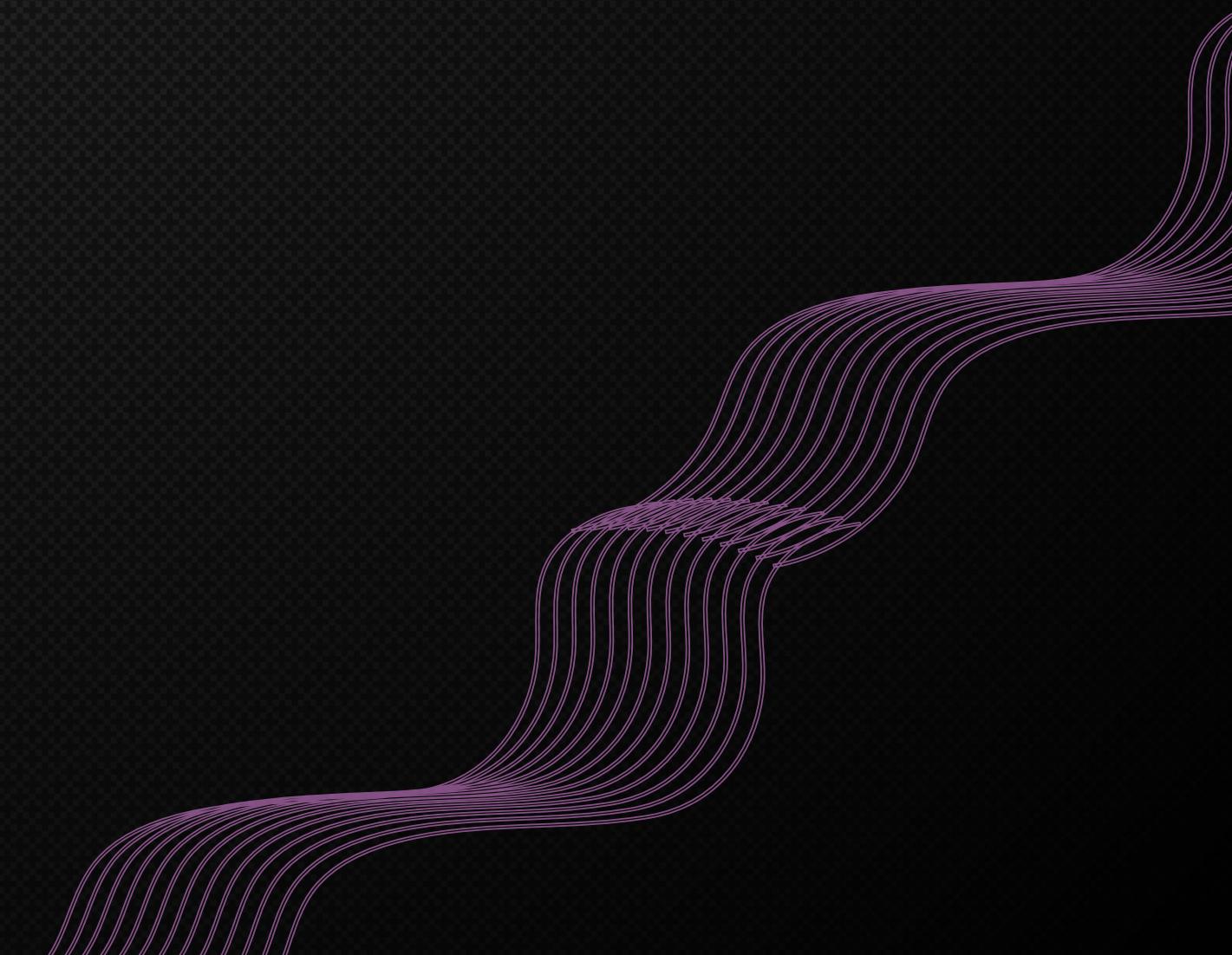


# **JAVA OOP BASICS**

## **PART I**

# AGENDA

- 1 **Package**
- 2 **Classes and Objects**
- 3 **Enum**
- 4 **Methods**
- 5 **Access modifiers**



# What is an object?



25% off Deal

Ring Alarms (2nd Gen)



Up to 41% off Deal

DEWALT 20V Tools and Kits



Up to 75% off Deal

Star Wars and Mandalorian Funko Pops



Up to 71% off Deal

Home and Kitchen Deal from Lush Decor,  
KitchenAid, and more



Up to 48% off Deal

Utopia Bedding Zippered Mattress Encasement  
- Waterproof Mattress Protector



39% off Deal

TAGRY Bluetooth Headphones True Wireless  
Earbuds 60H Playback LED Power Display...

# Product class

```
public class Product {  
    2 usages  
    String name;  
    2 usages  
    double price;  
    2 usages  
    Category category;  
    2 usages  
    ArrayList<Review> reviews;  
}
```



# Product Object

- Name: Apple watch
- Category: ELECTRONICS
- Price: 500\$
- Reviews: []

```
Product product = new Product(  
    name: "Apple Watch",  
    price: 500,  
    category: "ELECTRONICS");
```



# ENUM

```
public enum Category {  
    1 usage  
    BOOKS,  
    1 usage  
    ELECTRONICS,  
    2 usages  
    CLOTHING,  
    1 usage  
    FOOD,  
    1 usage  
    TOYS  
}
```

```
Product product = new Product(  
    name: "Apple Watch",  
    price: 500,  
    Category.ELECTRONICS);|
```

# INNER OBJECTS

```
public class Review {  
    no usages  
    private String author;  
    no usages  
    private int rating;  
    no usages  
    private String comment;  
}
```

```
Product product = new Product(  
    name: "Apple Watch",  
    price: 500,  
    Category.ELECTRONICS);  
Review review = new Review(  
    author: "John",  
    rating: 5,  
    comment: "This is a great product");  
product.addReview(review);
```

# Constructor

```
public Product(String name, double price, Category category) {  
    this.name = name;  
    this.price = price;  
    this.category = category;  
    this.reviews = new ArrayList<>();  
}  
  
no usages  new *  
public Product(String name, Category category) {  
    this.name = name;  
    this.category = category;  
}  
  
no usages  new *  
public Product(String name, double price, Category category, ArrayList<Review> reviews) {  
    this.name = name;  
    this.price = price;  
    this.category = category;  
    this.reviews = reviews;  
}
```

# Create a Class



Dell  
Processor: Intel i9, 8 core  
Display: FHD, 15.6inch  
Ram: 16gb

MacBook  
Processor: M2 chip, 8 core  
Display: Retina, 13.3inch  
Ram: 16gb



# Method

1

**Retrieving data  
from class**

2

**Setting class  
field**

3

**Interacting  
with other  
objects**

4

**Breaking down  
complex  
functionality**

# Retrieving data from class

```
public double getPrice() {  
    return price;  
}
```

## Setting fields

```
public void setPrice(double price) {  
    this.price = price;  
}
```

## Interacting with other objects

```
public void addReview(Review review) {  
    this.reviews.add(review);  
}
```

## Breaking down complex functionality

```
public Integer getAverageRating() {  
    if (reviews.size() == 0) {  
        return null;  
    }  
    int sum = 0;  
    for (Review review : reviews) {  
        sum += review.getRating();  
    }  
    return sum / reviews.size();
```

Lets write some  
code...