

Objectius: <ul style="list-style-type: none">- Conèixer com llegir i escriure fitxers JSON amb Java
Instruccions: <ul style="list-style-type: none">- Responen a l'espai de cada pregunta, si ho feu amb diapositives enganxeu la diapositiva en aquest mateix espai.- Es valorarà la presentació i els comentaris al codi
Criteris d'avaluació: <ul style="list-style-type: none">- Cada exercici té la mateixa puntuació
Entrega: <ul style="list-style-type: none">- Un arxiu .zip anomenat: PRx.y-NomCognom.zip<ul style="list-style-type: none">• PRx.y correspon al codi de la pràctica, per exemple PR1.1• NomCognom correspon al nom i primer cognom de cada participant- L'arxiu .zip conte:<ul style="list-style-type: none">• Aquest document emplenat en format .pdf anomenat memoria.pdf• Els arxius necessaris per fer anar la pràctica

Nom i Cognom: [Joel Berzal](#)

URL repositori GitHub: <https://github.com/joelberzalgithub/PR1.4b-BerzalJoel>

Materials:

- Necessiteu una eina per programar en JAVA
- Feu servir Google per buscar els tutorials que us serveixin millor
- Repositori de referència: <https://github.com/jpala4-ieti/DAM2-MP06-Repo-Referencia>

Tasques, a cada exercici feu l'explicació i captures que cregueu convenients.

Preparació

Si no l'has creat, crea un 'Main.java' amb un menú per cridar cada una de les classes amb funció "main" d'aquesta activitat (consulta anterior activitats).

```
17 public class Main {
18
19     static Scanner in = new Scanner(System.in); // System.in és global
20
21     public static void main(String[] args) throws InterruptedException, IOException {
22
23         try {
24
25             boolean running = true;
26
27             while (running) {
28
29                 System.out.println("\n*****\n\r
30                 |\n0) Lectura del fitxer 'llibres_input.json' +
31                 |\n1) Modificar l'any de publicacio del llibre amb id 1 a 1995" +
32                 |\n2) Afegir un nou llibre amb id 4, titol 'Histories de la ciutat', autor 'Miquel Soler', any 2022" +
33                 |\n3) Esborrar el llibre amb id 2\n100) Sortir\n");
34
35                 try {
36
37                     int opcio = Integer.valueOf(llegirLinia("Opcio: "));
38
39                     switch (opcio) {
40                         case 0: lectura();
41                             break;
42                         case 1: modificar();
43                             break;
44                         case 2: afegir();
45                             break;
46                         case 3: esborrar();
47                             break;
48                         case 100: running = false;
49                             break;
50                         default: System.out.println("\nOpcio fora del rang!");
51                             break;
52                     }
53
54                 } catch (Exception e) {
55                     System.out.println("\nOpcio no numerical!");
56                 }
57             }
58             in.close();
59         } catch (Exception e) {
60             System.out.println("\nOpcio no numerical!");
61         }
62     }
63 }
```

Exercici: Gestió d'una Llibreria Digital

Crea un programa anomenat `PR143GestioLlibreriaMain.java` que compleixi la següent especificació:

Context:

Ets el desenvolupador d'una petita llibreria digital. Les dades dels llibres estan emmagatzemades en un fitxer JSON anomenat `llibres_input.json`. Cada llibre té un títol, autor, any de publicació i un identificador únic.

Contingut del fitxer `llibres_input.json`:

```
[
  {"id": 1, "títol": "Els vents del nord", "autor": "Clara Rojas", "any": 1990},
  {"id": 2, "títol": "Muntanyes màgiques", "autor": "Joan Peris", "any": 2005},
  {"id": 3, "títol": "El secret del mar", "autor": "Laura Vidal", "any": 2010}
]
```

Tasques a realitzar:

- **Lectura del fitxer (2 punts):** Llegeix el fitxer `llibres_input.json` i carrega'l a una estructura de dades en memòria.

Per poder dur a terme aquest exercici, primer he hagut de crear un objecte `ObjectMapper`.

```
68 public static void lectura() {
69
70     try {
71         // Crea un objecte ObjectMapper de Jackson
72         ObjectMapper mapper = new ObjectMapper();
```

Després, he hagut de llegir l'arxiu JSON i carregar-lo en una llista de `HashMaps`.

```
73
74         // Llegeix l'arxiu JSON i carrega el seu contingut en una llista de HashMaps
75         List<Map<String, Object>> llibres = mapper.readValue(new File("data/llibres_input.json"), List.class);
76     }
```

Finalment, he hagut de crear un bucle *for* per poder imprimir les dades de l'arxiu JSON.

```
77         // Imprimeix les dades de l'arxiu JSON
78         System.out.println("");
79         for (Map<String, Object> llibre : llibres) {
80             System.out.println(llibre);
81         }
82     } catch (IOException e) { e.printStackTrace(); }
84 }
```

- **Modificació (2 punts):** Canvia l'any de publicació del llibre amb id 1 a 1995.

Per poder dur a terme aquest exercici, primer he hagut de crear un objecte `ObjectMapper`.

```
86 public static void modificar() {  
87     try {  
88         // Crea un objecte ObjectMapper de Jackson  
89         ObjectMapper mapper = new ObjectMapper();  
90     }
```

Després, he hagut de llegir l'arxiu `llibres_input.json` i carregar-lo en una llista de `HashMaps`.

```
91     // Llegeix l'arxiu JSON i carrega el seu contingut en una llista de HashMaps  
92     List<Map<String, Object>> llibres = mapper.readValue(new File("data/llibres_input.json"), List.class);  
93  
94 }
```

Seguidament, he hagut de modificar l'any de publicació del llibre amb id 1 a 1995.

```
95     // Modifica l'any de publicació del llibre amb id 1  
96     for (Map<String, Object> llibre : llibres) {  
97         if (llibre.containsKey("id") && Integer.parseInt(llibre.get("id").toString()) == 1) {  
98             llibre.put("any", 1995);  
99             break;  
100         }  
101     }
```

Finalment, he hagut de desar aquests canvis a l'arxiu `llibres_output.json`.

```
103     // Desa l'arxiu JSON actualitzat  
104     mapper.writeValue(new File("data/llibres_output.json"), llibres);  
105     System.out.println("\nDades guardades amb exit a llibres_output.json!");  
106  
107 } catch (IOException e) { e.printStackTrace(); }  
108 }
```

- **Afegit (2 punts):** Afegeix un nou llibre amb id 4, títol “Històries de la ciutat”, autor “Miquel Soler”, any 2022.

Per poder dur a terme aquest exercici, primer he hagut de crear un objecte `ObjectMapper`.

```
110 public static void afegir() {  
111     try {  
112         // Crea un objecte ObjectMapper de Jackson  
113         ObjectMapper mapper = new ObjectMapper();
```

Després, he hagut de llegir l'arxiu `llibres_input.json` i carregar-lo a un `ArrayNode`.

```
115  
116         // Llegeix l'arxiu JSON actual i carrega el seu contingut en un ArrayNode  
117         ArrayNode node = (ArrayNode) mapper.readTree(new File("data/llibres_input.json"));  
118     }
```

Seguidament, he hagut de comprovar si ja el llibre que es demana a l'enunciat ja estava creat, i per fer-ho he creat un bucle `for` el qual, en detectar que dins de l'arxiu `llibres_input.json` existia un llibre amb id 4, canvia el valor d'un booleà anomenat `bookExists` a `true`.

```
119         // Comprova si ja existeix un llibre amb id 4  
120         boolean bookExists = false;  
121         for (JsonNode jsonNode : node) {  
122             if (jsonNode.has("id") && Integer.parseInt(jsonNode.get("id").toString()) == 4) {  
123                 bookExists = true;  
124                 break;  
125             }  
126         }
```

Finalment, he hagut crear el llibre i desar-lo a l'arxiu `llibres_output.json` en cas que el booleà `bookExists` tingués valor `false`, ja que del contrari s'imprimiria per pantalla un missatge indicant que el llibre ja existeix.

```
128         if (!bookExists) {  
129             // Crea un nou HashMap en el qual s'insereixen les dades del nou llibre  
130             Map<String, Object> map = new LinkedHashMap<>();  
131             map.put("id", 4);  
132             map.put("titol", "Histories de la ciutat");  
133             map.put("autor", "Miquel Soler");  
134             map.put("any", 2022);  
135  
136             // Agrega el nou llibre a l'ArrayNode  
137             node.add(mapper.valueToTree(map));  
138  
139             // Desa l'arxiu JSON actualitzat  
140             mapper.writeValue(new File("data/llibres_output.json"), node);  
141             System.out.println("\nDades guardades amb exit a llibres_output.json!");  
142         }  
143         else {  
144             System.out.println("Aquest llibre ja existeix!");  
145         }  
146  
147     } catch (IOException e) { e.printStackTrace(); }  
148 }
```

- **Esborrat (2 punts):** Esborra el llibre amb id 2.

Per poder dur a terme aquest exercici, primer he hagut de crear un objecte `ObjectMapper`.

```
150 public static void esborrar() {  
151  
152     try {  
153         // Crea un objecte ObjectMapper de Jackson  
154         ObjectMapper mapper = new ObjectMapper();
```

Després, he hagut de llegir l'arxiu `llibres_input.json` i carregar-lo en una llista de `HashMaps`.

```
155  
156         // Llegeix l'arxiu JSON i carrega el seu contingut en una llista de HashMaps  
157         List<Map<String, Object>> llibres = mapper.readValue(new File("data/llibres_input.json"), List.class);  
158
```

Seguidament, he hagut de crear un bucle `for` per poder cercar el llibre amb id 2 i eliminar-lo.

```
159         // Cerca el llibre amb id 2 i l'elimina de la llista  
160         for (Map<String, Object> llibre : llibres) {  
161             if (llibre.containsKey("id") && Integer.parseInt(llibre.get("id").toString()) == 2) {  
162                 llibres.remove(llibre);  
163                 break;  
164             }  
165         }
```

Finalment, he hagut de desar aquests canvis a l'arxiu `llibres_output.json`.

```
167         // Desa l'arxiu JSON actualitzat  
168         mapper.writeValue(new File("data/llibres_output.json"), llibres);  
169         System.out.println("\nDades guardades amb exit a llibres_output.json!");  
170  
171     } catch (IOException e) { e.printStackTrace(); }  
172 }
```

- **Guardat (2 punts):** Guarda les dades modificades en un fitxer nou anomenat `llibres_output.json`.

Aquest exercici l'he dut a terme durant tota la pràctica, ja que com he mostrat en tots els casos anteriors (menys en el primer), sempre que he hagut de fer alguna modificació, l'he desat a l'arxiu `llibres_output.json`.

Expectatives després de l'execució del programa:

```
[  
    {"id": 1, "títol": "Els vents del nord", "autor": "Clara Rojas", "any": 1995},  
    {"id": 3, "títol": "El secret del mar", "autor": "Laura Vidal", "any": 2010},  
    {"id": 4, "títol": "Històries de la ciutat", "autor": "Miquel Soler", "any": 2022}  
]
```