



Trabajo Práctico N° 8: USO DE CLASES BASE

1. Explica los siguientes métodos de la clase String:

- a. `charAt(int index);`
- b. `equals(String anotherString);`
- c. `equalsIgnoreCase(String anotherString)`
- d. `length();`
- e. `substring(beginIndex, endIndex);`
- f. `split(String regex)`
- g. `toLowerCase();`
- h. `toUpperCase();`
- i. `trim();`

Elegí 3 de ellos y desarrollá un ejemplo de código en dónde los usarías.

2. Explica los siguientes métodos de la clase Math:

- a. `abs(int a);`
- b. `abs(double a);`
- c. `abs(float a);`
- d. `abs(long a);`
- e. `pow(double a, double b)`
- f. `random()`
- g. `round(float a);`
- h. `round(double a);`
- i. `sqrt(double a);`

Elegí 3 de ellos y desarrollá un ejemplo de código en dónde los usarías.

3. Modifica la clase `CajeroAutomatico` desarrollada en el Trabajo Práctico N° 2, incorporando un menú de opciones que interactúe con el usuario y le permita realizar las siguientes acciones:

- a. `DEPOSITAR;`
- b. `EXTRAER;`
- c. `CONSULTAR;`
- d. `SALIR;`

Asegurate que al ingresar la opción deseada, el sistema valide que el valor ingresado es alguno de los esperados. En caso contrario requerí se ingrese el valor nuevamente.

4. Desarrolla una clase llamada `Password` que siga las siguientes condiciones:

- Que tenga los atributos `longitud` y `contraseña`. Por defecto, la longitud será de 8.



- Los constructores serán los siguiente:
 - Un constructor por defecto.
 - Un constructor con la longitud que nosotros le pasemos. Generará una contraseña aleatoria con esa longitud.
 - Los métodos que implementa serán:
 - `esFuerte()`: devuelve un booleano si es fuerte o no, para que sea fuerte debe tener más de 2 mayúsculas, más de 1 minúscula y más de 5 números.
 - `generarPassword()`: genera la contraseña del objeto con la longitud que tenga.
 - Método `get` para contraseña y longitud.
 - Método `set` para longitud.
5. Un `Tragamonedas` está compuesto por tres `Tambores`. Cuando el `Tragamonedas` se activa, giran los tres `Tambores`. Cada tambor se detiene en una posición comprendida entre 1 y 8. El `Tragamonedas` entrega un premio cada vez que, luego de ser activados los tres `Tambores` se detienen en la misma posición. Implementar las clases `Tragamonedas` y `Tambor`, y verificar su correcto funcionamiento.
 6. Crear una clase `Empleado` que contenga el atributo `fechaDeIngreso` y un método `calcularDiasDeVacaciones()` Sabiendo que después de los 6 meses de antigüedad se le otorgan 14 días, después de los 5 cinco años 21 días y más de 10 años 28 días.
 7. Modificá el ejercicio 2 del TP 4, haciendo que los días de la semana se puedan validar a través de un atributo de tipo `enum`.
 8. Modificá el ejercicio 3 del TP 4, haciendo que los distintos tipos de temperatura tomen sus valores validándolos a través de un tipo `enum`.
 9. Desarrollá la clase `PruebaCalculadora`, que ofrezca todas las funciones disponibles de la misma. Luego ofrece al usuario la posibilidad de ingresar los valores para realizar los cálculos correspondientes, mostrando por pantalla los resultados a cada uno.
 10. Modifica el ejercicio 4 del TP 6, de manera que la muestra pueda ingresarse por teclado. Tené en cuenta que los valores a ingresar deben ser validados, y sólo instancia a las personas una vez que cuentes con toda la información necesaria de la misma. Otra cosa, el valor de la muestra debe ser variable, con lo cual dejá que el usuario decida cuantas personas debe ingresar en la misma.