SOFTWARE DESIGN TECHNIQUES (CSCN72040)

Week-1

Introduction to Software Design

Petros Spachos

OUTLINE

Introduction

Explain why the software should be designed

Reviewing Object-Oriented Programming

What are Software Design Principles?

What are Software Design Patterns?

Benefits of Learning Design Patterns

Which Programming Language Will We Use in This Course?

Summary and Conclusion

INTRODUCTION

INTRODUCTION

WHY THE SOFTWARE SHOULD BE DESIGNED

Any one can build a software application that can last a month (short term),but

- Rigid ☹
- Fragile ☹
- Immobile ☹

INTRODUCTION

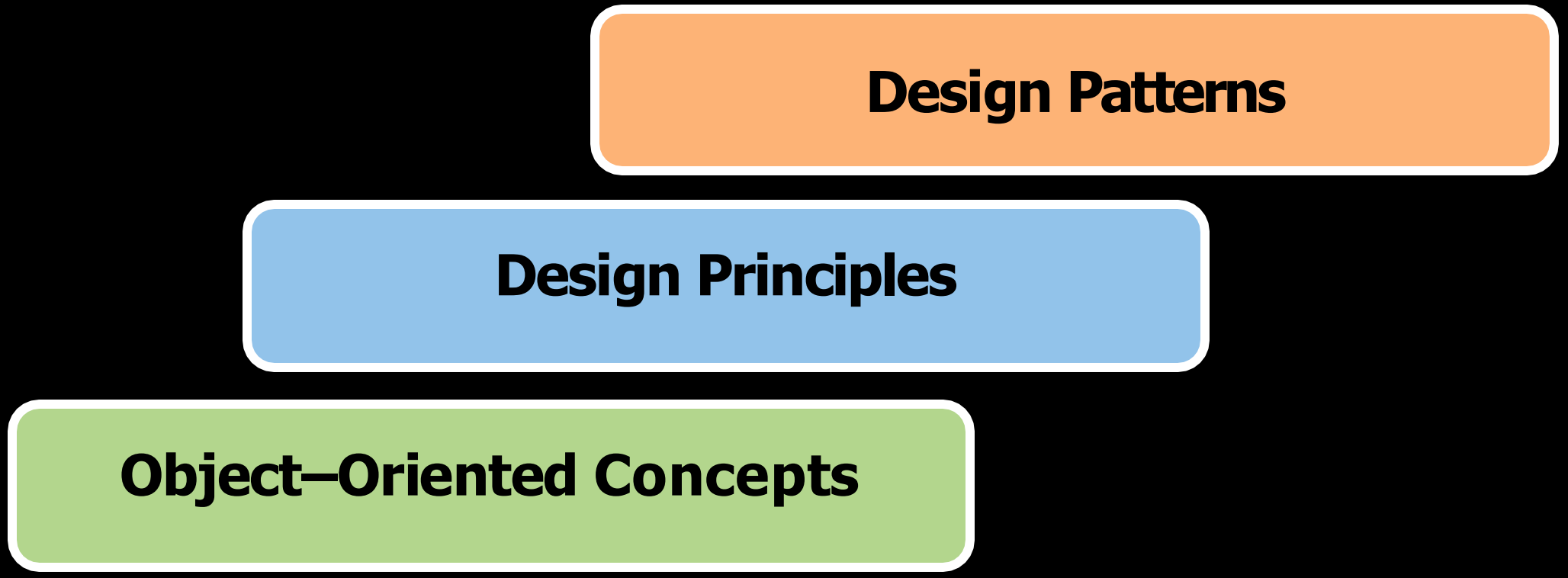
WHY THE SOFTWARE SHOULD BE DESIGNED

Real software applications build to be **stable and long- lived** applications.

- Flexible(Easy to change)
- Easy to maintain
- Reusable
- Easy to understand

INTRODUCTION

Phases of learning programming



OBJECT-ORIENTED PROGRAMMING

OBJECT ORIENTED PROGRAMMING

WHAT IS OBJECT-ORIENTED PROGRAMMING?

Object-Oriented Programming (OOP) is a programming paradigm based on defining, creating, and manipulating objects to develop reusable software.

- **A**bstraction
- **P**olymorphism
- 
- **I**nheritance
- **E**ncapsulation

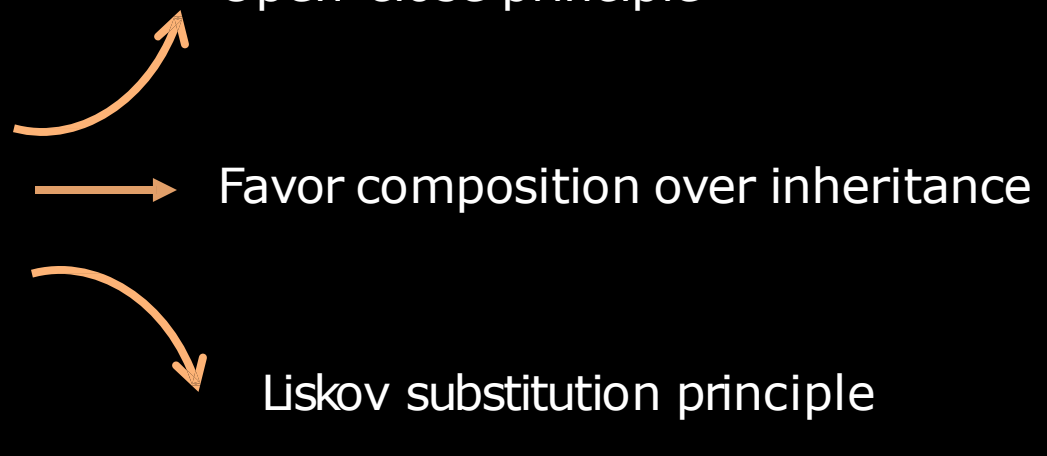
Object-Oriented Concepts

Design Principles

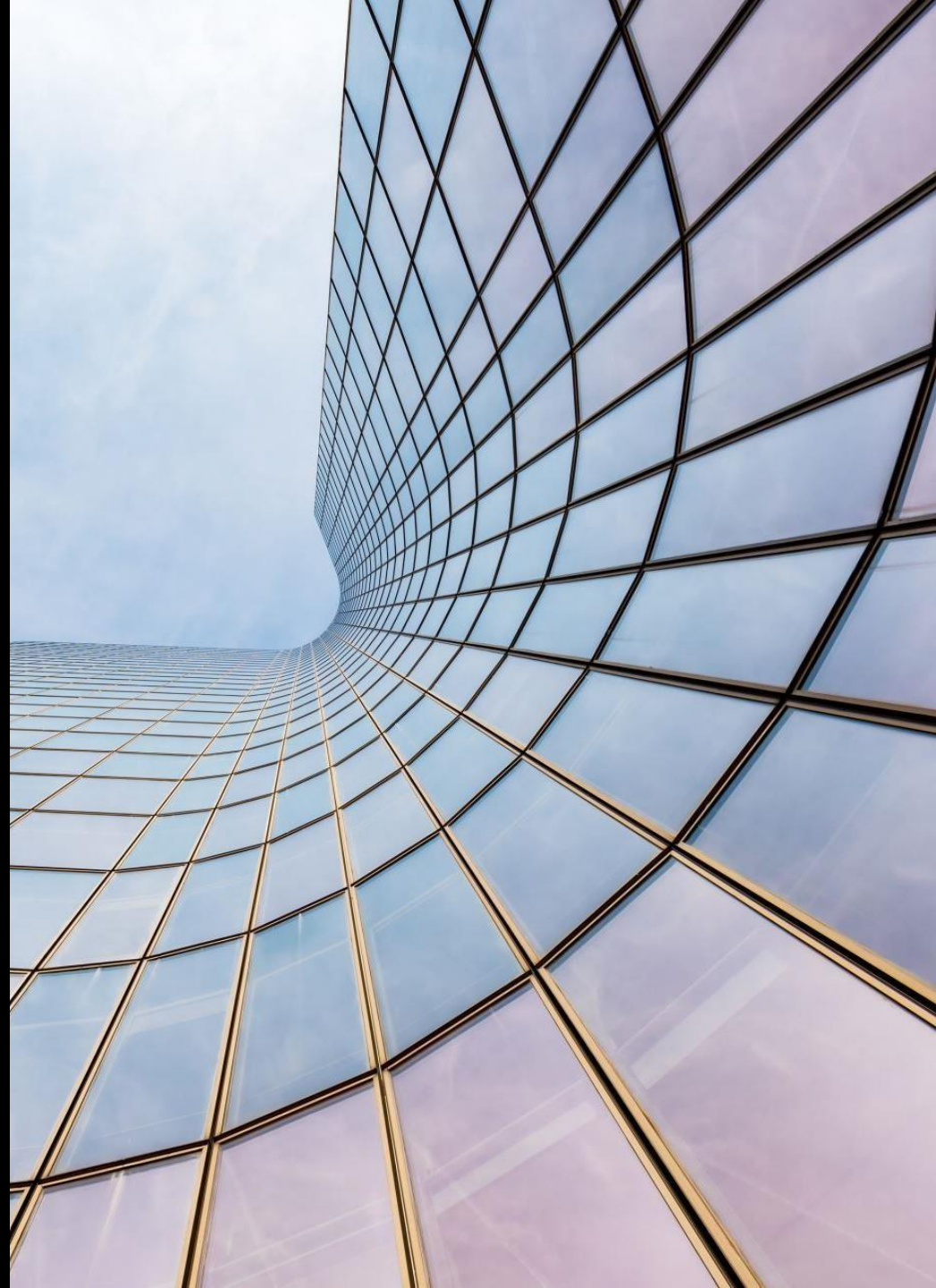
DESIGN PRINCIPLES

WHAT ARE DESIGN PRINCIPLES?

Design principles are a set of high-level guidelines which are used on the top of the object-oriented concepts to design better software application.



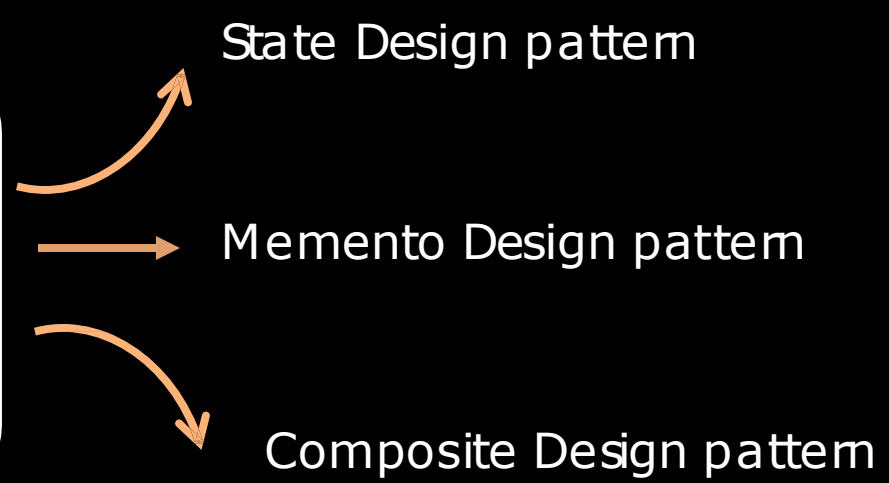
Design Patterns



DESIGN PATTERNS

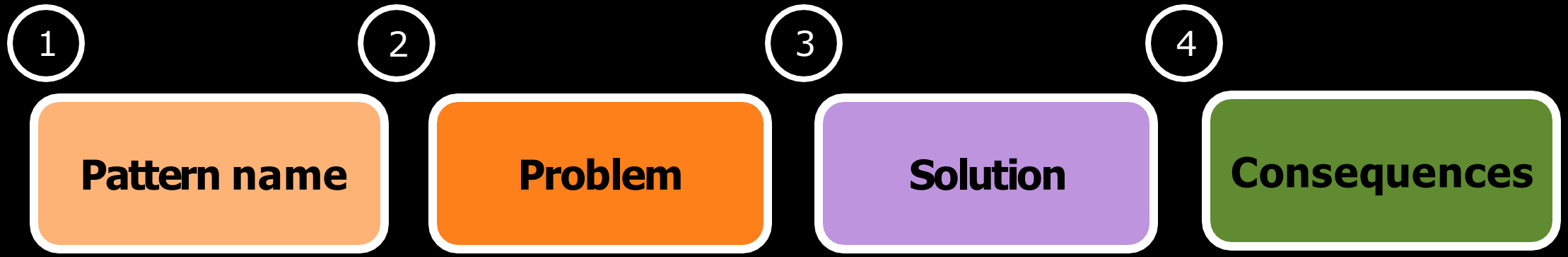
WHAT IS A DESIGN PATTERN?

Each pattern describes a problem which occurs repeatedly in our environment, and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice.



DESIGN PATTERN ESSENTIAL ELEMENTS

A pattern has four essential elements:



DESIGN PATTERN ESSENTIAL ELEMENTS

A pattern has four essential elements:

1 **Pattern name**

State Design Pattern

Composite Design pattern

The pattern name is a handle we can use to describe a design problem, its solutions, and consequences in a word or two. Naming a pattern immediately increases our design vocabulary. It lets us design at a higher level of abstraction. Having a vocabulary for patterns lets us talk about them with our colleagues, in our documentation, and even to ourselves. It makes it easier to think about designs and to communicate them and their trade-offs to others. Finding good names has been one of the hardest parts of developing our catalog [1].

DESIGN PATTERN ESSENTIAL ELEMENTS

A pattern has four essential elements:

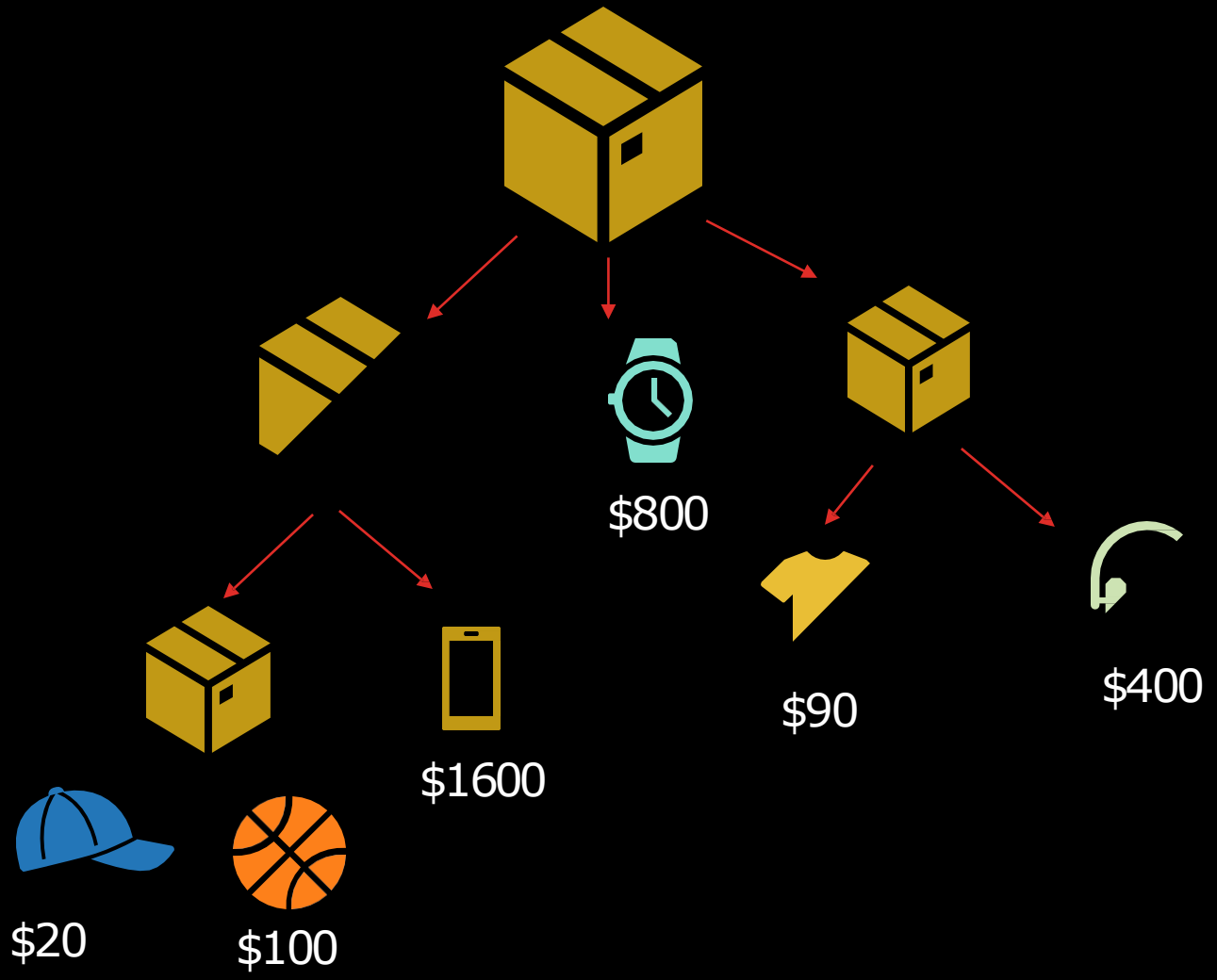
2 **Problem**

The problem describes when to apply the pattern. It explains the problem and its context. It might describe specific design problems such as how to represent algorithms as objects. Sometimes the problem will include a list of conditions that must be met before it makes sense to apply the pattern [1].

DESIGN PATTERN ESSENTIAL ELEMENTS

A pattern has four essential elements:

2 **Problem**



DESIGN PATTERN ESSENTIAL ELEMENTS

A pattern has four essential elements:

3 **Solution**

The solution describes the elements that make up the design, their relationships, responsibilities, and collaborations. The solution doesn't describe a particular concrete design or implementation, because a pattern is like a template that can be applied in many different situations. Instead, the pattern provides an abstract description of a design problem and how a general arrangement of elements (classes and objects in our case) solves it [1].

DESIGN PATTERN ESSENTIAL ELEMENTS

A pattern has four essential elements:

4 **Consequences**

The consequences are the results and trade-offs of applying the pattern. Though consequences are often unvoiced when we describe design decisions, they are critical for evaluating design alternatives and for understanding the costs and benefits of applying the pattern.
The consequences for software often concern space and time trade-offs [1].

Gamma, Erich, et al. "Design patterns: Elements of reusable object-oriented software addison-wesley." Reading, MA 2 (1995)

Which Programming Language will we use? and Why?

Java Programming Language

Java Programming Language

Java is a high-level object-oriented programming language. It is related to C++, which is a direct descendant of C. Much of the character of Java is inherited from these two languages. From C, Java derives its syntax. Many of Java's object-oriented features were influenced by C++. [1].

Schildt, Herbert. "Java: the complete reference." (2021).

Integrated Development Environment (IDE)

Programming Language

1 Java

Development tool

2

Text pad

Sublime Text

Eclipse

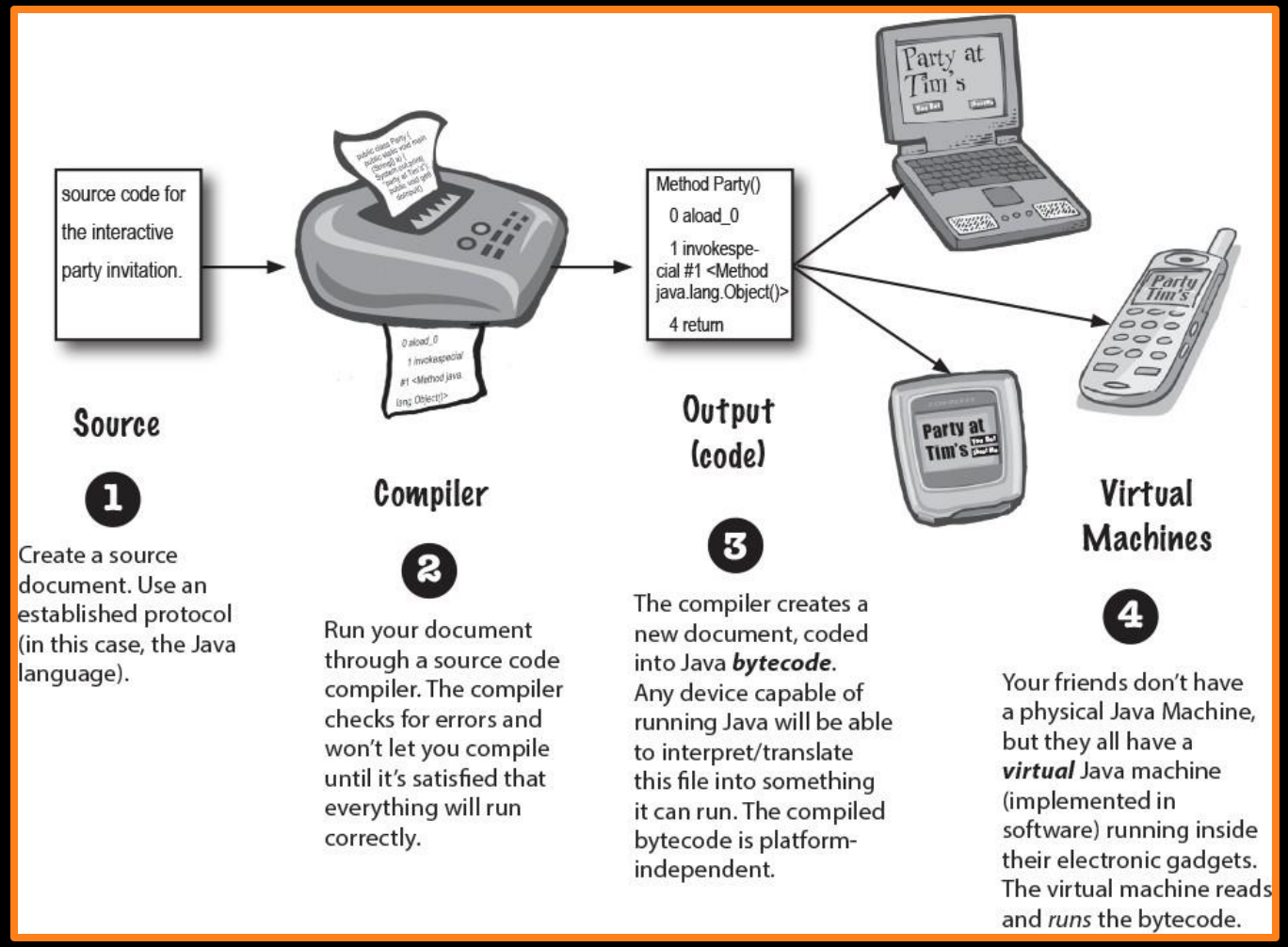
Intellij

Visual Studio

Netbeans

The Way Java Works

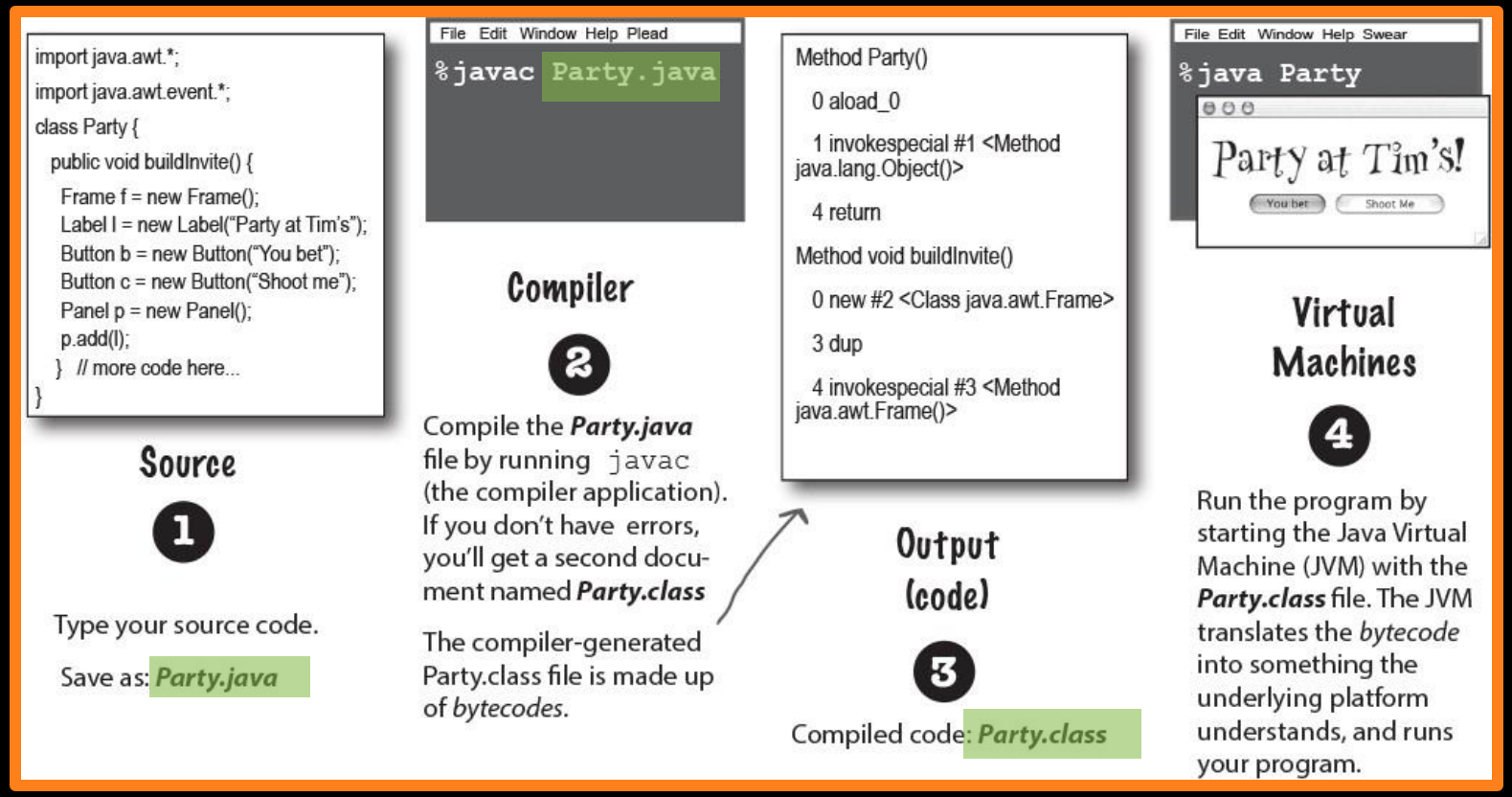
The goal is to write one application (in this example, an interactive party invitation) and have it work on whatever device!



Sierra, Kathy, and Bert Bates. *Head first java*. " O'Reilly Media, Inc.", 2021.

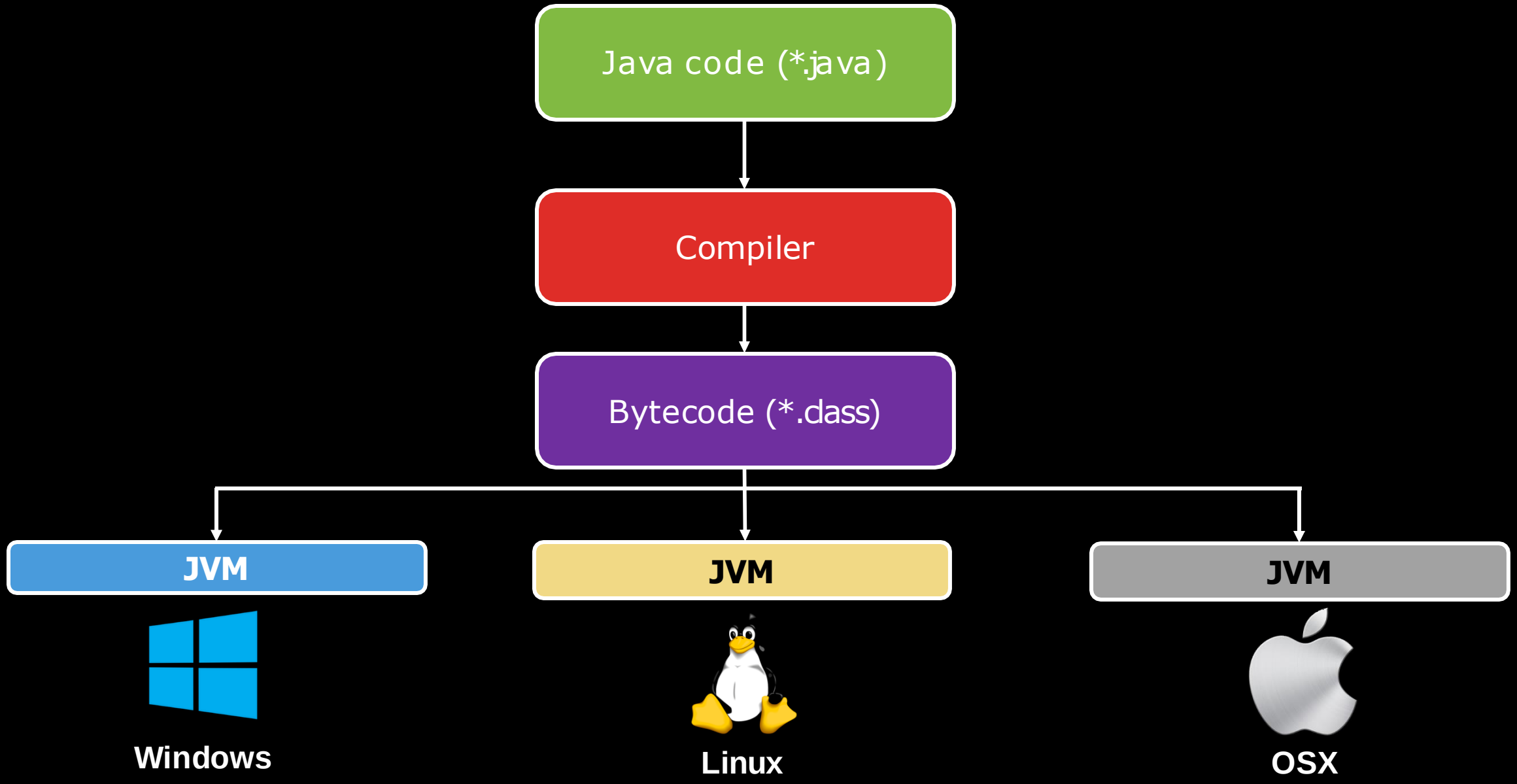
Java virtual Machine

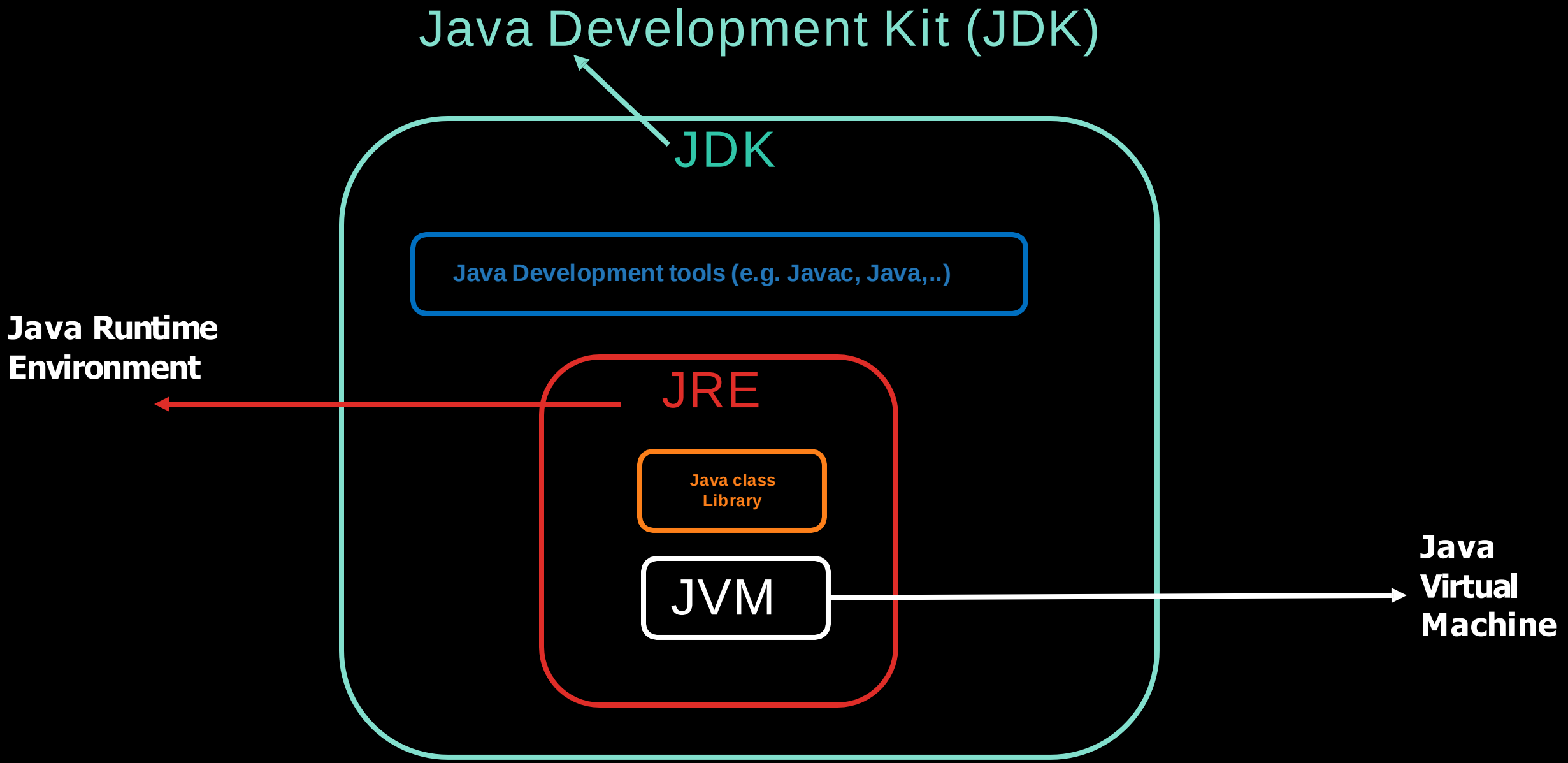
You'll type a source code file, compile it using the javac compiler, then run the compiled bytecode on a Java virtual machine



Sierra, Kathy, and Bert Bates. *Head first java*. " O'Reilly Media, Inc.", 2021.

Java virtual Machine





https://www.oracle.com/java/technologies/javase/jdk19-archive-downloads.html

Integrated Development Environments (IDE)

Eclipse is an integrated development environment (IDE) that simplifies writing, editing, running and debugging your code.

Eclipse IDE 2022-12

- **How to install Eclipse for Java on Windows:**

  https://www.youtube.com/watch?v=7DQykS3DVM8

- **Activate the autocomplete triggers for Java for Eclipse**

  https://www.youtube.com/watch?v=45t3flbRTzE

SUMMARY

SUMMARY

Explained why the software should be designed

Reviewed Object-Oriented programming

Define a Design principles

Define a design pattern

Explained the four essential elements of a design pattern

Introduced you to Java development environment

NEXT STEPS

To do

Further Reading

TO DO

- Download and install Eclipse IDE 2022-12

- **How to install Eclipse for Java on Windows:**

https://www.youtube.com/watch?v=7DQykS3DVM8

- **Activate the autocomplete triggers for Java for Eclipse**

https://www.youtube.com/watch?v=45t3flbRTzE

FURTHER READING

REFERENCES

1. Gamma, Erich, et al. "Design patterns: Elements of reusable object-oriented software (1995)
2. Schildt, Herbert. "Java: the complete reference." (2021).

THANK YOU