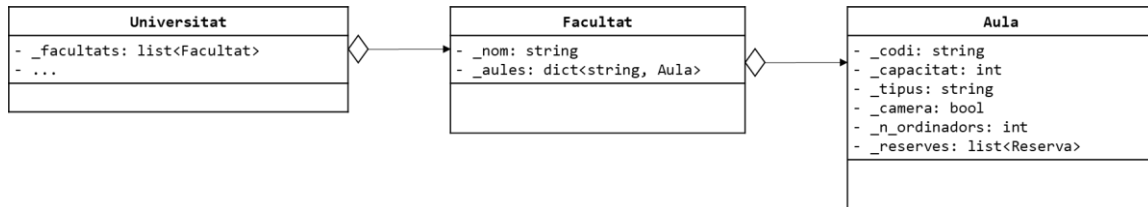


Volem implementar un conjunt de classes per poder gestionar l'ocupació de les aules i altres espais de la universitat (poder saber quines aules tenim a cada facultat, la seva capacitat, si estan reservades o no per un dia i hora concrets, etc.).

De moment hem creat aquest conjunt de classes:



Com podeu veure tenim creades aquestes 3 classes:

- **Aula:** és una classe que guarda la informació de cadascuna de les aules que hi ha a cada facultat de la universitat. Cada aula ve identificada per un codi i té una determinada capacitat (nº de persones que hi caben). Es guarda també el tipus d'aula (que pot ser laboratori o classe). Dels laboratoris guardem el nº d'ordinadors que hi ha al laboratori i de les classes si tenen càmera per enregistrar les sessions de classe. A més a més volem saber per cada aula quines reserves d'ocupació té fetes, és a dir, per quins dies i hores està ocupada. Aquesta informació la guardarem a l'atribut reserves que conté una llista d'objectes d'una classe Reserva que, de moment, no està creada.
- **Facultat:** és una classe que serveix per gestionar les aules de cadascuna de les facultats de la universitat. Cada facultat ve identificada pel seu nom i té com a atribut un diccionari per guardar totes les aules de la facultat, on la clau és el codi de l'aula i el valor l'objecte de la classe Aula corresponent.
- **Universitat:** és la classe que agrupa la informació de totes les facultats de la universitat. Té com a atribut una llista en què cada element és un objecte de la classe Facultat.

PRIMER PARCIAL

Nota: en aquesta primera part de l'examen només necessitareu treballar amb la classe `Aula`. Les classes `Facultat` i `Universitat` no són necessàries.

Volem **implementar les classes** necessàries per gestionar la **informació de les reserves** de les aules, **utilitzant herència** i tenint en compte les especificacions que us indiquem a continuació. Les reserves es poden fer per qualsevol dia de la setmana, de dilluns a divendres, en horari des de les 8 del matí fins a les 9 de la nit. Les reserves sempre es faran per hores completes començant i acabant a les hores en punt, però una reserva pot ser per més d'una hora consecutiva (per exemple des de les 9 del matí fins a les 12). El dia de la reserva s'indica especificant un nº de setmana i un dia dins de la setmana (de dilluns a divendres). Les setmanes s'identifiquen per un número que indica el nº de setmana des de l'inici del curs acadèmic. Quan es vol fer una reserva s'indica el nº de setmana, el dia de la setmana i l'hora d'inici i final (per exemple, reserva pel dimecres de la setmana nº 5, des de les 10 fins a les 12).

Les reserves podran ser de dos tipus: **reserves puntuals**, per un sol dia i hora concrets (tal com hem explicat abans) o **reserves periòdiques** que es repeteixen el mateix dia i hora de la setmana durant varies setmanes. En aquest darrer cas quan es fa la reserva s'ha d'especificar la setmana inicial i final de la reserva (per exemple reserva pel dimecres des de les 10 fins a les 12, des de la setmana nº 1 fins a la setmana nº 10).

Les reserves hauran de tenir mètodes que permetin fer les accions següents:

- **Llegir les dades de la reserva**, tenint en compte que haurà de ser diferent per les reserves puntuals (només necessitem un nº de setmana) que per les reserves periòdiques (necessitem setmana inicial i final).
- Poder **consultar** si la **reserva se solapa** amb una altra reserva potencial que es vol fer i que s'especifica indicant el dia de la setmana, l'hora d'inici i final i la setmana inicial i final (si la reserva és puntual, la setmana inicial i final seran iguals). Per exemple una reserva feta pel dimecres de la setmana nº 10 de 10 a 12, se solapa amb una altra reserva que es vulgui fer pels dimecres de les setmanes nº 1 fins la nº 12 de 11 a 13.

A partir d'aquestes especificacions **us demanem**:

1. **(2'5 punts) Completeu el disseny de classes** que us hem donat afegint les classes que siguin necessàries per gestionar les reserves. Dibuixeu el **diagrama de classes UML** complet. Tingueu en compte d'aplicar tots els conceptes que hem anat explicant:
 - Separació privat/públic.
 - Incloure els getters/setters necessaris (només els que siguin necessaris)
 - Utilitzar correctament l'herència per gestionar els diferents tipus de reserves.
 - Pensar i especificar bé els paràmetres i valor de retorn dels mètodes que faci falta declarar.
2. **(3 punts) Feu la implementació completa de la classe o classes** que afegiu al diagrama de classes per guardar les reserves.
3. **(2,5 punts) Feu la implementació d'un mètode de la classe `Aula`** que permeti **fer una nova reserva** per aquella Aula. Aquest mètode ha de rebre com a paràmetres el dia de la setmana, l'hora d'inici i final de la reserva, el nº de persones que han de poder cabre a l'aula, i el nº de setmana inicial i final de la reserva. Si el nº de setmana inicial i final són iguals la reserva s'ha de fer com una reserva puntual i si no, com una reserva periòdica. Per fer la reserva, primer s'ha de comprovar que l'aula tingui capacitat suficient i que no estigui prèviament reservada pel període indicat (és a dir que no hi hagi cap reserva ja existent per aquesta aula que se solapi amb el període de la nova reserva). Si la reserva es pot fer s'haurà de crear (del tipus que correspongui) i afegir a la llista de reserves de l'aula.

4. **(2 punts)** Feu la **implementació** d'un **mètode de la classe Aula** que rebi com a paràmetre un nº de setmana i **retorni un array en numpy** que indiqui la **disponibilitat de l'aula** per aquella setmana en funció de les reserves que hi ha fetes. L'array haurà de tenir tantes files com hores possibles (des de les 8 del matí fins a les 9 del vespre) i tantes columnes com dies de la setmana (des de dilluns fins a divendres). A cada posició de l'array hi haurà un 0 si l'aula està disponible en aquell dia i hora i un 1 si hi ha alguna reserva per aquell dia i hora.