JAVAGURU INTRODUCTION TO

JAVA 1

ABOUT US

▸ Teaching more than 8 years

▸ Over 1000 students graduated

▸ Constantly expanding list of available courses

▸ Assisting students to get a job after graduation

HOMEWORK ASSIGNMENTS

▸ Required for course completion

   ▸ Level 1 assignments

▸ Everything else

   ▸ Optional, but highly recommended

▸ Upload to JavaGuru Moodle

MATERIALS AND RESOURCES

▸ Course materials

  ▸ JavaGuru Moodle: http://www.javaguru.lv/moodle/

▸ Books

  ▸ Head First Java (2nd edition) - Bert Bates, Kathy Sierra

  ▸ Grokking Algorithms - Aditya Y. Bhargava

  ▸ Thinking in Java (4th edition) - Bruce Eckel

▸ Documentation & help

  ▸ Oracle's Java Tutorial: https://docs.oracle.com/javase/tutorial/

  ▸ Q&A: https://stackoverflow.com/questions/tagged/java

GETTING STARTED

▸ Toolbox

    ▸ **JDK 8**: https://adoptopenjdk.net/

    ▸ **Intellij IDEA Community Edition**: https://www.jetbrains.com/idea/download/

    ▸ **Git**: https://git-scm.com/downloads

▸ Collaboration

    ▸ **GitHub**: https://github.com/

▸ Communication

    ▸ **Skype:** https://www.skype.com/en/get-skype/

BRIEF HISTORY

▸ **Green Team** (James Gosling, Mike Sheridan, Patrick Naughton)

▸ Initiated language project in June 1991

▸ Originally designed for small embedded systems in electronic appliances

▸ Initially called **Greentalk**, shortly after renamed to **Oak**

▸ A few years later renamed to Java

REASONS BEHIND NAMING

▸ **Reflection** of the **essence** of the technology

  ▸ Revolutionary and dynamic

  ▸ Easy to spell and fun to say

▸ Suggested names were **Dynamic, Silk, Jolt, DNA**

▸ **Java** is an island where first coffee was produced

PRINCIPLES

▸ Simple and robust

▸ Portable and platform independent

▸ High performant, multithreaded and secured

▸ Following object oriented language features

WHAT JAVA IS USED FOR

▸ Mobile phones

   ▸ Google's **Android** OS

▸ Point of sale systems

   ▸ **McDonald's** checkout computers

▸ Video games

   ▸ **Minecraft -** one of the most popular games of all time

▸ Trading applications

   ▸ **Murex -** application used for front to back connectivity is utilized by many banks

▸ Big data technologies

   ▸ **Elasticsearch** - distibuted search and analytics engine

JAVA COMPONENTS: JVM & JRE & JDK

▸ Java Virtual Machine (**JVM**) is an abstract computing machine that enables computer to run a Java program

▸ Java Runtime Environment (**JRE**) is a software package that contains what is required to run a Java program. It includes a Java Virtual Machine implementation together with an implementation of the Java Class Library

▸ Java Development Kit (**JDK**) is a superset of a JRE and contains tools for Java programmers, e.g. a java compiler

RELEASE TIME LAPSE

- JDK 1.1 Major Changes

  - **Extensive retooling** for AWT event model

  - Introduced **inner classes**

  - Addition of **JDBC** and **JavaBeans**

- J2SE 1.2 Major Changes

  - **Rebranded** as Java 2

- J2SE 1.3 Major Changes

  - Bundled with **Hotspot JVM**

  - **JavaSound**, Java Naming and Directory Interface

  - Java Platform **Debugger** Architecture

1997
JDK 1.1

▼

1998
J2SE 1.2

▼

2000
J2SE 1.3
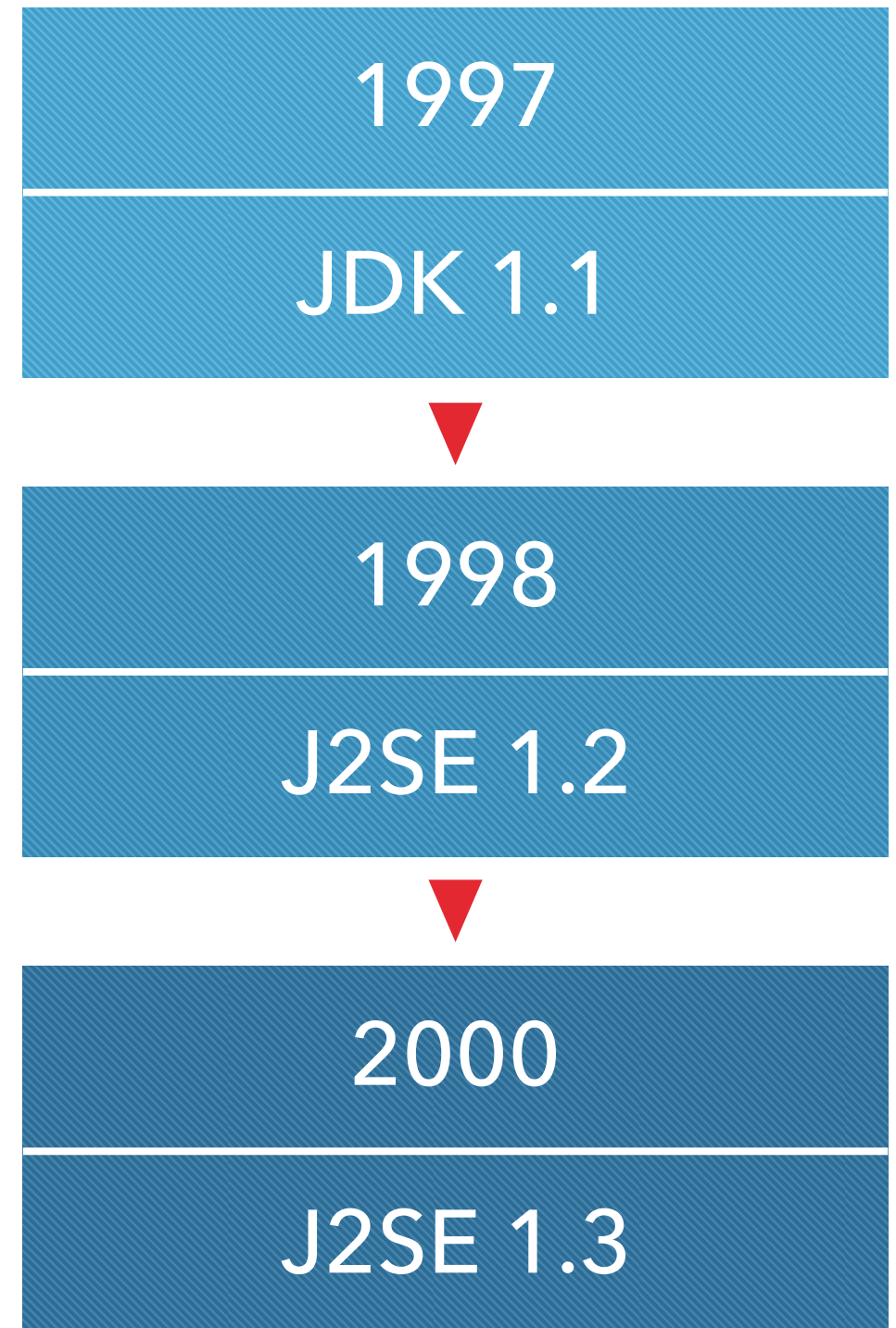
RELEASE TIME LAPSE

▸ JDK 1.4 Major Changes

  ▸ First release developed under **Java Community Process**

  ▸ Inclusion of **regular expressions**

▸ J2SE 5.0 Major Changes

  ▸ For-each loop

  ▸ Generics & Auto-boxing

  ▸ Var-args

▸ Java SE 6 Major Changes

  ▸ **Bundled** with database manager
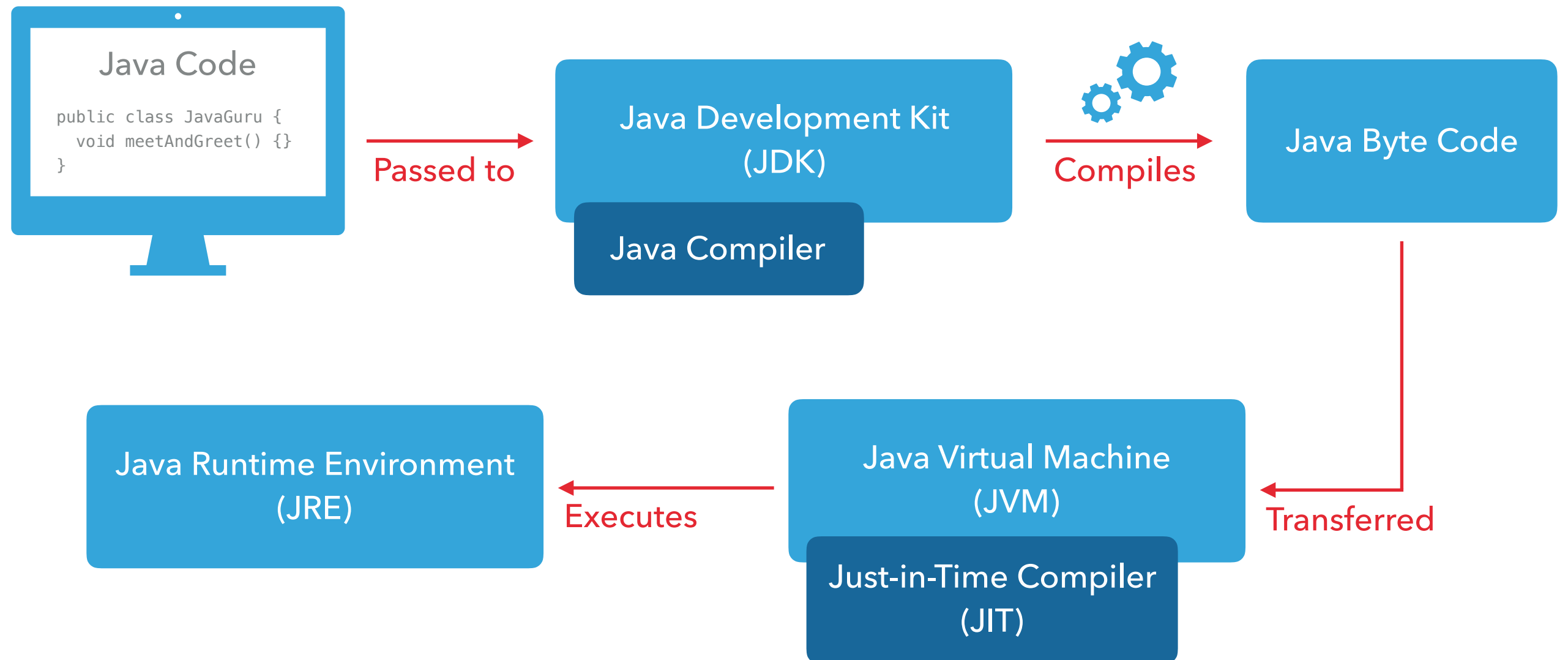
  ▸ **Facilitates** use of scripting languages with JVM

| 2002 |
| J2SE 1.4 |

| 2004 |
| J2SE 5.0 |

| 2006 |
| JAVA SE 6 |

RELEASE TIME LAPSE

▸ Java SE 7 Major Changes

  ▸ Strings in **switch**

  ▸ JVM **support** for dynamic languages

▸ Java SE 8 Major Changes

  ▸ **Lambda expressions** & Streams API

  ▸ Default methods

  ▸ **Reworked** dates API

▸ JDK 9 Major Changes

  ▸ Module system

  ▸ **Extended** Streams API

2011

JAVA SE 7

2014

JAVA SE 8

2017

JDK 9

INTERNAL WORKFLOW

Java Code

```
public class JavaGuru {
  void meetAndGreet() {}
}
```

Passed to →

Java Development Kit
(JDK)

Java Compiler

Compiles →

Java Byte Code

Transferred

Java Virtual Machine
(JVM)

Just-in-Time Compiler
(JIT)

Executes ←

Java Runtime Environment
(JRE)

OBJECT-ORIENTED
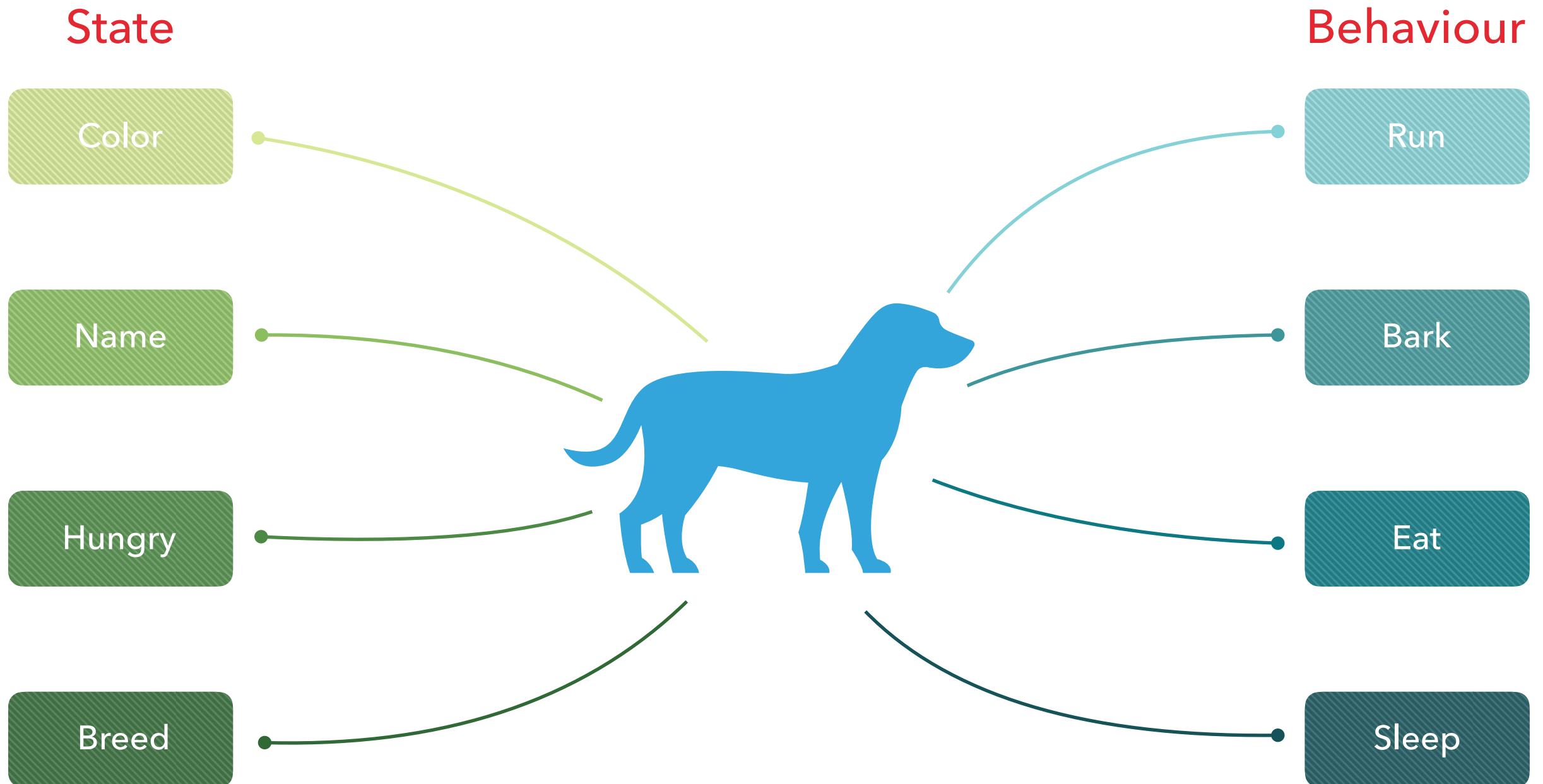# PRINCIPLES OVERVIEW

WHAT IS AN OBJECT?

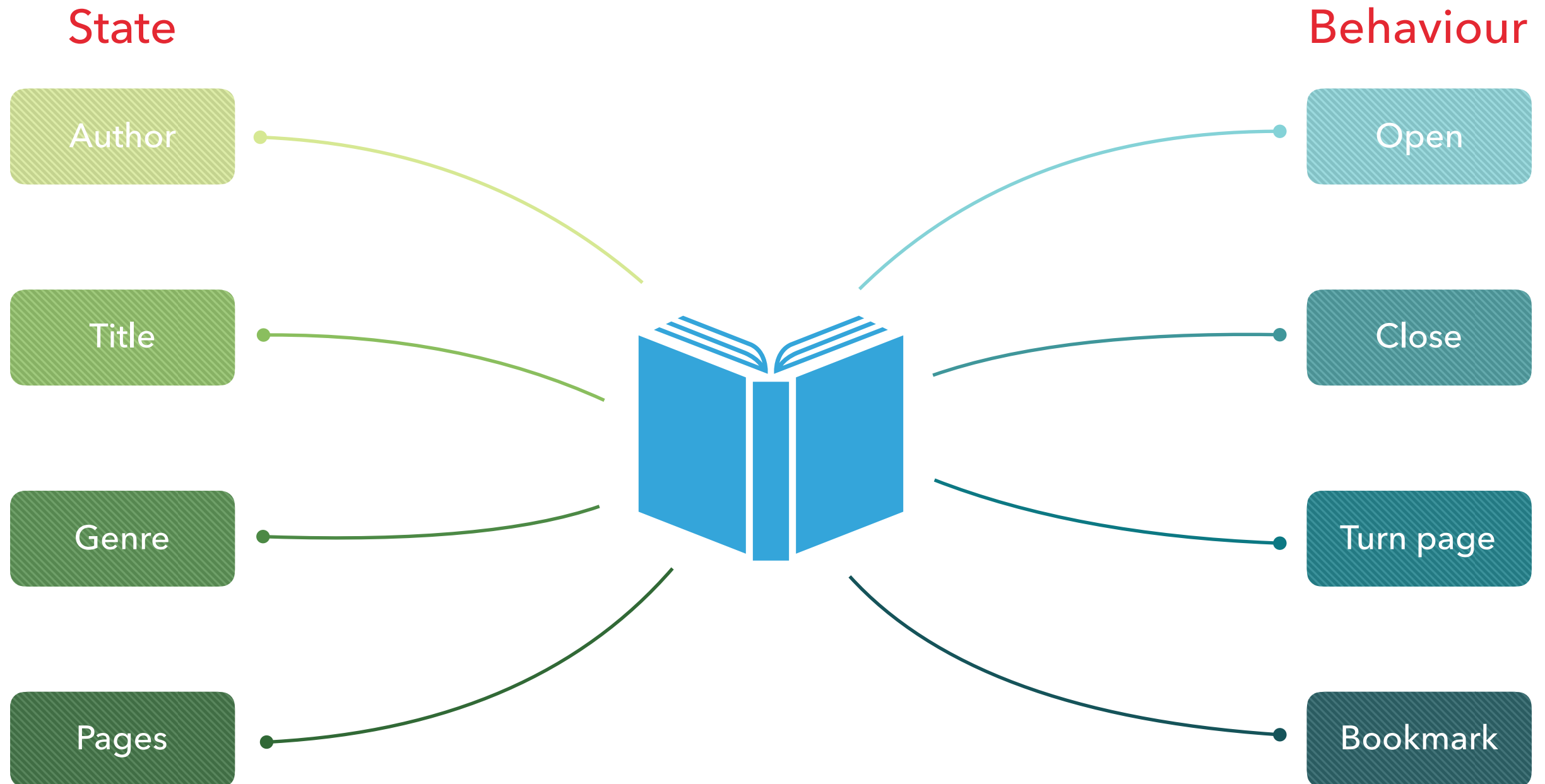▸ Just look around… Everything is an object!

CHARACTERISTICS

▸ Every object share two characteristics:

    ▸ Object has some sort of state

    ▸ Object might have behaviour

OBJECT BREAKDOWN: DOG

State

Color

Name

Hungry

Breed

Behaviour

Run

Bark

Eat

Sleep

OBJECT BREAKDOWN: BOOK

State

Behaviour

Author

Title

Genre

Pages

Open

Close

Turn page

Bookmark

ALGORITHMS

DEFINITION OF AN ALGORITHM

▸ An algorithm is a procedure or formula for solving a problem, based on conducting a sequence of specified actions

   ▸ In mathematics and computer science, an algorithm usually means a small procedure that solves a recurrent problem

▸ A computer program can be viewed as an elaborate algorithm

ALGORITHM BREAKDOWN: TAXI RIDE

1. Contact with the taxi service

2. Provide pickup & destination addresses

3. Wait for the cab

4. Get in the car

5. Enjoy your ride

6. Pay for the service upon arrival

ALGORITHM BREAKDOWN: PLANT A PLANT

1. Dig a hole

2. Put roots into soil

3. Cover the hole

4. If soil is wet then end

5. Water the plant