

# Programación Orientada a Objetos

## Curso 2023/24

### Sesión 9

#### Ordenación de colecciones

En esta sección se proponen modificaciones al ejercicio de la **sesión 7** (viajes) para incluir la ordenación de algunas colecciones:

- Implementa el orden natural (interfaz Comparable) en la clase Viaje. El criterio de ordenación está basado en la fecha de salida. Ten en cuenta que el tipo de datos LocalDate dispone de orden natural que corresponde al orden cronológico.
- En el programa principal ordena la lista de viajes una vez se han introducido los tres viajes. Muestra en la consola el contenido de la lista.
- Implementa un criterio de ordenación (interfaz Comparator) diferente para los viajes, de acuerdo al orden alfabético de la propiedad ruta. Las cadenas disponen de orden natural que corresponde a la ordenación alfabética ascendente. Después del último bucle del programa principal, ordena la colección de viajes atendiendo a este nuevo criterio y muestra la colección en la consola
- Declara en la clase Viaje un método sobrecargado para la consulta de reservas (getReservas) que esté parametrizado con un criterio de ordenación (interfaz Comparator).
- Implementa dos criterios de ordenación para las reservas. El primer criterio está basado en el orden natural de la propiedad usuario y en caso de empate en el orden natural de la fecha de reserva. El segundo criterio de ordenación se apoya en el orden natural de los códigos de reserva.
- En el programa principal:
  - o Añade la reserva de una plaza del usuario "Antonio" al viaje selectivo.
  - o Una vez hechas las reservas, muestra por la consola las reservas del viaje selectivo ordenadas de acuerdo a los dos criterios programados anteriormente.