

# COMMENT YOUR CODE!

David Wyant

A dark blue diagonal gradient bar that starts from the bottom left corner and extends towards the top right corner, covering the lower half of the slide.

I've know something that all code needs...

I've know something that all code needs...

COMMENTS!!!

This is my handwriting.

Hello,

my name is David

I am 22 years old

I really like coding a lot

but sometimes it's really hard

writing code is very  
important

the more you understand  
the better, meaning

This is my handwriting.

Would you like to know  
what it says?

Hello,

my name is David

I am 22 years old

I really like coding a lot

but sometimes it's really hard

Learning code is very  
important

It helps us understand  
the world, creating

# Translation:

//Hello,

Hello,

//My name is David

my name is David

//I am 22 years old

I am 22 years old

//I really like coding a lot

I really like coding a lot

//but sometimes its really hard

but sometimes its really hard

//Commenting code is very important

commenting code is very important

//It helps us understand what others are thinking

It helps us understand what others are thinking

# Commenting Code:

Most Languages:

// For a Line

/\* Or this for everything in between \*/

HTML: <!-- -->

Haskell: {- -}

Python: #

# Why Do It?



# Why Do It?

- Inform your teammates what your code does, so they can work with it.

# Why Do It?

- Inform your teammates what your code does, so they can work with it.
- allow employers to understand what you did by looking at your work.

# Why Do It?

- Inform your teammates what your code does, so they can work with it.
- allow employers to understand what you did by looking at your work.
- Inform yourself what you did last week.

# Why Do It?

- Inform your teammates what your code does, so they can work with it.
- allow employers to understand what you did by looking at your work.
- Inform yourself what you did last week.
- Temporarily cut out code that causes problems.

# Why Do It?

- Inform your teammates what your code does, so they can work with it.
- allow employers to understand what you did by looking at your work.
- Inform yourself what you did last week.
- Temporarily cut out code that causes problems.
- make checklists or notes on what needs to be done.

```

import processing.sound.*; //allows for music and sound effects
SoundFile curMusic; //current music being played
String curTheme; //name of the song currently being played
SoundFile swing; //swing sound effect
SoundFile bouncey; //sound effect for bouncing off walls and sinking into the hole
PImage mc; //special mouse cursor
ArrayList<Level> levels; //list of all levels in the game
String[] data; // Data from text file
HUD hud; //overlay of the game
Ball ball; //ball that is used in the game
Level cl; //current level selected
String state; //current state the game is in (TITLE, SWING, WAIT, Etc)
boolean aMove; //true if the ball can be swung at right now

//setup method
void setup(){
    mc = loadImage("Club.png");
    swing = new SoundFile(this,"Swing.mp3");
    bouncey = new SoundFile(this, "Bounce.mp3");
    size(700,700);
    data = loadStrings("Level_Info.txt");
    state = "TITLE";
    hud = new HUD();
    levels = new ArrayList();
    for(int i = 0; i < hud.numholes; i++){ // assembles the levels based on data from the text file
        String[] info = split(data[i], ' ');
        float[] ballco = float(split(info[2] + " " + info[3] + " " + info[4] + " " + info[5] + " " + info[7] + " " + info[8], ' '));
        boolean slide = boolean(info[6]);
        levels.add(new Level(info[0],info[1],ballco[0],ballco[1],ballco[2],ballco[3],slide,new PVector(ballco[4],ballco[5])));
    }
    cl = levels.get(0);
    ball = new Ball(cl.startX,cl.startY);
    aMove = false;
    curTheme = "Title.mp3";
    curMusic = new SoundFile(this,curTheme); //starts the music
    curMusic.play();
    curMusic.loop();
}

```

```

//if the mouse is pressed, and the ball is able to be moved, checks if the mouse is on the ball and changes aMove as such
void mousePressed() {
    if (!ball.air && (ball.velocity == ball.velocity.mult(0))){
        aMove = ball.moveStart();
    }
}

//if the mouse is released and the ball is able to be moved, the methods for the ball being shot are run through
void mouseReleased(){
    if (!ball.air && (ball.velocity == ball.velocity.mult(0)) && aMove){
        aMove = false;
        hud.incstroke();
        ball.shoot();
        swing.play();
        state = "WAIT";
    }
}

//if a key is pressed, this method is called
void keyPressed(){
    if(key == 't'){ //returns the user to the title screen
        state = "TITLE";
        hud.reset();
        cl = levels.get(hud.holenum-1);
        curTheme = "Title.mp3";
        changeSong();
    }
}

//build a path to show the amount of movement the ball will undergo, and adjusts the trajectory to accomodate this
void buildPath(){
    PVector loc = new PVector(mouseX,mouseY);
    int am = int(sqrt(sq(loc.x-ball.position.x)+sq(loc.y-ball.position.y)))/15;
    fill(random(255),random(255),random(255));
    for(float i = 1; i <=am;i++){
        ellipse(((ball.position.x-loc.x)*(i/am))+loc.x,((ball.position.y-loc.y)*(i/am))+loc.y,5,5);
    }
    ellipse(loc.x,loc.y,7,7);
    ball.setTrajectory(ball.position.x-loc.x+ball.position.x,ball.position.y - loc.y + ball.position.y);
}

```

In Conclusion:

In Conclusion:  
**COMMENT YOUR CODE!**