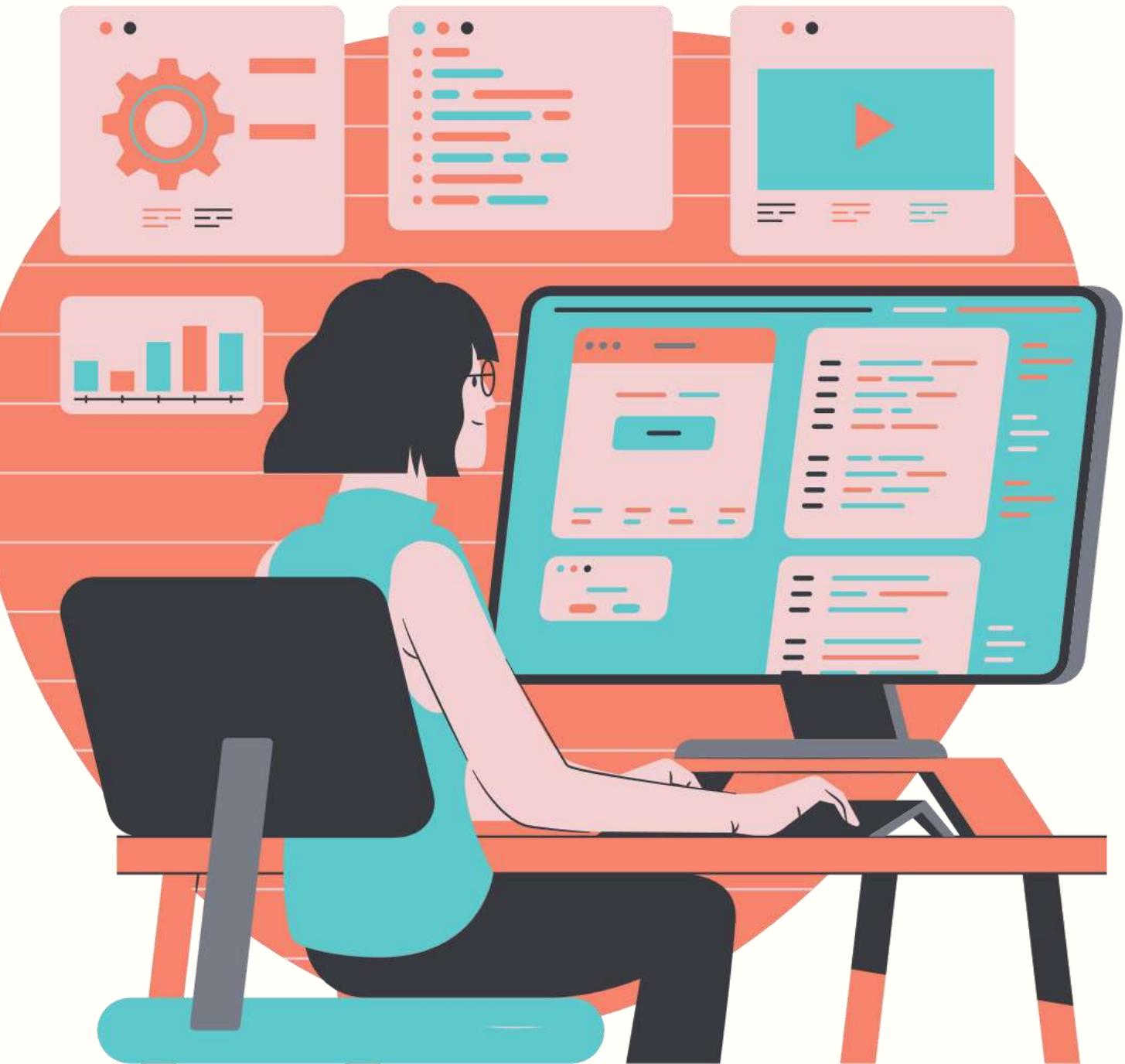


# Research About: vibe coding

**Created by:**  
**Ibrahim Qahtan**  
**Mustafa Musab**  
**Abdulazeez Qusay**  
**Kozhin Kamal**

**Supervisor:**  
**DR.Nawin Najat**





# TABLE OF CONTENTS

01

Introduction

02

Importance

03

Coding Style  
Conventions

04

Principles of Clean  
Code and Aesthetics

05

Subjectivity, Team  
Culture, and Tools

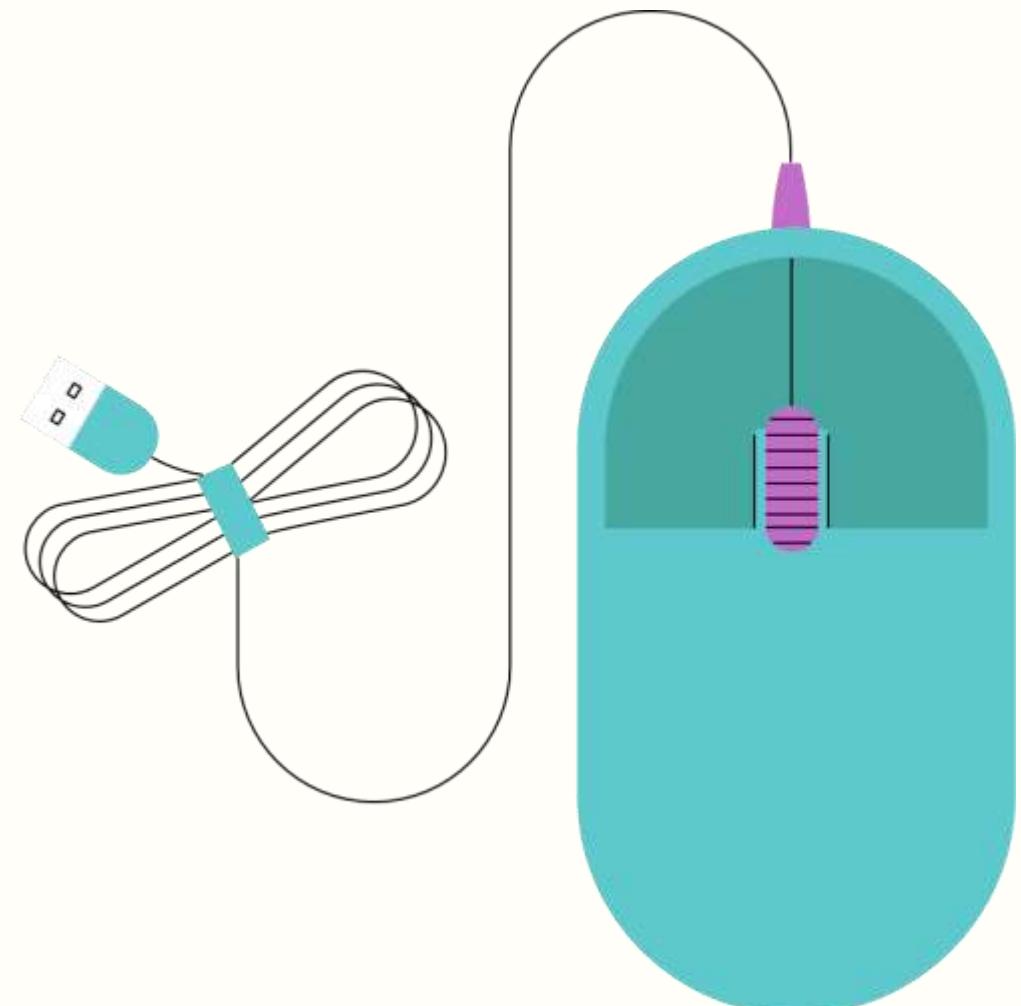




01

# INTRODUCTION

Vibe coding is an AI-assisted programming method using natural language prompts for code generation. Popularized by Andrej Karpathy in February 2025, highlighting the role of AI tools like ChatGPT. Focuses on expressing intention in plain language, allowing AI to generate executable code. Aims to free developers to concentrate on creativity and problem-solving. Interpretations include "code first, refine later," intent-based specification, and an evolution of low-code platforms.





# 02

## The Importance of Code Fundamentals

**Code Readability:** Essential for understanding and modifying code, accelerating feature addition and bug fixing. Reduces the learning curve for new team members and enhances collaboration. Crucial for maintainability, stability, and security of software. Poor readability can be indicated by longer bug fix times and onboarding times.

**Software Maintainability:** Ease with which code can be understood, corrected, adapted, and enhanced. Key principles include readability, modularity, documentation, consistent coding standards, and thorough testing. Benefits include software longevity, simplified updates, reduced technical debt, and improved collaboration.



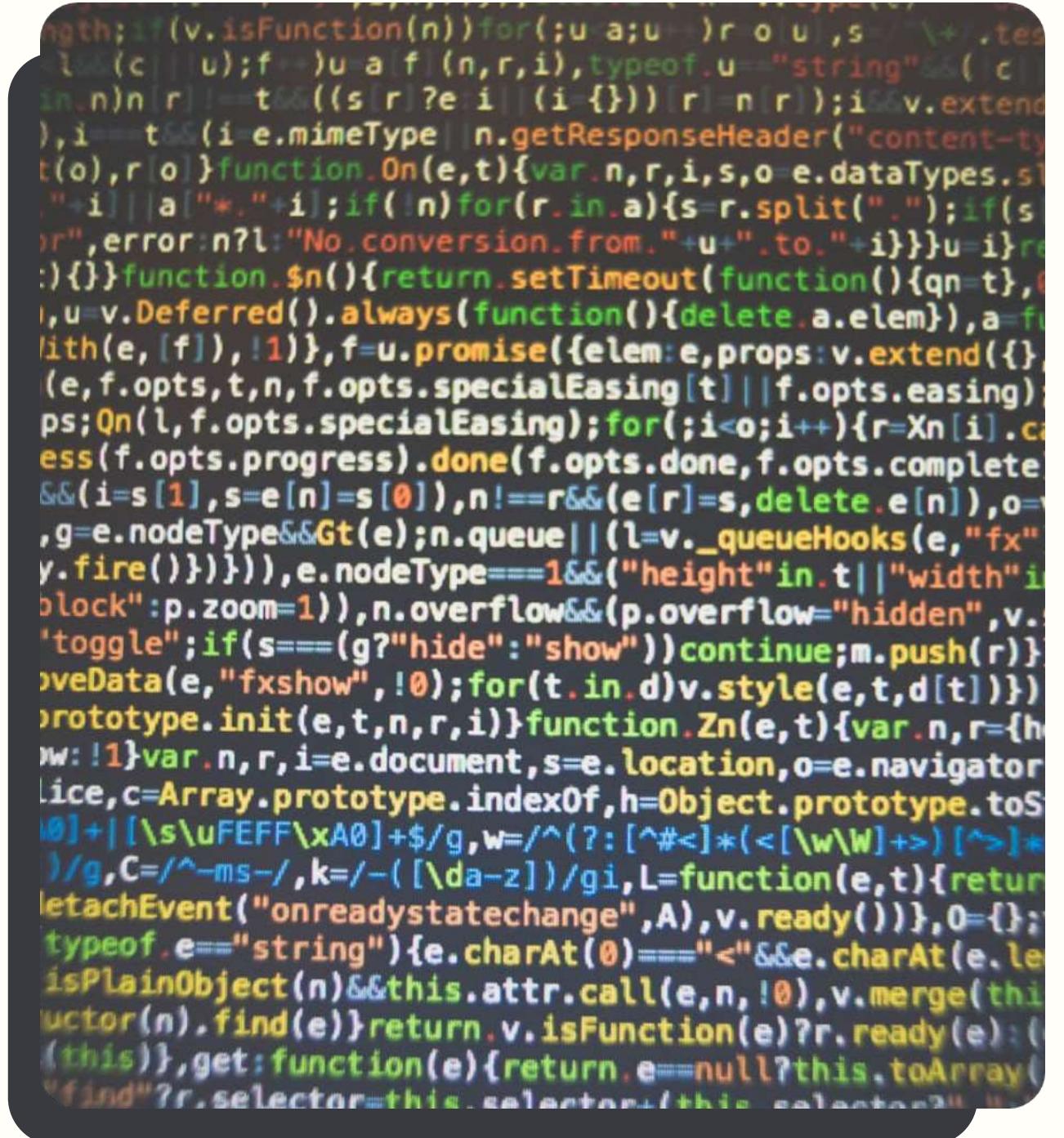
## 03

# Coding Style Conventions

**Python (PEP 8):** Emphasizes readability with 4-space indentation, line length limits, blank lines, and consistent naming conventions (lowercase, snake\_case, CamelCase).

**Java:** Common conventions include 3-4 space indentation, specific brace placement, whitespace usage, CamelCase naming, and Javadoc style comments for documentation.

**JavaScript (Standard Style):** Recommends 2-space indentation, single quotes for strings, no unused variables, space after keywords, and camel case naming; enforces rules for equality, commas, and more.



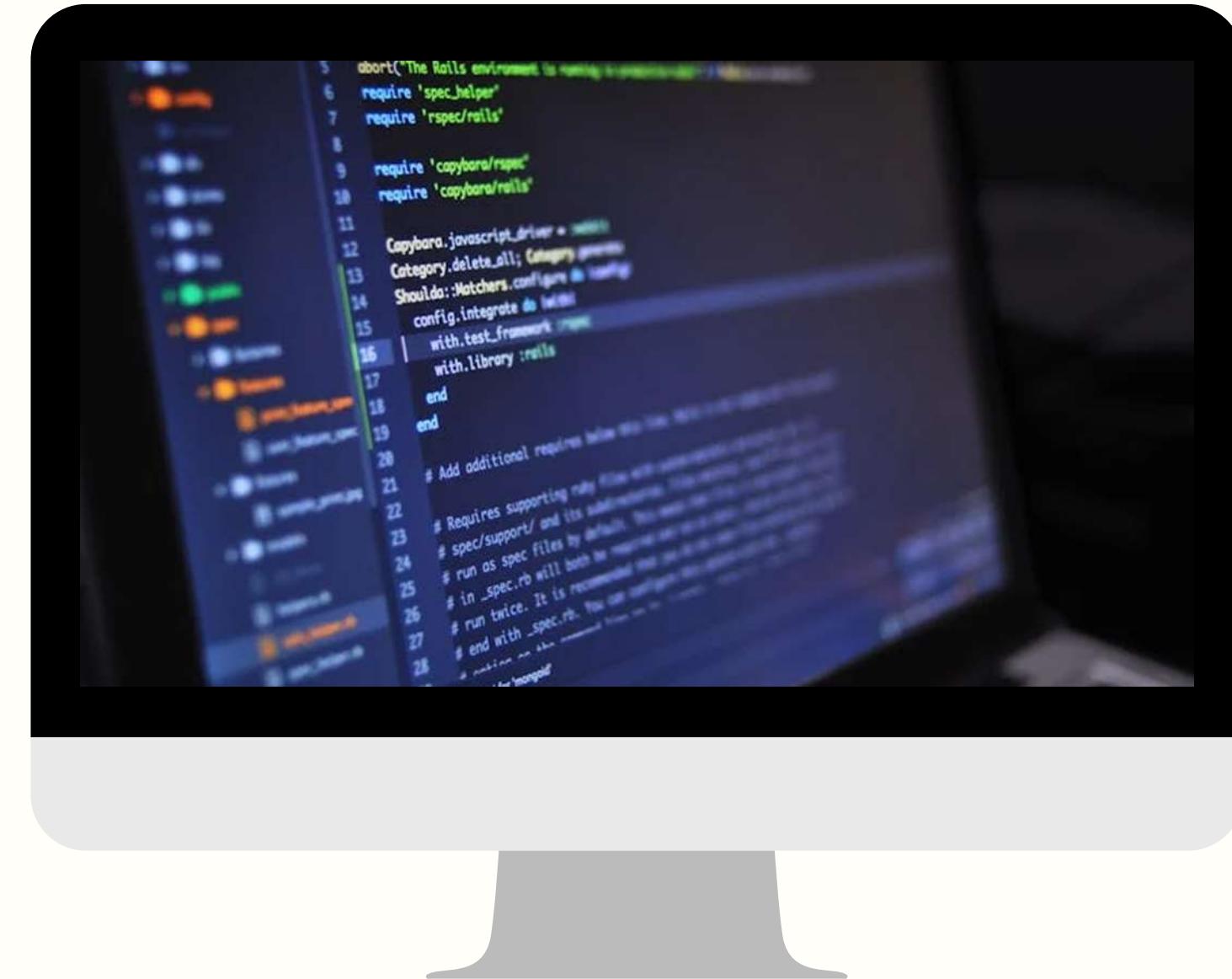
```
ngth;if(v.isFunction(n))for(;u.a;u++)r.o[u],s=\+.te  
l(c|u);f+=u.a.f(n,r,i),typeof.u=="string"&&(c  
in.n)n.r=t&((s|r?e|i(i-{}))|r=n(r));i56v.extend  
,i=t&(i.e.mimeType n.getResponseHeader("content-t  
t(o),r)o}function.On(e,t){var n,r,i,s,o=e.dataTypes.s  
"i||a["*,"+i];if(!n)for(r.in.a){s=r.split(".");if(s  
or",error:n?l:"No.conversion.from."+u+".to."+i})}u=i}re  
:(){}}function.$n(){return.setTimeout(function(){qn=t},  
,u=v.Deferred().always(function(){delete.a.elem}),a=f  
ith(e,[f]),!1)},f=u.promise({elem:e,props:v.extend({}  
,(e,f.opts,t,n,f.opts.specialEasing[t]||f.opts.easing)  
ps;Qn(l,f.opts.specialEasing);for(;i<o;i++){r=Xn[i].c  
ess(f.opts.progress).done(f.opts.done,f.opts.complete  
&&(i=s[1],s=e[n]=s[0]),n!=r&&(e[r]=s,delete.e[n]),o=  
,g=e.nodeType&&Gt(e);n.queue||(l=v._queueHooks(e,"fx"  
y.fire()))}),e.nodeType==1&&("height"in.t||"width"in  
block":p.zoom=1)),n.overflow&&(p.overflow="hidden",v.  
"toggle";if(s==(g?"hide":"show"))continue;m.push(r)  
oveData(e,"fxshow",!0);for(t.in.d)v.style(e,t,d[t]))  
prototype.init(e,t,n,r,i)}function.Zn(e,t){var n,r={h  
w:!1}var n,r,i=e.document,s=e.location,o=e.navigator  
ice,c=Array.prototype.indexOf,h=Object.prototype.toS  
@]+|[\\s\\uFEFF\\xA0]+$/g,w=/^?:[^#<]*(<[\\w\\W]+>)[^>]*  
)/g,C=/^~ms-/i,k=-/(\\da-z)/gi,L=function(e,t){retur  
etachEvent("onreadystatechange",A),v.ready()),0={};  
typeof.e=="string"){e.charAt(0)<"&&e.charAt(e.le  
isPlainObject(n)&&this.attr.call(e,n,!0),v.merge(th  
uctor(n).find(e)}return visFunction(e)?r.ready(e):  
(this)},get:function(e){return e==null?this.toArray()  
"find"?r.selector=this.selector+(this.selector?"/":  
"/>
```

# 04

## Principles of Clean Code and Aesthetics

**Clean Code:** Focuses on code that is easy to read, understand, and modify, avoiding complexity and redundancy. Principles include meaningful names, short and focused functions (Single Responsibility Principle), judicious comments, consistent formatting, DRY (Don't Repeat Yourself), and robust error handling. Practical applications include using named constants, meaningful naming, short functions, and refactoring for reusability.

**Code Aesthetics:** Visual aspects like indentation, spacing, and naming conventions that impact developer experience. Aesthetically pleasing code improves readability, reduces cognitive load, and increases productivity and job satisfaction. Consistent aesthetics enhance team collaboration, facilitate code reviews, and minimize merge conflicts.

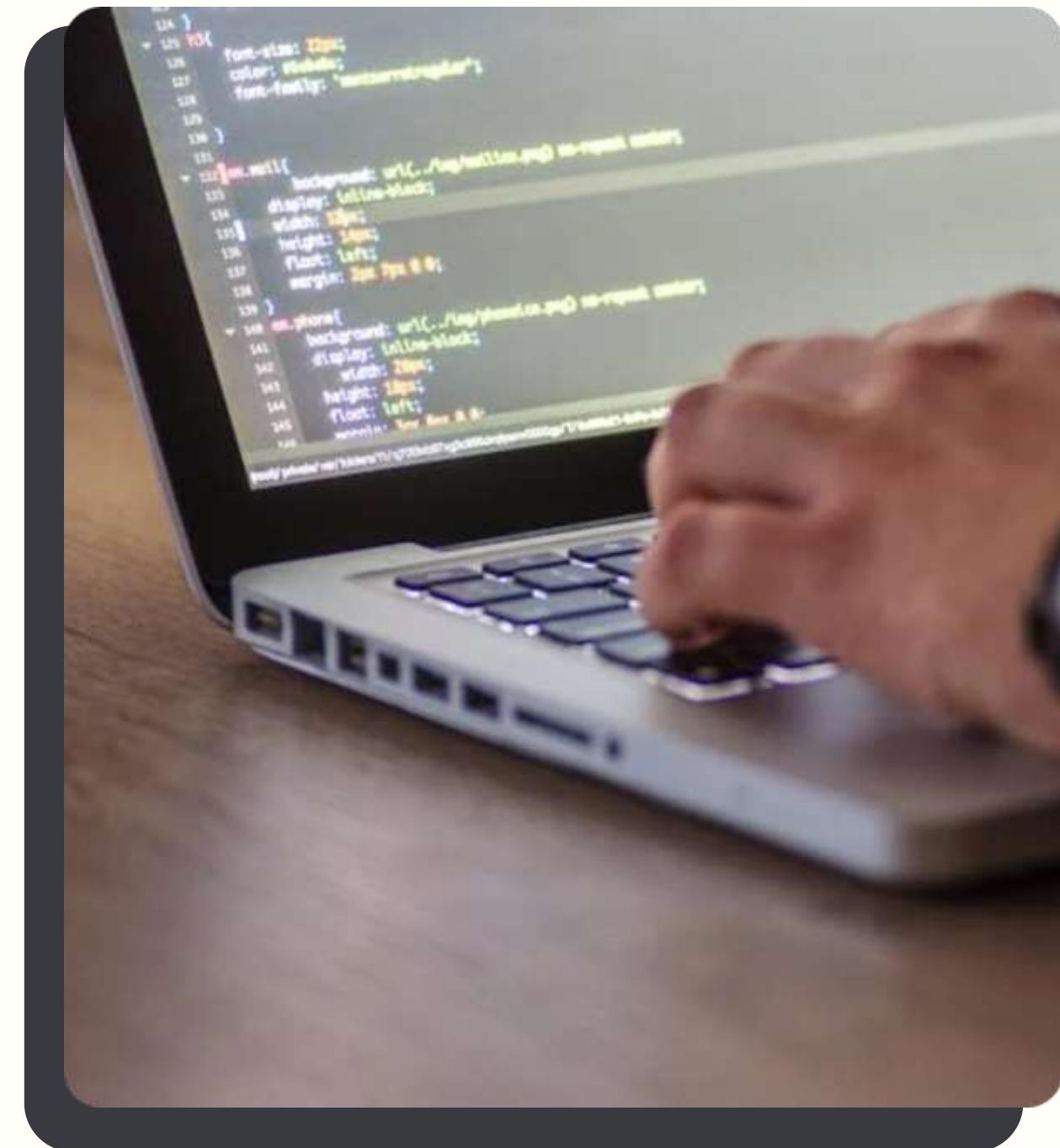


# 05

## Subjectivity, Team Culture, and Tools

**Emotional and Subjective Aspects of Coding:** Coding involves emotions like curiosity, frustration, and satisfaction; subjective interpretations of code exist. Satisfaction from writing good code motivates developers; helping others increases satisfaction.

**Personal Preferences:** Influence coding style in structuring functions, naming, error handling, and formatting. **Team Culture:** Shapes coding standards through shared values, communication, continuous learning, emotional intelligence, and inclusion.





# RESOURCE PAGE

1.What is Vibe Coding? From IBM

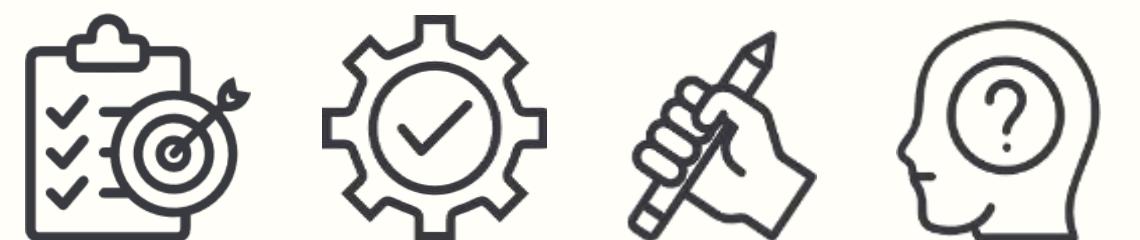
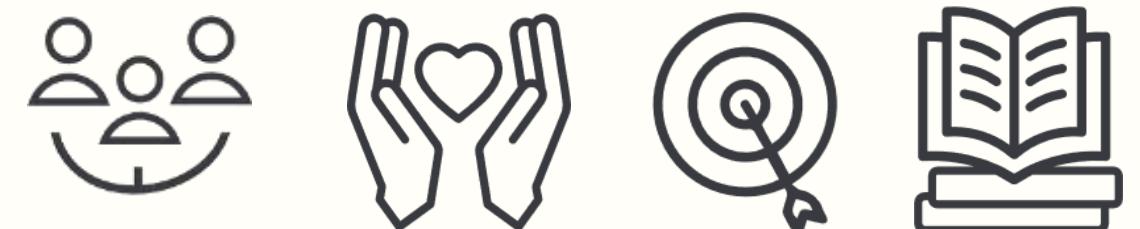
<https://www.ibm.com/think/topics/vibe-coding>

2.blog.replit.com

<https://blog.replit.com/what-is-vibe-coding#:~:text=Vibe%20coding%20refers%20to%20the,getting%20stuck%20in%20technical%20details.>

3.tiiny.host

<https://tiiny.host/blog/vibe-coding/#:~:text=Vibe%20coding%20represents%20a%20shift,the%20AI%20handles%20code%20generation.>



# THANK YOU!

