

**SOPORTE A LA GESTION DE DATOS CON PROGRAMACION VISUAL – 2018**  
**TRABAJO PRÁCTICO Nº 2**

**Ejercicio 1**

Escribir una clase llamada rectángulo que contenga una base y una altura, y que contenga un método que devuelva el área del rectángulo.

**Ejercicio 2**

Escribir una clase llamada círculo que contenga un radio, con un método que devuelva el área y otro que devuelva el perímetro del círculo.

**Ejercicio 3**

Escribir una clase llamada Persona que cumpla las siguientes condiciones:

- Atributos: nombre, edad, sexo (H hombre, M mujer), peso, altura.
- Métodos:
  - es\_mayor\_edad(): indica si es mayor de edad, devuelve un booleano.
  - print\_data(): imprime por pantalla toda la información del objeto.
  - generar\_dni(): genera un número aleatorio de 8 cifras y lo guarda dentro del atributo DNI. Llamar desde el método \_\_init\_\_, solo una vez.

**Ejercicio 4**

Escribir una clase Estudiante, que herede de Persona, y que agregue las siguientes condiciones:

- Atributos: nombre de la carrera, año de ingreso a la misma, cantidad de materias de la carrera, y cantidad de materias aprobadas.
- Metodos:
  - avance(): indica que porcentaje de la carrera tiene aprobada.
  - edad\_ingreso(): indica que edad tenia al ingresar a la carrera.

**Ejercicio 5**

Escribir una funcion que tome como parámetro una lista de Estudiantes, y devuelva un diccionario con las carreras como keys, y la cantidad de estudiantes en cada una de ellas como values.

**Ejercicio 6**

Programar una clase Persona con un constructor donde reciba una fecha de nacimiento (objeto datetime.datetime).

# La clase además debe contener un método edad, que no recibe nada y devuelva la edad de la persona (entero).

# Para obtener la fecha actual, usar el metodo de clase "now" de la clase datetime (ya importada).

```
class Persona:
    def __init__(self, nacimiento):
        pass

    def edad(self):
        pass
```