Análisis

1. Propiedades y Métodos de cada clase:

Clase TicketManager:

Propiedades:

* ArrayList<Localidad> localidades: Almacena las diferentes localidades del evento.
* Comprador compradorActivo: Almacena el comprador que está actualmente interactuando con el sistema.

Métodos:

* void nuevoComprador(String nombre, String email, int presupuesto): Crea un nuevo objeto Comprador y lo asigna como el comprador activo.
* void nuevaSolicitudBoletos(int cantidad): Procesa una solicitud de compra de boletos y verifica si el comprador puede realizarla.
* void consultarDisponibilidadTotal(): Muestra la disponibilidad de boletos para cada localidad.
* void consultarDisponibilidadIndividual(int numeroLocalidad): Muestra la disponibilidad de boletos para una localidad específica.
* void reporteCaja(): Calcula y muestra el total de ingresos por ventas de boletos.
* void codigoEspecial(int codigo): Procesa la compra especial con el código proporcionado.
* int[] generarSecuenciaFibonacci(): Genera y devuelve la secuencia de Fibonacci requerida para el código especial.

Clase Localidad:

Propiedades:

* int numeroLocalidad: Número identificador de la localidad.
* int precio: Precio de los boletos en esta localidad.
* int espacioDisponible: Espacio disponible para boletos en esta localidad.
* int boletosVendidos: Número de boletos vendidos en esta localidad.

Métodos:

* boolean validarEspacio(): Verifica si hay espacio disponible para más boletos.
* boolean validarDisponibilidad(int cantidad): Verifica si hay suficientes boletos disponibles para la cantidad deseada.
* boolean validarPrecio(int presupuesto): Verifica si el presupuesto del comprador es suficiente para comprar en esta localidad.
* void venderBoletos(int cantidad): Actualiza las propiedades cuando se realiza una compra de boletos.

Clase Comprador:

Propiedades:

* String nombre: Nombre del comprador.
* String email: Dirección de correo electrónico del comprador.
* int presupuesto: Presupuesto disponible para comprar boletos.
* int boletosComprados: Número total de boletos comprados por el comprador.

Métodos:

* void comprarBoletos(Localidad localidad, int cantidad): Realiza una compra de boletos en una localidad específica y actualiza las propiedades correspondientes.

2. ¿Qué tipo deben tener las propiedades y métodos de cada clase?

TicketManager:

* Propiedades: ArrayList<Localidad> y Comprador
* Métodos: Todos los métodos son void, excepto generarSecuenciaFibonacci, que devuelve un arreglo de int.

Localidad:

* Propiedades: int para numeroLocalidad, precio, espacioDisponible y boletosVendidos
* Métodos: Todos los métodos son de tipo boolean, excepto venderBoletos, que es void.

Comprador:

* Propiedades: String para nombre y email, e int para presupuesto y boletosComprados
* Métodos: void para comprarBoletos.

3. ¿Cuáles deben ser los modificadores de visibilidad de los miembros en cada clase?

TicketManager:

* Propiedades: private para localidades y compradorActivo
* Métodos: public para todos los métodos

Localidad:

* Propiedades: private para todas las propiedades
* Métodos: public para todos los métodos

Comprador:

* Propiedades: private para todas las propiedades
* Métodos: public para comprarBoletos

4. ¿Qué parámetros serán requeridos por los métodos en sus clases?

TicketManager:

* nuevoComprador: String nombre, String email, int presupuesto
* nuevaSolicitudBoletos: int cantidad
* consultarDisponibilidadIndividual: int numeroLocalidad
* codigoEspecial: int codigo

Localidad:

* venderBoletos: int cantidad

Comprador:

* comprarBoletos: Localidad localidad, int cantidad

5. ¿Cómo proveerá de valores iniciales a sus objetos? ¿Qué valores iniciales les asignará?

Los valores iniciales se proporcionarán a través de los constructores de cada clase.

Para TicketManager, no es necesario un constructor explícito ya que no tiene propiedades directas.

Para Localidad, el constructor aceptará int numeroLocalidad, int precio y int espacioDisponible.

Para Comprador, el constructor tomará String nombre, String email e int presupuesto.