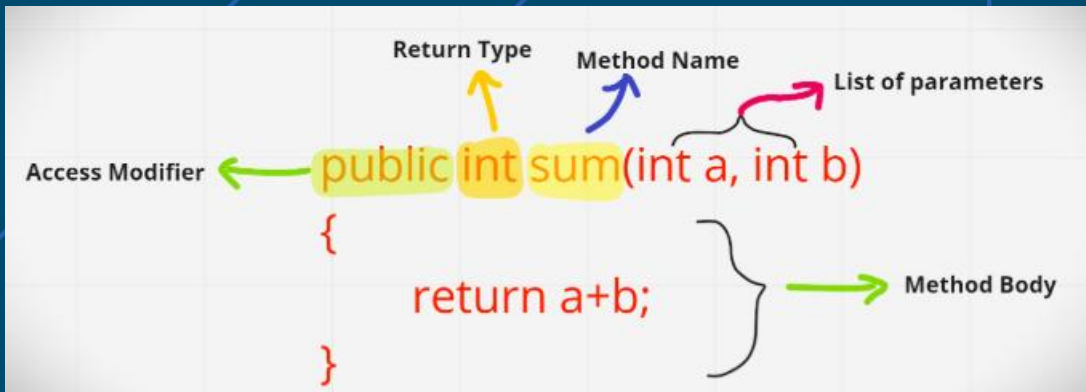


Féidearthachtaí as Cuimse Infinite Possibilities



Methods in Java (and OO)

Object Oriented programming
Colette Kirwan

Methods

- ```
class className
{
 // attributes
 // methods
 ...
}
```

# What are methods?

- A **method** in Java (and OOP in general) is a **block of code that belongs to a class and performs a specific task.**

Methods define **behaviour** for objects and allow **code reuse, modularity, and encapsulation.**

Very basic example !!!!

```
public class Car
{
 // Method with parameters
 public void setModel(String newModel) {
 model = newModel;
 }
}
```

# Example :

- Car class
  - How many attributes
  - How many methods?
  - Write code to use them (i.e. call them)
  - Note whether a method takes in or return data

# Method signature

**Method signature** is the method name and the number and type of its parameters

```
public int methodName(int a, int b)
{
 // body
}
```

(note:

return type and access modifier are not part of the method signature

# Method Overloading

- Sometimes
  - need multiple versions of the **same method** in the same class
  - Ok –provided that **method signatures** are different
    - Example: “car” class
      - `setDetails()`
      - `setDetails(String model)`
      - `setDetails(String model, int year)`

# Can these be in same class?

```
public int addNumbers(int a, int b)
{
 // body
}

public int addNumbers(int a, String b)
{
 // body
}

public int addNumbers(int a, int b, int c)
{
 // body
}

public void addNumbers(int x, int y, int z)
{
 // body
}

public String addNumbers(int a, String b)
{
 // body
}

public int addNumbers(int temp, String region)
{
 // body
}
```

# Different method signatures

```
public int addNumbers(int a, int b)
{
 // body
}

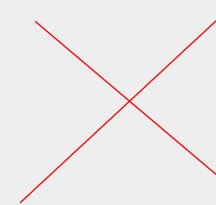
public int addNumbers(int a, String b)
{
 // body
}

public int addNumbers(int a, int b, int c)
{
 // body
}

public void addNumbers(int x, int y, int z)
{
 // body same as 3rd one
}

public String addNumbers(int a, String b)
{
 // body same as second one
}

public int addNumbers(int temp, String region)
{
 // body same as second one
}
```





# Methods

- Using a method –
  - `objectname. methodname (..)`
- – example –

# UML

- In industry, Unified Modelling Language used to specify design.
- UML **class diagrams** often used

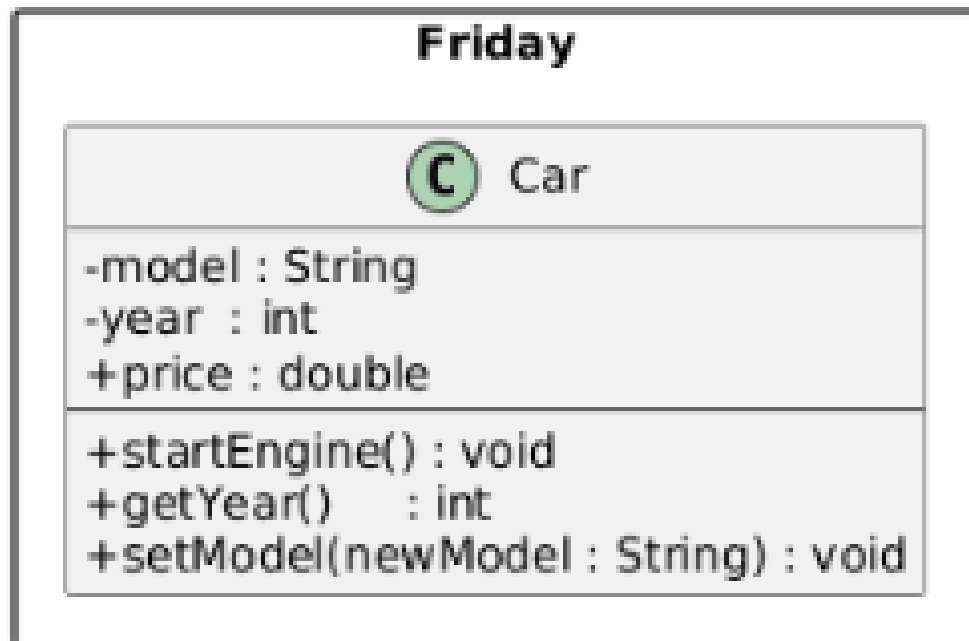
Note: +s/-s,  
method  
signature,  
parameter list,  
return types

| Person                                         | Class Name           |
|------------------------------------------------|----------------------|
| - name: String                                 | attributes           |
| + setName(name: String)<br>+ getName(): String | operations (methods) |

# Draw UML

- For `Car` class

# Answer



# Covered

- Methods definitions
- Method signatures
- How to use methods
- UML representation of classes