**UNIVERSIDAD DE CARTAGENA**

**FACULTAD DE INGENIERÍA**

**INGENIERÍA DE SOFTWARE**

**ENTREGADO POR:**

**LUIS FERNANDO VILLADIEGO SABALZA**

**ASIGNATURA:**

**ESTRUCTURA DE DATOS**

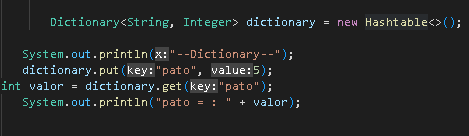
**TUTOR:**

**JOHN CARLOS ARRIETA ARRIETA**

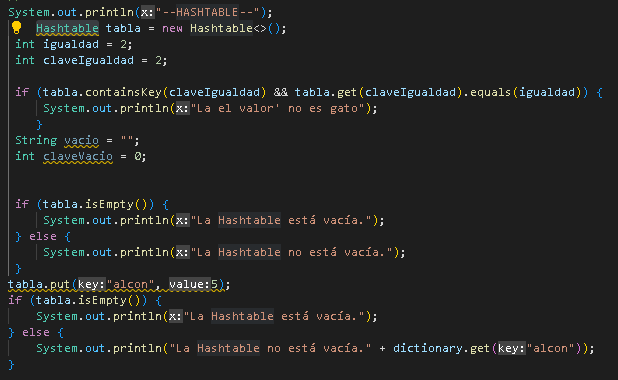
**III SEMESTRE**

**CARTAGENA DE INDIAS D. T. Y C**

**2023**

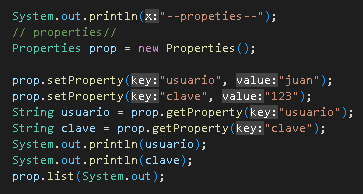
1. Investigue la clase Dictionary:  
   Dictionary es una de las tantas estructuras de datos que nos brinda java, esta clase es un poco vieja y fue remplazado por otros sistemas parecidos que usan el mismo mecanismo de llave-valor como lo sería el hashmap entre sus métodos tenemos unos cuantos que seria elements(); el cual nos retorna un listado de los valores que se encuentran en la estructura, getobject(); que nos sirve para manipular los datos que almacenamos, empty(); para verificar existencia de elementos, remove(); para eliminar tanto llave como valor, y su contraparte put()para poner una llave y un valor   
   
2. Investigue la clase HashTable:

Esta clase al igual que la anterior funciona por clave-valor la única diferencia es que solo trabaja con elementos que son diferentes a nulos, o bueno preferiblemente ya que se pueden crear excepciones esta sería la versión “mejorada” de dictionary tiene los mismos métodos dichos anteriormente put(); poner elementos, remove(); quitar elementos etc., en esta se agrega el método tostring(); que sirve para convertir los valores en cadenas de texto



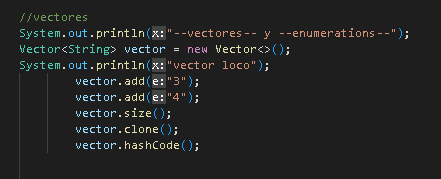
1. Investigue la clase property:

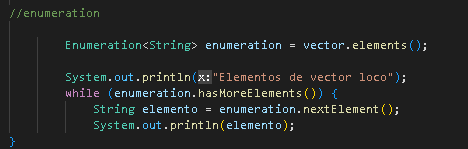
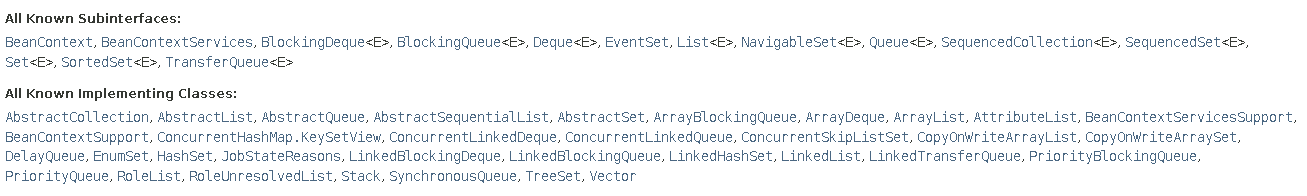
La clase property nos permite pasarle propiedades a el valor que queramos para seguidamente poder guardarse y hacer uso de la persistencia para llamar a las propiedades que guardamos entre sus métodos tenemos los de persistencia que consisten en loadformxml(); nos permite cargar las características desde un archivo XML

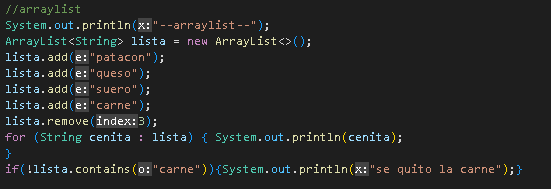
Getproperty(); que nos permite tomar un valor puesto en la lista con setproperty();  


1. Investigue la clase vectores:

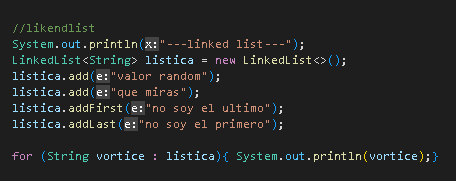
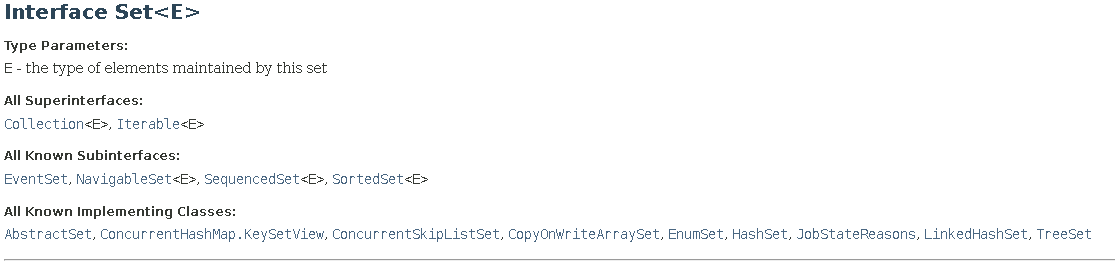
Los vectores o mejor llamados arrays dinámicos son arreglos que tienen la característica de ser muy flexibles ya que pueden apuntar a un valor desde cualquier lugar usa el método add(): para agregar elementos tiene un método para clon(); para clonarlos puede retornar el numero de elementos que hay con elements(); puede aumentar su tamaño durante tiempo de ejecución si lo necesita con ensurecapacity(): además de los métodos anteriormente mencionados

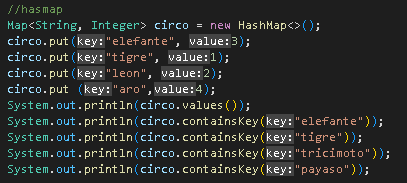


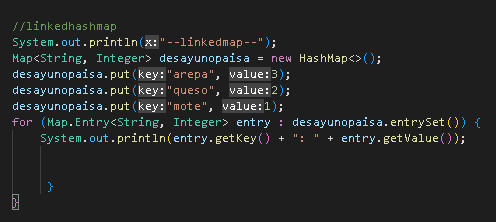
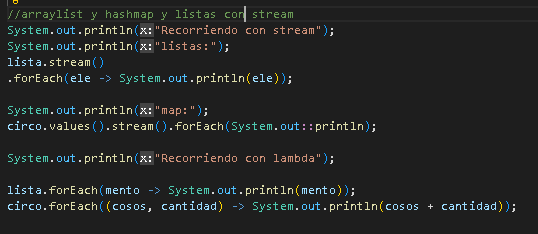
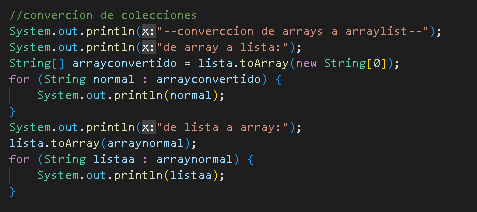
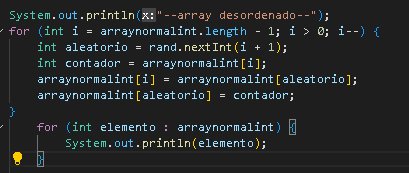
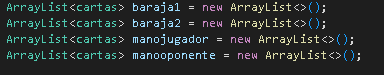
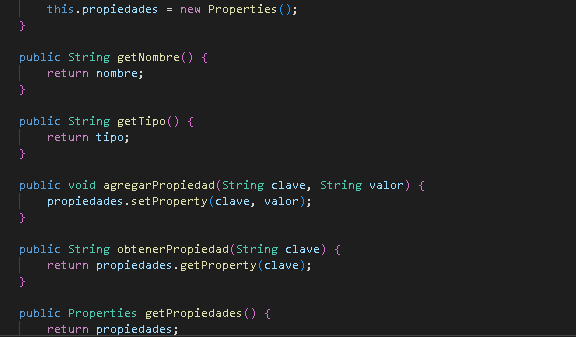
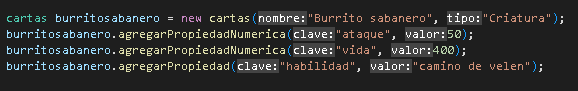
1. Investigue la clase enumeration:  
   esta es una interfaz la cual tiene su uso principalmente en las colecciones donde esta es usada para hacer interacciones de los antes mencionados cuenta con métodos como asinterator(); que devuelve el iterador al cual esta vinculado la interfaz hasmorelementes() el cual es un booleano y comprueba si interfaz se pasa de su límite nextelement(); que devuelve el numero siguiente si este lo tiene.  
   
2. Investigue la clase collections:  
   collections es uno de los frameworks predilectos de java que agrupa muchos sistemas de estructuras de datos como lo son los arrays y las listas e incluye interfaces como map que se encarga de que no sirve para para facilitar el manipula miento de grupos de datos a continuación las principales.  
   
3. Clase arraylist:  
   arraylist es un array que usa la interfaz list de java útil una de las diferencias de este tipo de array es que no se necesita especificar índices, sino que se va haciendo mas grande a medida que se agregan los datos



1. Interfaz list:  
   la interfaz list hace de puente para la interfaz collection que tiene la mayoría de métodos antes vistos, como los son .add(); y size(): y su funcionamiento es igual al arraylist la diferencia es que a la hora de hacer conversiones list al ser la interfaz que implementa y hereda otros sistemas de colección se vuelve mas flexible y manejable junto a otros sistemas  
   

* Clase Linkedlist:  
  siendo una interfaz de list y cumpliendo las mismas funciones de arraylist este es un estructura que se caracteriza por poder hacer operaciones en medio de la lista ya que esta retorna tanto el elemento siguiente como el que pre-sigue aunque este puede consumir mas recursos, cuenta con los mismos métodos y unos adicionales como addfirst() para agregar antes de un valor y addlast(); y con la forma add(“pocicion”,”elemento ”) de todo array.  
  
* Interfaz set:  
  a diferencia de list no permite elementos duplicados y no están necesariamente ordenados este tiene la estructura treeset la cual se almacena en una especie de árbol con un orden igual al de inserción el cual puede ser modificado.  
    
  incluye también hashset el cual a diferencia del treeset si mantiene el orden de inserción de los elementos ambos cuenta con los métodos principales de la clase collections add();. Remove(); etc  
  

1. Investigue hashmap  
   hashmap comparte la misma características de los otros arreglos como dictionary tratándose de un arreglo llave-valor el cual al igual que sus parecedios usa los métodos   
   put() y get para asignar valores y containkey(); para ver si la llave existe.  
   

1. Investigue Linkedhasmap:  
   linkedhasmap al igual que su variante linkedlist este comparte los mismas caracterisitcas y métodos de hashmpa con la diferencia que los datos almacenados quedan con le orden en que fueron insertados gracias a los métodos entrygetkey( ); y entry ( ); ger value  
   
2. Recorriendo arreglos antes creados con lambda y stream  
   
3. Convirtiendo colecciones   
   
4. Desordenado coleccion  
     
     
     
     
     
   -Programa java  
   el siguiente programa usa. listas, linkedlist, hashmap, , properties y set para simular le sistema de extraer cartas de una barja a tu mano y que esta tenga a su vez propiedades que pueden ser usada en diferentes algoritmos para simular diferentes encuentros con otras cartas   
     
     
   se crearon 4 clases las 2 barajas a elegir y la mano de cada jugador   
     
     
   se usó properties para pasar las estadísticas de las cartas   
     
     
   luego se llamó al método y se le especificaron estas:  
     
     
   una vez ya definidas todas las cartas se invoca el método shuffle de la clase colletions para mezclar las barajas y seguido se usa una función para remover de la barja y ponerla en la mano del jugador una se encarga de la mano del oponente y otra de la mano del jugador   
     
   