

Day 1  July 31

Web

Design

Code

Hw

Day 2  Aug 1

Web

Design

Code

Hw

Day 3  Aug 2

Web

Design

Code

Hw

Day 4  Aug 3

Web

Design

Code

Hw

Day 5  Aug 4

Web

Design

Hw

Dorkshop  Aug 5

Day 6  Aug 7

Web

Design

Code

Hw

Day 7  Aug 8

Web

Design

Code

Hw

Day 8  Aug 9

[Web](#)

[Design](#)

[Code](#)

[Hw](#)

[Day 9](#)  [Aug 10](#)

[Web](#)

[Design](#)

[Code](#)

[Hw](#)

[Day 10](#)  [Aug 11](#)

[Web](#)

[Design](#)

[Code](#)

[Hw](#)

[Friday Pizza Night](#)

[Dorkshop](#)  [Aug 12](#)

[Day 11](#)  [Aug 14](#)

[Web Special Topic: p5.play](#)  [Aug 12](#)

[Design](#)

[Code Library Day: sound with Minim](#) ~ [Aug 12](#)

[Code Library Day: Computer Vision](#)  [Aug 12](#)

[Day 15](#)  [Aug 18](#)

[Presentation](#)

[Zackery Bruce](#)

[Chavisa Rojratanaadumrong](#)

[Bryan Collinsworth](#)

[Ramsey Nasser](#)

[Yuemei Zhang \(张月美\)](#)

[Ke Qing](#)

[Digests](#)

[Appendix I](#)  [HCD Project](#)

[Mindmapping](#)

[Design Question](#)

[HMW](#)

[Goals](#)

[Design Statements](#)

[Resources](#)

[Persona](#)

[Scenario](#)

[Prototype](#)

Appendix II ➔ Text Adventure: Fake AI

Introduction

Flowchart

Snapshots

Implementation

Source Code

License

Appendix III 📈 Project Catastrophe

Observation

Appendix IV 🔺 Sound of Shape (Bootcamp Code Final Project)

Inspirations

Neil Harbisson

Research

"Deaf in America: Voices from a culture" by Padden and Humphries

Do people who are born deaf know what sound is?

Do Deaf people understand the concept of sound? In what way?

Human speaking frequency

Design Statement

Tools

Minim sound library with Processing

Processing functions

Process

Appendix V 🔥 Creative Coding

12 Principles of Animation

Learning Processing

Chapter 1: Pixels

Color range

Chapter 2: Processing

Chapter 3: Interaction

mouseX, mouseY, pmouseX, pmouseY

Chapter 4: Variables

System variables

Chapter 5: Conditionals

Chapter 6: Loops

Use constrain() to exit loops

Chapter 7: Functions

Chapter 8: Objects

Chapter 9: Arrays

Resize using append()

Processing frame functions: frameRate(), frameCount(), and frameRate

Chapter 10: Algorithms

[dist\(\)](#)
[Max size of arrays = 2*31 = 2147483647](#)
[Rain drop](#)
[Chapter 11: Debugging](#)
[Chapter 12: Libraries](#)
[Chapter 13: Mathematics](#)
[Noise generation](#)
[map\(\)](#)
[How to draw circle](#)
[How to draw wave](#)
[Stroke vs. fill](#)
[Chapter 14: Transformations and 3D](#)
[translate\(\)](#)
[P3D vs. OPENGL](#)
[Custom shapes](#)
[3D Coordinates](#)
[Rotation](#)
[rectMode\(\)](#)
[Example: solar system](#)
[PShape](#)
[Chapter 15: Images and Pixels](#)
[Images](#)
[Pixels](#)
[Image Processing](#)
[Chapter 12: Libraries](#)
[Chapter 12: Libraries](#)
[Chapter 12: Libraries](#)
[Chapter 12: Libraries](#)
[The Nature of Code](#)

Day 1 ➡ July 31

Web

Credits: [Lucien Huang](#)

Components of Internet

App: html/css/js

Server: python/js/php

Database: MySQL/MongoDB

How to put a website online

Domain name

Server

FTP (File Transfer Protocol): a standard network protocol used for the transfer of computer files between a client and server on a computer network

What happens when I go to a website

1. Browser sends out request of a specific page
2. Use **HTTP (Hypertext Transfer Protocol)** protocol to request the page
3. The server responds by sending back the metadata, followed by the page source
4. Other protocols include the **HTTPS (Hypertext Transfer Protocol Over Secure Socket Layer)**

www.giphy.com/artists

Protocol Domain Path

5. Server fetches the page via **URL (Uniform Resource Locator)**

6. Which is located with a unique IP address

7. *for Mac you can find out your IP address by typing

`ifconfig`

Web Application

Front-end languages: HTML/CSS/JS

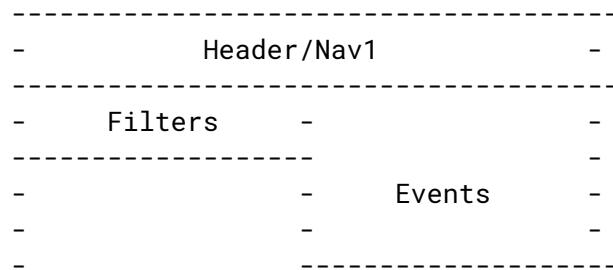
HTML: HyperText Markup Language (HTML) is the building-blocks of web pages

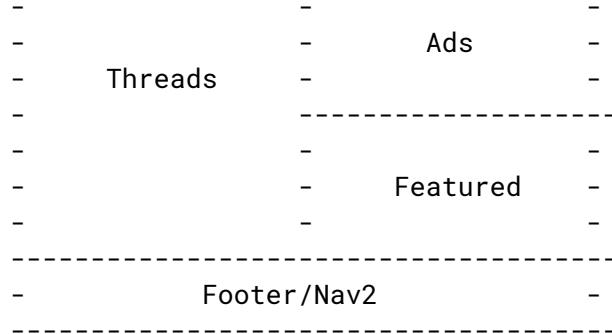
CSS: Cascading Style Sheets (CSS) dictate the website's look and feel

JS: JavaScript (JS) dictates how the website behaves based on events (mainly by the user)

Wireframing of a website

<https://stackoverflow.com>





*wireframing softwares: Balsamiq.com

Homework

1. Draw wireframing for your portfolio site
2. Email pdf by tonight 12AM
3. Find two or three websites to share, post them to the blog
4. Codecademy // Introduction to HTML lessons 1-3
codecademy.com/en/tracks/web

Design

Credits: [Tuba Ozkan](#)

What is creativity?

Introduce yourself with 3 words?

Course Overview

- Mastery Project (Day 1-3)
- User-Centered Design Project & Prototyping (Day 4-6)
- Project Catastrophe (Day 6-10)
- Final Project (Week 3)

Required Readings and Videos

- [Italo Calvino, "Invisible Cities"](#)
- [Don Norman, "The Design of Everyday Things"](#)
- [What do prototypes prototype?](#)
- [Ellen Isaacs, "Ethnography" at TEDx \(Video\)](#)
- [Rick & Dick](#)
- [J. Paul Neeley, Speculative Design and The Future of UI \(Video\)](#)
- [Hillary Collins, "Creative Research"](#)

Additional resources

- [Google for entrepreneurs: Rapid Prototyping](#)
- [What is design ethnography?](#)

[Anthony Dunne, "What If... Crafting Design Speculations" \(Video\)](#)
[WingKee, "Speculative Design Fiction" \(Video\)](#)
[New Media Art](#)
[Bernard Tschumi, "Manhattan Transcripts"](#)

Mastery Project

What is Mapping?

The act or process of making a map, a diagram used to visually organize information

Types of mapping

Domain map, mind map...

Mind Map

Begins with only one work or idea

Radial/tree structure

Quick tool to transfer idea into a visual context

An easy way to brainstorm

Concept Map

Similar to mind mapping

Connections of multiple words or ideas

Connections between concepts in more diverse structures

Domain Map

It is a scope or an expression for you to plan and visually map out a project

It's the domains that your ideas fall underneath

Domain is an academic expression or discipline related to your interests or projects

Mind Mapping Process

1. Generate a focal point then develop related components

Domain Map

A tool that allows you to structure your ideas into specific categories that your ideas fall underneath

In design process in particular, it is an academic expression or discipline related to your interests or projects

Help define/structure your ideas

Questions to start with domain mapping

1. What larger and broader topics does my idea fit in
2. What topic does it relate/overlap onto
3. What discipline can I categorize
4. Who is my audience
5. What practices branch out in my ideas

Tools for mapping

1. Pen and paper
2. softwares

Homework

1. [Don Norman, "The Design of Everyday Things"](#) Chapter 1 pp.1-34, moderated by Jason, Nana, Ting & Kevin
2. **Mastery Project:** create a mind map and a short project based on the Invisible Cities in relation to your design skills
3. Present your mind map on Tue and submit your project on Wed

Code

Credits: [Andrew Cotter](#)

What is coding?

Stores information as 1s and 0s
Communicate with computers

Languages and Platforms

Swift: iOS
JS: web, p5.js
C++: openFrameworks, Unity
Java: Processing

Great artists use code

Casey Reas
Raven Kwok

Physical computing: Arduino

Color Code

Rgb
Rgba

Variables

Types of variables: Int, Float, Bool, String
Scope of variables
Syntax sugar: width and height are predefined variables as the width/height of the canvas

Processing

[Processing References!](#)

Pseudo Code first

Coding Resources

[The Coding Train](#)
[Codecademy](#)
[Learning processing](#)
[The Nature of Code](#)
[Stackoverflow](#)
[Khan Academy](#)
[Code Combat](#)

My Processing Development Setup

*the processing original IDE sucks, so I choose Atom to code and compile to processing

1. Download [Atom Text Editor](#)
2. Search and install Packages: processing (run processing in atom), processing-language (highlights processing syntax), and processing-autocomplete (autocomplete and suggests the semantics)
3. Hit `ctrl+alt+b` to compile
4. Enjoy

Example

https://github.com/kevinfan23/parsons_bootcamp/tree/master/code/day1/lecture_1

Homework

1. Watch video
2. Replicate [this image](#) in processing,
3. sample answers:

https://github.com/kevinfan23/parsons_bootcamp/tree/master/code/day1/homework_1

Hw

Norman Questions

1. What examples of bad designs you have seen in your life?
2. "Designs are at fault, not the operators", what are your takes on it?

Reading synopsis

Don Norman: *The design of everyday things*

1. Two of the most important characteristics of good design: **discoverability** and **understanding**
2. All artificial things are designed.
3. The field of design is relatively new
4. Products should fulfill human needs while being understandable and usable. In the best of cases, the products should also be delightful and enjoyable
5. Design focuses

Industrial design: The professional service of creating and developing concepts and specifications that optimize the function, value, and appearance of products and systems for the mutual benefit of both user and manufacturer (from the *Industrial Design Society of America's* website).

Interaction design: The focus is upon how people interact with technology. The goal is to enhance people's understanding of what can be done, what is happening, and what has just occurred. Interaction design draws upon principles of psychology, design, art, and emotion to ensure a positive, enjoyable experience.

Experience design: The practice of designing products, processes, services, events, and environments with a focus placed on the quality and enjoyment of the total experience.

6. It is the duty of machines and those who design them to understand people.
7. "We are people ourselves," engineers think
8. We have to accept human behavior the way it is, not the way we would wish it to be
9. Design was at fault, not the operators
10. Even experts make errors. So we must design our machines on the assumption that people will make errors
11. But even though much has improved, the rapid rate of technology change outpaces the advances in design
12. **HCD** (Human Centered Design): an approach that puts human needs, capabilities, and behavior first, then designs to accommodate those needs, capabilities, and ways of behaving

TABLE 1.1. The Role of HCD and Design Specializations	
Experience design	
Industrial design	These are areas of focus
Interaction design	
Human-centered design	The process that ensures that the designs match the needs and capabilities of the people for whom they are intended

13. Designers need to focus their attention on the cases where things go wrong, not just on when things work as planned...this is where the most satisfaction can arise
14. This understanding comes about primarily through observation, for people themselves are often unaware of their true needs, even unaware of the difficulties they are encountering

15. Experience is critical, for it determines how fondly people remember their interactions.
16. Discoverability results from appropriate application of five fundamental psychological concepts: **affordances, signifiers, constraints, mappings, feedback + conceptual model of the system**

Affordance

The presence of an affordance is jointly determined by the qualities of the object and the abilities of the agent that is interacting

Affordance is not a property...is a relationship

Visible affordances provide strong clues to the operations of things

Signifier

If an affordance or anti-affordance cannot be perceived, some means of signaling its presence is required

Gibsonian psychology, an ecological approach to perception...that people simply picked them up through "direct perception"

Signifiers can be deliberate and intentional, such as the sign push on a door, but they may also be accidental and unintentional

Mapping

Mapping is a technical term, borrowed from mathematics, meaning the relationship between the elements of two sets of things

Natural mapping, by which I mean taking advantage of spatial analogies, leads to immediate understanding

Controls should be close to the item being controlled

Note that there are many mappings that feel "natural" but in fact are specific to a particular culture

Conceptual Model

A conceptual model is an explanation, usually highly simplified, of how something works

17. **System Image:** People create mental models of themselves, others, the environment, and the things with which they interact
18. Good conceptual models are the key to understandable, enjoyable products: good communication is the key to good conceptual models
19. **The Paradox of technology**
20. Quite often each discipline believes its distinct contribution to be most important...The hard part is to convince people to understand the viewpoints of the others, to abandon their disciplinary viewpoint and to

think of the design from the viewpoints of the person who buys the product and those who use it, often different people

Day 2 ➡ Aug 1

Web

Markup Language

A set of markup tags, which describe the document content, ex. **HTML** (HyperText Markup Language)

<body></body> <head></head> <video>...

Tags

1. <html><head><body><title>
2. Inline tags: <a><video>
3. Block tags: <nav><p><h1...h6><table>
4. *<script><link> can be included in <head>
5. *non-breaking space

6. Line break

7. Thematic break

<hr>

8. Use id attribute for relative links

DOM (Document Object Model)

The hierarchy structure of the html is called DOM

HTML Tag Anatomy

<tagname attribute="value">content</tagname>

*for more attributes, check [here](#).

Use inspect in your Browser (Safari, Chrome) to see the web source

CSS (Cascading Stylesheet)

Used to style the DOM

Linked in the header

Homework

1. Make a hyperlinked website that introduces yourself

Design

Don Norman: The design of everyday things

Affordance

Actionable properties

Exist naturally, do not have to be visible, known or desirable

Independent of human's perception and do not change if our needs

and the goals change - Gibson

Perceived properties may change over time

result s from the mental interpretation of things, based on past knowledge or experiences - Norman

Research

Why

1. Helps to find subject matter
2. Gives insight into topic and domain
3. Help you make informed design decisions

Primary research: Interviews, surveys, questionnaires

Secondary research: secondhand info, offers different perspectives, analysis, and conclusions

Quantitative data

Length, percentage, age, average, temp

Making is researching + Iterations

1. Create personas
2. Prototyping
3. User testing

Starting Your Research

1. Ask the right question
2. Topic of interest
3. Identify a design problem
4. Google, library, JSTOR, Proquest, etc.
5. NYU & Cooper Union libraries

Inspirational websites

[Creative Applications](#)

[Hover States](#)

[NYC Media Lab](#)

[Designer Inspiration](#)

Background Information

1. Info about domain

2. Info about topic
3. Target audience
4. Secondary data
5. Precedent (conceptual, technical, aesthetic precedents)
6. **OPVL (Origin,, Purpose, Value and Limitation)**

Documentation

1. Shows the iterative process
2. Help others to understand your process
3. Document everything
4. Scopes: maps, research, sketches/storyboards, brainstorm notes, prototype iterations, code screenshots, user testing data (actual event, outcome), **final outcome**

Homework

1. Prepare to present Mastery Project
2. Assign: Post one project found on the listed websites, or introduce a new reference website

Code

Binary and Boolean

0s and 1s
Only true or false

Conditionals

If else statement
Conditional operators: &&, ||

Example

https://github.com/kevinfan23/parsons_bootcamp/blob/master/code/day2/lecture_2/lecture_2.pde

How to Draw Flowchart Diagram

<http://creately.com/blog/diagrams/flowchart-guide-flowchart-tutorial/>

Homework

1. Write out a paper story for a game
2. Requirements
 - A planned storyline
 - User input
 - Minimum number of stages: 3
 - Use canvas to show storyline illustration
 - Use sound as a bonus
 - Presentation

Hw

Portfolio Site

[An old version of my portfolio site](#)

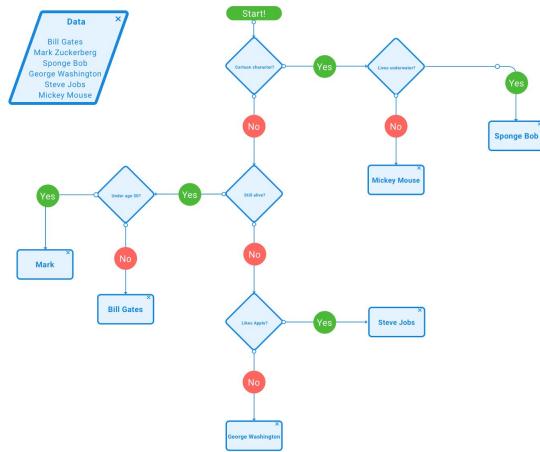
Mastery Project

[Project in static mode](#)

[On Codepen](#)

Paper story for a game

[Flowchart](#)



Day 3 🤞 Aug 2

Web

Some Useful Online Resources for CSS

[Google Fonts](#)

[Adobe Color CC, color wheel](#)

[W3 School Org](#)

[CSS Tricks](#)

CSS Percentage Value

https://www.w3schools.com/cssref/pr_dim_width.asp

CSS Box Model

Box-sizing: <https://css-tricks.com/box-sizing/>

Universal border-box

```

*, *:before, *:after {
  box-sizing: border-box;
}
  
```

To Center Text Inside the Div

<https://stackoverflow.com/questions/5703552/css-center-text-horizontally-and-vertically-inside-a-div-block>

To Center A Div Inside the Div

<https://css-tricks.com/centering-css-complete-guide/>

CSS FlexBox

<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>

CSS Selectors

https://www.w3schools.com/cssref/css_selectors.asp

Unix Style File/Directory Path (the one CSS uses)

[https://en.wikipedia.org/wiki/Path_\(computing\)](https://en.wikipedia.org/wiki/Path_(computing))

CSS Positioning

Reference: <https://css-tricks.com/almanac/properties/p/position/>

Z-index: <https://css-tricks.com/almanac/properties/z/z-index/>

Homework

1. Stylize the website that you built with HTML
2. Post screenshots to blog

Design

Homework

1. Watch Google for Entrepreneur Rapid Prototyping video

Good Presentation

1. Show + visual, videos?
2. Don't read the slides
3. Know your material and practice (time, record)
4. Know your audience (organize your language and presenting styles)
5. Go to your site before presenting, know your environment
6. Backup your files (Google Drive, dropbox)

Good Critiques

1. Mix positive feedbacks with negative ones
2. Contextualizing
3. Be specific
4. Offer next steps
5. Offer alternative approaches
6. Listen, respect (not defensive), ask
7. Get critiques from multiple sources (academic background, demographic, groups, previous knowledge, etc)
8. Ask for the feedbacks you want

Mastery Project Critiques



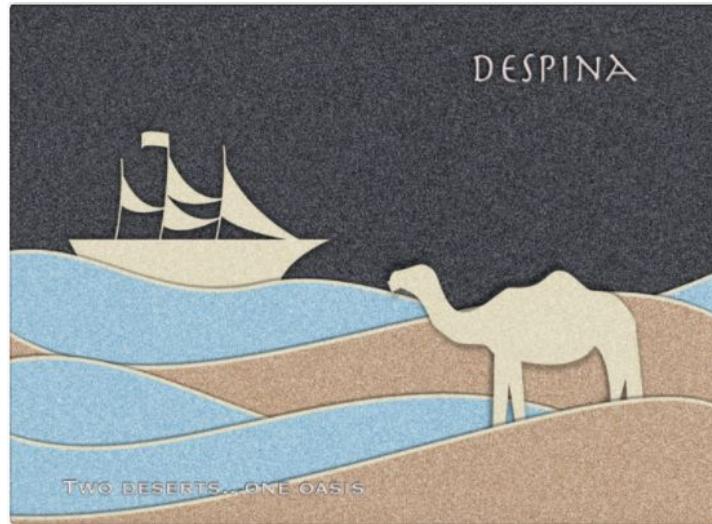
Barsha: Time in still

!! Monk in the crowd
Indra Chowk St (weather god)



Milan: Globe

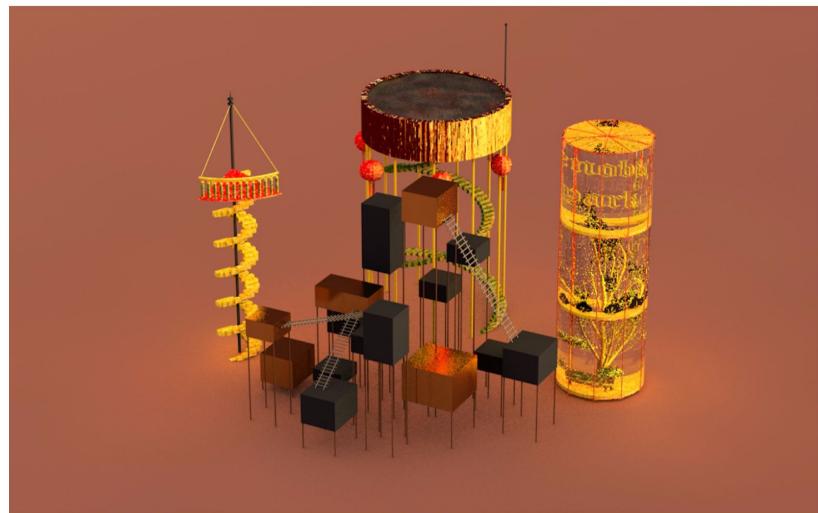
!! Eye
Infinity
The city of Phedora
Different speed, different directions



Srishti: Two desert...one oasis

!! Sand texture

3D shadow added depth to the image



Jason

!! Texture, 3d, miniature

Code

Processing Input Interactions

```
keyPressed -> bool  
keypressed() -> function gets called as key pressed  
CODED = The variable keyCode is used to detect special keys such as the arrow  
keys (UP, DOWN, LEFT, and RIGHT) as well as ALT, CTRL, and SHIFT.  
keyReleased() -> function gets called as key released  
mousePressed  
mousePressed()
```

For more key interactions, check [Processing References](#).

Using Assets

Images

Supported format: jpeg, gif, png, tga, and
[**PImage**](#): the datatype for storing/manipulating images in Processing
Use function [`image\(\)`](#) to draw image

Font

Supported format: jpg, gif, png, tga, and
[**PFont**](#): the datatype for storing/manipulating images in Processing
[`loadFont\(\)`](#): Loads a .vlw formatted font into a PFont object
[`createFont\(\)`](#): Dynamically converts a font to the format used by
Processing from a **.ttf or .otf** file inside the sketch's "data"
folder or a font that's installed elsewhere on the computer.

Example

https://github.com/kevinfan23/parsons_bootcamp/blob/master/code/day3/lecture_3/lecture_3.pde

Homework

1. Finish first two stages of the code video game

Hw

Design

[Google for Entrepreneur Rapid Prototyping](#)

1. Prototype early and often
2. Prototyping types: sketch & paper prototyping, digital prototyping, native prototyping
3. Design -> Prototype -> Test -> Validate -> Iterate
4. Communicate your vision: Ideas + Clarity + Stakeholders + Investors + Buy-in

Sketch & Paper Prototyping

5. Sketch & paper prototyping: paper or postnote
6. Sketch outline of steps, use flow, or interactions => wireframe
7. High fidelity sketches: button, images, and color

8. Paper prototyping: key user interactions, purposeful use of colors, explore elevations and shadows
9. Create gifs with paper prototyping to further explore the user flow

Digital Prototyping

10. Get the hang and feel of product without engineering expense
11. Purposes: get the feel and an artifact for user testing

Native Prototyping

12. Be ready to throw it away.
13. Great way to test the development environment
14. Explore your technology + Validate your direction + Communicate your vision

Code

https://github.com/kevinfan23/parsons_bootcamp/tree/master/code/code_videogame

Day 4  Aug 3

Web

CSS Transition

[Reference](#)

[Easing functions cheat sheet](#) (Not supported in CSS, but can convert to [cubic bezier](#))

CSS Transformation

[Reference](#)

Design

Lenses

1. Critical
2. Conceptual
3. Speculative
4. Human-centered
5. Non-human centered design

Different Design Disciplines

Industrial design

function, value, and appearance of products and systems for the mutual benefit of both user and manufacturer

Interaction design

The goal is to enhance people's understanding of what can be done, what is happening, and what has just occurred.

User experience design

Enhancing user satisfaction with a product by improving the usability, accessibility, and pleasure provided in the interaction with the product

HCD (Human-Centered Design)

An approach that puts human needs, capabilities, and behavior first, then designs to accommodate those needs, capabilities and ways of behaving.

HCD Process

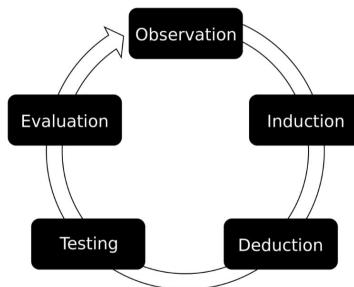
Inspiration -> Ideation -> implementation (by IDEO)

Where to begin

1. Brainstorming
2. Mind mapping
3. Design question
4. How might We's
5. Challenges & goals
6. Design statements

Observation Research

1. Survey
2. Interviews
3. AEIOU
Actions
Environment
Interactions
Objects
Users



Persona

Specify: help us specify which groups of users most important to target

Communication needs: tell story about different types of users and their needs

Transcend archetypes: defines a class or type of user and realize

Derived from behaviors: based on observations of real people we are targeting as users + supporting quantitative research about them

=> Persona = behaviors + patterns + goals

How to Create Personas

1. Identify user characteristics: activities, attitudes, aptitudes, motivations, skills
2. Map and identify behaviors: look for clusters of people that occurs across multiple behavior ranges or variables
3. Patterns: most critical details, both by observing actions or by analyzing subject responses to goal-oriented interview questions
4. Identify the logical connections between people's behaviors

Storyboard

Visual representation of your scenario and personas

Help people better connect how your product/service will work

Write it all out as steps first in a scenario. Make one scenario for each persona => storyboard

Homework

1. Read "How Do Prototypes Prototype"
2. Send ideas for HCD
3. Create a Human Centered Design Project
 - a. Mind mapping, design questions, HMWs, goals, design statements
 - b. What resources do you want to look at (2 sources)
 - c. Create a persona (or personas) of your main intended users and their goals
 - d. Create a scenario (written or visual), imaging how users will use your design

Code

Function

Reusability

Return-type & value

Hw

Design

1. [HCD project](#)
2. [How Do Prototypes Prototype](#)

Code

1. [Text adventure](#)

Reading Synopsis

How Do Prototypes Prototype

Prototyping audiences

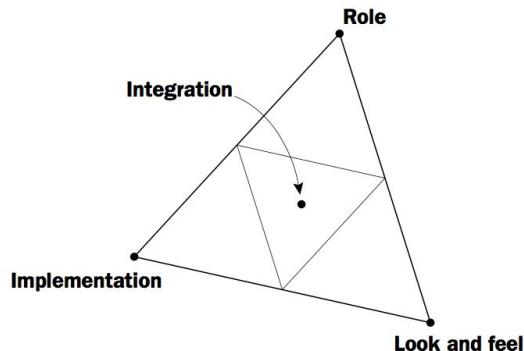
1. Intended users: feedbacks on evolving design
2. Design teams: critiquing prototypes of alternate design directions
3. Supporting organizations: indicate progress and direction

- What is significant is not what media or tools were used to create them, but *how they are used by a designer* to explore or demonstrate some aspect of the future artifact.

Terminologies

- Resolution:** the amount of detail
- Fidelity:** closeness to the eventual design
- Prototype: any representation of a design idea, regardless of medium
- Artifact: the interactive system being designed

What prototypes prototype



- Three processes developed in parallel
- Role -> Storyboard (how it integrates into user's life)
- Define "prototype" broadly.
- Build multiple prototypes
- Know your audience.
- Know your prototype; prepare your audience

Day 5 ➡ Aug 4

Web

CSS Float

[Reference](#)

CSS Units

[Reference](#)

Homework

- Finish your personal website

Design

Understanding Prototypes

Prototypes are

Building Prototype

1. Determine the type of challenges
2. Decide which types of prototype you need: experimental, technical, aesthetic

Experimental prototype

Tools

Aesthetic prototype

Look/sound

Tools

Technical prototype

Possible/feasible

What components are ready-made

Tools: mechanical and digital tools

Homework

1. Watch Ethnography by Ellen Isaacs at TEDx
2. Create a prototype to the HCD project they worked on, and make a presentation on the entire project
3. Upload your project

Hw

Web

[Portfolio site](#)

Design

1. HCD Project
2. [Ellen Isaacs, "Ethnography" at TEDx](#)

Reading Synopsis

Ellen Isaacs, "Ethnography" at TEDx

1. Alan Kay (Xerox PARC) and Doug Engelbart invented GUI and the mouse respectively
2. Alto, 1973, the first graphical personal computer

Dorkshop Aug 5

openFrameworks

[Official website](#)

[MAC OSX Xcode setup](#)

[Start a new project](#)

Array

Declaration

```
float size[500];  
Array indexing: start from size[0]...size[499] => 0-indexed
```

Examples

1. [Example 1](#)
2. [Example 2](#)

Day 6 ➔ Aug 7

Web

Homework

1. Keep working on the website

Design

Ethnography

Definition: The study and systematic recording of human cultures; also: a descriptive work produced from such research

Design ethnography

1. to ultimately understand more of the user's perception of the object, environment, system, or service the user is engaged with.
2. Takes place in the field
3. Observation is primary data collection technique
4. Re-shadowing, surveys and interviews useful for clarifying observation

Represent your observation

1. Note-taking
2. Video recording
3. Audio recording
4. Mapping
5. Drawing sketches

Mapping

An emerging act in contemporary discourse to understand, criticize, and re-imagine complex cultural, social, and physical relationships in the built environment

Event, space, movement

Project Catastrophe

Scenarios

1. Higher temperature
2. Continuing floods
3. Post-earthquake situation

What to design?

Communications

Design stories to help people imagine new ways of living amidst climate change. Help them imagine the changes they need to make in their lives to adapt to shortages or disasters caused by climate change

Products and spaces

The objects, buildings and places designers create can empower people to deal with the changes they will experience in their supply of water, food, and energy, and the threats to their home and health

Services

A human centered approach is crucial for finding new ways to connect people online and offline to improve (or invent) services around water, food, energy, housing and healthcare, that help people to adapt to climate change.

Systems

Design system transformations that accelerate people's adaptation to climate change. This is needed to make the many systems most people rely on (from food, water, healthcare, housing and energy) more resilient to climate change

Homework

1. [J. Paul Neeley, Speculative Design and The Future of UI](#)
2. [Speculative Design Fiction](#)

Code

Array

1. [Documentation](#)
2. Definition: A list of variables
3. Extremely efficient in memory, uses adjacent memory
4. Declaration

```
Int[] number = {90, 150, 30};  
String[] names = {"Neil, Vivian..."};
```
5. [zero-indexed](#)

For Loops

Statement 1 | statement 2 | statement 3

HSB

<http://www.tomjewett.com/colors/hsb.html>

Code Final Project

1. Concept
2. Precedent/References
3. Process
4. Flowchart
5. Challenges
6. Future iteration
7. Live demo

Creative Coding Demo

1. openProcessing.org
2. [Creative Applications](#)

Homework

1. Create patterns using for loop and arrays

Hw

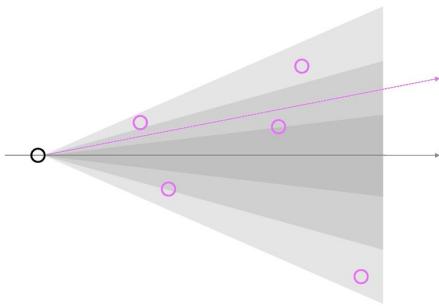
Design

Reading Synopsis

J. Paul Neeley, *Speculative Design and The Future of UI*

1. **Speculative design ≈ critical design**

- a. Not fantasy
- b. Not Utopic or Dystopic // complexity
- c. Messy people
- d. Not predictive // work in multiples
- e. Unclear contexts // emerging tech



2. How would self-driving cars change the world?
3. Only 4% utilization rate for cars
4. Think about AI and VR would affect your business

Code

[Patterns](#)

Day 7 ➡ Aug 8

Web

Javascript

1. Event driven, click, load
2. Link js file

```
<script src="js/main.js"></script>
```

3. How to include the script tag, [Google's explanation](#)
4. Rules for naming variables: always start with letters, & or _
5. Alert and ConcoleLog

```
        alert("text");
        console.log("text")
```
6. Comparison operators ==, !=
7. Strict comparison operators, ===, !===, [explanation](#)
8. Logical operators, ||, &&, !
9. **getElementById** => variable, **getElementsByClassName** => array
10. Manipulate the DOM, use **style**

```
document.getElementById(id).style.property = new style
```

Inspirations

1. <http://arkade.london>
2. <http://www.makemepulse.com>

Design

Homework

1. Prepare to present notes/sketches/interviews/ mappings tomorrow, present analysis and observations + preliminary scenario
2. [Rick & Dick](#)

Code

PVector

[Reference](#)

Animate Movement

1. Position
2. Speed
3. Acceleration
4. Friction

Homework

1. [Lauren McCarthy talk, the creator of P5.js](#)
2. [Neil Harbisson TedX talk](#)

Hw

Design

Reading Synopsis

Dick & Rick: A visual primer for social impact design / community-engaged design

1. Do your research
2. Interview people and develop projects with the community

Day 8 ➡ Aug 9

Web

Javascript Cont.

1. For loop
2. If-else statement

Homework

1. Add JS to your portfolio site

Design

Homework

1. Finalize prototype and presentation for Project Catastrophe

Code

Class

Class example

```
Class SuperHero {  
    String name;  
    Int age;  
    String superPower;  
    String homeTown;  
    String lover;  
  
    Void saveTheWorld();  
}
```

Example

https://github.com/kevinfan23/parsons_bootcamp/tree/master/code/day8/lecture_8

Homework

1. Pseudo code and MAKE a class.

Hw

Day 9 ➡ Aug 10

Web

Javascript Cont.

1. Toggle states

2. Create gallery

Example

https://github.com/kevinfan23/parsons_bootcamp/tree/master/web/gallery

Homework

1. Starting on final project

Design

New Media Art

1. [Lundin Norway: Breaking the Surface \(waves\)](#)
2. [New Media Art Performing: "The Inheritance" \(dance\)](#)
3. [INORI - PRAYER \(creepy\)](#)
4. [CULTIVOS de Gilberto Esparza \(digital bots\)](#)
5. [Cybercrime: Split and Compare \(data wheel\)](#)
6. [Step Inside a Painting with Virtual Reality \(in love with VR\)](#)
7. [Coldplay - Up&Up](#)

Homework

1. Finalize prototype and presentation for Project Catastrophe

Code

Example

https://github.com/kevinfan23/parsons_bootcamp/tree/master/code/day9/lecture_9

Hw

Design

Day 10 ➡ Aug 11

Web

Example

https://github.com/kevinfan23/parsons_bootcamp/tree/master/web/gallery

Design

Project

[Project Catastrophe](#)

Speculative Design

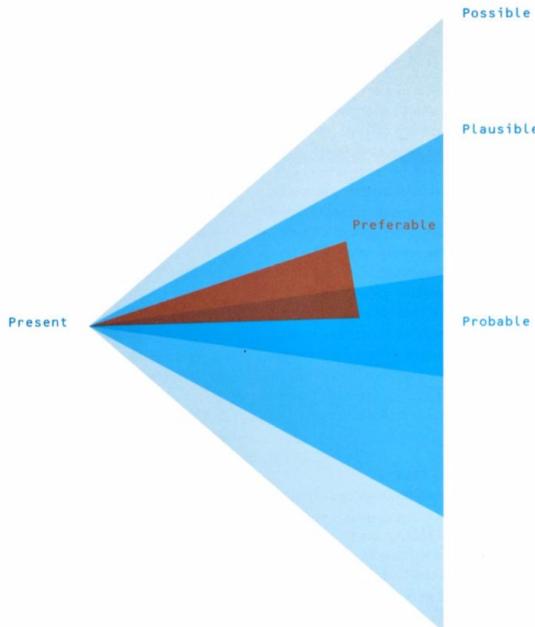
1. [Speculative Everything](#)

“ When people think of design, most believe it is about problem solving.

There are other possibilities for design: **one is to use design as a means of speculating how things could be**

”

Speculative Design Should



1. Open up all sorts of possibilities that can be discussed, debated, and used to collectively define a preferable future for a given group of people: from companies, to cities, to societies
2. Intended to be **provocative** rather than predictive or prescriptive
3. Illustrate of **relationships of production, consumptions, and circulation**
4. Instantiated in object as well as words

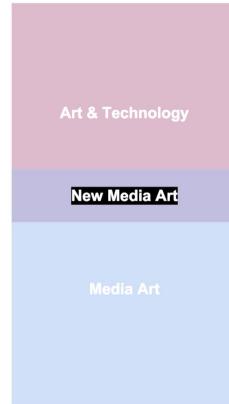
Categories of Arts

1. Fine/Traditional arts
2. Modern arts: include photograph, film making, pre-film, film based, video art, kinetic art, Mixed media
3. New media arts

New Media Art is the intersection of two different domains

- **Art & Technology:** art practices that involve technologies that are new but not necessarily media related
- **Media Art:** art practices that use old and new communication technologies or are concerned with platforms used to disseminate information (media)

Art in the Age of Digital Distribution
Mark Tribe and Reena Jana



Feedbacks

1. Prototype has to serve a purpose
2. Give audience a context at the beginning of your presentation
3. Analysis of field trip has to be related to your solutions
4. Use more pictures from yourself, instead of internet images

Code

Concepting

visualization -> social media -> sound, color, data

=> Person walking on wire, using voice to control left and right x

Hw

Design

Creative Research

What makes a good design topic

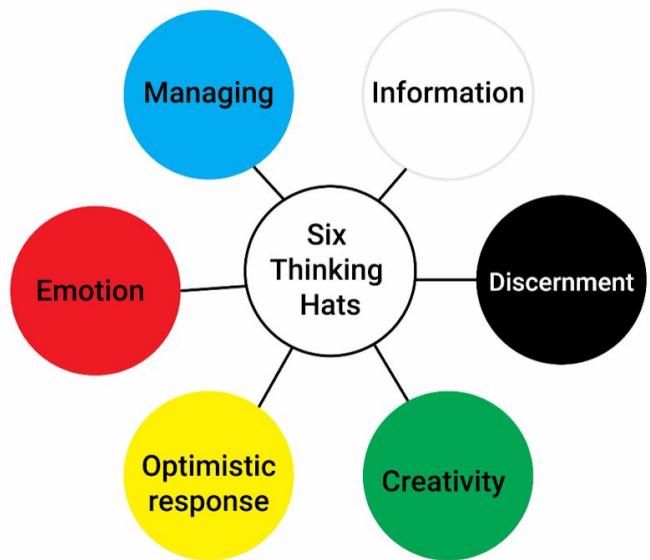
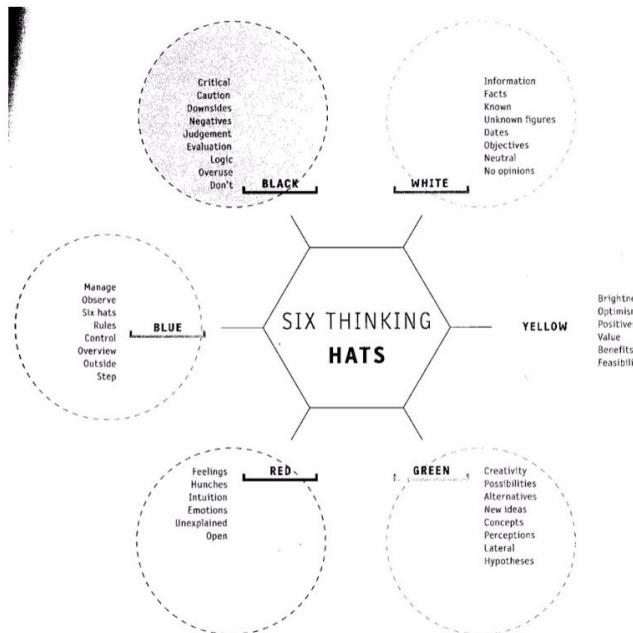
1. Interest
2. Focus
3. Available knowledge
4. Data availability
5. Career goals

Generating and refining research ideas

1. Finding connections: any ideas, no matter how small, is an association between previous established ideas
2. Being prolific, then is the key to having great ideas. => no trade off between quality and quantity
3. **Brainstorming => create idea pools**
 - a. Defer judgement, think about workable later
 - b. Go for quantity
 - c. Look for unusual or wild ideas
 - d. Combine and build on ideas
 - e. Random word/object technique
 - f. Mind maps
4. **Movement ladder => dig deeper**

- a. Decrease turnover
- b. Increase differentiation
- c. Increase competitiveness
- d. Improve the image of the organisation

Six-Thinking Hats



Friday Pizza Night

Morry Galonoy

Effectiveness + efficiency + satisfaction => Usability

Strategy

1. Define goals
2. ID & involve users
3. Understand context
4. Be curious, ask a lot of questions
5. Ask more questions
6. Don't jump to technical solution, let idea drive technology

Dorkshop 🚧 Aug 12

Documentation

1. Steps taken to reach the conclusion
2. Authentication of your work
3. Types of documentation
 - a. Phone, take landscape mode => better for screen
4. Wireframes, materials, sketches, mind maps,
5. Show logic between your works and your conclusions
6. Shows recruiters how you work and how you can incorporate into the culture

Day 11 👉 Aug 14

Web Special Topic: [p5.play](#) 🎨 Aug 12

P5.js

[Reference](#)
[P5 Online Editor](#)

p5.play

[Source code](#)
[Reference](#)
[createSprite\(\)](#)

Example

[Setup Node.js local server \(Mac\)](#)

1. Open terminal on your mac
2. Install Homebrew

```
ruby -e "$(curl -fsSL
```

```
https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

3. Install npm

```
brew install node
```

4. Check if npm and node are installed on your Mac



```
daves-mbp:install-node dave$ node -v
v0.10.31
daves-mbp:install-node dave$ npm -v
1.4.27
daves-mbp:install-node dave$
```

A screenshot of a macOS terminal window titled "1.bash". The window shows a command-line session where the user has run "node -v" and "npm -v" to check the versions of Node.js and npm installed on their machine. The output indicates Node.js version 0.10.31 and npm version 1.4.27.

5. Install node snippet http-server

```
npm install -g http-server
```
6. Go to your directory/folder where the web files are in

```
http-server
```
7. You should see in your console

```
Starting up http-server, serving ./  
Available on:  
  http://127.0.0.1:8080  
  http://149.31.224.61:8080  
Hit CTRL-C to stop the server
```
8. Open your web browser, go to

```
localhost:8080
```

Example

Design

Final Project

1. Design statement

I'm looking for a new way to visualize the city's data by creating a piece of drawing using the data and let users to interact with them through a console game.

Code Library Day: sound with Minim ≈ Aug 12

Minim

[Reference](#)

Fast Fourier Transform (FFT)

[Definition](#)

[Processing fft](#)

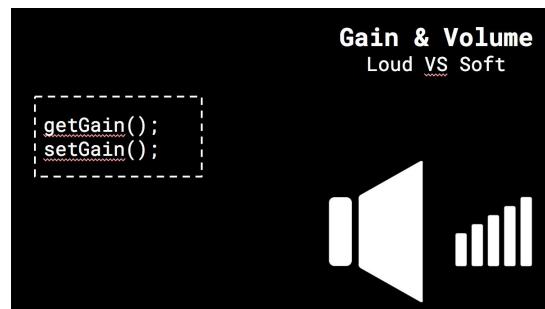
Pan



[setPan\(\)](#)

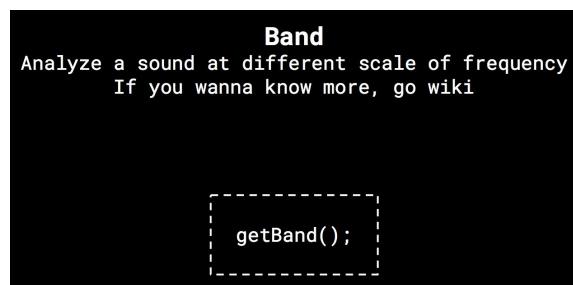
[getPan\(\)](#)

Gain



[getGain\(\)](#)
[setGain\(\)](#)

Band



[getBand\(\)](#): Returns the amplitude of the requested frequency band.
[setBand\(\)](#): Sets the amplitude of the i^{th} frequency band to a . You can use this to shape the spectrum before using `inverse()`.

Examples

[here](#)

Code Library Day: Computer Vision Aug 12

Processing Video Library

[Reference](#)

[Capture class](#)

OpenCV

[Official site](#)

[OpenCV with Processing](#)

[OpenCV vs. OpenGL](#)

Example

[here](#)

Day 15 ↗ Aug 18

Presentation

Zackery Bruce

[Soundcloud](#)

[Brooklyn blue grass](#)

musician

Chavisa Rojratanaadumrong

[Site](#)

Communication design, graphic design

P5.js for web (urban city builder), great taste, great learning with p5

Bryan Collinsworth

[Site](#)

Sharp, developer/designer

Ramsey Nasser

CS/game designer

Yuemei Zhang (张月美)

[behance](#)

Graphic design

Great taste, clean

Ke Qing

[site](#)

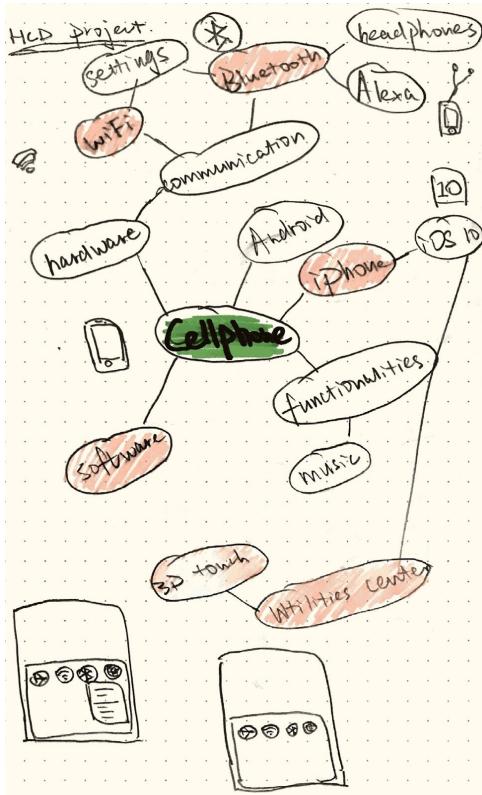
Communication, psychology, and digital media

Digests

1. Mix warm and cool colors to eliminate the time aspect
2. [The Interaction of Color](#)
3. Don't look at things too large, philosophical, something that cannot be solved
4. When you're designing something. You're building upon something people already built and looking for other people to build upon yours

Appendix I ➔ HCD Project

Mindmapping



Design Question

How to pair our increasingly important wireless devices with our phones (iPhone)? Ex. headphones, smart home devices, watches

HMW

Can we use 3D touch in the native iOS 10? -> could be added easily by Apple

Can we use gestures in the lock screen? -> accidentally touch

Can we use a user-defined buttons or gestures to pair devices? -> extra work + iphone accessibility

Goals

To be able to pair devices without unlocking the screen/go to settings

Design Statements

I'm exploring easier ways to pair bluetooth/wifi devices with cellphone because I want to find out how to pair devices at the home screen in order to avoid the hassle of going to the iphone settings every time.

I'm approaching this project from the lens of HCD.

Resources

1. [iOS HCI Guideline](#)
2. [Android Bluetooth Doc](#)

Persona



briefing on Alexa

Kristina

Jobs: Writer,
photographer
Location: Based at NYC

Wireless devices:

1. Canon 5D camera
2. Apple watch
3. Amazon echo dot
4. Beats wireless earphone

Uses phone for:

1. Download pics from Canon 5D camera
2. Listen to music with wireless earphones
3. Plays music/flash

Scenario

Morning: Kristina wakes up in the morning, use [Alexa](#) to play music/flash briefing of the news. Hops on the subway, she connects to her [wireless earphones](#) and uses her [Apple watch](#) to check texts at times

Afternoon: puts on her [earphones](#) to watch her favorite show during lunch break. Connects to her [camera](#) to check the photos she took in the morning.

Evening: listens to [Alexa](#) for flash briefing of the news. Organize photos again with her [camera](#).

Prototype

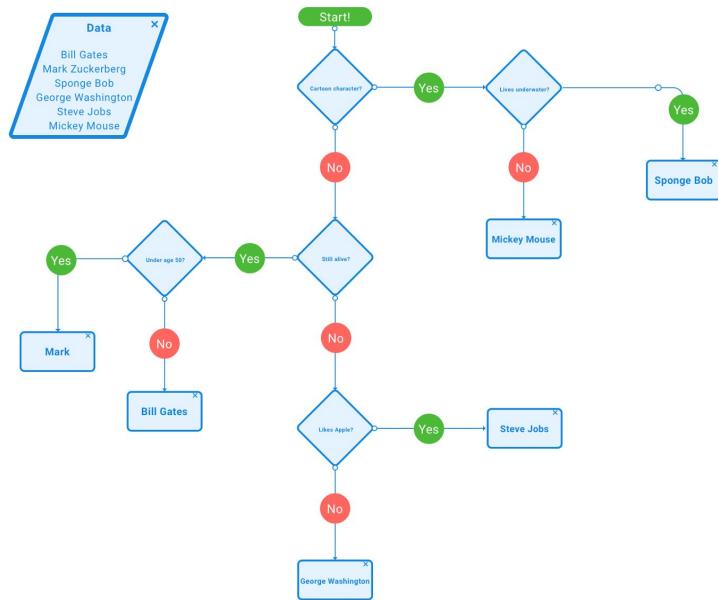


Appendix II ➔ Text Adventure: Fake AI

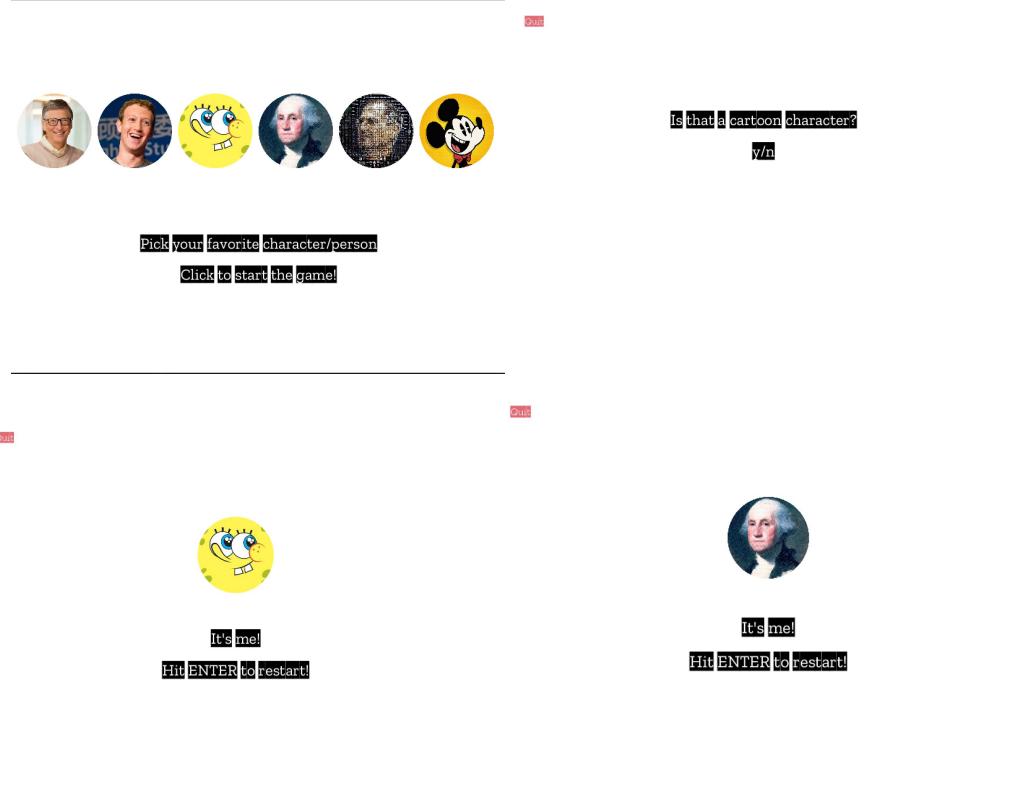
Introduction

Fake AI is a game in [Processing](#) that guesses your favorite character/person from a range of 6 people/characters provided by the system. Users can answer y/n to each prompt and the computer will be able to find out your favorite character/person within 3 steps.

Flowchart



Snapshots



Implementation

Fake AI uses Processing to generate graphics and text flow for the game. The game load assets through `loadImage()` and `createFont()` functions. For detailed implementations, refer to the **source code** in the following section.

Source Code

[Github](#)

License

MIT license

Appendix III Project Catastrophe

Observation

1. Mapping: dynamics, intensity, spread (Kevin)
2. Bridge design... design & history? (Jason) -> illustration & history reference
3. Heat/sound/smell (Kevin)
4. User engagement with technology...phone, camera, self sticker, circled net
5. Preliminary scenario: earthquake (evacuation)
6. Analysis: 2 findings each



Appendix IV ▲ Sound of Shape (Bootcamp Code Final Project)

Inspirations

[Neil Harbisson](#)

Research

["Deaf in America: Voices from a culture" by Padden and Humphries](#)

The tension between Deaf people's views of themselves and the way the hearing world views them finds its way into their stories, which include tales about their origins and the characteristics they consider necessary for their existence and survival.

[Do people who are born deaf know what sound is?](#)

[Do Deaf people understand the concept of sound? In what way?](#)

[Human speaking frequency](#)

[Music note frequency](#)

volume (gain) + pitch (frequency) => size of shape + color

Design Statement

Tools

[Minim sound library with Processing](#)

Thanks Jeana!

Processing functions

[triangle\(\)](#)
[rect\(\)](#)
[ellipse\(\)](#)
[scale\(\)](#)

Process

Appendix V 🔥 Creative Coding

12 Principles of Animation

[Wikipedia](#)

[Illustration](#)

Squash and stretch | Anticipation | Staging | Straight ahead and pose to pose animation | Follow through and overlapping action | Slow-out and slow-in | Arcs | Secondary action | Timing | Exaggeration | Solid drawing | Appeal

Learning Processing

[Website](#)

Chapter 1: Pixels

Color range

```
colorMode(RGB, 100, 500, 10, 255);
```

Chapter 2: Processing

Chapter 3: Interaction

```
mouseX, mouseY, pmouseX, pmouseY
```

Chapter 4: Variables

System variables

width—Width (in pixels) of sketch window.

height—Height (in pixels) of sketch window.

frameCount—Number of frames processed.

frameRate—Rate that frames are processed (per second).

screen.width—Width (in pixels) of entire screen.

screen.height—Height (in pixels) of entire screen.

key—Most recent key pressed on the keyboard.

keyCode—Numeric code for key pressed on keyboard.

keyPressed—True or false? Is a key pressed?

mousePressed—True or false? Is the mouse pressed?

mouseButton—Which button is pressed? Left, right, or center?

Chapter 5: Conditionals

Chapter 6: Loops

Use constrain() to exit loops

Examples

```
void draw()
{
    background(204);
    float mx = constrain(mouseX, 30, 70);
    rect(mx-10, 40, 20, 20);
}
```

Description

Constrains a value to not exceed a maximum and minimum value.

Chapter 7: Functions

Chapter 8: Objects

Chapter 9: Arrays

Array declaration and creation

```
int [] arrayOfInts = new int [42];
```

The "new" operator means we're making a "new" array.

Type Size of array

Resize using [append\(\)](#)

Processing frame functions: [frameRate\(\)](#), [frameCount\(\)](#), and [frameRate](#)

Chapter 10: Algorithms

[dist\(\)](#)

[Max size of arrays](#) = $2 \times 31 = 2147483647$

[Rain drop](#)

Chapter 11: Debugging

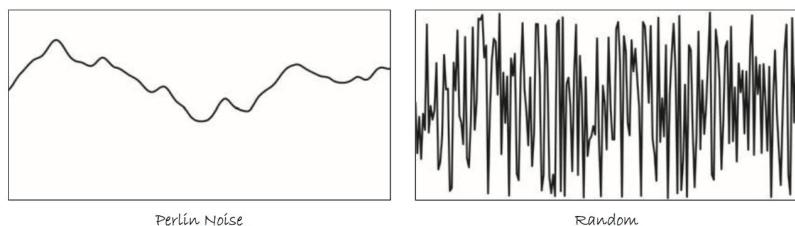
Chapter 12: Libraries

Chapter 13: Mathematics

Noise generation

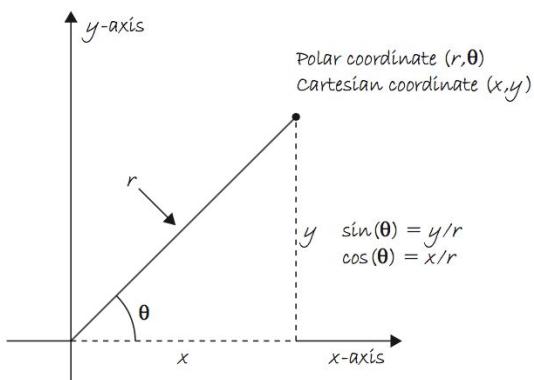
[Perlin Noise](#)

noise() vs. random



[map\(\)](#)

[How to draw circle](#)



[How to draw wave](#)
[Stroke vs. fill](#)

Chapter 14: Transformations and 3D

[translate\(\)](#)

P3D vs. OPENGL

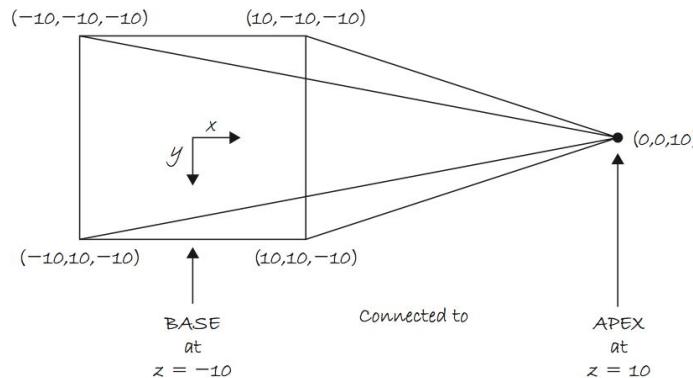
P3D—P3D is a 3D renderer developed by the creators of *Processing*. It should also be noted that anti-aliasing (enabled with the `smooth()` function) is not available with P3D.

OPENGL—OPENGL is a 3D renderer that employs hardware acceleration. If you have an OpenGL compatible graphics card installed on your computer (which is pretty much every computer), you can use this mode. Although at the time of the writing of this book, there are still a few, minor kinks to be worked out with this mode (you may find things look slightly different between P3D and OPENGL), it may prove exceptionally useful in terms of speed. If you are planning to display large numbers of shapes onscreen in a high-resolution window, this mode will likely have the best performance.

Custom shapes

[beginShape\(\)](#)
[vertex\(\)](#)
[endShape\(\)](#)

3D Coordinates



Rotation

`rotateX(), rotateY(), rotateZ()`
Set origin to center

```
// Translate origin to center  
translate(width/2, height/2);
```

[rectMode\(\)](#)

The default mode is `rectMode(CORNER)`, which interprets the first two parameters of `rect()` as the upper-left corner of the shape, while the third and fourth parameters are its width and height.

[Example: solar system](#)

```
// pushMatrix()  
// rotate()  
// translate origin  
// popMatrix()
```

[PShape](#)

```
// PShape shape
```

```
// shape = createShape()  
// shape.beginShape()  
// shape.vertex(x, y)  
// shape.endShape()
```

Chapter 15: Images and Pixels

Images

PImage

[image\(\)](#): The img parameter specifies the image to display and by default the a and b parameters define the location of its upper-left corner.

[imageMode\(\)](#)

[createImage\(\)](#): Creates a new PImage (the datatype for storing images). This provides a fresh buffer of pixels to play with. Set the size of the buffer with the width and height parameters

[random\(a, b\)](#): starting at a , and up to, but not including b

Pixels

[loadPixels\(\)](#): Loads a snapshot of the current display window into the **pixels[]** array. This function must always be called before reading from or writing to pixels[].

[updatePixels\(\)](#): Updates the display window with the data in the **pixels[]** array. Use in conjunction with loadPixels().

2-D Pixel Array (use **x + y * width**)

```
// Loop through every pixel column  
for (int x = 0; x < width; x++) {  
    // Loop through every pixel row  
    for (int y = 0; y < height; y++) {  
  
        // Use the formula to find the 1D location  
        int loc = x + y * width;  
  
        // If even column  
        if (x % 2 == 0) {  
            pixels[loc] = color(255);  
            // If odd column  
        } else {  
            pixels[loc] = color(0);
```

Image Processing

red(), **blue()**, **green()**, **hue()**, **saturation()**, **brightness()**, **alpha()**

[filter\(\)](#): Filters the display window using a preset filter or with a custom shader. Using a shader with filter() is much faster than without. Shaders require the P2D or P3D renderer in size().

Chapter 12: Libraries

Chapter 12: Libraries

Chapter 12: Libraries

Chapter 12: Libraries

The Nature of Code

[Html version](#)