

Pràctica 1 Programació 2

Continguts:

1. [Llista de tasques](#)
2. [Explicar les classes implementades](#)
3. [Diagrama de relacions](#)
4. [Explicar quins són els atributs de la classe Reserva i perquè](#)
5. [Expliqueu on s'implementen els mètodes correcteFuncionament i perquè.](#)
6. [Expliqueu en quines classes heu implementat el mètode toString i perquè.](#)
7. [Expliqueu si seria correcte que es creïn el soci i el servei quan es crea una reserva.](#)
8. [Detallar les proves realitzades per comprovar el correcte funcionament de la pràctica, resultats obtinguts i accions derivades](#)
9. [Observacions generals.](#)

1. Llista de tasques

- o Implementar la classe PistaTennis (James Chen)
- o Implementar la classe PistaFutbol (James Chen)
- o Implementar la classe Gimas (Albert Villanueva)
- o Implementar la classe LlistaReserves (James Chen)
- o Implementar la classe Pista (Albert Villanueva)
- o Implementar la classe Reserva (James Chen)
- o Implementar la classe SalaFitness (Albert Villanueva)
- o Implementar la classe Servei (James Chen)
- o Implementar la classe Soci (Albert Villanueva)
- o Implementar la classe GestorServeisGimnas (Albert Villanueva)

2. Cal explicar les classes implementades

Classe GestorServeisGimnas:

La classe GestorServeisGimnas és la classe principal del projecte, on es troba el Main, on s'omplen les dades. En el Main es crea un objecte de la classe gimnas anomenat gimnasMediterrani, la classe gimnas adquireix les dades per mitjà del mètode omplirDadesModel, on afegeix les variables necessàries per crear els camps de tenis, futbol i gimnàs a més dels socis. Aquestes dades recopilades pel mètode, s'utilitzen per al mètode ferReserves, el qual agafa la identificació del servei i la persona, i el temps de la reserva per fer la reserva.

Classe Gimnas:

Totes les dades recopilades per omplirDadesModel són emmagatzemades dins d'aquesta classe. Aquí s'hi troba el seu constructor on s'accepta el nom del gimnàs, els mètodes getters getNom que retorna el nom del gimnàs, i getNumServeis que retorna el nombre de serveis del gimnàs. Els mètodes per afegir serveis, afegirSoci, afegirPistaTenis, afegirPistaFutbol, afegirSalaFitness i afegirReserva. Els mètodes calculMidaTotalPistes i calculServeisOperatius.

Classe Servei:

La classe servei és una classe abstracta que serveix per a representar els diferents tipus de serveis que ofereix el gimnàs. Mitjançant els mètodes getters i setters, permeten poder utilitzar les variables de Servei les subclasses heretades d'aquesta (Pista, PistaFutbol, PistaTenis).

Classe Soci:

La classe soci utilitza les variables nom i DNI per donar-li informació als mètodes buscarSoci i afegirSoci.

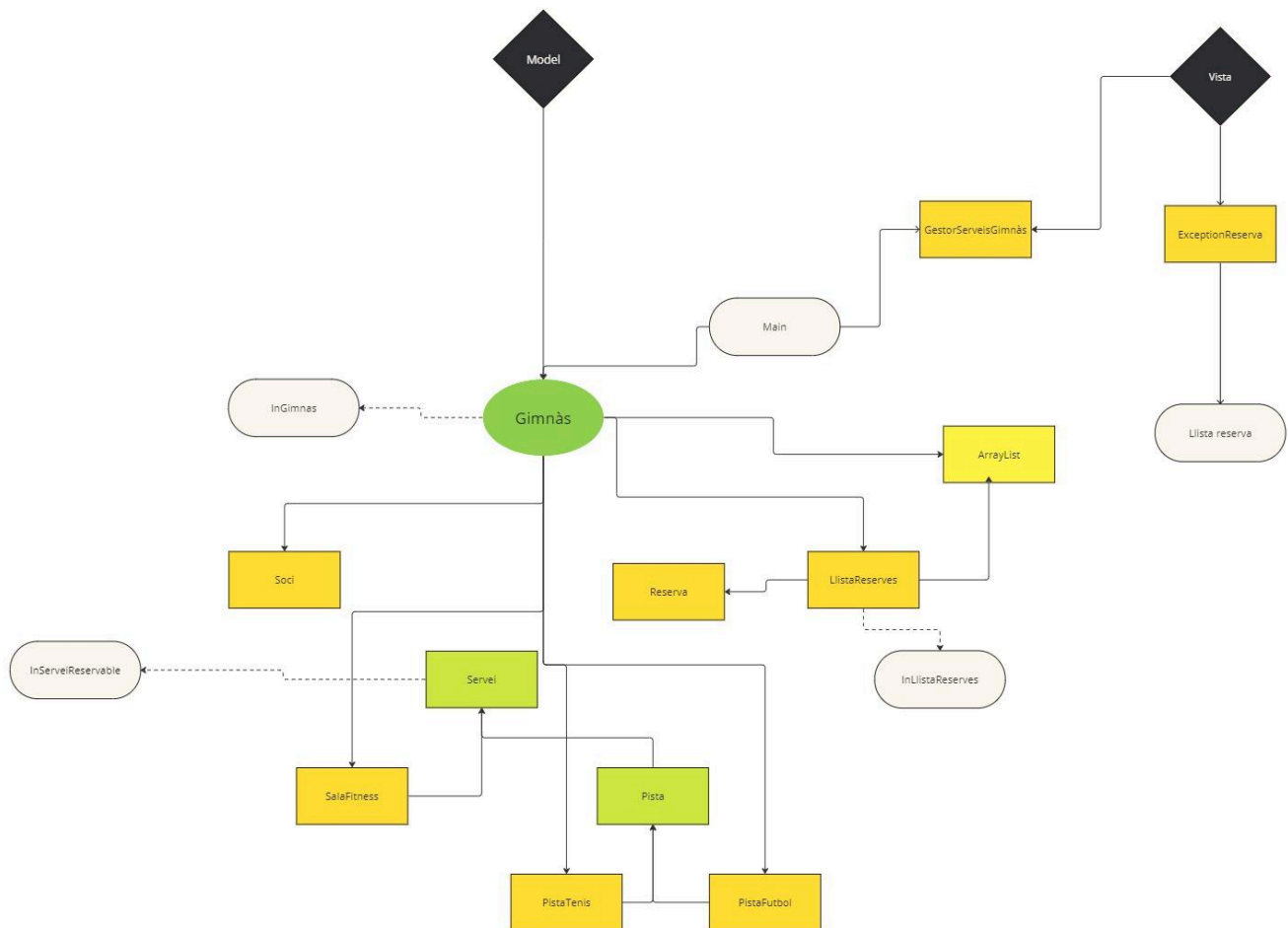
Classe Reserva:

La classe reserva amb les dades emmagatzemades proporcionades per les classes servei, soci i data representa una reserva específica realitzada per un soci per un servei en particular en un moment donat.

Classe LlistaReserves:

Aquesta classe utilitza les dades proporcionades per reserva on decideix si posarà les dades que ha rebut per mitjà dels mètodes `afegirReserva` i `serveiDisponible`, on verifiquen la disponibilitat del servei dins del rang de temps que el soci vol fer la reserva.

3. Diagrama de relacions



4. Explicar quins són els atributs de la classe Reserva

Reserva es defineix per 3 atributs: soci, servei i data. Soci és un atribut de la classe Soci, aquest serveix per saber quina persona vol fer la reserva. Servei és un atribut de la classe Servei, aquest serveix per saber quin tipus de servei està fent servir el soci, i per saber quan vol fer la reserva el soci s'utilitza l'atribut data, que realitza la reserva a l'hora determinada al servei corresponent.

Es necessita l'atribut data per emmagatzemar el temps que es vol fer reservar el servei. A més, cada servei té el seu horari amb reserves, per tant, per diferenciar les diferents dates dels serveis és necessari tenir un atribut de la classe servei per guardar tots els serveis. I per saber qui vol fer servir el servei es necessita l'atribut de classe Soci.

5. Expliqueu on s'implementen els mètodes correcteFuncionament i perquè

Els mètodes correcteFuncionament s'implementen al mètode calculServeisOperatius per saber el nombre de serveis que es poden utilitzar que compleixen els requisits. També s'implementa al mètode afegirReserva on verifica si el servei que es vol reservar es pot utilitzar.

6. Expliqueu en quines classes heu implementat el mètode toString

El mètode toString s'ha implementat a les classes pistaFutbol, pistaTenis, salaFitness, Reserva, Servei i Soci. Aquest mètode recopila les dades que s'han fet servir a la classe i les imprimeix.

7. Expliqueu si seria correcte que es creïn el soci i el servei quan es crea una reserva

El problema del codi de l'enunciat és que s'està creant un nou índex de l'array de servei i soci cada vegada que es crea una reserva, i com no es comprova si el servei i soci existeixen, o si

el servei o soci estan repetits, hi haurà distints índexs de socis i serveis de la mateixa persona i servei, i és un problema perquè al fer una nova reserva d'un servei, els solapaments dins d'aquest servei no es poden detectar, ja que totes les reserves del servei es situen en diferents índexs de serveis.

Per exemple, si una persona reserva una pista de futbol a les 14:00, i una altra persona reserva la mateixa pista també a les 14:00, i com es crea un nou servei per aquesta reserva, no detecta que la pista de futbol ja va ser reservada per la primera persona.

8. Detallar les proves realitzades per comprovar el correcte funcionament de la pràctica, resultats obtinguts i accions derivades

Per comprovar el correcte funcionament de la pràctica hem utilitzat aquestes dues estratègies:

Primer, hem provat tots els mètodes per separat. Hem comprovat el seu funcionament amb dades i situacions similars que en la pràctica en entorns externs a la pràctica.

Després hem depurat el projecte i hem obtingut errors principalment derivants de la implementació de variables en la jerarquia de classes. Això ens ha permès veure que no vam entendre completament el funcionament de la jerarquia de classes. Arran d'aquesta situació, vam fer els canvis necessaris.

Finalment, vam comparar els nostres resultats amb els dels nostres companys i vam corregir algunes incorreccions que permetien el funcionament del codi, però, no eren del tot eficients o eficaços.

9. Observacions generals

En aquesta pràctica hem après a desenvolupar un sistema informàtic per gestionar els serveis d'un gimnàs i permetre als seus socis reservar aquests serveis.

Aquest projecte, sent limitat en les seves aplicacions reals, proporciona una petita mostra del que és el desenvolupament de programes orientats a objectes que és integral pel software de l'actualitat.

Des del punt de vista tècnic, hem après a treballar amb herència de classes i polimorfisme. A més, ha sigut el primer projecte de programació en el que hem treballat amb una multitud de classes abstractes i fitxers diferents. Proporcionant un cop d'ull en què és el desenvolupament d'aplicacions simples.

En definitiva, creiem que amb aquesta pràctica hem après més quant als aspectes tècnics que hem mencionat que si haguéssim intentat aprendre aquests conceptes de manera teòrica.