



I'm not robot



Continue

Collections in java pdf ebook

This lesson explains the hierarchy of a collection as well as the framework of Java's collection in details (JCV), the benefits of various interfaces and JCFs; hopefully, it is very helpful to promote your knowledge of Java to these eloquent lessons on this Java series. Our previous tutorial will help more on the /O operation in Java in short. In this lesson we will learn in detail about the framework of Java's collection (JCV). The Java Collection Framework (JCF) consists of interfaces, summary classes, and classes that can provide architecture for storage and collection of a group of items. Video Lesson:Java collection of the firmirokkanuoladj list in Java: Set interface ↦ row interface with real time examples in Java: Hashana map, trackpi and mix table in Java: So far, we have seen variables that store different types of data. These are the only units and you are not very helpful when you have huge data to store and deal with. As we know, data is the most important constitution of computer processing. You must be able to extract and collect request data, find useful information and also move it to follow it and fro so it is used effectively to reach solutions. In this software world, we store data that needs to be managed in a particular fashion so that software programs are able to store, read, process and finally produce results. This is done using the data structure. Data structures supporting almost all programming languages. The structure of data can be described as representing data configuration. Programming languages provide a PI (application programming interface) functions to harass data structures. To firmiroc and need java 1.2, java programming language support arrays like data structure, vitor or hash tables. These data structures were not working on that was not a normal interface as these data structures were not easy to perform operations on. All of the programmers will work for data structure that is used to find difficult to write algorithms and was a big problem. There was a need for a common interface that will work continuously on all data structures and also operate sedately. Later, Java came up with a collection framework that can be performed on the data structure, consisting of a group and interface consisting of different operations. In the tutorial of this Java collection, we will usually discuss the framework of the Java collection. We'll find each of the ingredients collected in our upcoming lesson. Java collectors firmirocbyetion we explain the framework of a Java collection, let's get to know the meaning of the collection and a framework. A collection is usually a unit consisting of more than one object. A framework is a software that has a ready-made functionality interface or architecture and also contains a set of classes and interfaces to use with the provided interface, a collection Work is defined United, the architecture produced by the following. #1) Algorithms are set by statements that can help programmers in finding, arranging, and processing methods or collected data. The collection is packed with the interfaces, classes and algorithms. Applications developed by the legacy of the collection framework have access to these algorithms and use already defined methods and procedures. #2) Using the interface interface in Java, the user is not aware of the process, but can look at the methods and data necessary to write an application. The collection interface has many interfaces that can be applicable by programmers to write their own classes. #3) Class collection framework classes are data structures that can be applicable in a program. These classes apply the collection interface and thus inherit all the methods and definitions of the collection interface. A collection framework is used to store and store this collection, which is the object group. Java Collecting Framework consists of high performance algorithms that carry standard operations such as search, setting and processing. It provides various standard implementations including The Lankadrest, the Treest and Hashanah set for which the interface is provided. Next, let's understand the Java collection ranking. Java Collection Herarchical a class and interface shown in the following Java collection rankings. util. * Belongs to the package. As shown in the above format, the Java collection rating consists of different classes and interfaces. As you can see, inherit every class from an interface and inherit all classes ↦ interfaces, in turn, from a single collection interface. Let's discuss some common methods in the collection interface with a short introduction about each of the classes and interfaces that have been shown in the above ranking. The collection of the interfaqathi is the moaja root moaja. Apply all classes in the collection framework to the collection interface. This means that every collection methods will be announced in the collection interface. Some methods of the deposit interface are given below. The Aitrabi Interfacathi Aitrabali interface is also the framework base interface of the collection. All other classes have parent interfaces that extend the collection interface atrabali interface. So all classes apply a combination interface as well as an atrabali interface. The only way in the Aitrabali interface is the Aitratore () which comes back to The Aitratore () which you can use to repeat on the elements of type T. Aitrare and Aitratore (j&T&T; The Aitrare and Interfacathi Ateratchi interface provides the ability to repeat on elements in a forward direction. This interface is supported in the following ways. List of Interfaaktelst is the Moaja collection of moaja heritage. List &T;T&T;Data contains the structure that is used to store the structure or collection of items. These data structures are the type of list. These data structures are not apply to the list interface or have the duplete values. The list of moaja contains the methods used for access, insert or remove elements from the list object. The different classes that apply the list interface are as follows: Arelistalankedelastatorstorsatkoa will argue briefly each of these classes. Our subsequent topics will have detailed discussion stake in the framework classes of each one. #1) The Aralastaria list is the easiest process of the interface. The location in its list is safe and not synchronized. The general definition of the Aralast Data Structure (collection) is as follows: List &T;data-type&T;list1 = New List of Faxes (); Once the list is defined, you can use the 'additional' method to include elements. Note that internally the art list uses dynamic saine mechanisms. See the following examples that create an arylist collection of colors. Import Java. Classmen (Public Static Falsehood (Silk args []) {/String List of items ↦T; String ↦T; color_list = New Arylastist ↦T; String ↦T; Add (red); color_list. Add (green); color_list. Add (blue); color_list. Add (yellow); Aitrator. Addition (yellow);/creating list atrator = color_list. System. Outside the contents of the partin (are); while (tr) {system. perantion (tr. next ()); } Output: Displays the arealist collection in the above program. We add color values to the collection and then the atrator. #2) To display individual values in the collection using the following. The collection of The LinkadalaStoiases a double-connected list mechanism for internally protected elements. It can contain the dupletelements. The transfer of elements is not required as the Gather Sanctodlasi is accelerated. The common definition of creating an attached list is as follows: List &T;data-type&T;list2 = new Linkadlaist (); The following program displays a linked list collection of number names. Import Java. Classmen (Public Static Invalid (Silk args []) {/linked linked linked list Silk Object LinkDalist ↦T; Silk ↦T; linkedist_num = New Link (); linkedist_num. Add (a); linkedist_num. Addition (two); linkedist_num. Addition (three); linkedist_num. Addition (four); linkedist_num. Addition (five); Add The Aterator ↦T; ↦Sterling; tr = linkedist_num/tefade more than the list attached to show. System. Out. Perantion (the contents of the attached list are); while (tr) {system. perantion (tr. next ()); } Output: Thus, we create an attached list and then add elements using the 'additional' method. Then using an attetower, we moved the attached list and every element What. #3) is a like and uses a dynamic. &T;data-type&T; &T;data-type&T; &T;data-type&T; To store such elements. But The Character supports many independent methods besides this collection which makes it a better choice of being a favorite combination. The common definition for the character collection is: List &T;data-type&T;list3 = New Character (); Note that although The Connector and my list use the same mechanism of dynamic arrays, the character elements are synchronized. Below is evidence of the use of the connector in the Java program-collecting framework. Import Java. Important of public class (public static falsehood (string args []) {vector ↦T; string ↦T; subject_vector = new ↦T; string ↦T; (); subject_vector. Addition (Hindi); subject_vector. Addition (mathematics); subject_vector. Addition (knowledge); Aterator ↦T; String ↦T; tr = subject_vector. Aitrator (); System. Perantion (Character Material); while (tr) {System. Perantion (tr. Next ()); } Output: In this program, we've explained a character collection containing articles. We add different articles and then use an attetower to outhe elements. #4) The first (alpha) method of inserting elements into the stoxtak data structure is to apply. Stack class is a subclass of the character (refer to the organizational structure of the above collection). In addition to its own methods, stack collection also supports the ways of the Character collection. The general definition of stack collection is: List &T;data-type&T;list4 = new stack (); The following program stake collection is applicable. Import Java. Important public class (public static falsehoods (silks args []) {thack ↦T; salm number ↦T; even_stack = new thack ↦T; molecular number ↦T; (); even_stack. Push (2); even_stack. Push (4); even_stack. Push (6); even_stack. Push (8); System. Perantion (element removed from stack +even_stack.pop ()); Aterator ↦T; Salam Number ↦T; tr = even_stack. Aitrator (); System. Out. Perantion (stack materials are); while (tr) {system. printing (tr. next ()); } } Output: As you can see in the above process, new elements are added to the stack using push operation. Stack is called a single entry point' at the top of the stack' and elements are pushed to the top stack. Thus, the last element involved is the top of the stack. Like the included, the elements are also removed from one end i.e. the top of the stack. A process pop is used to remove elements. This way if you make pop calls (), the top element of the stack will be the removed. In the output above, we call element 2, 4, 6, 8, 10, and then pop (so that 10 is removed. Row interface combinations from row interface follow the first (fifo) order in the first. The elements are put at one end and removed from the other end. So the inserted element has to be the first element that is removed. Below are the combinations that support the question interface. Each of them of the apostovedida intrafaqarayadyaqelate talks about them. #1) The row collection of the perantion work. &T;data-type&T; &T;data-type&T; The store is implemented based on their preferences. You cannot store invalid values in the priority queue. The general definition of the preferred row is as follows: Row&T;data-type&T; q1 = New PriorityQueue (); The following program lists the priority queue. Import Java. Important public class (public static invalid (silks args []) {PropertyQueue ↦T; String ↦T; color_queue = New QualityQueue ↦T; String ↦T; (); color_queue add (red); color_queue. Add (green); color_queue. Addition (blue); color_queue. Addition (qormas); color_queue. Addition (yellow); System. Out. Perantion (preferred row elements.); system. Outside. Perantion (Head + color_queue. Blink ());System. Perantion (other elements in the preferred row.); Aitrator (tr = color_queue. Aitrator (); while (tr) {system. printing (tr. next ()); } system. Out. Perantion (); color_queue. Remove (); System. Perantion (after removing the element, new head -<color_queue. element ()); system. Out. Perantion (); color_queue. System. Perantion (after removing another element, preferred row); atrator ↦T; sterling ↦T; list2 = color_queue. Aitrator (); while (tr2) {system. Print (tr2. Next ()); } } Output: Once again we use the color as elements of the priority row. In the above program, we added methods to add to the queue and used to remove and remove an element, in addition to the above program. We use the (blink) method that returnamelement that indicates its priority queue. Finally, elements of the preferred queue are displayed using an attestation. #2) Deque or a 'double finish queue' is a data structure that allows you to add and remove elements from both ends. Java Collection framework has deque interface that increases the queue interface. It provides the functionality of the deque and it is a class from a 'caste Aydethe' legacy. #3) apply the data). The general definition is as follows:&T;data-type&T; Deayd = New Edico&T;data_type&T; (); Allows you to use The Ay's functionality. Unlike other collections such as the one list or stack, there is no restriction on its ability. For the following example, the process shows the Import ArrayDique Java. Important public class (public static false (args) {/Creating configuration and elements day↦T; molecular number ↦T; Deay = new indacy ↦T; Salam Number ↦T; (); day. Add (10); day. Add (30); System. Out. For the perantion (Deque elements.); For passing elements (digital number: deque {system. perantion (num.); } } Output: In the above program, we define an ArrayDique collection of the type of molecular number and add digital elements to it using the Add procedure. Then using the set for construction of the deposit. Set the interface to the interface of the interface is a part of Java. Util package and collection interface extension. Set is a structure that allows the collection to be more than the duplete values and more than one price. The following classes apply &T;data_type&T; &T;data_type&T; &T;data-type&T;Set the moaja Hassetta-Ankadhassitreset #1), which apply the set interface contains the custom values stored. This collection uses a hash table to store items and store ingredients. The general definition of the Hashanah set collection is shown below. Set&T;data-type&T; s1 = new Hashanah set&T;data-type&T; (); We have implemented this in the following program. Import Java. Basic of public class (public static falsehood (silks args []) {/Create-make Hashana set_subjects Hashana ↦T; String ↦T; set_subjects = New Hashanaset ↦T; String ↦T; (); system. Out of the partin (given hasshasset_subjects has elements.); add set_subjects (mathematics); set_subjects add (computer); add set_subjects (English); set_subjects. Add (French); Aterator ↦T; Silk ↦T; tr = set_subjects down. Aitrator (); while (tr) {System. Output: In the above program, we create an indelme collection of articles and then move it using a means to display elements in the atrator. #2) The Lockadasset-Adahasset application sat and extends on a set interface (refer to the collecting organizational structure). The Link list of the Lankadhasset interface set is represented. The Laenkadhasset contains unique elements but allows for invalid values. The general definition of The Language sat is given below. Set &T;data-type&T; s2 = new Linkadhasset&T;data-type&T; (); The implementation for The Lankadhasset is given below. Import Java. Public Classmen (Public Static Invalid (Silk Args []) {Linkadhasset ↦T; String ↦T; set_subjects = New Linkadhasset ↦T; String ↦T; (); System. Out. Partion (set_subjects Elements in Lankadhasset); set_subjects (Mathematics); add set_subjects (computer); add set_subjects (English); set_subjects. Add (Sanskrit); Aterator ↦T; Silk ↦T; tr = set_subjects. Aitrator (); while (tr) {system. Output: Once again we use the book title to create The CalendarSit. As you can see from the production, also retained by The Lankadhasset. The SourceSet Interfacethat interface allows full command of elements. It contains methods that providea natural command of elements. The elements that apply the Sordiset interface in the collection are organized in a growing order. Treestclass is one of the examples that apply the Surtidset interface. The general definition of the trias is as follows: Set&T;data-type&T; s3 = New Treest&T;data-type&T; (); Treestset contains the SourceDSAT interface and contains unique elements. Storage and resinareare very fast and then the elements are organized in a growing order. Import Java. The basic (public static false (string args []) of the public class is denoting (/create-saivanana which has a number of ↦T; digital ↦T; set_oddnms = new triano↦T; salm number ↦T; (); set_oddnms. Add (1); set_oddnms. Add (3); set_oddnms. Addition (9); کو افقی. كواقی. برانظن (الن [ن اجزاء میں عناصر لیں): کو افقی. كواقی. برانظن (الن [ن اجزاء میں عناصر لیں): &T;data-type&T; &T;data-type&T; &T;data-type&T; &T;data-type&T; Using The Aitrator Aterator ↦T; Digital ↦T; tr = set_oddnms. Aitrator (); While (tr) {system. printing (tr. next ()); } } Output: In the above program, we created and added odd numbers to the collection using the methods involved. Then using an attetower, we output elements in the collection. The Benefits of Java Collection Firmiroktradrk Programming: The collection comes with all interfaces and classes containing framework methods that can allow programmers to write an effective program. Thus, programmers need not focus on programming too much. Continuous methods and API: All classes are common methods that follow data. These APIs are similar throughout the class and programmers need not worry about many different methods for each class. Add speed and accuracy: You can write down highly effective programs using a combination framework and also provides complete functionality to apply the framework data structure and collection of the collection as fast ↦ accurate programs. Software facilitates reuse: Data structures and java plus framework algorithms are re-supported. Interplay between unrelated APIs: The collection also allows apply to be inter-related between APIs. Fewer efforts to design new APIs: Programmers can use standard collection APIs and design new APIs based on it. They don't struggle to write a complete new APIs. Frequently Asked Questions #1) What is the use of a collection framework in Java? A: The collection framework offers pre-paged algorithms, interfaces, and classes that allow programmers to write high-effective programs that can store and process data. #2) Why is the collection a framework? A: There is a framework collection of classes, interfaces, and algorithms. The collection framework has a base interface while the 'collection' is a class. Both Java. Util are a part of the package. * * * More details and examples of java collection framework: * * * List, comparison between set and row. Top ClassAn Sarni, memory allocation is set. But the list inside, memory can be allocated dynamically. It maintains the order of location in the social list class and you can insert the duplete elements. Demo Program for Form List: Public Class List Demo (Public Static Falsehood (Args) {Theo List List = New Group List (); Li. Add (1); Li. Add (2); Add (3);..... System. Out. Perantion (= = = = =); Your list [= new eqon list (); Li. Add (text1); Li. Add (text2); Li. Add (text3); For (String = l) {system. perantion (tempo) } system. Out. Perantion (= = = = =); Aryalist al = new monolist (); Add (1); Add (2); (a) (&T; System. Out. Perantion (A); The list of the) Language Class-Selankad consists of data structure Node and this node will consist of two parts: Dathifranka is not a separate church for the next alimanth first church. There is only one reference and it is called sir. The last church is invalid. Demo Program: Public Class Linkadalistdemo (Public Static Falsehood (Args) {Linkadalist List = New Linkadalist (); List. Addition (22); List. Addition (44); List. Addition (46); List. Included (46); For (L number tempo: list) {System. Perantion (tempo); Aitrataeand it = List. Aitrataevar (); while (it) {System. Perantion (it. next ()); } System. Out. Perantion (= = = = =); for (nti = 0; i&T;list.size()-1) {= system.out.println(get(i) =); } = } queuea= queue= is= frst= in= frst= out = (fifo)= data = s the texture = if you= call= the= remove= method.= always= the= frst= inserted= element= will= be= removed= from= the = queue.= so.= queue= is= used= in= real = time= application= where= the= data= has = to= be= retrieved= in= the= inserted= order= example= program= public= cls= queuedemo = {= public= stack = void= main (string[] = args) = {= principaqueue= pq=new principaqueue()= pq.add(1)= pq.add(2)= pq.add(3)= pq.add(4)= pq.add(5)= pq.add(6)= pq.add(7)= for(int= temp:= ; pq)= {= system.out.println(temp)= ; pq.remove()= system.out.println(queue= after= removing= a= element)= system.out.println(pq)= ; } = } seta= set= collection= will= not= allow= duplicate= elements.= you= can= use= a= set= data= structure.= when= you= want= to= process= the= collection= of= data= without= duplication = and= when= there= is= no= need= to= maintain= the= insertion= order= demo= program= public= class= demohashset= {= public= static= void= main= (string[] = args) = {= hashset= hs=new hashset()= hs.add(23)= hs.add(24)= hs.add(25)= hs.add(26)= hs.add(27)= hs.add(28)= hs.add(29)= hs.add(293)= hs.add(203)= hs.add(243)= hs.add(243)= hs.add(243)= hs.add(243)= (int= temp)= {= system.out.println(temp)= ; }= hs.remove(24)= hs.remove(23)= system.out.println(after= removing= a= element =); }= system.out.println(=====); hs.foreach(tempo)=&T;System. perantion (tempo); } Map Interface you want to assemble elements process with important and price pairs, then you can use the map data structure because the map is an objection in which keys of values can be mapping. The map cannot contain the mesoory keys. Map is a map of the main implementations, trackable, the Linkadhashamp, the mix table. Difference between, Tremix, Linkadhashamp, and Hashana Table: Demo Program: Public Class Hashanamap Demo (Public Static Invalid Important (Wire [] args) {Muhammd&T;Integer. String=&T; Map&T;Integer. String&T; m. Put (1, one), m. Put (2, two), m. Put (3, three), m. Put (false, one), M. Put (invalid, one), System. Out. TreeMap&T;String. string=&T; tm = New strings=&T; (); tm. Jls (R. ع.م), tm. Jls (B. ب.م), tm. Jls (G. ج.م), tm. Jls (H. ه.م), tm. Jls (T. ت.م), tm. Jls (D. د.م), tm. Jls (S. س.م), tm. Jls (Z. ز.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م), tm. Jls (L. ا.م), tm. Jls (P. ا.م), tm. Jls (V. ا.م), tm. Jls (F. ا.م), tm. Jls (X. ا.م), tm. Jls (C. ا.م), tm. Jls (D. ا.م), tm. Jls (S. ا.م), tm. Jls (Z. ا.م), tm. Jls (A. ا.م), tm. Jls (I. ا.م), tm. Jls (O. ا.م), tm. Jls (U. ا.م), tm. Jls (E. ا.م), tm. Jls (Y. ا.م), tm. Jls (Q. ا.م), tm. Jls (W. ا.م), tm. Jls (M. ا.م), tm. Jls (K. ا.م), tm. Jls (N. ا.م