

C#

- Exercice
 - C# POO

Exercise

Exercice

C# POO

Ex 8 :

Polymorphisme

Ex 9 :

```
namespace test2;

public class Circle
{
    private readonly double radius;
    private readonly double pi = System.Math.PI;

    public Circle(double radius)
    {
        this.radius = radius;
    }

    public double GetArea()
    {
        return pi * radius * radius;
    }

    public double GetPerimeter()
    {
        return 2 * pi * radius;
    }
}

public class Program
{

```

```
static void Main()
{
    Circle c = new Circle(6.25);
    double area = c.GetArea();
    double perimeter = c.GetPerimeter();
    Console.WriteLine(area); // Doit retourner 122.718463030851.
    Console.WriteLine(perimeter); // Doit retourner 39.2699081698724.
}
}
```

Ex 10 :

```
namespace test2;

public class Student : Person
{
    public void GoToClasses()
    {
        Console.WriteLine("I'm going to class.");
    }

    public void DisplayAge()
    {
        Console.WriteLine($"My age is: {this.age} years old");
    }
}

public class Teacher : Person
{
    private string subject;

    public void Explain()
    {
        Console.WriteLine("Explanation begins");
    }
}
```

```
public Teacher(string subject)
{
    this.subject = subject;
}
}
```

```
public class Person
{
    protected int age;

    public void SetAge(int n)
    {
        this.age = n;
    }

    public virtual void SayHello()
    {
        Console.WriteLine("Hello");
    }
}
```

```
public class Test
{
    static void Main()
    {
        Person person = new Person();
        person.SayHello();

        Student student = new Student();
        student.SetAge(15);
        student.SayHello();
        student.GoToClasses();
        student.DisplayAge();

        Teacher teacher = new Teacher("math");
        teacher.SetAge(40);
        teacher.SayHello();
        teacher.Explain();
    }
}
```

```
}
```

BONUS :

```
class A
{
    public void Afficher()
    {
        Console.WriteLine("A");
    }
}

class B : A
{
    public new void Afficher()
    {
        Console.WriteLine("B");
    }
}

class C : B
{
    public new void Afficher()
    {
        Console.WriteLine("C");
    }
}

public class Program
{
    static void Main()
    {
```

```
A a = new A();  
B b = new C();  
a.Afficher();  
b.Afficher();  
}  
}
```

résultat :

A B

pour avoir AA : il suffit juste de supprimer les 2 "new"