

Módulo Profesional 03:

Programación II

Actividades Prácticas Evaluables UF4

CICLO FORMATIVO DE GRADO SUPERIOR EN

DESARROLLO DE APLICACIONES MULTIPLATAFORMA

MODALIDAD ONLINE



Ejercicios Prácticos

Objetivos

- Seleccionar y emplear lenguajes, herramientas y librerías, interpretando las especificaciones para desarrollar aplicaciones multiplataforma con acceso a bases de datos
- Identificar los cambios tecnológicos, organizativos, económicos y laborales en su actividad, analizando sus implicaciones en el ámbito de trabajo, para mantener el espíritu de innovación

Competencias asociadas:

- Configurar y explotar sistemas informáticos, adaptando la configuración lógica del sistema según las necesidades de uso y los criterios establecidos

Metodología

- Preparación individual
- Para la realización del ejercicio se deberán visualizar todos los videotutoriales del curso.

Entrega

27 de octubre del 2020.

Se deberá entregar un único proyecto en Java o C++ comprimido en **Zip** con la nomenclatura

MP03_PII_UF04_APE.zip

(el resto de requisitos de entrega se describen en la sección descripción de la actividad).

Dedicación estimada

10 horas

Documentos de referencia

Videotutoriales de la UF.

Libro de referencia de la asignatura

Resultados de aprendizaje

- Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.

Criterios de evaluación

- Se han identificado los bloques que componen la estructura de un programa informático.
- Se han utilizado entornos integrados de desarrollo.
- Se han clasificado, reconocido y utilizado en expresiones los operadores del lenguaje.
- Se ha comprobado el funcionamiento de las conversiones de tipo explícitas e implícitas.
- Se han introducido comentarios en el código.

Enunciados

1. Crear una clase dentro de un proyecto llamada Persona con los siguientes atributos nombre, edad, dni, fecha caducidad, (dia, mes y año), dni, sexo, peso y altura. Crear todos los getters y setters, y crea un programa que cree tantos objetos como miembros de tu familia rellenando todos los atributos.
2. A partir del ejercicio anterior crear los siguientes métodos dentro de la clase,
 - a. esMayorDeEdad(): devuelve un booleano, true si es mayor a igual 18 años y false en caso contrario.
 - b. esMujer(): devuelve un booleano, true si es mujer y false si es hombre.
 - c. Print() devuelve el valor de todos los parametros separada por guiones.
3. A partir del ejercicio 2 añadir el siguiente método:
 - a. esDNICorrecto() devuelve un booleano
 - i. Un DNI está formado por 9 caracteres alfanuméricos: 8 números seguidos de una letra. Para poder validarlo he establecido las siguientes premisas:
 - ii. Debe tener nueve caracteres.
 - iii. Los primeros ocho deben ser números.
 - iv. Y el último debe ser una letra.
4. A partir del ejercicio 1, 2 y 3 crear un programa que permita comprobar si el dni es correcto para los miembros de tu familia.
5. A partir del ejercicio anterior crear un constructor sobrecargado.
 - a. Persona()
 - b. Persona(nombre, edad, dni, sexo, peso, altura)
 - c. Persona(dni)
6. Crear una clase Familia la cual es una clase abstracta de la cual hereda Persona. La clase Familia posee los atributos numero de libro de familia, direccion, población y provincia. También posee un método abstracto llamado IMC() que calcula el peso ideal de la persona (al ser abstracto se debe implementar en las clases hijas).

7. Crear la interfaz Humano, que debe implementar la clase Familia, que se compone de las siguientes declaraciones de métodos:
 - a. `actualizaDNI(dia, mes, anyo)` devuelve true si la fecha de la caducidad es superior a la introducida y false en caso contrario.
 - b. `esAlto()` devuelve un booleano y devuelve true si es superior a 1.80 y false en caso contrario.