



APELLIDOS, Nombre

TITULACIÓN Y GRUPO

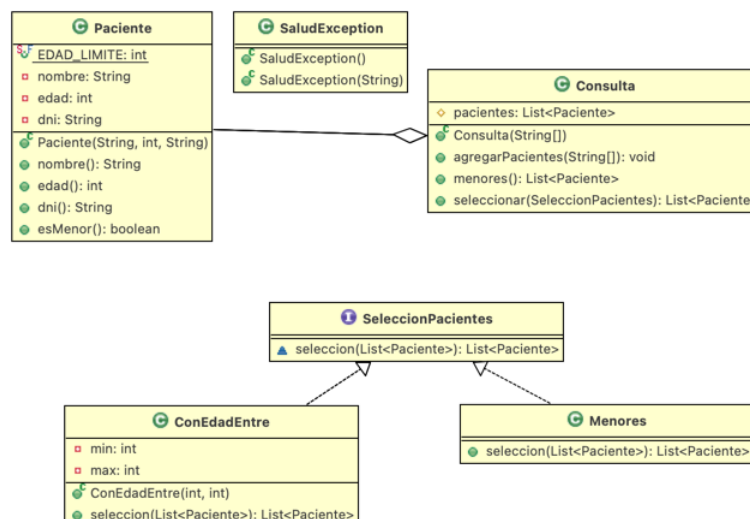
MÁQUINA

NOTAS PARA LA REALIZACIÓN DEL EJERCICIO:

- Al inicio del contenido de cada fichero realizado deberá aparecer un comentario con tus apellidos y nombre, titulación y grupo.
- Los diferentes apartados tienen una determinada puntuación. Si un apartado no lo sabes hacer, *no debes pararte en él indefinidamente*. Puedes abordar otros.
- **Está permitido:**
 - Consultar los apuntes (CV), y la guía rápida de la API (CV).
 - Añadir métodos privados a las clases.
- **No está permitido:**
 - Intercambiar documentación con otros compañeros.
 - Recibir ayuda de otras personas. Se debe realizar personal e individualmente la solución del ejercicio propuesto.
 - Añadir métodos no privados a las clases.
 - Añadir variables o constantes a las clases.
 - Modificar la visibilidad de las variables, constantes y métodos que se indican.
 - Modificar el código suministrado.
- Una vez terminado el ejercicio, debéis subir (a la tarea creada en el campus virtual para ello) un fichero comprimido de la carpeta **src** que hayáis realizado y usáis vuestros apellidos y nombre para su denominación (**Apellido1Apellido2Nombre.rar** o **.zip**).
- La evaluación tendrá en cuenta la claridad de los algoritmos, del código y la correcta elección de las estructuras de datos, así como los criterios de diseño que favorezcan la reutilización.
- Para la corrección del ejercicio se utilizarán **programas de detección de copias/plagios**.
- Con posterioridad a la realización del ejercicio, el profesor podrá convocar a determinado/as alumno/as para realizar **entrevistas personales** con objeto de comprobar la autoría de las soluciones entregadas.

Proyecto prPacientes

Se va a crear una aplicación para modelar consultas en un centro de salud. Para ello se crearán en el paquete salud las clases que siguen: SaludException, Paciente y Consulta; además de la interfaz SeleccionPacientes y las clases ConEdadEntre y Menores. También hay que implementar la clase PruebaPaciente en el paquete por defecto. Se proporciona una versión simplificada de la clase Paciente, correspondiente a una solución de la anterior prueba de seguimiento.



1) **(0,25 puntos)** Defínase la clase `SaludException` que represente excepciones **no comprobadas** para tratar las situaciones excepcionales en el proyecto.

2) **(1,50 puntos)** Se debe modificar la clase `Paciente` para que tenga la funcionalidad que se indica a continuación, y que incluya una constante de clase (`final` y `static`) que almacene una edad límite de 14 años (de tipo `int`) que se utilizará en el apartado (c):

a) Modifíquese adecuadamente el constructor de la clase para que se lance una excepción de tipo `SaludException` cuando la edad sea negativa y cuando el DNI no tenga 9 caracteres o el último no sea una letra. Para esto último, utilícese el método `isLetter` del envoltorio `Character`.

b) Dos pacientes se considerarán iguales cuando sus nombres coincidan (independientemente de mayúsculas o minúsculas) y también sus DNI (independientemente de mayúsculas y minúsculas).

c) Un método:

```
boolean esMenor()
```

que devuelva `true` cuando la edad sea menor o igual que 14 (utilícese la constante incluida en la clase).

3) **(1,25 puntos)** Debe definirse una aplicación (clase distinguida) `PruebaPaciente`, en el paquete por defecto, que pruebe la clase anterior, creando dos objetos de la clase `Paciente`, con los datos siguientes (nombre, edad, DNI):

```
"pablo ruiz picasso", 18, "12345678A"
```

```
"PABLO RUIZ PICASSO", 81, "12345678a"
```

Posteriormente, se mostrarán en consola ambos pacientes y un mensaje indicando si los pacientes son iguales o no.

A continuación, debe crearse un tercer objeto de la clase `Paciente` con el nombre, edad, y DNI proporcionados, en ese orden, en el array de argumentos del `main`. Posteriormente, debe mostrarse en consola.

Como es posible que la edad que se proporcione en los argumentos no sea numérica o sea negativa, o que el número de argumentos sea insuficiente, deben capturarse las correspondientes excepciones, mostrando sobre la variable `System.err` un mensaje indicando el tipo de incidencia específica en cada caso.

Por ejemplo, si los argumentos que se pasan son:

```
Cervantes XX 87654321Z
```

la salida puede ser algo como:

```
p678
```

```
P678
```

```
Los pacientes sí son iguales
```

```
La edad debe ser numérica
```

4) **(4,5 puntos)** La clase `Consulta` pretende representar consultas de un centro de salud. Deberá mantener información en una lista con los pacientes que deben ser atendidos (`List<Paciente>`). La clase incluirá:

a) Un constructor con un argumento que es un array de cadenas de caracteres, donde cada una incluirá información sobre pacientes a ser atendidos, con el siguiente formato:

```
Nombre(edad)DNI
```

Con esa información deberán crearse pacientes y añadirlos a la lista de pacientes.
(Nota: Utilícese el método del siguiente apartado).

- b) Un método para agregar los pacientes cuya información se encuentra en el array de cadenas de caracteres que se pasa como argumento:

```
void agregarPacientes(String[] infoPacientes)
```

Cada elemento del array tendrá el formato indicado en el apartado anterior. En caso de que el formato de algún elemento del array no sea correcto, o que la edad no sea un número entero o sea negativo, se lanzará una excepción de tipo `SaludException`, con un mensaje identificativo del tipo de anomalía detectada. Por ejemplo, dependiendo del caso:

```
Faltan datos: Cervantes(68)
```

```
Edad no numérica: Cervantes(XX)87654321Z
```

```
La edad no puede ser negativa: Cervantes(-68):87654321Z
```

- c) Un método

```
List<Paciente> menores()
```

que devuelva una lista con los pacientes considerados menores (con menos de 14 años).

- d) La representación textual de una consulta vendrá dada por la secuencia de la representación textual de los pacientes, separados por " : " y encerrados entre llaves. Por ejemplo:

```
{ Pablo Ruiz Picasso 678 : Salvador Rueda Santos 321 : María Zambrano  
Alarcón 234 : Miguel de Cervantes 222 }
```

Utilícese para ello `StringJoiner` o `StringBuilder`, según se considere.

- 5) **(3 puntos)** Con objeto de no tener que definir en la clase `Consulta` distintos métodos para seleccionar pacientes que cumplan determinado criterio (por ejemplo, el que se utiliza en el apartado (c) del ejercicio anterior), se deben definir:

- a) La interfaz `SeleccionPacientes`, que incluirá un solo método:

```
List<Paciente> seleccion(List<Paciente> pacientes)
```

con la intención de devolver una lista con los pacientes que cumplen un determinado criterio entre los que componen la lista que se pasa como argumento. Ese criterio será definido por clases (como las de los apartados siguientes) que implementen la interfaz.

- b) La clase `ConEdadEntre` implementará la interfaz anterior, de forma que incluirá:

- Dos variables de instancia (`int`) que almacenarán dos valores la edad mínima y máxima entre las que se seleccionarán los pacientes.
- Un constructor que permita inicializar esas variables de instancia, asegurando que el mínimo es menor o igual que el máximo. Si no fuese así, se debe lanzar una excepción de tipo `SaludException`.
- La implementación del método `seleccion`, que devuelva una lista con los pacientes de la lista que se pasa como argumento cuya edad esté entre el mínimo y el máximo considerados, ambos inclusive.

- c) La clase `Menores` implementará también la interfaz del apartado (a) de forma que incluirá:

- La implementación del método `seleccion`, que devuelva una lista con los pacientes considerados menores (con 14 o menos años).

- d) Incluir en la clase `Consulta` el método:

```
List<Paciente> seleccionar(SeleccionPacientes seleccionador)
```

que devuelva la lista de pacientes que cumplan el criterio de selección que se pasa como argumento.

Se proporciona una clase de pruebas para incluir en el paquete por defecto, PruebaConsulta, para ilustrar el uso de las clases propuestas, y cuya salida debería ser algo como:

```
{ Pablo Ruiz Picasso 678 : Josefa Flores Manzano 765 : Salvador Rueda Santos
321 : Maria Zambrano Alarcón 234 : Miguel de Cervantes 222 }
Pacientes menores: [Salvador Rueda Santos 321]
Pacientes entre 15 y 50 años: [Pablo Ruiz Picasso 678, Josefa Flores Manzano
765, Maria Zambrano Alarcón 234]
Pacientes menores: [Salvador Rueda Santos 321]
```