|  |  |
| --- | --- |
|  |  |
| **Unit:** Basics | **Turn In List:** **1. Terms (this file), 2. ColDet.pde** |
| *“I will understand how to open, configure, write simple code and compile in the Processing IDE.”* | |

**Collision Detection Program: Getting to Know the Coding Environment**

**Content Objectives:** Students will create a working application and eventually publish it to run on Windows, OSX, Linux or within a webpage.

|  |
| --- |
| **Add to Shapes Sketch** |
| Draw vertical and horizontal reference lines every 100 pixels in the shapes sketch. Add text to mark each of the lines. |

|  |  |
| --- | --- |
| **Key Terms:** | |
| IDE | Integrated development environment – app to create other apps |
| Comment | Is a line of code ignored by the IDE - // |
| Compile | Changing source code to machine code |
| Debug | Stopping the compiler to inspect variables |

|  |  |
| --- | --- |
| **Application Terms:** | |
| Sketch | Any new project |
| Console | Dark area at the bottom that shows errors |
| pde | File extension for processing sketches |
| Development Modes | Changes behaviour of IDE for other languages. |

|  |
| --- |
| **History and Background:** |
| The Processing initiative began in 2001-02 at MIT by Ben Fry and Casey Reas. It has grown into a large community of open source developers that write apps to run on Windows, OSX, Linux, the Web, Android and soon to be iOS as well as other adaptations.  Casey Reas is a renowned designer/artist, lecturer, thinker, and of course one half of the Processing design/development team. He was a student of John Maeda's at the MIT media lab, where he and Ben Fry helped develop the DBN programming language/environment. Processing was, to a degree, a natural outgrowth of DBN. Reas originally studied design at the University of Cincinnati before attending MIT, and was one of the founding professors at Interaction Design Institute Ivrea. At Ivrea, Reas worked with an international student body to develop a new arts pedagogy. It was during this period in Ivrea that he and Fry initiated Processing. |

|  |
| --- |
| **Assignment Tutorial (Setup and Class Demonstration):** |
| Students will locate and open the Processing 2.0 IDE and set the mode to experimental. Save the new sketch as ColDet - Run the application with nothing typed in the text window… what do you see? |

|  |
| --- |
| **Assignment:** |
| // Code the following in a new Processing sketch and do not worry about NOT understanding everything!  **Macintosh HD:Users:kappter:Desktop:Screen Shot 2013-06-27 at 11.41.11 AM.png** |
| CHALLENGES FOR STUDENTS: MODIFY THE CODE TO DO THE FOLLOWING  1. DOUBLE THE SIZE OF THE BALL  2. MAKE THE CANVAS AS LARGE AS YOUR DISPLAY  3. SET THE SPEED OF THE ANIMATION TO HALF  4. MAKE THE BALL GO EXACTLY UP AND DOWN IN THE SAME X POSITION  5. SHOW THE BALL PATH  6. INVERSE THE COLOR OF THE BALL AND BACKGROUND  7. STEP THROUGH THE ANIMATION WATCHING XPOS AND YPOS CHANGE  8. MAKE ANOTHER BALL  9. WHAT IS THE FORMATTING ERROR IN THE CODE SAMPLE ABOVE? |

Notes:

|  |
| --- |
| int xspeed, yspeed;  int xpos, ypos, wdth, ht;  //this runs once after program starts  void setup() {  size(displayWidth, displayHeight);  background(255);  xspeed = 3;  yspeed = 10;  wdth = 20;  ht = 20;  noStroke();  xpos = width/2;  ypos = height/2;  frameRate(30);  }  //this runs on a infinite loop  void draw() {  background(0);  fill(255);  ellipse(xpos, ypos, wdth, ht);  ellipse(xpos-50, ypos-50, wdth, ht);  xpos += xspeed;  ypos += yspeed;  if (xpos >= width-wdth/2 || xpos <= wdth/2) {  xspeed \*= -1;  }  if (ypos >= height-ht/2 || ypos <= ht/2) {  yspeed \*= -1;  }  } |