

Tema: 1

Apellido y Nombre: _____

Dada la clase Pasajero:

```
public class Pasajero {  
  
    private int dni;  
    private String nombre;  
    private String apellido  
    ;  
    public Pasajero(int dni, String nombre, String apellido) {  
  
        this.dni = dni;  
        this.nombre = nombre;  
        this.apellido = apellido;  
    }  
  
    public String getApellido() {  
        return apellido;  
    }  
  
    public void setApellido(String apellido) {  
        this.apellido = apellido;  
    }  
  
    public int getDni() {  
        return dni;  
    }  
    public void setDni(int dni) {  
        this.dni = dni;  
    }  
  
    public String getNombre() {  
        return nombre;  
    }  
  
    public void setNombre(String nombre) {  
        this.nombre = nombre;  
    }  
  
    public String toString() {  
        return "Pasajero [dni=" + dni + ", nombre=" + nombre + ", apellido=" +  
apellido + " ]";  
    }  
}
```

NOMBRE Y APELLIDO

DNI

FIRMA

1. Desarrollar la clase Vuelo, siguiendo la descripción de cada método.

```
public class Vuelo {

    private final int CANTIDAD_DE_FILAS = 32;
    private final int CANTIDAD_DE_ASIENTOS_POR_FILA = 6;

    private String origen;
    private String destino;
    private Pasajero[] pasajeros;
    private Pasajero[][] asientos;

    public Vuelo(String origen, String destino, int cantidadPasajero) {

        /*
         * El constructor debe generar las acciones necesarias para
         garantizar el correcto funcionamiento de los objetos de la clase
         */

    }

    public boolean agregarPasajero(Pasajero pasajero) {

        /*
         * Incorpora un nuevo pasajero a la lista de pasajeros del vuelo.
         Devuelve true si se pudo agregar o false si la capacidad del vuelo se
         encuentra completa.
         */

    }

    public boolean verificarDisponibilidadAsiento(int fila, int columna) {

        /*
         * Verifica si el asiento indicado se encuentra disponible.
         */

    }

    public Pasajero buscarPasajero(int dni) {

        /*
         * Busca un pasajero en la lista de pasajeros a partir del DNI. Si no
         lo encuentra devuelve null.
         */

    }

}
```

```
public boolean asignarAsiento(Pasajero pasajero, int fila, int columna) {  
    /*  
     * Asigna el asiento al pasajero recibido por parámetro. Devuelve  
     * true si lo pudo asignar o false en caso que el asiento no se encuentre disponible.  
     */  
}  
  
public void ordenarListaDePasajerosPorDNI() {  
    /*  
     * Ordena la lista de pasajeros por DNI.  
     */  
}  
  
public Pasajero[] getPasajeros() {  
    /*  
     * Devuelve la lista de pasajeros  
     */  
}  
  
public String toString() {  
    /*  
     * Devuelve un mapa de los asientos del vuelo indicando por cada uno  
     * si se encuentra libre "L" u ocupado "O".  
     */  
}  
}
```

2. Desarrollar la clase checkin

```
public class Checkin {

    public static void main(String[] args) {

        System.out.println("Bienvenido ");
        int CantidadPasajero = 192;
        Vuelo vuelo = new Vuelo("Buenos Aires", "Bariloche", CantidadPasajero);

        int opcion = 0;
        do {
            opcion = seleccionarOpcion();
            switch (opcion) {
                case 1:
                    registrarPasajero(vuelo);
                    break;
                case 2:
                    elegirAsiento(vuelo);
                    break;
                case 3:
                    listarPasajeros(vuelo);
                    break;
                default:
                    System.out.println("Opción Invalida");
                    break;
            }
        } while (opcion != 9);
    }

    private static int seleccionarOpcion() {
        Scanner teclado = new Scanner(System.in);
        int opcionSeleccionada=0;

        System.out.println("*****");
        System.out.println("Menu de opciones\n");
        System.out.println("1 - Registrar pasajero");
        System.out.println("2 - Elegir Asiento");
        System.out.println("3 - Listar pasajeros");
        System.out.println("9 - Salir");
        System.out.println("*****");
        System.out.println("Ingrese una opcion");

        opcionSeleccionada = teclado.nextInt();

        return opcionSeleccionada;
    }
}
```

```
public static void registrarPasajero(Vuelo actual) {  
  
    /*  
     * Interfaz de usuario encargada de cargar un pasajero al vuelo. Debe  
     * ingresar los datos necesarios e informar por pantalla el resultado de la operación  
     */  
  
    Scanner teclado = new Scanner(System.in);  
    String nombre, apellido;  
    int dni;  
  
}  
  
private static void elegirAsiento(Vuelo actual) {  
  
    /*  
     * Interfaz de usuario encargada de gestionar la asignación de asientos en  
     * el vuelo. Debe permitir el ingreso de los datos y mostrar por pantalla el  
     * resultado de la operación  
     */  
  
    Scanner teclado = new Scanner(System.in);  
    int dni, fila, columna;  
    Pasajero pasajeroASentar;  
  
}  
  
private static void listarPasajeros(Vuelo actual) {  
  
    /*  
     * Se debe mostrar la lista de pasajeros registrados para éste vuelo  
     * ordenados por DNI  
     */  
  
}  
}
```