

# Objektum orientált programozás Java-ban

## Osztályok definiálása

```
class Car {
    String licensePlate;
    double speed;
    double maxSpeed;
}
```

## Objektumok létrehozása (new)

```
class Car {
    String licensePlate;
    double speed;
    double maxSpeed;
}

Car c;
c = new Car();
Car c = new Car();
```

## Objektum attribútumainak elérése

```
class Car {
    String licensePlate;
    double speed;
    double maxSpeed;
}

Car c = new Car();
c.licensePlate = "New York A45 636";
c.speed = 70.0;
c.maxSpeed = 123.45;
System.out.println(c.licensePlate + " is moving at " + c.speed +
    " kilometers per hour.");
```

## Egy Car objektum használata egy másik osztályban

```
class Car {
    String licensePlate;
    double speed;
    double maxSpeed;
}

class CarTest {
    public static void main(String args[]) {
        Car c = new Car();
        c.licensePlate = "New York A45 636";
        c.speed = 70.0;
        c.maxSpeed = 123.45;
        System.out.println(c.licensePlate + " is moving at " + c.speed +
            " kilometers per hour.");
    }
}
```

## Attribútumok (tagváltozók)

```
class Car {
    String licensePlate; // tagváltozó
    double speed; // tagváltozó
    double maxSpeed; // tagváltozó
}
```

## Metódusok (tagfüggvények)

```
class Car {
    String licensePlate; // e.g. "New York A456 324"
    double speed; // kilometers per hour
    double maxSpeed; // kilometers per hour

    // accelerate to maximum speed
    // put the pedal to the metal
    void floorIt() { // metódus
        this.speed = this.maxSpeed;
    }
}
```

## A metódusok hívása

```
class Car {
    String licensePlate; // e.g. "New York A456 324"
    double speed; // kilometers per hour
    double maxSpeed; // kilometers per hour

    // accelerate to maximum speed
    // put the pedal to the metal
    void floorIt() {
        this.speed = this.maxSpeed;
    }
}

class CarTest2 {
    public static void main(String args[]) {
        Car c = new Car();
        c.licensePlate = "New York A45 636";
        c.speed = 0.0;
        c.maxSpeed = 123.45;

        System.out.println(c.licensePlate + " is moving at " + c.speed +
            " kilometers per hour.");
        c.floorIt();
        System.out.println(c.licensePlate + " is moving at " + c.speed +
            " kilometers per hour.");
    }
}
```

## Tagváltozók használata metódusokban

```
class Car {
    String licensePlate; // e.g. "New York A456 324"
    double speed; // kilometers per hour
    double maxSpeed; // kilometers per hour

    void print() {
        System.out.println(licensePlate + " is moving at " + speed +
            " kilometers per hour ");
    }
}
```

## Argumentumok átadása a metódusoknak

```
class Car {
    String licensePlate; // e.g. "New York A456 324"
    double speed; // kilometers per hour
    double maxSpeed; // kilometers per hour

    void accelerate(double deltaV) {
        this.speed = speed + deltaV;
        if (speed > maxSpeed) {
            speed = maxSpeed;
        }
        if (speed < 0.0) {
            speed = 0.0;
        }
    }
}
```

## A this referencia használata - hivatkozás az objektumra

```
class Car {
    String licensePlate; // e.g. "New York A456 324"
    double speed; // kilometers per hour
    double maxSpeed; // kilometers per hour

    void accelerate(double deltaV) {
        this.speed = this.speed + deltaV;
        if (this.speed > this.maxSpeed) {
            this.speed = this.maxSpeed;
        }
        if (this.speed < 0.0) {
            this.speed = 0.0;
        }
    }
}
```

```
void setNameURLDescription(String s1, String s2, String s3) {
    this.name = s1;
    this.url = s2;
    this.description = s3;
}
```

## Példa a paraméterek átadására

```
class Car {
    String licensePlate; // e.g. "New York A456 324"
    double speed; // kilometers per hour
    double maxSpeed; // kilometers per hour

    // accelerate to maximum speed
    // put the pedal to the metal
    void floorIt() {
        this.speed = this.maxSpeed;
    }

    void accelerate(double deltaV) {
        this.speed = this.speed + deltaV;
        if (this.speed > this.maxSpeed) {
            this.speed = this.maxSpeed;
        }
        if (this.speed < 0.0) {
            this.speed = 0.0;
        }
    }
}

class CarTest3 {
    public static void main(String args[]) {
        Car c = new Car();
        c.licensePlate = "New York A45 636";
        c.speed = 0.0;
        c.maxSpeed = 123.45;

        System.out.println(c.licensePlate + " is moving at " + c.speed +
            " kilometers per hour.");
        for (int i = 0; i < 15; i++) {
            c.accelerate(10.0);
            System.out.println(c.licensePlate + " is moving at " + c.speed +
                " kilometers per hour.");
        }
    }
}

% java CarTest3
New York A45 636 is moving at 0.0 kilometers per hour.
New York A45 636 is moving at 10.0 kilometers per hour.
New York A45 636 is moving at 20.0 kilometers per hour.
New York A45 636 is moving at 30.0 kilometers per hour.
New York A45 636 is moving at 40.0 kilometers per hour.
New York A45 636 is moving at 50.0 kilometers per hour.
New York A45 636 is moving at 60.0 kilometers per hour.
New York A45 636 is moving at 70.0 kilometers per hour.
New York A45 636 is moving at 80.0 kilometers per hour.
New York A45 636 is moving at 90.0 kilometers per hour.
New York A45 636 is moving at 100.0 kilometers per hour.
New York A45 636 is moving at 110.0 kilometers per hour.
New York A45 636 is moving at 120.0 kilometers per hour.
New York A45 636 is moving at 123.45 kilometers per hour.
New York A45 636 is moving at 123.45 kilometers per hour.
New York A45 636 is moving at 123.45 kilometers per hour.
```

## Tagváltozók értékének beállítása metódusok segítségével

```
class Car {
    String licensePlate; // e.g. "New York A456 324"
    double speed; // kilometers per hour
    double maxSpeed; // kilometers per hour

    // setter method for the license plate property
    void setLicensePlate(String licensePlate) {
        this.licensePlate = licensePlate;
    }

    // accelerate to maximum speed
    // put the pedal to the metal
    void floorIt() {
        this.speed = this.maxSpeed;
    }

    void accelerate(double deltaV) {
        this.speed = this.speed + deltaV;
        if (this.speed > this.maxSpeed) {
            this.speed = this.maxSpeed;
        }
        if (this.speed < 0.0) {
            this.speed = 0.0;
        }
    }
}
```

## Metódusok visszatérési értéke

```
String getLicensePlate() {
    return this.licensePlate;
}
```

## Metódusok visszatérési értéke - példa

```
class Car {
    String licensePlate; // e.g. "New York A456 324"
    double speed; // kilometers per hour
    double maxSpeed; // kilometers per hour

    // getter (accessor) methods
    String getLicensePlate() {
        return this.licensePlate;
    }

    double getMaxSpeed() {
        return this.maxSpeed;
    }

    double getSpeed() {
        return this.speed;
    }

    // accelerate to maximum speed
    // put the pedal to the metal
    void floorIt() {
        this.speed = this.maxSpeed;
    }

    void accelerate(double deltaV) {
        this.speed = this.speed + deltaV;
        if (this.speed > this.maxSpeed) {
            this.speed = this.maxSpeed;
        }
        if (this.speed < 0.0) {
            this.speed = 0.0;
        }
    }
}
```

## Konstruktorok - I.

```
Car c = new Car();
```

```
Car() {
    this.licensePlate = "";
    this.speed = 0.0;
    this.maxSpeed = 120.0;
}

Car(String licensePlate, double speed, double maxSpeed) {
    this.licensePlate = licensePlate;
    this.speed = speed;
    this.maxSpeed = maxSpeed;
}

Car(String licensePlate, double maxSpeed) {
    this.licensePlate = licensePlate;
    this.speed = 0.0;
    this.maxSpeed = maxSpeed;
}
```

## Konstruktorok - II.

```
class Car {
    String licensePlate; // e.g. "New York A456 324"
    double speed; // kilometers per hour
    double maxSpeed; // kilometers per hour

    Car(String licensePlate, double maxSpeed) {
        this.licensePlate = licensePlate;
        this.speed = 0.0;
        this.maxSpeed = maxSpeed;
    }

    // getter (accessor) methods
    String getLicensePlate() {
        return this.licensePlate;
    }

    double getMaxSpeed() {
        return this.maxSpeed;
    }

    double getSpeed() {
        return this.speed;
    }

    // setter method for the license plate property
    void setLicensePlate(String licensePlate) {
        this.licensePlate = licensePlate;
    }

    // accelerate to maximum speed
    // put the pedal to the metal
    void floorIt() {
        this.speed = this.maxSpeed;
    }

    void accelerate(double deltaV) {
        this.speed = this.speed + deltaV;
        if (this.speed > this.maxSpeed) {
            this.speed = this.maxSpeed;
        }
        if (this.speed < 0.0) {
            this.speed = 0.0;
        }
    }
}
```

## A konstruktorok használata

```
class CarTest4 {
    public static void main(String args[]) {
        Car c = new Car("New York A45 636", 123.45);
        System.out.println(c.licensePlate + " is moving at " + c.speed +
            " kilometers per hour.");
        for (int i = 0; i < 15; i++) {
            c.accelerate(10.0);
            System.out.println(c.licensePlate + " is moving at " + c.speed +
                " kilometers per hour.");
        }
    }
}
```

## A speed tagváltozó használata az objektumon belül

```
void accelerate(double deltaV) {
    this.speed = this.speed + deltaV;
    if (this.speed > this.maxSpeed) {
        this.speed = this.maxSpeed;
    }
    if (this.speed < 0.0) {
        this.speed = 0.0;
    }
}

Car(String licensePlate, double maxSpeed) {
    this.licensePlate = licensePlate;
    this.speed = 0.0;
    if (maxSpeed >= 0.0) {
        this.maxSpeed = maxSpeed;
    }
    else {
        maxSpeed = 0.0;
    }
}

Car c = new Car("New York A234 567", 100.0);
c.c.speed = 150.0;
```

## Tagváltozók és metódusok láthatóságának (elérhetőségének) szabályozása (access protection)

```
Car c = new Car("New York A234 567", 100.0);
c.c.speed = 150.0;

public class Car {
    private String licensePlate; // e.g. "New York A456 324"
    private double speed; // kilometers per hour
    private double maxSpeed; // kilometers per hour
    static String version = "1.0";

    public Car(String licensePlate, double maxSpeed) {
        this.licensePlate = licensePlate;
        this.speed = 0.0;
        if (maxSpeed >= 0.0) {
            this.maxSpeed = maxSpeed;
        }
        else {
            maxSpeed = 0.0;
        }
    }

    // getter (accessor) methods
    public String getLicensePlate() {
        return this.licensePlate;
    }

    public double getMaxSpeed() {
        return this.maxSpeed;
    }

    public double getSpeed() {
        return this.speed;
    }

    // setter method for the license plate property
    public void setLicensePlate(String licensePlate) {
        this.licensePlate = licensePlate;
    }

    // accelerate to maximum speed
    // put the pedal to the metal
    public void floorIt() {
        this.speed = this.maxSpeed;
    }

    public void accelerate(double deltaV) {
        this.speed = this.speed + deltaV;
        if (this.speed > this.maxSpeed) {
            this.speed = this.maxSpeed;
        }
        if (this.speed < 0.0) {
            this.speed = 0.0;
        }
    }
}
```

## Példa: A hozzáférési jogok szintjei - Static tagváltozók használata

```
public class Car {
    private String licensePlate; // e.g. "New York A456 324"
    private double speed; // kilometers per hour
    private double maxSpeed; // kilometers per hour
    static String version = "1.0";

    public Car(String licensePlate, double maxSpeed) {
        this.licensePlate = licensePlate;
        this.speed = 0.0;
        if (maxSpeed >= 0.0) {
            this.maxSpeed = maxSpeed;
        }
        else {
            maxSpeed = 0.0;
        }
    }

    // getter (accessor) methods
    public String getLicensePlate() {
        return this.licensePlate;
    }

    public double getMaxSpeed() {
        return this.maxSpeed;
    }

    public double getSpeed() {
        return this.maxSpeed;
    }

    public static String getVersion() {
        return version;
    }

    // setter method for the license plate property
    public void setLicensePlate(String licensePlate) {
        this.licensePlate = licensePlate;
    }

    // accelerate to maximum speed
    // put the pedal to the metal
    public void floorIt() {
        this.speed = this.maxSpeed;
    }

    public void accelerate(double deltaV) {
        this.speed = this.speed + deltaV;
        if (this.speed > this.maxSpeed) {
            this.speed = this.maxSpeed;
        }
        if (this.speed < 0.0) {
            this.speed = 0.0;
        }
    }
}
```

## Hivatkozás a static tagváltozókra

```
Car c = new Car("New York", 89.7);
String s = c.getVersion();
String s = Car.getVersion();
Error: Can't make static reference to method void print() in class test.
```

## Static (class) tagváltozók

```
class Alkalmazott {
    int életkor;
    int fizetés;
    static int hivatalosMinimálbér;

    int mennyiFizetésJárNeki() {
        if (hivatalosMinimálbér > Math.round(fizetés * (1 + 6letkor/100.0))) {
            return hivatalosMinimálbér;
        }
        else {
            return (int) Math.round(fizetés * (1 + 6letkor/100.0));
        }
    }
}

Alkalmazott = béla new Alkalmazott();
Alkalmazott = marci new Alkalmazott();
Alkalmazott = ilona new Alkalmazott();
```



## Öröklés

```
class Alkalmazott {
    int életkor;
    int fizetés;
}

class Manager extends Alkalmazott {
    int autoAzonosító;
}

class Nagyfőnök extends Manager {
    int titkárnőAzonosító;
    int villaAzonosító;
}

Alkalmazott = béla new Alkalmazott();
Manager = marci new Manager();
Nagyfőnök = ilona new Nagyfőnök();
```

