

## Design e Arquitectura de Software

2025/2026

### Lab. GoF Classes - 1

#### - Creational Patterns

1. Implement the **Factory Method** pattern as illustrated in slide nr. 12.

An example of expected output is:

```
A tiger is created.  
It says: Roar!  
It loves to roam in a jungle.  
A dog is created.  
It says: Bow-Wow.  
It prefers barking.
```

2. Implement the **Abstract Factory** pattern:

A) as illustrated in slides nr.17-18.

B) As illustrated in slide nr. 19, using a Factory provider.

Here is an example of expected output

```
You opt for a wild animal factory.  
A wild dog with white color is created.  
A wild tiger with golden and cinnamon color is created.  
The wild dog says: I prefer to roam freely in jungles.Bow-Wow.  
The wild tiger says: I prefer hunting in jungles.Roar.  
The wild tiger says: I saw a wild dog in the jungle.  
*****  
You opt for a pet animal factory.  
  
A pet dog with black color is created.  
A pet tiger with yellow color is created.  
The pet dog says: Bow-Wow. I prefer to stay at home.  
The pet tiger says: Halum. I play in an animal circus.  
The pet tiger says: I saw a pet dog in my town.
```

3. Implement the **prototype** pattern:

a) as described in slide 21

b) as described in slide 22 (using a maker object)

Here is an example of expected output

```
Model: Nano XM624 cc  
Price: 5003  
-----  
Editing a cloned model:  
Model: Nano XM624 cc  
Its on-road price: $5103  
-----
```

```
Model: Ford Aspire
Price: 40020
-----
Editing a cloned model:
Model: Ford Aspire
Its on-road price: $40120
-----
```

4. Implement the **Builder** pattern,
  - a) As described in slides 30-31
  - b) As described in slides 32 (with a fluent interface)

Here is an example of expected output

```
We are about to make a Ford car.
These are the construction sequences:
Making the car body.
4 wheels are added to the car.
Adding the car brand: Ford
We are about to make a Honda motorcycle.
These are the construction sequences:
Adding the brand name: Honda
Making the body of the motorcycle.
2 wheels are added to the motorcycle.
```

5. Implement the **Singleton** pattern,
  - a) As described in slides 36-37 (lazy initialization)
  - b) As described in slide 38 (eager initialization)

```
Trying to make a captain for your team.
A new captain is elected for your team.
Trying to make another captain for your team:
You already have a captain for your team.
Send him for the toss.
Both captain1 and captain2 are the same.
```