

Trabajo práctico integrador.

Objetos

1. ¿Qué es una clase?
2. ¿Qué hacemos al instanciar?
3. ¿Qué es un objeto?
4. Definición de estado y comportamiento.
5. ¿Cuáles son los cuatro pilares de orientación a objetos? Definirlos.
6. ¿Cuál es la particularidad de una clase abstracta?
7. Escribir la estructura y componentes de un mensaje. Definir los tipos de mensajes.
8. Diferencia entre Self y Super.

C1		C2		C3		C4	
m6	^self m8	m9	^self m2	m2	^2	m1	^self m9
m10	^9	m8	^self m7	m5	^self m1	m3	^5
m13	^self m1	m12	^super m13	m7	^super m10	m4	^super m6
						m11	^super m12

Donde C1 es padre de C2, C2 de C3, y C3 de C4. (C1>C2>C3>C4).

Evaluar que devuelven las siguientes líneas

c:= C4 new.

c m4.

c m1.

c m11.

La UTN de La Plata necesita un sistema de gestión para llevar un registro de los alumnos, los docentes y las cátedras a las que están asociados.

9. Realizar un cartel que diga “Bienvenido al Sistema de Gestión UTN”.
10. Generar la clase persona (legajo, nombre, apellido, materias que cursa/dicta) y sus subclases alumno y docente (sueldo), junto con la clase cátedra (nombre, conjunto de alumnos, docente, comisión). Nota: Asumir que un docente puede dictar una única cátedra y que un alumno puede cursar varias asignaturas distintas.
11. Generar una clase “Cursada” que permita guardar un alumno, el nombre de una cátedra, el año en que se cursó y la nota obtenida.

12. Generar un menú, y por medio de un input permitir seleccionar la opción con la cual se va a operar, siendo cada uno de los incisos restantes (inciso 13 al 20) cada una de las opciones del menú.
13. Generar una opción que permita cargar tantos alumnos como el usuario quiera.
14. Idem anterior pero para cargar docentes y otro para cátedras.
15. Generar una validación que permita asegurarse de que no se repita el legajo del docente o alumno (- 💡 : *usar detect*).
16. Generar opciones para listar alumnos, docentes y materias (tres opciones por separado, no todas juntas).
17. Recuperar a todos los alumnos que están cursando Paradigmas de Programación en la S21.
18. Recuperar a todos los alumnos que promocionaron Sistemas Operativos en 2022.
19. Generar una colección de docentes excluyendo a aquellos que dicten Sistemas Operativos.
20. Hacer una comparativa desde 2015 hasta 2020 indicando la cantidad de alumnos que cursaron Paradigmas en cada año. ¿Hubo un incremento en el número de alumnos?. Utilizar un diccionario para guardar la cantidad por año.

Funcional

1. Dada una lista de enteros, retornar el promedio de los mismos.
2. Dada una lista de listas, reemplazar todos los caracteres 'i' por la letra 'y'.
3. Dada una lista de listas, retornar una lista que contenga los 3eros elementos de cada lista, siempre y cuando estos sean mayores a 5.
4. Dado una lista de listas, concatenar las listas en la que todos sus elementos sean pares.

Lógico

Dada la siguiente base de conocimientos:

```

catedra(1512,"Paradigmas de Programacion").%catedra(codigo, nombre)
catedra(1509,"Sistemas Operativos").
catedra(1511,"Sintaxis y Semántica del Lenguaje").
catedra(1513,"Ingeniería y Sociedad").
catedra(1510,"Análisis Matemático II").
catedra(1508,"Física II").
catedra(1514,"Análisis de Sistemas").

alumno(1,"Rodríguez","Emilio").%alumno(legajo,apellido,nombre)
alumno(2,"Ohara","Maximiliano").
alumno(3,"Herrera","Esteban").
alumno(4,"Iturriaga","Julia").
alumno(5,"Treval","Cecilia").
alumno(6,"Gimenez","Iara").
alumno(7,"Gonzalez","Sonia").
alumno(8,"Martinez","Roberto").

docente(1,"Acosta","Iara").%docente(legajo,apellido,nombre)
docente(2,"Luna","Darío").
docente(3,"Dorado","Luciano").
docente(4,"Salaberri","Tomás").
docente(5,"Bianchi","Stefania").
docente(6,"Draga","Tiara").
docente(7,"Yanini","Pablo").
docente(8,"Carrillo","María").

promociono(5,1509,10).%promociono(legajoAlumno,codigoCatedra,nota)
promociono(2,1512,6).
promociono(4,1512,9).
promociono(6,1508,10).
promociono(4,1511,7).
promociono(3,1513,8).
promociono(1,1511,10).
promociono(1,1514,8).
promociono(1,1508,9).
promociono(7,1509,6).
promociono(8,1512,7).
promociono(8,1511,7).

dicta(1,1514,"S24").%dicta(legajoDocente, codigoCatedra, comision)
dicta(2,1513,"S21").
dicta(7,1511,"S21").
dicta(6,1509,"S24").
dicta(3,1510,"S22").
dicta(4,1512,"S23").
dicta(5,1508,"S21").
dicta(8,1512,"S21").

curso(1,1513,"S23").%curso(legajoAlumno,codigoCatedra,comision)
curso(2,1513,"S23").
curso(1,1509,"S24").
curso(8,1513,"S23").
curso(3,1511,"S21").
curso(3,1512,"S23").
curso(5,1514,"S24").
curso(6,1514,"S24").
curso(8,1514,"S24").
curso(4,1508,"S21").
curso(7,1508,"S21").
curso(2,1510,"S22").
curso(1,1510,"S22").
curso(1,1512,"S21").
curso(3,1514,"S24").

```

1. Mostrar el legajo del alumno "Ohara".
2. Mostrar el nombre del docente "Bianchi".
3. Mostrar el legajo de los alumnos que cursen en la S23.
4. Generar una regla que dado un legajo retorne nombre y apellido de un docente.
5. Generar una regla que dado un nombre de cátedra, retorne el código de la misma junto con el apellido del docente a cargo.
6. Mostrar el código de la materia "Paradigmas de programación".
7. Mostrar nombre y apellido del docente que dicta la asignatura "Sistemas Operativos".
8. Generar una regla que muestre nombre y apellido de los alumnos que cursen en una comisión dada.
9. Generar una regla que dado un apellido de un alumno, retorne las materias que promocionó.
10. Dado un docente, retornar el nombre y apellido de los alumnos que cursan la materia que dicta.