

```

"""
Este indice crea un map cuya llave es el autor del libro
"""

catalog['authors'] = mp.newMap(800,
                              maptype='CHAINING',
                              loadfactor=4.0,
                              comparefunction=compareAuthorsByName)

```

- a) ¿Qué estructura de datos se usa para este índice?  
El tipo de tabla de hash es “Separate Chaining”
- b) ¿Cuántos elementos se espera almacenar inicialmente?  
Inicialmente se esperan almacenar 800 elementos
- c) ¿Cuál es el factor de carga máximo?  
El factor de carga maximo es 4.0

```
mp.put(catalog['bookIds'], book['goodreads_book_id'], book)
```

- a) ¿Qué hace la instrucción “mp.put(...)”?  
Agrega una pareja llave-valor a la tabla seleccionada (En este caso catalog['bookIds'])
- b) ¿Qué papel cumple “book['goodreads\_book\_id']” en esa instrucción?  
Es la llave con que se va a guardar el valor en la lista.
- c) ¿Qué papel cumple el tercer parámetro “book” en esa instrucción?  
Es el valor que va a tener la llave book['goodreads\_book\_id'] en la lista.

```

def getBooksByYear(catalog, year):
    """
    Retorna los libros publicados en un año
    """
    year = mp.get(catalog['years'], year)
    if year:
        return me.getValue(year)['books']
    return None

```

- a) ¿Qué hace la instrucción “mp.get(...)”?  
Retorna la pareja llave-valor de la tabla que esta en el primer parametro y que su llave sea igual al segundo parametro si no lo encuentra, devuelve None
- b) ¿Qué papel cumple “year” en esa instrucción?  
Es la llave que se va a buscar.
- c) ¿Qué hace la instrucción “me.getValue(...)”?  
Devuelve el valor que esta en la llave.