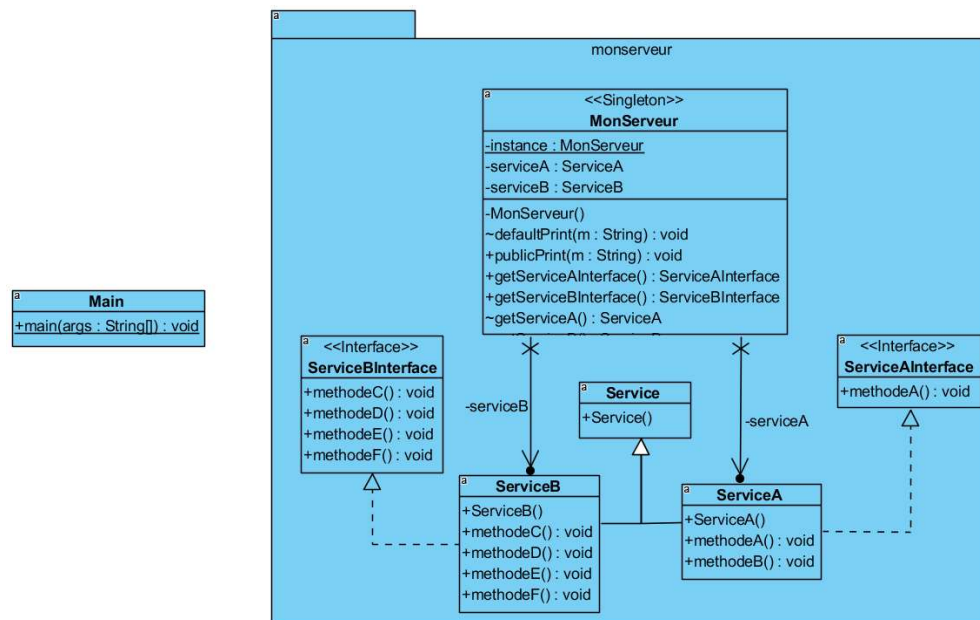


Patron #1 : Singleton



Contraintes

- MonServeur est un singleton
 - Le constructeur instancie les deux services
 - La méthode **defaultPrint** affiche « defaultPrint » + le paramètre « m »
 - La méthode **publicPrint** affiche « publicPrint » + le paramètre « m »
 - La méthode **getServiceAInterface** retourne serviceA via son interface
 - La méthode **getServiceBInterface** retourne serviceB via son interface
 - La méthode **getServiceA** retourne serviceA
- Les méthodes des services
 - La méthode **methodeA** envoie « A » à la méthode **monServeur.defaultPrint**
 - La méthode **methodeB** envoie « B » à la méthode **monServeur.publicPrint**
 - La méthode **methodeC** envoie « C » à la méthode **monServeur.defaultPrint**
 - La méthode **methodeD** envoie « D » à la méthode **monServeur.publicPrint**
 - La méthode **methodeE** appelle la méthode **ServiceAInterface.methodeA**
 - La méthode **methodeF** appelle la méthode **ServiceA.methodeB**
- Recoder la procédure « main » suivante

```

public class Main {

    public static void main(String args[]){

        MonServeur.getInstance().publicPrint( m: "main");
        MonServeur.getInstance().getServiceAInterface().methodeA();
        MonServeur.getInstance().getServiceBInterface().methodeC();
        MonServeur.getInstance().getServiceBInterface().methodeD();
        MonServeur.getInstance().getServiceBInterface().methodeE();
        MonServeur.getInstance().getServiceBInterface().methodeF();

    }

}

```

```

publicPrint : main
defaultPrint : A
defaultPrint : C
publicPrint : D
defaultPrint : A
publicPrint : B

```