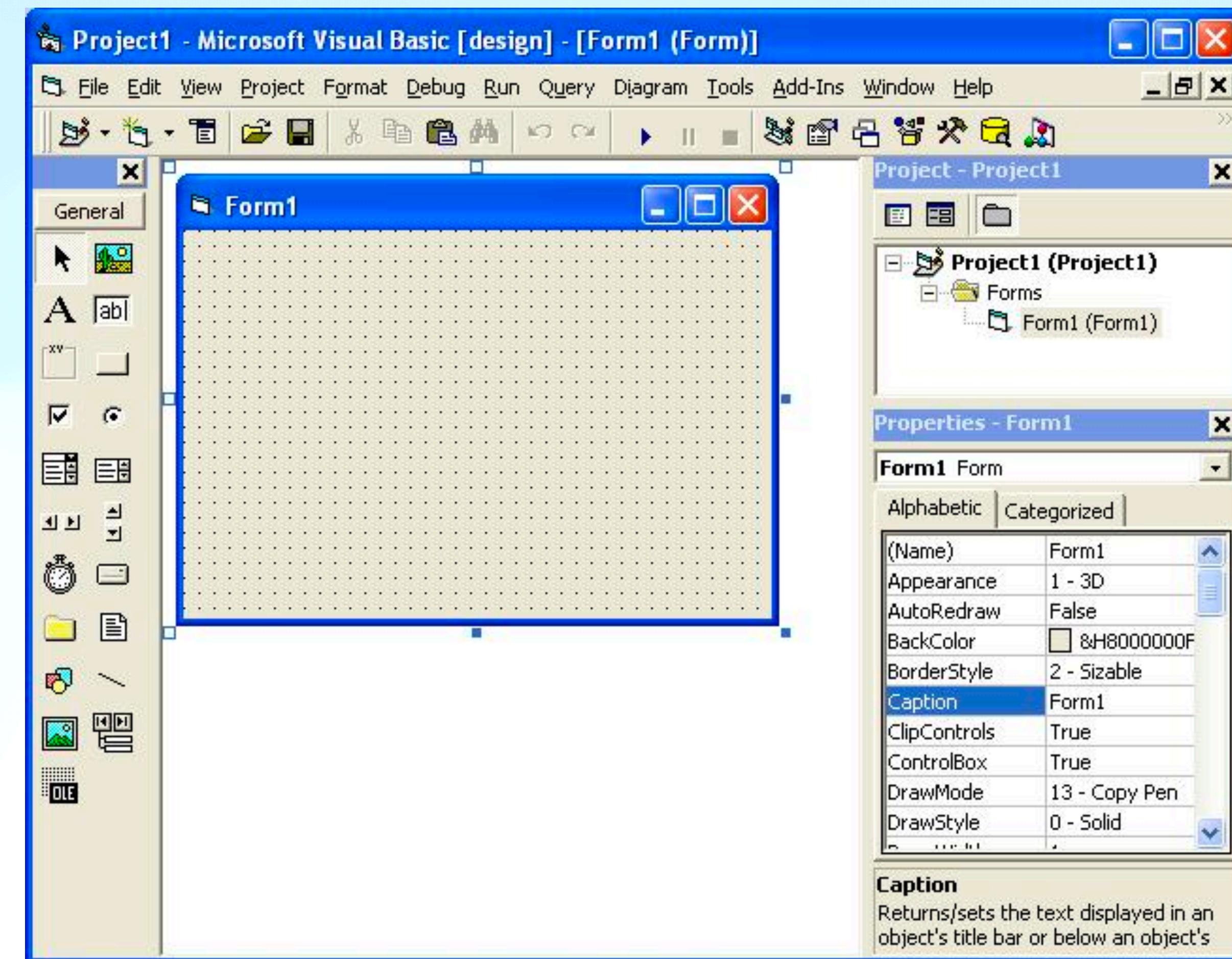


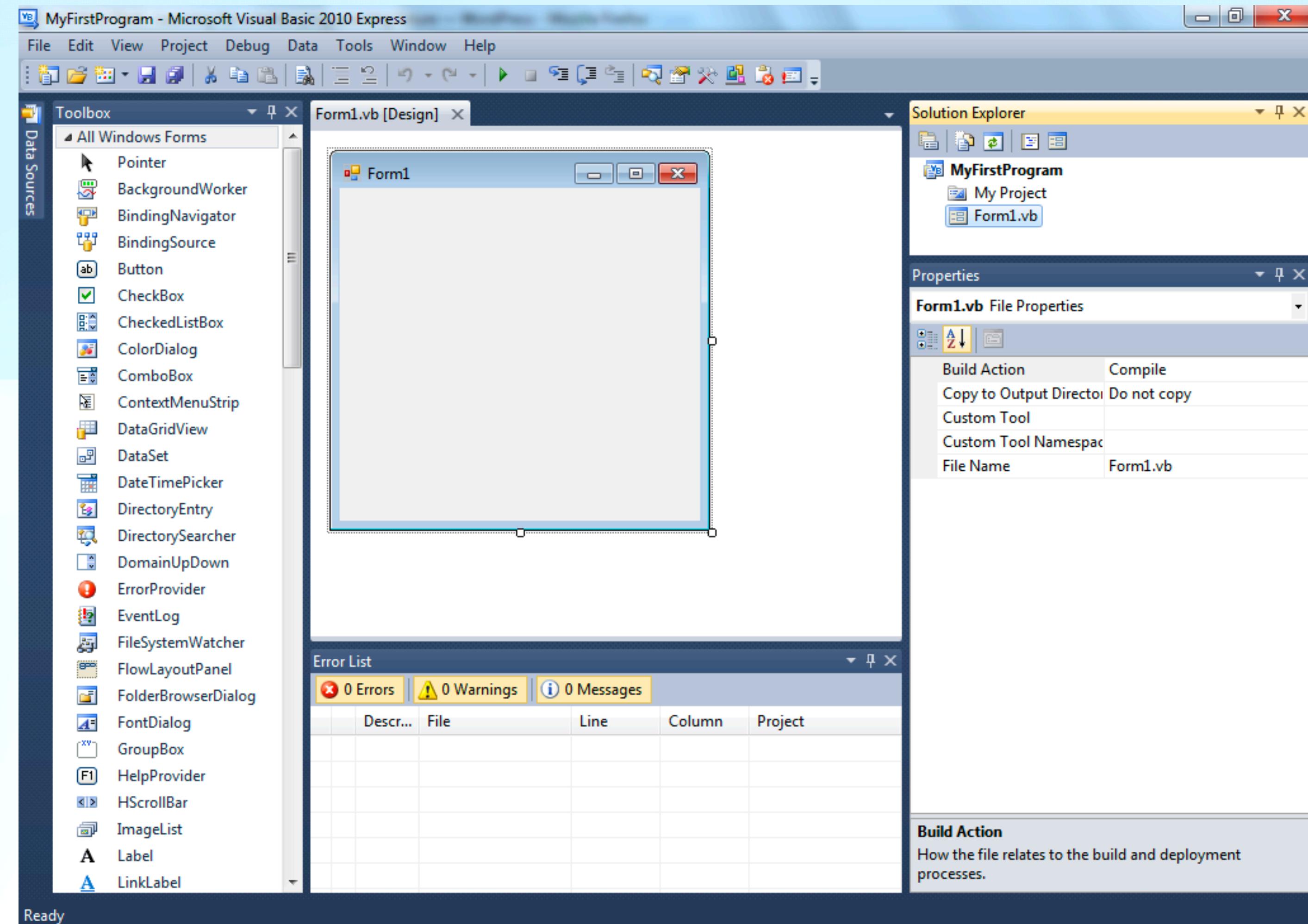
My programming experience

Microsoft Visual Basic 6.0 (VB6) - 1380 - 1382



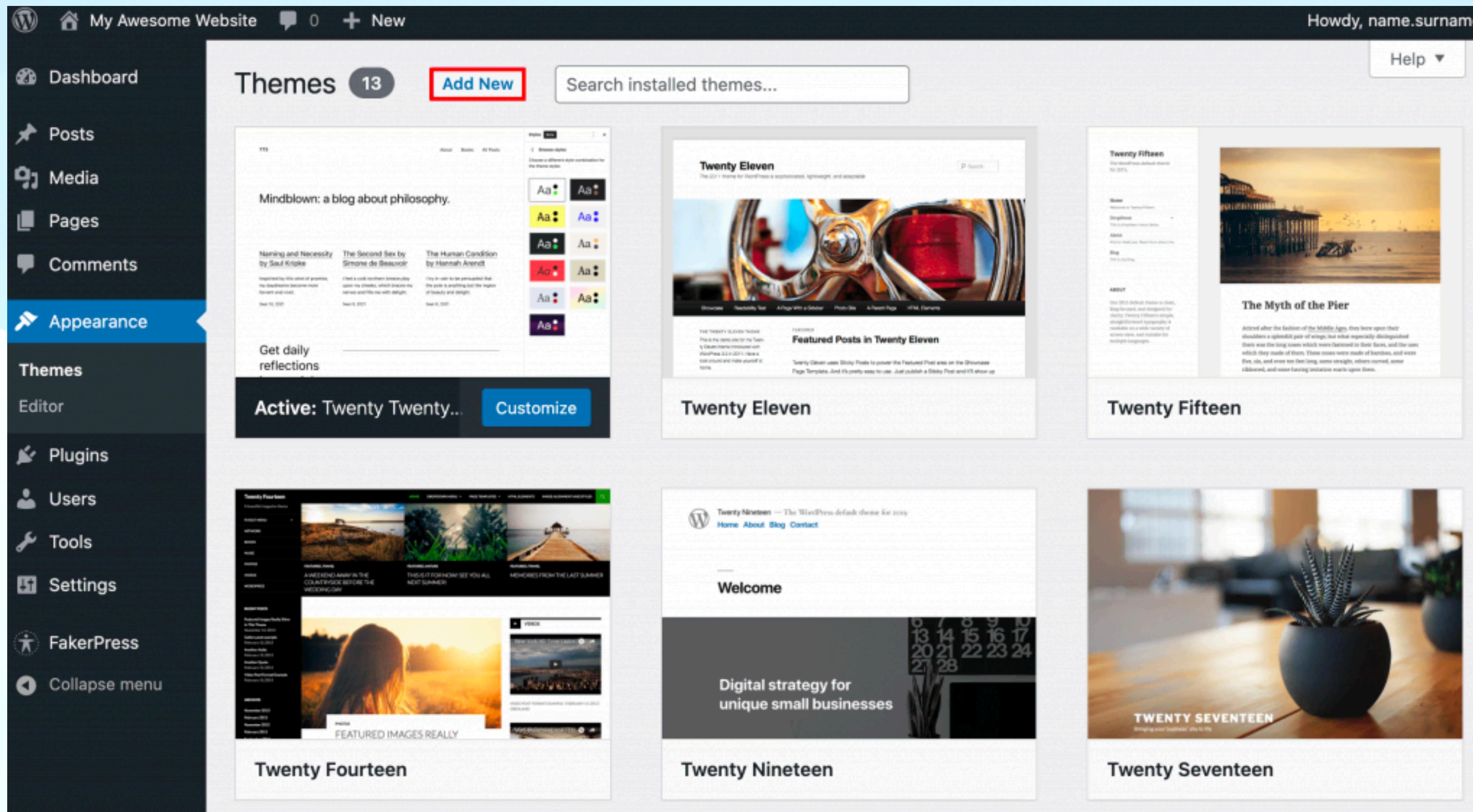
My programming experience

Visual Studio .net , VB, C#, J#



My programming experience

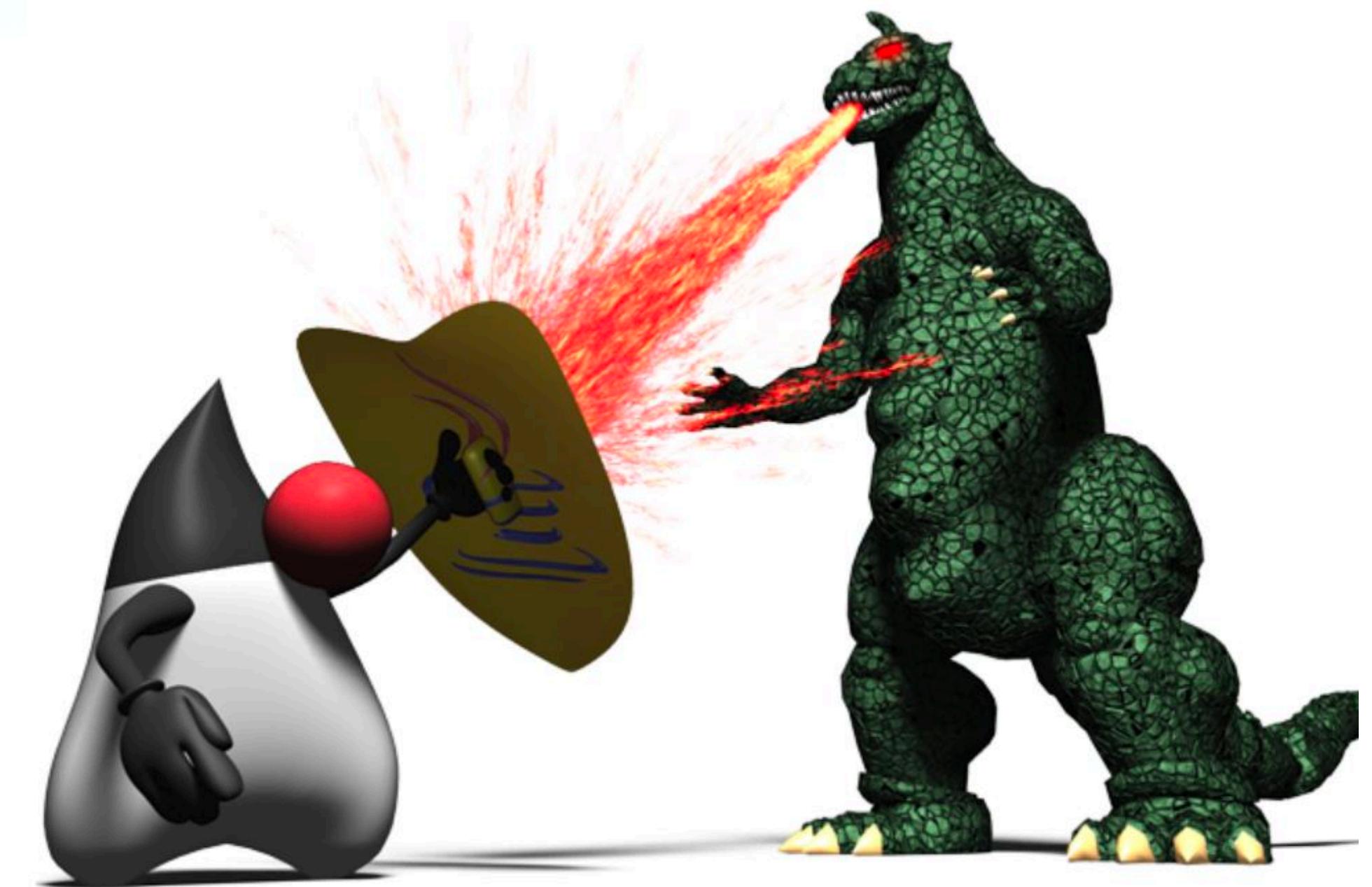
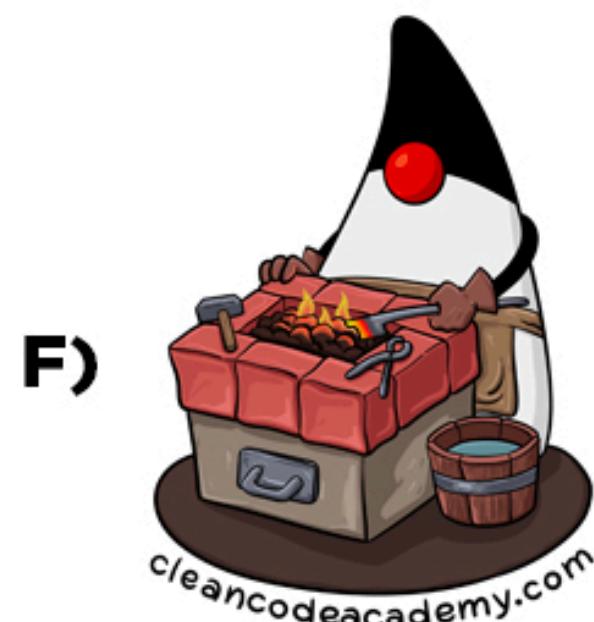
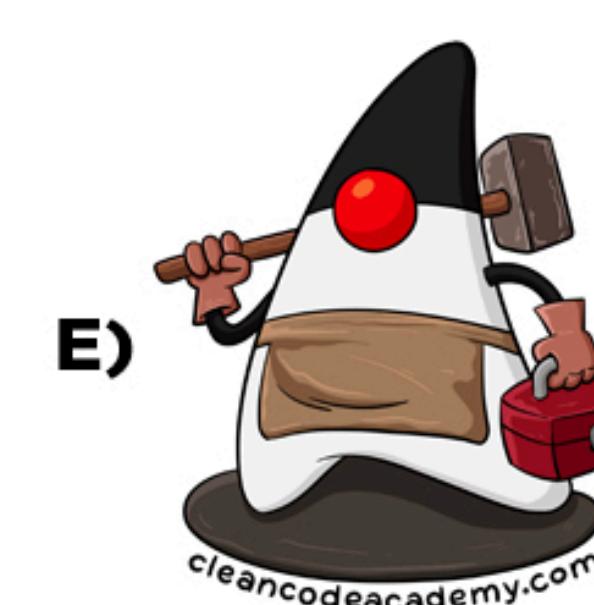
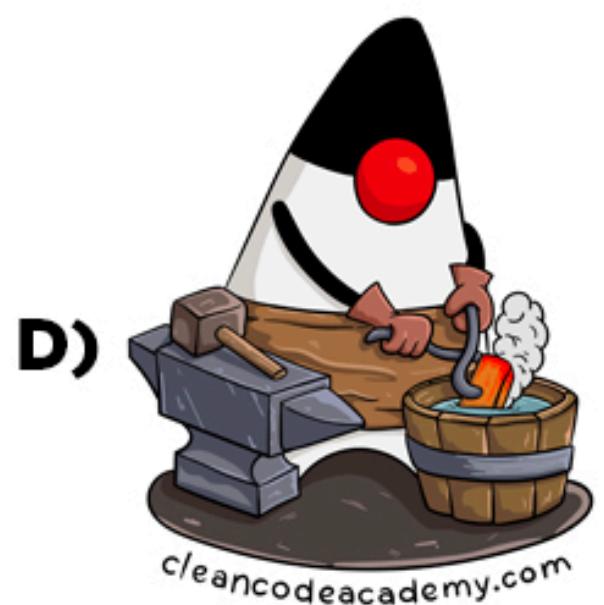
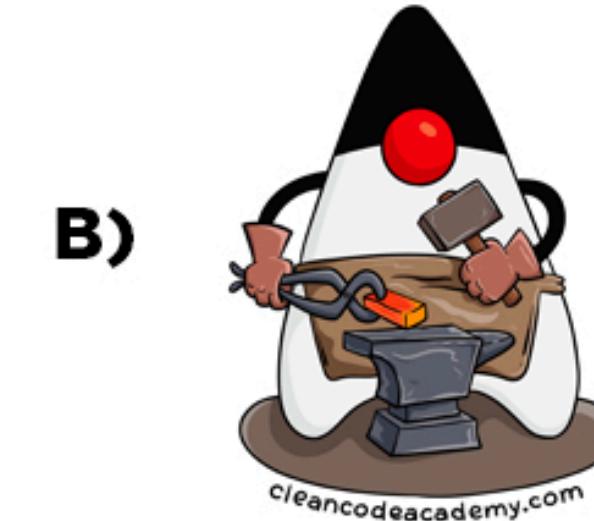
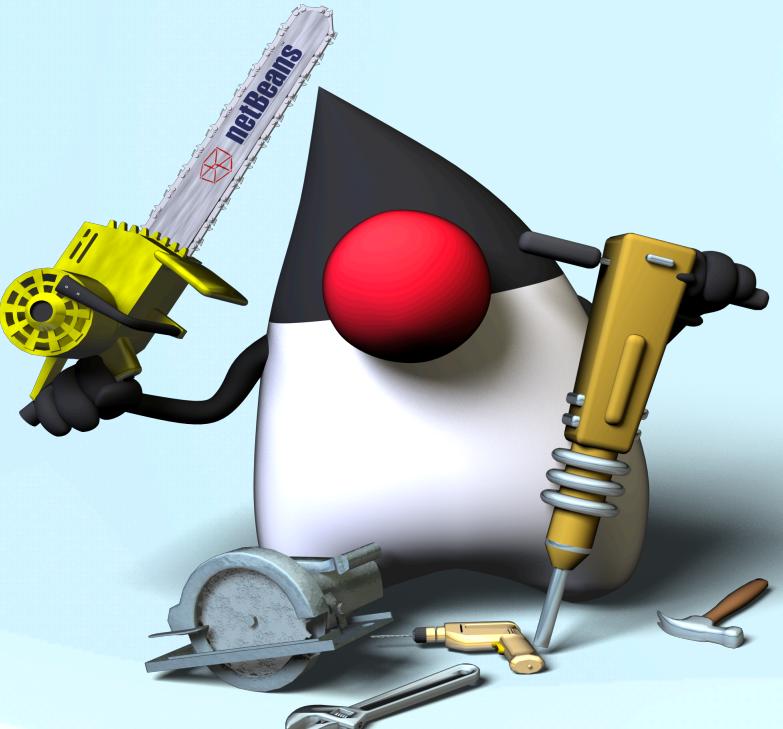
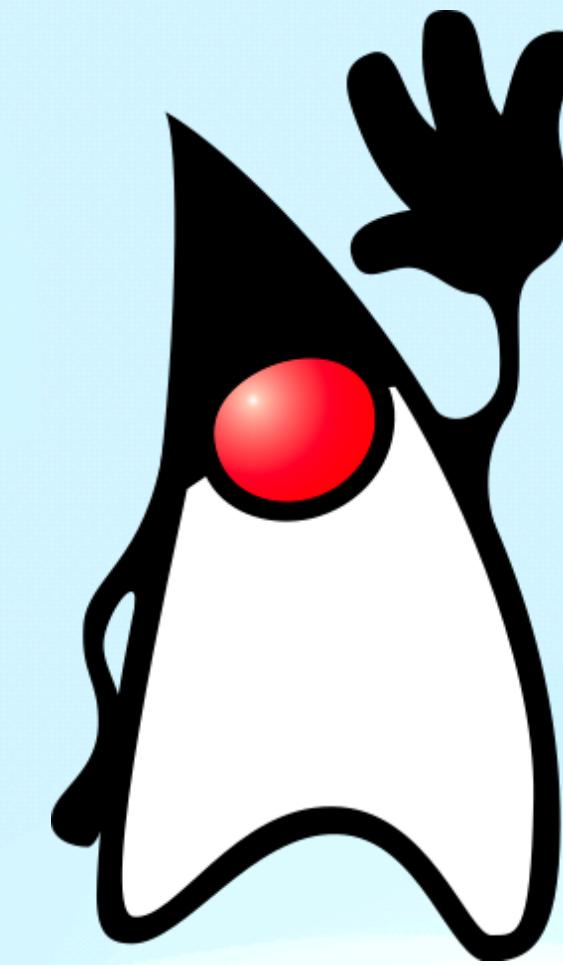
PHP Wordpress



My programming experience

JAVA

<https://www.tiobe.com/tiobe-index/>



Programming language history

Sample 2 + 5

Machine code

	ARM	X86
X = 2	E3A00002	b8 02 00 00 00
X = X + 5	E2800005	83 c0 05

Programming language history

Sample 2 + 5

Assembly

	ARM	X86
X = 2	E3A00002	b8 02 00 00 00
X = X + 5	E2800005	83 c0 05

Assembly (ARM)	Assembly (X86)
mov r0, #2	mov eax, 2
add r0, r0, #5	add eax, 5

Programming language history

Assembler

Assembly (ARM)

```
mov r0, #2  
add r0, r0, #5
```

ARM Assembler

ARM

```
E3A00002  
E2800005
```

ARM

```
2+5=7
```

Assembly (X86)

```
mov eax, 2  
add eax, 5
```

X86 Assembler

X86

```
b8 02 00 00 00  
83 c0 05
```

X86

```
2+5=7
```

Programming language history

Sample 2 + 5

C

	ARM	X86
X = 2	E3A00002	b8 02 00 00 00
X = X + 5	E2800005	83 c0 05

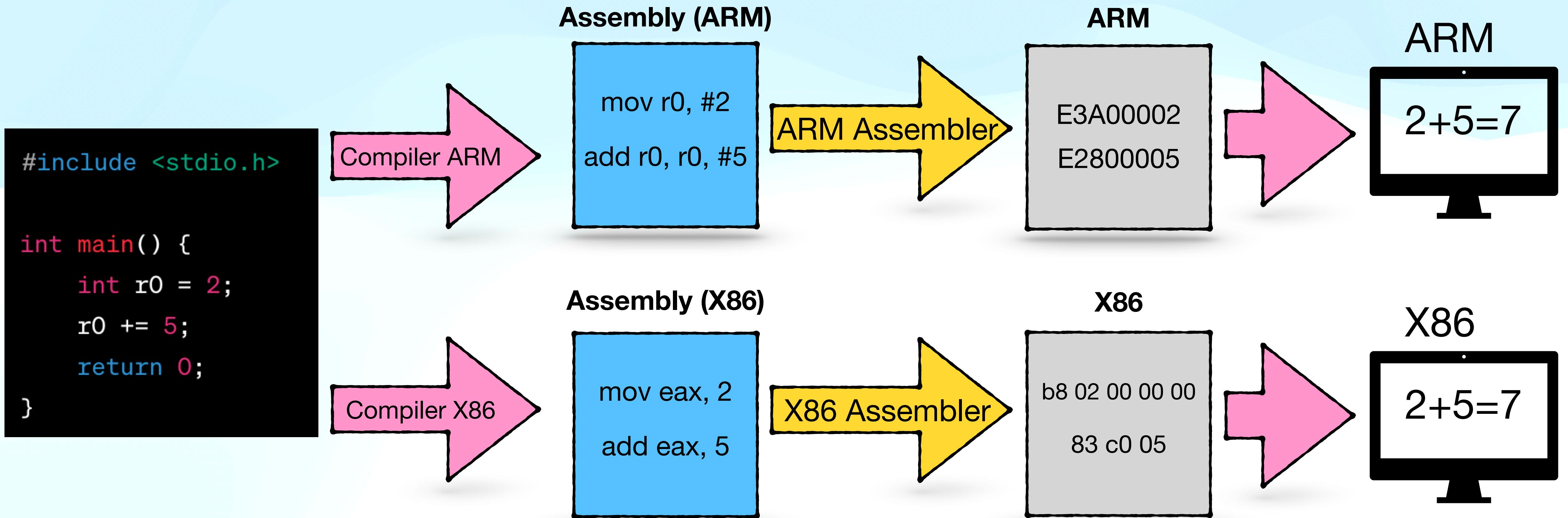
Assembly (ARM)	Assembly (X86)
mov r0, #2	mov eax, 2
add r0, r0, #5	add eax, 5

```
#include <stdio.h>

int main() {
    int r0 = 2;
    r0 += 5;
    return 0;
}
```

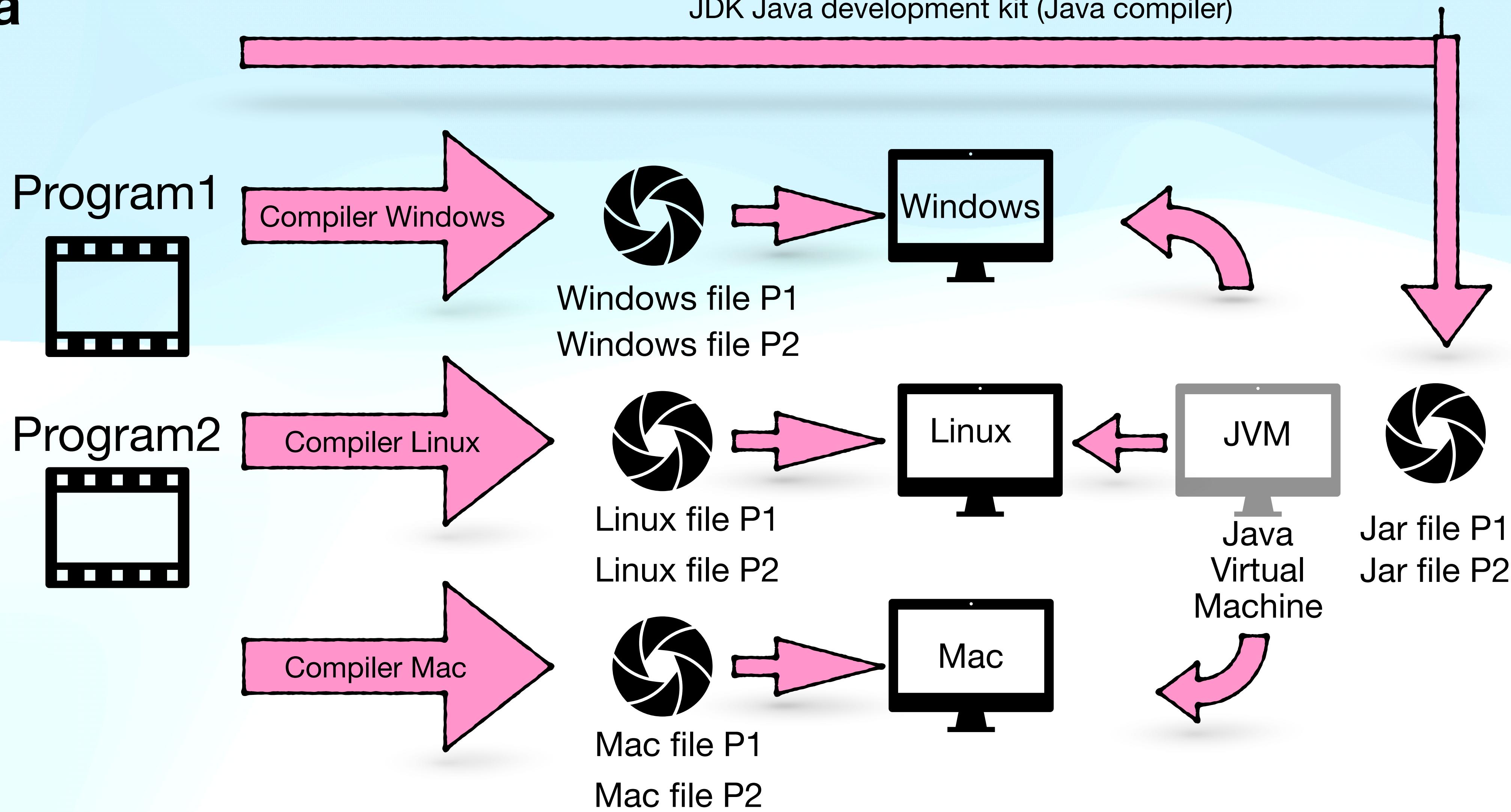
Programming language history

Compiler



Programming language history

Java

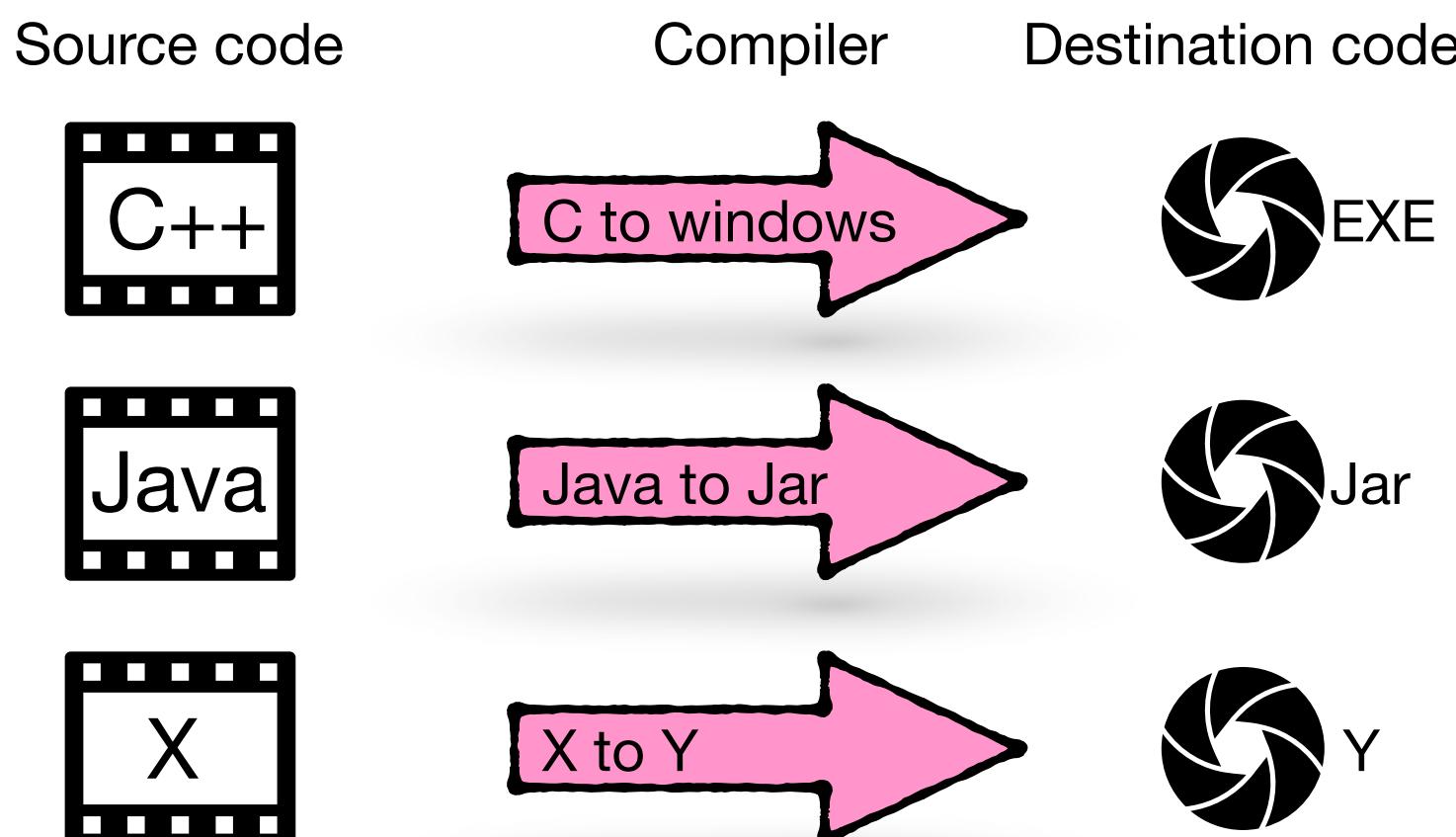


Programming language history

Compiler

What is a fundamental function of a compiler in programming?

1. Converts programming code to machine language.
2. Transforms programming code to assembly language.
3. Translates source code from one programming language to another.
4. Run or execute a programming code on a machine.



Programming language history

JVM JRE JDK

What is the difference between JDK, JRE, and JVM?

JVM Java Virtual Machine

It is an abstract machine which provides the runtime environment in which Java bytecode can be executed.

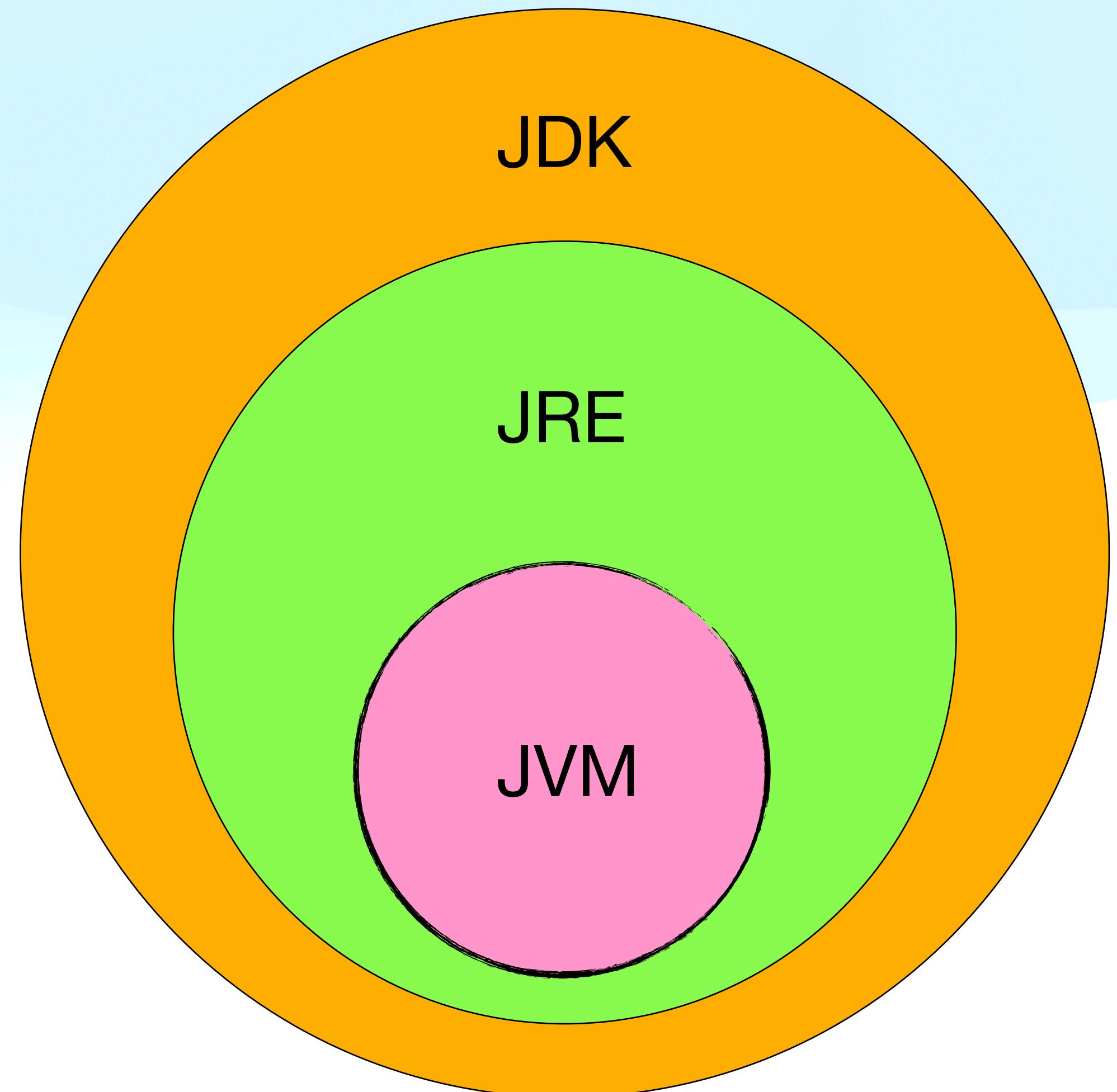
Its implementation is known as JRE.

JRE Java Runtime Environment

It is the implementation of JVM.
It physically exists.
It contains a set of libraries that JVM uses at runtime.

JDK Java Development Kit

It is a Java development environment.
It physically exists.
It contains JRE + development tools.



Programming language history

Linkedin skill assessment java quiz

The _ runs compiled Java code, while the _ compiles Java files.

1. IDE; JRE

2. JDK; IDE

3. JRE; JDK

4. JDK; JRE



Programming styles

Procedural Programming (Line-by-Line Execution)

Example Language: BASIC

In BASIC, the program is executed sequentially, line by line.

Each line typically represents a single statement or command.

basic

```
10 PRINT "Enter your name: ";
20 INPUT NAME$
30 PRINT "Hello, "; NAME$
40 END
```

Programming styles

Introducing GOTO for Control Flow

Example Language: Assembly

Assembly languages often involve explicit control flow using GOTO or jump instructions.

GOTO allows the program to transfer control to a different part of the code based on conditions.

```
assembly

    MOV AX, 5
    CMP AX, 0
    JL LessThanZero
    ; Code to execute if greater than or equal to zero
    JMP EndProgram
LessThanZero:
    ; Code to execute if less than zero
EndProgram:
```

Programming styles

Functional Programming

Example Language: Javascript

Functional programming treats computation as the evaluation of mathematical functions.

Each Function responsible for an action and the program can be a set of actions.

```
javascript

// Function to calculate the square of a number
const square = (x) => x * x;

// Function to calculate the sum of two numbers
const add = (x, y) => x + y;

// Function to compose the square and add functions
const compose = (f, g) => (x) => f(g(x));

// Example usage
const result = compose(square, add)(2, 3);

console.log("Result:", result);
```

Programming styles

Object-Oriented Programming (OOP)

Example Language: Java

Object-oriented programming organizes code around objects.

Abstraction, encapsulation, inheritance, and polymorphism

```
// OOP Classes
class Animal {
    String name;

    Animal(String name) {
        this.name = name;
    }
}
```

```
class Dog extends Animal {
    Dog(String name) {
        super(name);
    }

    String speak() {
        return "Woof!";
    }
}
```

```
public class Example {
    public static void main(String[] args) {
        // OOP Style
        Animal myDog = new Dog("Buddy");
        System.out.println(myDog.speak()); // Output: Woof!
    }
}
```

Programming language history

Linkedin skill assessment java quiz

Object-oriented programming is a style of programming where you organise your program around _____ and data, rather than _____ and logic.

1. functions; actions 

2. objects; actions 

3. actions; functions 

4. actions; objects 

Java history

Algol (ALGOrithmic Language)

When: Developed in the late 1950s.

Why: Algol was created to be a universal algorithmic language.

It aimed to provide a clear, precise, and well-defined language for expressing algorithms

Independent of a specific machine.

```
algol
BEGIN
    INTEGER a, b, sum;

    a := 5;
    b := 7;

    sum := a + b;

    OUTPUT("The sum is: ", sum);
END
```

Java history

Pascal

When: Developed by Niklaus Wirth in the late 1960s.

Why: Pascal was designed as a small and efficient language for:
teaching programming and structured programming concepts.

```
pascal  
  
program HelloWorld;  
begin  
    writeln('Hello, World!');  
end.
```

Java history

C

When: Developed by Dennis Ritchie at Bell Labs in the early 1970s.

Why: C was created to develop the UNIX operating system.

It aimed to provide a low-level programming language with features

allowed for efficient system programming.

```
c

#include <stdio.h>

int main() {
    printf("Hello, World!\n");
    return 0;
}
```

Java history

C++

When: Developed by Bjarne Stroustrup in the early 1980s.

Why: C++ was created as an extension of the C programming language.

It introduced the concept of “classes” for object-oriented programming (OOP).

```
cpp Copy code

#include <iostream>
#include <string>

class Person {
private:
    std::string name;
    int age;

public:
    Person(std::string n, int a) : name(std::move(n)), age(a) {}

    void displayDetails() const {
        std::cout << "Name: " << name << ", Age: " << age << '\n';
    }
};

int main() {
    Person person1("John Doe", 25);
    person1.displayDetails();
    return 0;
}
```

Java history

Java

When: Developed by James Gosling and his team at Sun Microsystems in the early 1990s..

Why: Java was created to address the challenges of programming in the networked environment of the time (early Internet).

The goal was to design a language that would be platform-independent, secure, and easy to use.

Java incorporated key features from C++ but aimed to eliminate certain issues.

```
java

public class HelloWorld {
    public static void main(String[] args) {
        System.out.println("Hello, World!");
    }
}
```

Java history

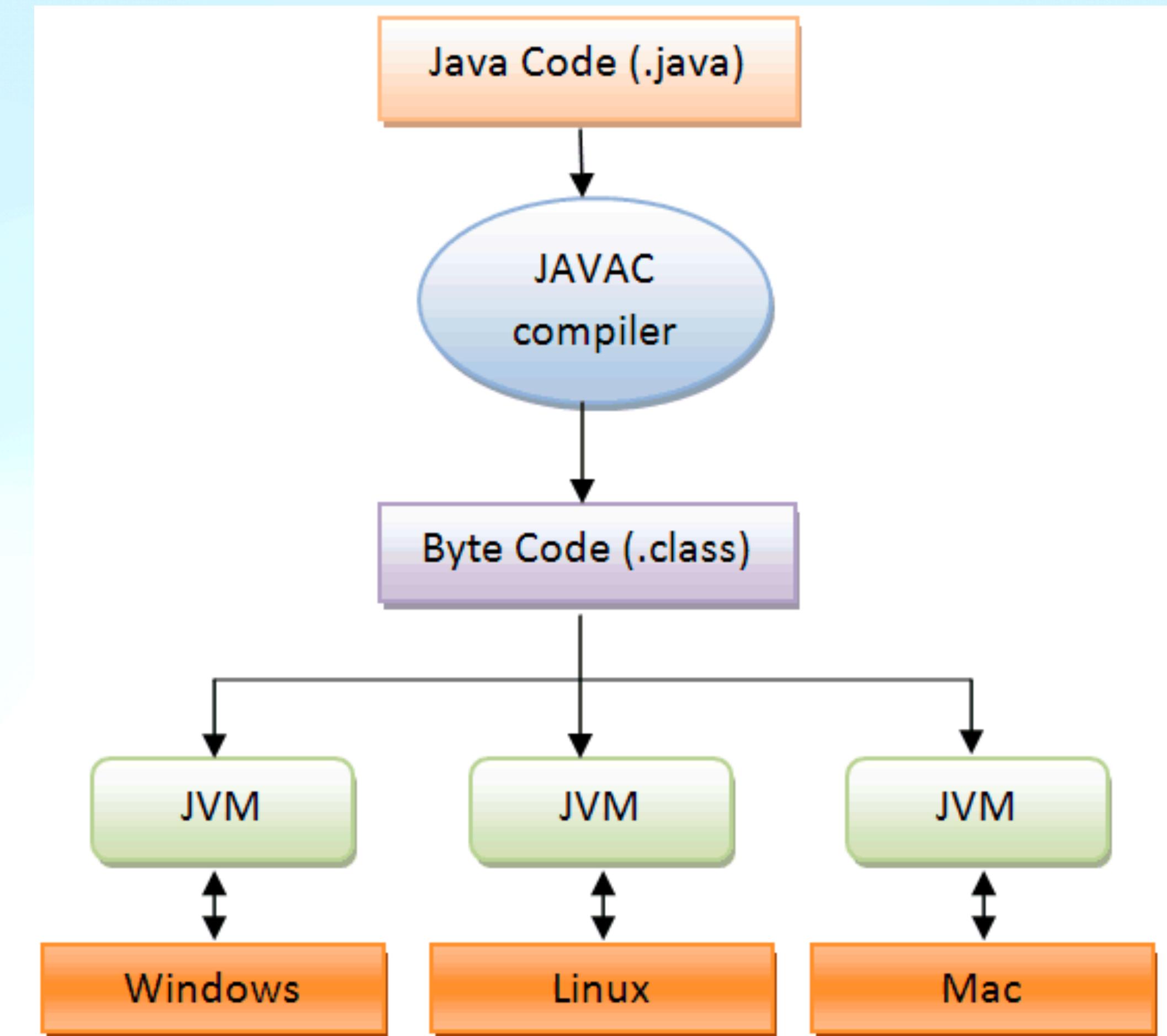
Java vs C++

Ariane 5 On June 4th, 1996

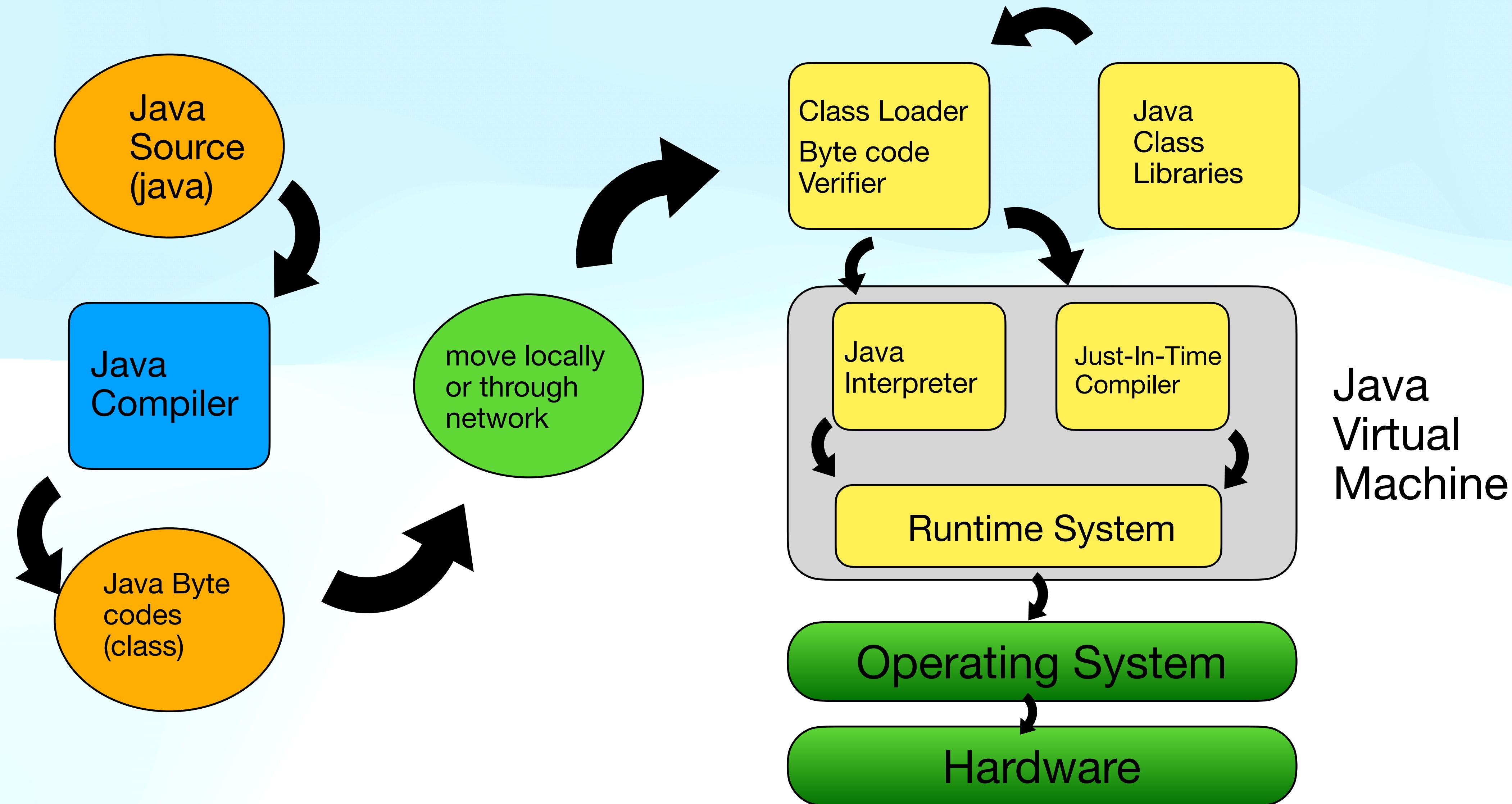
Comparison Index	C++	Java
Platform-independent	platform-dependent	platform-independent
Mainly used for	system programming	web-based, enterprise and mobile
Goto	supports the goto	doesn't support the goto
Pointers	supports pointers	supports internally. You can't write the pointer program.

Java

JAVA

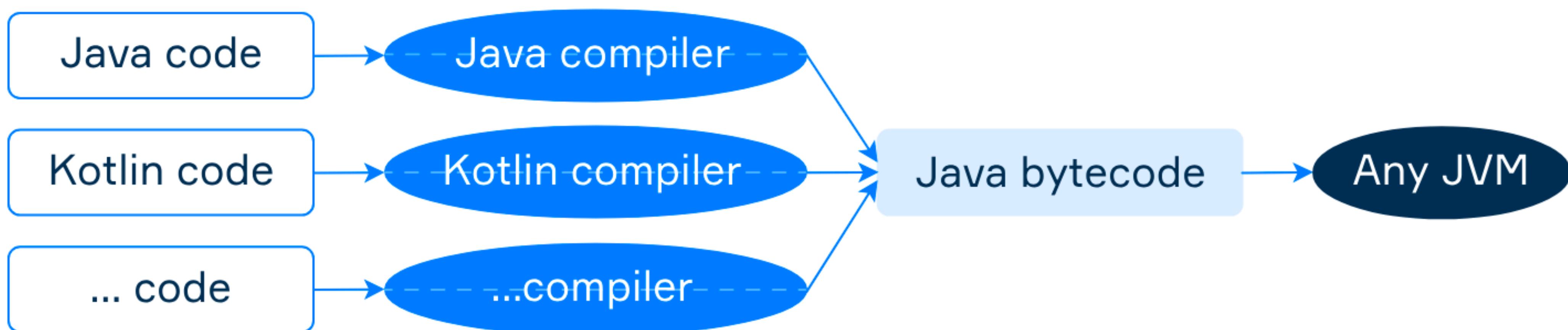


Java Development and Runtime Environment



Java

JAVA Kotlin



Java

JAVA JVM

