execute 100 times  ofDrawCircle(x,y, 100 - i);  } | |
| **Colors in RGBA space (red-green-blue-alpha)**  A channel is a number between 0 (off) and 255 (on)  Alpha means transparency, 255 being opaque (default)  ofColor color = ofColor(r, g, b);  ofSetColor(color);  ofSetColor(r, g, b);  ofSetColor(ofColor::black); | | | **Randomness & Noise**  ofRandom(-10, 10); //random nr between 0 & 10  ofRandomWidth(); //random nr in screen width  ofRandomHeight(); //random nr in screen height  ofNoise(float x); //1D noise between 0 & 1  ofNoise(float x, float y); //2D noise [0 & 1]  ofNoise(ofGetElapsedTimef());//noise on time  ofSignedNoise(float x, ...); //noise [-1 & 1] | | **Vectors: collection of objects**  vector<float> nums; //create vector of floats  nums.push\_back(10.5); //add a float to vector  float value = nums[0]; //element at position 0  nums.erase(nums.begin()); //erase first element  int size = nums.size(); //number of elements  nums.clear(); //clear all elements | |
| **Background**  Set background color in setup():  ofSetBackgroundColor(ofColor(r, g, b));  Disable background in setup():  ofSetBackgroundAuto(false); | | | **Conditional structures**  Do something only in certain conditions:  if (value > limit) {  // code if value is higher then limit  } else {  // code if the above was false  } | | **Increasing quality in setup()**  Tricks to increase visual quality at the cost of cpu usage  ofEnableAlphaBlending(); //enable transparancy  ofEnableAntiAliasing(); //enable anti-aliasing  ofEnableSmoothing(); //enable line smooth  ofSetCircleResolution(100);//make nicer circles | |
| **Declaring Variables**  int number; //declares variable "number"  number = 1; //sets value 1 to number  number += 6; //adds 6 to number (matching 7) | | | **Setting sketch resolution**  In “main.cpp” source file:  ofSetupOpenGL(1024, 768, OF\_WINDOW);  ofSetupOpenGL(1920, 1080, OF\_FULLSCREEN); | | **Saving images**  ofImage image;  image.grabScreen(0, 0, width, height);  image.saveImage("file.jpg"); //save to disk | |

Complete tutorials at <https://github.com/Kj1/OF_COURCE_ICON> and <http://openframeworks.cc/ofBook/chapters/foreword.html>