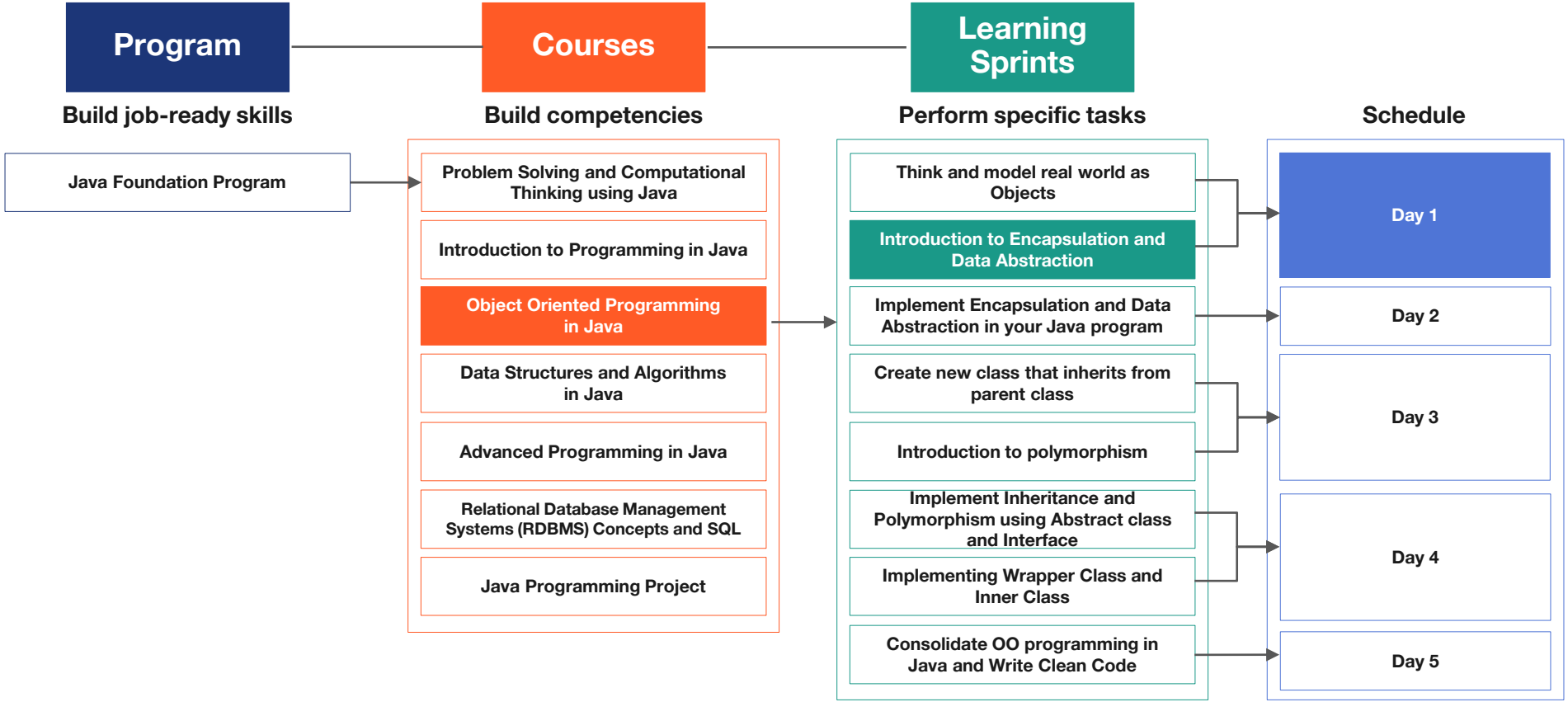
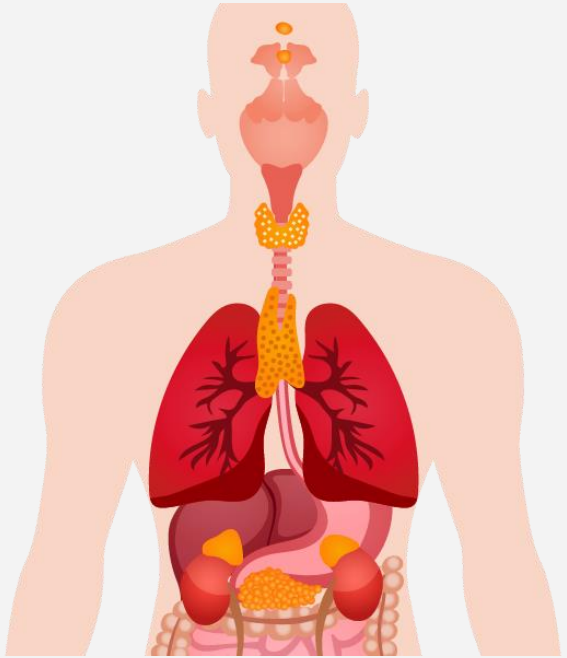
Java Program: Course 3: Plan



Think and Tell

How does our body work?



Digital Wallet

When was the last time you had used your digital wallet?

Why do you use a digital wallet?

Digital Wallet

These days digital wallets are commonly used for the purchases we make.

How do you think the process of payments made through your mobile works?

Digital Wallet

What features of a mobile are hidden from the end users?

Smart Speakers

Have you ever seen a smart speaker work?

How does it respond to your instructions?

https://spectrum.ieee.org/

Smart Speakers

https://www.news18.com/

How does the voice control mechanism of a speaker work?

Introduction to Encapsulation and Data Abstraction

Learning Objectives

- Define encapsulation and data abstraction
- Develop a class structure in Java
- Create objects
- Define and list the types of constructors
- Use “this” keyword

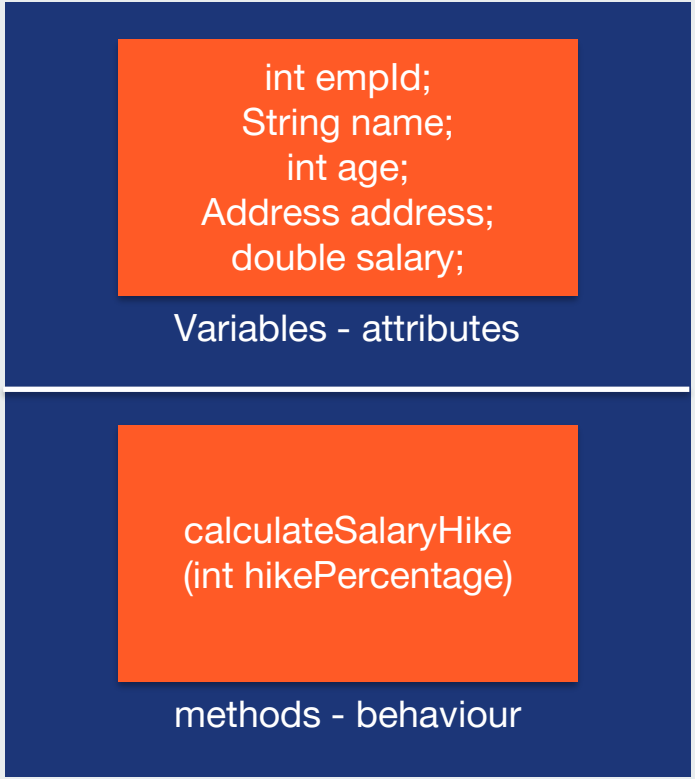
A Startup Company

- A startup company wants to manage its HR operations, like
  - People management
  - Compensation benefits to employees
  - Recruitment
- You, as a software programmer need to design a part of the application that manages the employee compensation benefits.
- Design a model of the application using OOP

Class

- A class is a blueprint or template from which objects can be created.

- The components of a class are variables and methods.



Class Structure

Class structure in Java

```
class Employee{
    int empId;
    String name;
    double salary;
    Address address;

    double calculateSalaryHike(float hikePercentage)
    {
        return salary * hikePercentage;
    }

}
```

Creating Objects

Objects are created from a defined class.

Declaration:

'employeeSam', 'employeeTom' are names given to the objects of type **Employee.**

```
Employee employeeSam;
Employee employeeTom;
```

Creating Objects

Instantiation:

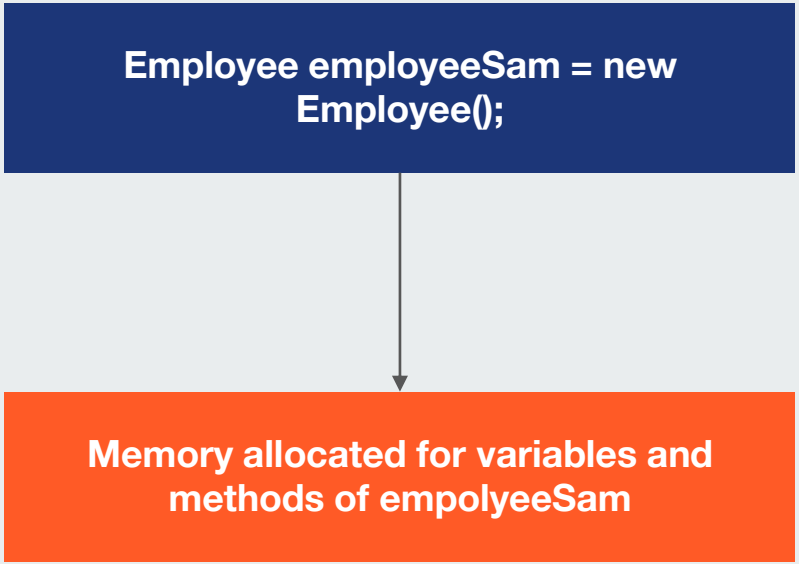
The new keyword is a Java operator that creates the object.

```
Employee employeeSam = new Employee();
Employee employeeTom = new Employee();
```

Creating Objects

Initialization:

The new operator is followed by a call to a constructor, which initializes the new object.



Constructor

- A class contains constructors that are invoked to create objects from the class blue print.
- Constructor declarations look like method declarations—except that they use the name of the class and have no return type

```
Employee()
{


}
```

A default constructor

Types of Constructors

- Default constructor
- No-argument constructor
- Parameterized constructor

```
Employee()
{
    empId = 101;
    name =  "Sam";
    salary = 2000;
}
```

No-argument Constructor

```
Employee(int empId, String name, double salary, Address address) {
    this.empId = empId;
    this.name = name;
    this.salary = salary;
    this.address = address;
}
```

Parameterized Constructor

The this keyword

```
class Employee{
    int empId;
    String name;
    double salary;
    Address address;

    Employee(int empId, String name, double salary, Address address) {
        this.empId = empId;
        this.name = name;
        this.salary = salary;
        this.address = address;
    }
    double calculateSalaryHike(float hikePercentage)
    {
        return salary * hikePercentage;
    }
}
```

Class member variable

Parameter variable

The value of the parameter variable is set to the class member variable

Accessing Members of a Class

```
Employee employeeSam = new Employee();

employeeSam.empId = 101;
employeeSam.name = "Sam";
employeeSam.salary = 2000;
employeeSam.address = new Address(20,"Marble
Drive","Evansville",47705);
```

The “.” operator is used to access data members of a class

Interactive Demo

A startup company wants to manage its HR operations, like

People management
Compensation benefits to employees
Recruitment

Write a program to calculate the hike that each employee will get as the company has increased its profits immensely and plans to give a 20% hike in salary to all its employees. Display the names of the employees of the company. Use appropriate access specifiers and write accessors and mutator method.

Access Modifiers

- Modifiers determine how data members and methods are used in other classes and objects
- The two frequently used modifiers permitted in Java are:
  - Static - Defines the class members that belong to a class, but not to any particular object of the class
  - Final – Indicates that the data member cannot be modified

Interactive Demo

Write a program to count the number of users who have logged in to the website.

Key Takeaways

- Classes in Java
- Creating objects
- Constructors and its types
- Modifiers permitted in Java
- “this” keyword