

What will we be learning in this Lesson?

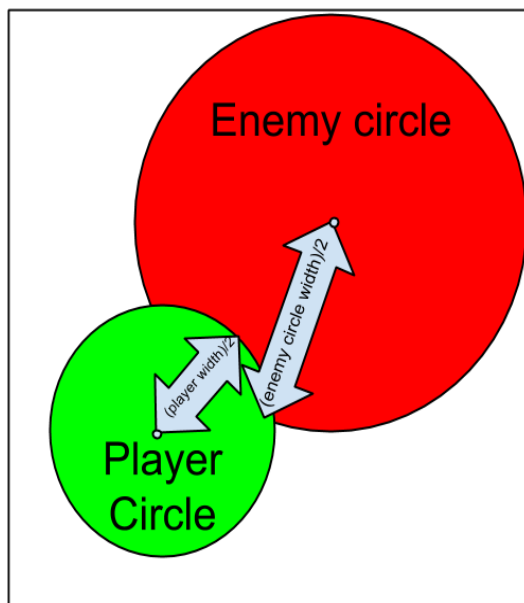
- How to use the `dist()` function.
- How to use what you already know to make collision detection.
- Improving our knowledge of `if()` statements.

Collision Detection:

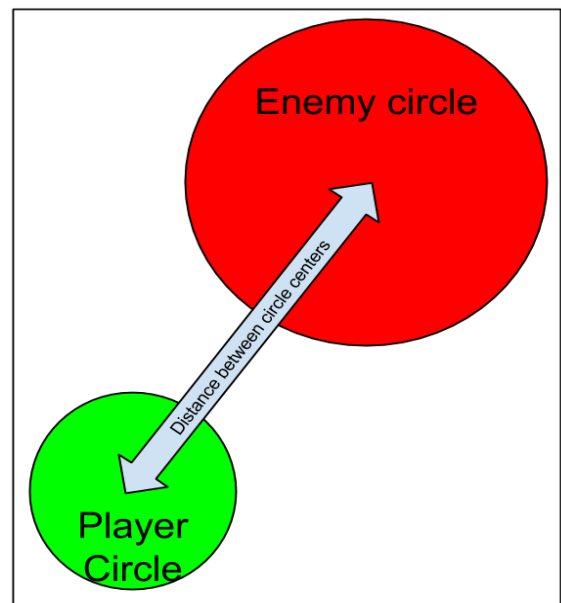
Collision detection is the term used to describe code, whose job it is, to test if two or more objects are touching. Collision detection is used all the time in games, to detect: the player collecting power ups, bullets hitting players, racers colliding with the edge of the track, and a lot more!

How to do Collision Detection between two circles:

You will need 2 things for this type of collision detection: The distance between the two circles and the radius of each circle. If the **length of the two radii added together** *is greater* than the **distance between their centres**, the circles have collided.



Distance < Radii = collision



Distance > Radii = no collision

Collision Detection Formula:

The formula that we use for the collision detection is the following:

$$\text{COLLISION: (playerRadius + enemyRadius) > distanceBetweenCentres}$$

Below is the partially finished sample code for the collision detection.

```
float ballOneX;
float ballOneY;
float ballTwoX;
float ballTwoY;
float ballOneDiameter;
float ballTwoDiameter;
float distance ;

void setup() {
  size(700, 700);
}

void draw() {
  background(0,255,0);
  distance = dist(ballOneX,ballOneY,ballTwoX,BallTwoY);

  if((ballOneDiameter + ballTwoDiameter)/2 >= distance ) {
    background(255,0,0);
  }

  ellipse();
  ellipse();
}
```

Task:

1. Finish the code above by adding in values for the variables and adding them into the `ellipse()` functions.
2. Make one of the ellipses move with your mouse so you can test collision.
3. Write code to show the distance between circles and radius of each circle. You can show text on a screen using the `text()` function. You'll need 3 parameters: The message and the position on the screen.

Bonus task:

Change the program to use a **boolean** to do the collision detection. Add another circle and do the collision detection for that circle as well. For your new circle, move it with W,A,S,D and make collision work for it.

What does the `dist()` function calculate:

The `dist()` function purpose is to calculate the shortest distance between two points on the screen, using the distance of a line formula $\sqrt{(X2 - X1)^2 + (Y2 - Y1)^2}$. The parameters that we put into the `dist()` correspond to the variables in the formula.

Questions for the end of the lesson?

- What is collision detection
- What is the formulas for collision detection with two circles
- What does `dist()` measure?
- If two circles have a diameter of 100, and are at points (200,200) and (200, 400), are they touching?