**Comprensión del Mundo del Problema**

Podemos identificar tres entidades distintas en el mundo: avión, silla y pasajero. Cree el diagrama del mundo del problema:

**La Clase Pasajero**

**Objetivo:** Hacer la declaración en python de la clase Pasajero.

Complete la declaración de la clase Pasajero, incluyendo sus atributos, el constructor y los métodos que retornan la cédula y el nombre.

class Pasajero:

#-----------------------------------

# Atributos

#-----------------------------------

#-----------------------------------

# Constructor

#-----------------------------------

def \_\_init\_\_(self, pCedula, pNombre ):

#-----------------------------------

# Métodos

#-----------------------------------

Def darCedula( ):

def darNombre( ):

**La Clase Silla**

**Objetivo:** Completar la declaración de la clase Silla.

Complete las declaraciones de los atributos y las enumeraciones de la clase Silla y desarrolle los métodos que se le piden para esta clase.

class Silla:

#-------------------------------------------

# Enumeraciones

#-------------------------------------------

“““\*

\* Enumeradores para las clases de silla.

““

Enum Clase

# Representa la clase ejecutiva.

#Representa la clase económica.

“““

\* Enumeradores para las ubicaciones de las sillas.

“““

enum Ubicacion

“““

\* Representa la ubicación ventana.

“““

“““

\* Representa la ubicación centro.

“““

“““

\* Representa la ubicación pasillo.

“““

#-------------------------------------------

# Atributos

#-------------------------------------------

private int numero;

private Clase clase;

private Ubicacion ubicacion;

private Pasajero pasajero;

\_\_init\_\_ ( self, pNumero, pClase, pUbicacion ):

Self.\_\_numero = pNumero;

Self.\_\_clase = pClase;

Self.\_\_ubicacion = pUbicacion;

Self.\_\_pasajero = null;

Def asignarPasajero(self, pPasajero ):

# Asigna la silla al pasajero "pPasajero".

Def desasignarSilla (self ):

# Quita al pasajero que se encuentra en la silla, dejándola desocupada.

Def sillaAsignada( self ):

# Informa si la silla está ocupada.

Def darNumero( ):