O

Collection frame work

→ An Asimay is an indexed * Collection of fixed no. I homogeneous data elements. · L'imitations of Object assays: -xu Assays ask fixed in Size. i.e., one we Caleated an assay these is no Chana of increasing on decreasing size based on over sugaristement. Heng, to cese assorays Concept Compulsory we should know the size in advance, which may not possible always. (2) Assays Can hold only Homogeneous data elements, i.e., (Same type)) Student [] S = Dew Student (1000); 9 S[0] = new Student [];) S[1] = New Studental: .) S[2] = new Customer(1) X ce: Incompatiable types) - - found : Costomeza 9 Dispulsed & Student.) is But one Can elesoive this peroblem by using Object-types assays. 0 ex! 0 Object [] a = new Object[1000]; a[o] = new Student[]; 0 a[i] = new Costomer(): ~ 0 Ð (3) Assays Concept not built based on Some datastouchure. Hence Gredymed method supposit is not available. Foo every requirement. O u

Computasiony porgonammen is suspensible to covide the logic.

| → To sesolve The above post | olems. Sun people introduced Collections |
|---|--|
| Conapt. | 4 |
| ************************************** | ** |
| -> Advantages of Collections or | ves associates. |
| | (-) |
| (1) Collections asie genowable | in natural. Hence based on over |
| Stepuishement coe Can increase | |
| , The Carlot Treating | e Usi decilease. The close. |
| (2) Collections Can hold bolts | Homogeneous & Heterogeneous objects. |
| (2) Every Collection class is in | implemented based on Some chatastructures) |
| 5850 V.CC.) | pool is available for Every organisment. |
| dis of collections: | • |
| -> Performance point of view G | ollections are not recommended to use. T |
| This is the Limitation of Collect | |
| The same of the contract | 9 |
| defference blw assorage & collection | 987 |
| | · · · · · · |
| -Annay | Collections(AL, VL, LL) |
| of Aprovage assertised in Size | 1) Collections asie goodable in nature 3 |
| e) Memory point of view arrays | 2) memory point of view Gilections |
| Concept is not Decommended to use | Concept is highly secommended to use. |
| 3) performance point of view assays | 2) Deeple week Datel of a continue |
| Concept is highly Decemmended to use. | is not trecommended to use. |
| 4) Astronays Can hold Only homogeneous | |
| data dende | |
| data elemnis | & Heterogeneous objects. |
| 5) These is no underlying dis fina | 5) Underlying D.s is available for every |
| assays. Here steady med method Supposit is not avaliable | Collection Closs Hence Steadyrned method Support is available. |

A DESCRIPTION OF THE PROPERTY OF THE PROPERTY

→ Assocy's Can be used to population -> Collections Can be used to hold both poxemitives & Objects. only objects but not for potentitives. Collection :-→ A gonoup of Individual objects as a Single entity is called Collection Collection for ame coosin " -→ 2t defines Several classes & Interfaces, which can be used to represent a govern of objects as a Single Entity. Teaminology:-Java C++ Contained Collection Collection framework STL (Standard Template Library) 9-Key interfaces of Collection frame coork: . O Collection (Enterface): → 8F we want to suppresent a group of individual objects as a Single Entity then we should go for Collection. • Э → In generial Collection anterface is considered as shoot anterface of 0 Collection frame work.) → Collection Interfale defines The most Common methods which Can be

applicable for any Collection Object.

O

U

| Collection vs Collections :- | |
|--|------------|
| Collection vs Collections :- | |
| - Collection is an interface, can be used to suppresent a govern to | 155 |
| ACCURATION OF THE PROPERTY OF | |
| Individual object as a Single Entity, where as | 5.4 |
| | () |
| Collections is an Utilityclass, peresent in Javatulii package, to define | 3 |
| Sevenal Utility methods - from Collections. | . 1 |
| 3.1 | C) |
| 31 List (Interface): | () |
| - It is the child Enterface of Collection. | , |
| | () |
| → 28 coe want to shepresent a governo of individual object where | 0 |
| insention order is preserved & clubicates are allowed. Then we should | 0 |
| go fon List. | 0 |
| The American Company of the Company | O |
| Collection (I) | 9 : |
| 1.9 | 0 |
| | • |
| | .0 |
| List (1) | +) |
| 1.2 | • |
| | 0 |
| Vectorist | 0 |
| Association (in the distance of the section (ist to the section (i | 0 |
| (c) Ust (c) Legacy crasses | ာ |
| 1.2 | 0 |
| Stack(c) 1.0 | 0 |
| | 9 |
| -> Vector & Stack Classes asie one Engineered in 1.2 version to thet int | |
| Collection formine work | 0 |
| 511 (-59359)/// | 0 |
| | O |

.

TAMES OF THE OWNER.

| 3 Set (Interface): | 134 |
|---|-----------|
| -> It is the child anterface of Collection. | |
| -> 29 cue want to suppresent a goodp of Andividual objects a | oheste. |
| "charplicates asse not allowed & Phsexton orades is not posese we should go foor Set. | eved. The |
| 30-30 - 300-2013 | |
| Collection (I) | |
| | |
| | |
| Set (2) | |
| 1-2V | |
| | |
| HashSet (C) | |
| 1 1-2 V | |
| 10 To Section 19 | |
| Linked HashSet(C) | |
| | |
| (4) Soonted Set (I):- | |
| > BE is the child interface of Set. | adena L |
| if we count to suppresent a govoup of white objects, accome | 9 10 |

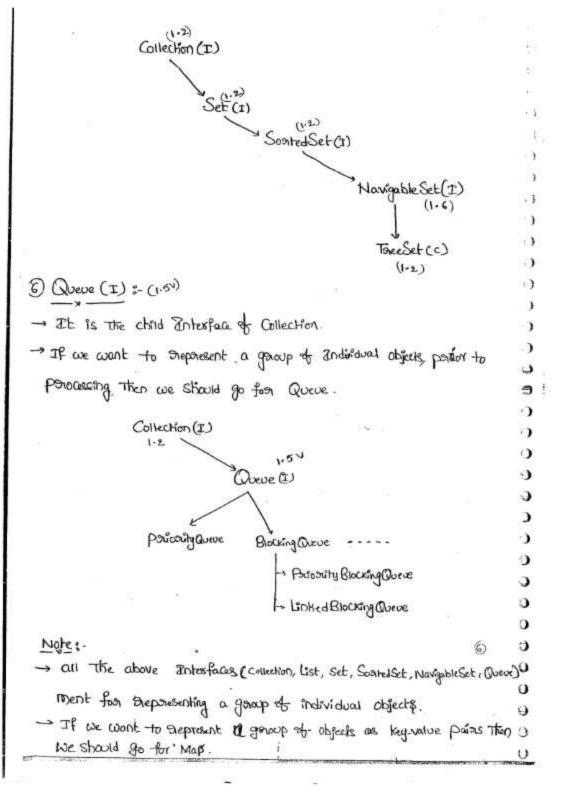
Some Souting andea then we should go from Souted Set.

(5) NavigableSet (1):

0

- It is the child interface of Scated Set, to posovide Several ornethods for Navigation prosposes. 0

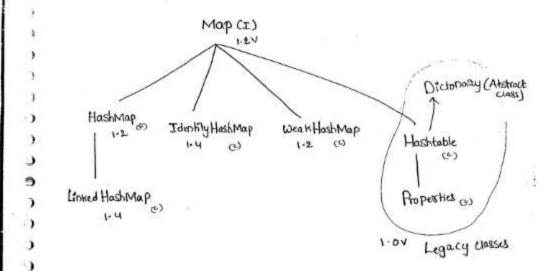
-> BE is introduced to 1.6 version.



(3) Map(1):-

- -> If we want to suppresent a group of objects as Key-value pains.

 Then we should go for Map.
- → Both Key & value asie objects only.
- duplicate keys asse not allowed. But values Can be duplicated.



) Note:-) -> "Map is not child Botosface of Collection.

(8) Soonted Map (I) !.

> 27 we want to expensent a group of objects as key-value pains

according to Some Scotting Oundern Then we Should go from Scotted Map.

3 → Soatisting Should be done only based on Keys. but not based-

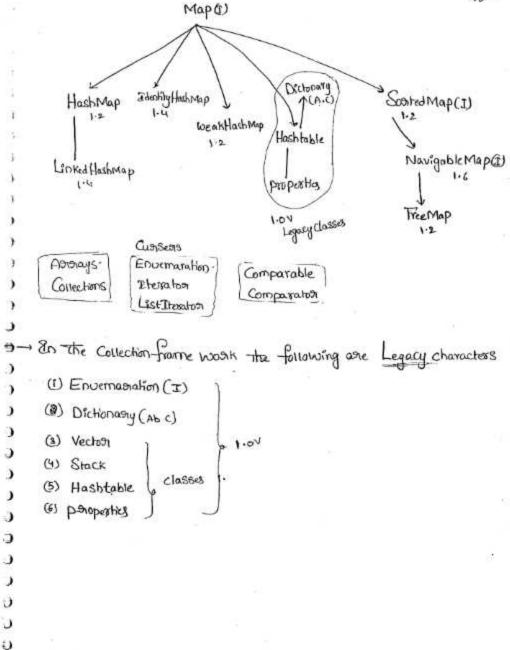
o on values.

U - SontedMap is child Enterface of Map.

| (| NbvigobleMa | b (1). | | | | | Ť. |
|----|---------------------------------|-----------------------|------------|----------|---------------|--------------------|---------------|
| | → &E is the | child Enterf | ace of | SostedM | 1ap & def | ine Sevenal n | nelfody |
| | -for Novigotion | | | | or 1000 | | |
| | 0 | | | | | 13 | - 3 |
| | | Мар С | 11.24 | | | | . 1 |
| | | | \ | | | | - 3 |
| | | | 1 | 124 | | | .) |
| | | | Sont | edMap(E) | | |) |
| | | | 8 | | | | 1) |
| | | | | Na | vigable Map (| 1) | .) |
| | | | | | 4 1.60 | ≈6 | -) |
| | | | | | 1 | | -) |
| | | | | - 7 | BeeMap (1) | | .) |
| | Summazy;- | | | | 1.24 | | 0 |
| | aa, | Colle | Kon (z) | | | | |
| | | | \ | | | | 9 |
| | | | | | 1.50 | | $\overline{}$ |
| | 1.2 | ٠, | | 1 | | | • |
| | List (2) | Set(I) | | ſ | Over (I) | | () |
| | 1 | "/\ | | (_ | / | | /3 |
| | LANAU | HashSet S | 1.1010 | , \h | Storily Queve | Blouryave | • |
| 41 | (C) VI | 1 1- 5 | ontedSet(1 | 0 | \ | - PatrockyBoo | (رووان |
| | 2 1.2 | 1 | Navigable | Set(I) | | → Linnsed Blocking | |
| | Stack | Linked Hookset 1.4 | 1 1. | 6 | 1 | |) |
| | | 41.4 | | | | |) |
| | Lagacy Closs | | 13 es Set | | | | • |
| | Lagaro | | 1-2 | | | | 3 |
| | | | | | | | • |
| | | | | | | | 3 |
| | | | | | | | 0 |
| | | | | | | 18 | O. |
| | | | | 2 | | | 9 |
| | | | | i | | | 1.4 |

The state of the state of the state of

the strategy



U

| Collection frame work: | |
|--|-----|
| | 80 |
| Collection (1): | 10 |
| - If we want to separement a group of individuou objects as a | - 1 |
| A CONTRACT OF THE CONTRACT OF | :) |
| Single entity then we should go for Collection. | 3 |
| -> Collection Interface defines the most Common methods which can be | . 1 |
| applied -fox any Collection object. | .) |
| |) |
| → The following is the list of methods present in Collection Interface. |) |
| 1 boolean add (Object o) | 0 |
| 1 boolean add All (Collection c) |) |
| | () |
| Object 6) | 0 |
| (boolean stimove All (Collection c) | 1.) |
| 6 boolean StetanAll (Collection c) | 9 |
| → To Sterrove all objects Except those possessent in C. | 0 |
| (Void Clean () | 0 |
| 1 boolean is Emply () |) |
| TO SECURE AND ADDRESS OF THE PROPERTY OF THE P | • |
| 1 int Side() 1 boolean contains (Object 0) |) |
| 520 03 |) |
| 6 boolean Contains All (Collection c) | 3 |
| · Object [] to Assonay() | 3 |
| (1) Iteration "tempon () | 0 |
| TO THE STATE OF TH | 0 |
| | O |
| | 0 |
| | 0 |
| T | 0 |
| The second secon | 0 |

The second secon

| · ® List (x):- | 137 |
|--|------------|
| -> List is the child Enterfag of Collection. | |
| -> 28 we want to stepstesent a group of individual objects when | e |
| deplicate Objects asse allowed & inscation Coidea is passesved. It | in coe |
| Should go-foor list. | |
| -> Ansestion Ocideor coin be pareserved by means of Index. | |
|) Years 3- and desplease objects by using Index. Hence Index | e nloca as |
| Very ampointant stole in List. | c pince as |
|) - List Enterface defines the following methods | |
| O boolean add (int index, Object o) | |
| 3 boolean add All (int index, Collection c) | - 0 |
| 3 Object Demone(int index) | |
|) @ Object get("bt "index) | |
| 927 (927 | |
| 5 Object Set (int index, Object new) | |
|) O int indexOf(Object o) | |
|) int last Index Of (Object 0) | |
| List I terration List I terrator () | |
| 2 Tt Contains 4 classes: | |

(1) Linked List (0):

U (II) Vectoristastastast

(Stack Co);

| (1) Asonay list (c): | • |
|---|----|
| → The underlying datastructure for Assaylist is Destrable Assay G | 0 |
| Growable Assay. | 3 |
| 3 | 1 |
| → Busentian Ondern P\$ ponescaved. | 1 |
| → dauplitate objects are allowed. | 1 |
| -> Heteageneous objects one allowed. | 3 |
| → Dull insertion is possible. | , |
| | |
| Constauctors:- | 3 |
| (1) | |
| (1) Association Al = Dew Association; | |
| - Caeates an Empty Asonaylist Object, with default initial Capacity 10. |) |
| | -3 |
| -> Once AL meaches 9t's max capacity then a new AL object will be | 9 |
| Greated with. | • |
| New Capacity = Casonent Capacity * 3 +1 | 0 |
| 1 - 5 - Constitution * 2 +1 | 1) |
| (2) Assay last 1 = new Assay last (int intra apacity); | •) |
| |) |
| -> Caeates an Empty Assaylest object with the Specified initial |) |
| Capacity, | .) |
| |) |
| (3) Assaylist l = new Assaylist (collection 5); |) |
| * | 3 |
| -> Greates on Equivalent Asmaylist object for the Given Collection obj | ず |
| ie, This Constauctor is for dancing blu Collection objects | J |
| I'm Couled alm Collected officer | O |
| | O |
| | U |

the state of the state of

manufacture of the section of the se

impost java. util. *; Ep1. Class Assay Let Demo P.S.V.m(Stazycz angs) Appaylist a = new Appaylist(). a. add ("A"). a. add (10): a. add ('A') a. add (noin): S.o.pin(a); [A,10, A, null] a. siemove (2); S-o-pln (a); [A 10, nui] a.add(2," m"); [A, 10, m, nau] a-add (" N"); [AHO, M, NA, N] S.o.pin(a); [A,10,M, non, M] 5.0 ph (a. 82800); 1/5) a-clearly; // [] a. addAll(a); //[A,10,M, Mx,N, A,10,M, MLX,N] -) Note: & Every Collection class to Stating() is oversolden to selven It's Content directly in the following formatt.) [obj1, obj2, obj3] 0 Usually we Can use Collection to Store & transfer Objects . to provide U Suppose for this Dequirement Every Collection Class implements U Sestalizable & Clonable anterfags. U 0 U

| → Asionaylist & Vector classes fir | pplements Random Access Britarfala, So tha | Ŀ |
|--|---|-----------------------|
| | n access with Some Speed. Hence, if | |
| CONT. 201 10 W 1000 | detailuable Operation Then best Suitable | Ŷ. |
| Clara Structura es Arragulist. | (Adlantage) | 8 |
| | operation on deletion, in the | 1 |
| | The coopest choice, because it required | -1 |
| differences blu Assocylist | | ·) |
| -Amonaylast | Vector |) |
| 1 No method is Synchronized 1 multiple threads Can access | Every method Ps Synchronized Only one Thread is allowed. | , , , , , |
| Assaylist Simultaneously, Fience Assaylist Object is not threadsfe | to Oberate on Vector Object at a time thence vector Object is Thread Safe. |) |
| (3) Threads asse not stequiled to wait, & Hence performance is | 3 at increases waithing time of threads Etteric performance is Low. |) |
| high. | | ú |
| 3 Antonoducid in 1.92 version & | @ Introduced in 10 version & Hence |) |
| -Hena "t is non-legacy | it is Legacy. |) |
| | | 9 |
| | |) J |
| | | J |

- 15 to the state of

| 200 | |
|---|------------|
| DLinked List @:- | i. |
| -> The workships of the object of dashly links like like | EA. |
| -> The Underlying datastructure is double Linked List. | 0.9 |
| ansertion condex is pereserved. | 21 |
| -> desplicate objects are allowed. | - 1 |
| → Heterogeneous " " | . 1 |
| → Dull insertion is possible. | -3 |
| | 1) |
| -> 8mplements Sescializable & Clonable Interfaces but not Mandomaccess- | . 1 |
| interfaces. | .) |
| → Bask Suitable PR own frequent operation insertion on deletion | 1 |
| for other state. | , |
| - Woodenest Choica of own frequent operation is Dietolival. | 0 |
| No. 21 | 3 |
| Constructions: | e : |
| 1 LinkedList & = Dew LinkedList(), | 0 |
| C LINEALISE 2 INCL |) |
| → Caeates an Smpty Linkedlist Object. | 0 |
| | • |
| @ Linkedlist & = new Linkedlist (Collection c) | <u>ي</u> |
| - for interconversion the Collection objects. | 3 |
| | o |
| Linkedlist Specific methods: | 5 |
| | |
| - Usually we an use Linkellist to implements Stacks & Queun | -) |
| to Suppose this sequipments linked list class define the following | :) |
| Se o a | J |
| Six Specific ornethody. | 0 |
| | O. |
| | () |

| + : | 0 | Aoig o | add First (Ob | ject o) | , | 12 | | 500,000 |
|-----|--------------|--------------|---------------|-----------|--------------|-------------|---------|---------|
| 8 | 0 | | add Last Cob | | | | | |
| 13 | (5) | | Stemove Fix | | | | | |
| | | | | | | | | |
| - | | | Stemove Las | | | | | |
| ¥5 | (6) | Object | get first () | 2 | | | | |
| ř. | (6) | Object | get Last () | | | EG. | | |
| , | ex! | | | | | | | |
|) | 75-333 | impoal | z gava uti | 1.*, | | | | |
|) | | class | LinkedListD | tmo | | | | |
| 1 | | £ | | | | | | |
|) | | P·s | -v-m (Stourn | act args |) | | | |
|) | | 1 | | | | | | |
|) | | 10 | okedList l | = new | Linked Listo | (); | | |
|) | | | | | | | | |
| • | | | add ("dunga") | , | | | | |
|) | | | add(30); | | | | | |
| , | | | add (nui); | | | software | 0 - 50 | dena |
|) [| In st, sycub | notion. C. | add ("dung | a"); | cce ve | agada. prin | 30 Dal | dunga |
| ΄. | SJW, 30, NY | I fegula, 11 | Set (0, "Se | twaste" | | | | |
| . 0 | denky, stu | BOOM SAP | ٠٠, م) ۵۵۵ | 40 × 411. | | | | |
|) - | V/10 7 | e from as | (0) | 2,00 | | | | |
|) | Mary | | PremoveLas | F(); | | | | |
| 6 | C, VATE | E. S. 6. | addFirst (" | ccc"). | | | | |
|) | | | opin(e); | | | | | |
|) | | 4 | · 111/c~3, | Lccc | , Venkey | , Software | , 30, n | [ווטו |
| , | | 4 | | | | | | |
|) | | 1 | | | | | | |
|) | | | | | | | | |
| , | | | | | | | | |

| in Wectoon (s): | |
|---|-------|
| | A. |
| - The Understying datastructure is Resideable assay or govocable | ť, |
| parea. | 4.1 |
| | (1 |
| -> Brisertian Onder 15 Preserved. | 1.3 |
| -> duplicate objects are allowed. | . 1 |
| → noll insertion is possible | |
| -> Heterogeneous Objects are allowed. | 1.3 |
| | () |
| → implements Sevializable, Clonable & RandomAccess Interfaces. | () |
| -> Best Suitable if our frequent operation is Retained & | 1.1 |
| evotionest choice if over frequent operation is insertion our | 0 |
| 1.011.00 | 3 |
| deletion in the middle. | 9 : |
| - Every method in vector is Synchronized. Hence vector object | .) |
| is ThreadSafe. |) |
| ······································ | 0 |
| Constructions: | .) |
| N N N N N N N N N N N N N N N N N N N | 0 |
| (1) Vector V = new Vector (); | 0 |
| CANADA AND DES BOS DAY IN IN IN MARKET WORLD CODE SAY THAT | • |
| - Cheates an Empty Vector object with default initial Capacity 10. | O |
| - One vector treaches it's max. Capacity a new vector object will |) |
| has a state of the second collect will |) |
| be Coneated with double Corpacity. | ٠) |
| new apacity = 2* Current apacity. | O |
| ® Vector V - and | J |
| Vector V = Dew Vector (int initial Corporatity); | O |
| (5) Vector V = new Vector (int initial Capacity, int incremental Capa | (a) : |
| (Vector V = New Vector (Connection c); | 0 |
| (Collection C); | 2,2 |

| į. | Vector Specific methods: |
|----|--|
| | → To add objects |
| | 0 ad/(a) |
| 1 | ① add(Object o) — C |
| 1 | @ add ("int "index, Object o) -> L |
| P | 3 add Element (Object obj) - v |
| 1 | - To Stemove Flomente - al |
| , | |
| 1 | (Object o) -> C Somewhore Element (Object o) -> V |
|) | e semove Element (Object o) -v |
|) | - Demove (int Index) |
|) | Semove Flement at a |
| J | Lo She move Element At (Int Index) -> v |
| • | Clean () |
|) | > @ nemove All Elements () → v |
|) | → To Stelestive elements |
|) | O get (int index) -> L |
|) | |
|) | element At (int index) → Y |
|) | 3 frest Element (); -> y |
|) | Last Element(); → v |
|) | |
|) | → Othea methods |
|) | O fot Side(); |
|) | 1 int Capacityu; |
|) | * 3 Enumeration elements(); |
|) | Enumeration elementsu; |

```
imposit Java. Util. *;
          class Demoi
           g b. 2· n· W (StaguidE) audis)
             Vector V = New Vector();
             S. o.pin(v. capacity (1);
             foo(int i=1; K=10; i++)
                V.add Element (i);
              S. o.pln(v. capacity ());
              V. addElemenE("A");
              S.o.pln (v. capacity ());
               8.0 pin(v);
      Opt
             [1, 1, 3, 4, 5,
V- Demove Element (9) // [1,2,3,4,6,6,7,8,10, A]
V. DiemoveEsement AEC 3) / [1, 2, 3, 5, 6, 7, 8, 10, A]
V. Fremove All Elemants () // []
                                                                               -)
```

| | 31% |
|--------|---|
| (4) | Stack (c): (LIFO) |
| • | > 21 is the child class of vector Contains only one Constauctor |
| | (1) Stack & = Dew Stack(); |
|) | methods: |
| 3 | (i) Object push (Object o) |
| 1 | To Possest an object into the Stack |
|) | (11) Object pop(); |
|) | To Demove and Dieturns top of Stack |
|) | (1) Object peek (); |
|) | To Defuvor top of the Stack. |
| 9 | (v) boolean empty(); |
|) | meturns true when Stack is Empty. |
|) | (Y) int Search (Object 0) |
|) J | surfusions the offset from top of the stack of the Object |
|) | is available, Othoscotse Stellins -1. |
|) | 84. impost java-Ufil.*; |
|) | Class StackDomo diffet S. search ("A"); 3 |
| .) | P.S. v. m (Stringt T args) S. Search ('c"); 1 |
|) | Stack s = new Stack(); |
|) | S. push ('A'); Speci (A & A |
| 5 | S. push ('R') |
| j | S. Posh C'c's; |
|) | S-o-pin (SI: / [A B C] S-o-pin (S. Search ("A")); |
| 3 | 3-0. pm (s. Search ('z')); -1 |

| Cuarsons !. | 400 |
|--|--------|
| | |
| Types of Cuasons !- | |
| W | 1 |
| -> 28 we want to get objects one by one from the Collection we s | . 1 |
| O B | hoold) |
| To food Curison. | 3 |
| → These are 3 types of Cuoisoons available in Java. | 3 |
| | .) |
| (i) Enumeration (1-04) |) |
| (#) Iterator (124) | .) |
| City I share |) |
| (m) ListItematon (1.24) | , |
| 'S F |) |
| (i) Enumeration (in 1-0 ver) |) |
| → IE is a Cuascon to metorieve Objects one by one from the | • |
| Collection. | 9: |
| |) |
| → ZE is applicable from legacy classes. | ó |
| → We Can Coneate Enumeriation object by Using elements(). | -) |
| | • |
| Public Enumeration elements(); | • |
| 9!. Enumeriation $e = 4$. elements(), | - 3 |
| | • |
| Vector object | • |
| VECTEON CO. | • |
| -> Enumeration Tel P. + P | - 3 |
| -> Enumeration anterface defines the following 2 methods. | :) |
| (i) Public booken has Moone Elements(); | J |
| | O |
| (ii) public Object nextElement(); | O |
| 59 | Э |
| The state of the s | U |

- (1) (+) (+) (=-1)

```
Ge:- imposit java.util. *;
             Class Enumeration Demo
              . p.s.v.m (staingt) args)
                 Vector V = new Vector();
                 for (int 1=0; i<=10; i++)
                   v.addElement(i);
                 S.o.ph(v); [0,1,2,3 --- «10]
                 Enumeration e = V·eluments();
                 While (e. has Mone Elements ())
                  Integea I = (Integea) e. next Element (1)
                   if (I%2 ==0)
                  S. o.pln(1);
                  S.o.pln(v);
                              [0,1,23,4,-.. 10]
              [0,1,2,3 ---- 10]
0
O
D
              Co , 1, 2,3 - - - 10]
```

| Limitations of Enumeriation: | 10 |
|---|----------|
| - Enumerolation Concept is applicable only for Legacy classes & | |
| hence it is not a Universal Cuissoon. | Ė |
| # 000 H | 1 |
| → By using Enumeration we can get only ReadAccess & we can't | 1 |
| Peofoom any overnove expensations. | 1 |
| → To over Come + | 73 |
| → To over Come these Limitations SUN people introduced | , |
| Iteration in 1.2 version. | |
| Tr 1 |) |
| Tteorator :- | 1 |
| ⇒ We Can apply Iteration Concept from any Collection object. | -) |
| ZE is a Utiliversal Cuerson. |) |
| was a second of the second of | |
| tohile Rhestating we Can Perstosim Sternove operation also, and | () () |
| to sead operation. |) |
| · Cue Can get Itemation object by Itemation (Ottober interface). |) |
| | • |
| Iterator "to = C- iterator() | -) |
| |) |
| Any collection object | |
| AND A CONTRACTOR OF STREET | J |
| · Thestation anterfall defined the following 3 methods. | , , |
| (i) Public boolean has Nexto; | , |
| an Dalle Ole to Dalle | 9 |
| | |

(i) public void

Demove();

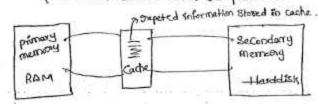
egs- imposit java . Utility. Class HashSetDemo Public Static void main (Starget args) HashSet h = new HashSetc); hadd ("B"); h-add ("c"); b.add ("D") h.add("z"). hadd (null) h.add Cio); S.o.pin(h.add (2)): / false S.o pln(b):/[[null , D , B , C , 10 , Z] Palse DUIL , D , B , C, 10, 2) Mote! Ensewhion order is not preserved (17) Linked Hosh Set (2);--> Lanked Hash Set 8s The Child Class of Hash Set. - It is exactly same as Hashset exapt the following differences. (4) -HashSet Linked Hashset J (1) The condentying D.S 9s -Hashtable 1) The Underlying D.S is a Combination of Hashtable & Linked List (i) In Seathan condean is not present F) ansertion condex is pereserved. U (Arm) Rotonoduced in 1-24 m) Introduced in 144

In the above program if we are replacing HashSet with Linked HashSet the following is the O/P.

YP! [B,C,D,Z, Dull, 10] re, insertion order is preserved.

Note:

→ The main impositant application asses of Linked Hash Set & Liskned-Hash Map is implementing <u>Cache</u> applications, where deplicates one not allowed & inscrition order must be preserved.



| 3) Coated Set (1):- | |
|---------------------|--|
| × | |
| | |

- -> 21 9s the child Interface to Set.
- The we want to suppresent a genoup of andividual objects actinding of the Some Southing contest. Then we should go food SouthedSet
- -> SoundSet Britishas defines the following 6 Specific methods
 - (1) object -first ()
 - → Dietuons the foots element of Sconted Set.
 - (1) Object Rost ()

 Shetward Last Clement of Shorted Set
 - (117) StratedSet headSet(Object obj)
 - Stefanns the SouldSele Whose elements are lessition obj

U

| * Toree Set Cas ! | |
|--|-----|
| -> The Underlying dataStructure is Balanced Times. | (6) |
| 9.00 FG | 200 |
| → duplicate objects agre not allowed. | |
| → Insection cooler is not precessorved because objects will be | e. |
| insexted according to Some Souting conden. | 1 |
| → Heteorogeneous objects asie not allowed otherwise Que coill get | ; |
| ClassCast Etemption: & NUIL insection is not possible - P. STYPOID - 8mpty Set. |) |
| |) |
| Constructions: | 3 |
| | .) |
| (1) ToneeSet t = new ToneeSet(); | .) |
| -> Coneates an Empty Theeset object whose the Southing condean | 9 |
| Ps defact Natural Scotling Obdes. |) |
| |) |
| The second secon |) |
| → Careates an Empty treasek object whose the Boarting objdex | .) |
| | • |
| is Customized Souting oudern Specified by Composition object. |) |
| (IT) Theeset t = new Theeset (Correction c) | , |
| | 1 |
| and ThereSet t = new ThereSet (Sonted Set C) | 3 |
| |) |
| Ens. imposit java. util. # | -) |
| Class Threset Demo |) |
| \(\tau\) | Ü |
| (Spin Capacital) on . v . 2 . d | 0 |
| | U |
| | 7.5 |

100 AND 100 PM

100

÷

```
Toree Set
                      t = new ToleeSet():
                t. add ("A");
                t. add ( "a");
                 Eadd (B)
                 Eadd ("z")
                t add ("L");
                1/2. add (new Integer(10)); //CCE Glass Cast Exception
                1/t.add (nui); 1/-> NOPE
                8.0. PIN(E); [A, B, Z, 1, a]
 )
 )
 ) On ull aceptance:-
   (1) from the NON-Empty Force Set if we agre togging to insent nour
    we com get Null Pointer Exaption (NPE).
  (3) foor the Emply Toreceset add the first Clement null insertion is always
      Possible.
) (61) But after inserting that null, if we agre trying to insert any-
     ether, we will get Numpointentraption (MPE).
)
          imposet Java-util +;
)
          Class TheeSetDemol
0
           P·S·V·M (String[] args)
O
            TENENSEL t = new TENENSEL();
               E. add (new Strong Boffer ("A");
0
                t.add (new Storing Roffer ("Z"))
U
                Eadd (new Storing Buffer (LY))
                E-add (new StorfigBoffer ("B"));
U
                                                   OLDI- : CLE
               S.o.pin(t);
()
```

| 40 28 we ask depending on Oblank natural Scotting order Compossar | " |
|--|------------|
| Objects should be Homogeneous & Compagnable othercoise we will | 4 |
| get ClassCastExaption (CCE) | 8 |
| N. S. | 3 |
| - An object is Said to be Companiable iff the Connesponding class | 8 - 8 |
| Prophements Companiable Britesfore. |) |
| Storing class & all worappear classes aloneady implements Composable | ٠, د |
| Enterface where as StoringBuffer down't implements Compassable Into fa | |
| Hence, an the above Example we got classCast Exception. | 1 |
| Compasiable Interface: | .) |
| - Foodble 2,700 pt. | :) |
| → This Interface posesent in jovalang package & Contains only |) |
| | ə : |
| One methodies, composito (). | -) |
| Public int Composito Object obj) |) |
| FORME THE COMPLICION OFFICE COD |) |
| Obj1. ComposieTo (obje) |) |
| |) |
| → 9 neturns -ve 1999 obj1 has to Come before obj2. | 0 |
| → Inclusions the 1999 obje has to Come after obje. |) |
| The April 1990 The Ap | -) |
| → 9 returns o iff objl & objl ane equal (dupliate) | .) |
| eg: impant Java-Uhl.*; |) |
| Class Test | .) |
| P.S. v. m Colinger args) | 0 |
| S.o.pln ("A". compose To("z")): // -ve -25 | 0 |
| S.o.pln (" z". Composeto (" K)) // +ve 15 | Ö |
| (8.0.pln("A". ComposeTo("A")); // 0 | Ü |
| 1) Go part a . Carippoetic A)); // 6 | () |

11.

The same special and the same

34744

147 → then were are depending on default natural Sorting corders internally Jum Calls Gom composition. -> Based on the stellan-type JUM identifies the Location of the element in Soating coaden. Obj1 . CompasieTo (obj2) already existing object which object weare in TheeSet. taying to add → hetuans -ve iff obj1 has to Come before obj2. - 9 setuans the TPF objet has to come after objet. → shekrans o 1969 objil & objil aske equal 9 TheeSet t = New TheeSet(); t. add ("z");) t. add ("K"); -> "K". compare To(z); -ve) t.add("D"); -> "D". Composito("x"); -ve E.add ("M"); → "M'. Comparato("D") → +ve 0 t-add ("D"); 'M' CompaneTo ("K") -> +ve -) "M'. composito ("z"); -> -ve / E. add (null). D'. Compasseto ('O') - 0) -) S.o.pln(t); .) 2 ChasCast Exception NDE [O,k,M,Z] 0 DULL - COMPANIETO ("O") -> RE => NAE

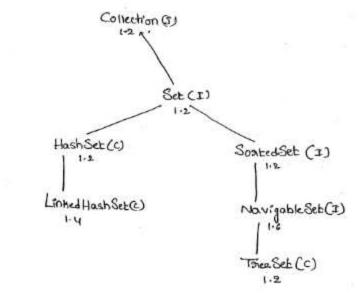
| → IP coe agre not Salfsfied with default naturnal Scotling condean | 1 |
|---|----------------|
| and if the natural Scorting order is not already Available. Then | (|
| we Can define own own Customized Scenting by using comparator | () |
| * Comparable ment from default national Scotting onder. * Comparation insent from Customized Scotting conders. | () () () |
| Companator (I): | • |
| - Companator EntexPage peresent in Java Util package & defines | .) |
| The following a methods. | 0 |
| | 3 |
| 1) public int compane (object obj., object obj.): | 9 : |
| -> stetusors -ve TPF obji has to Gome before obje | •) |
| -> shetusins the sife objit has to come after obje | • |
| - metuans o the objit & obje and equal (days cote). |) |
| i same of the ordinary | 9 |
| Obj 1 -> which object we agre togging to add |) |
| Obj2 => Alsheady existing object |) |
| | •) |
| @ Public boolean equals (Object obj) | C |
| |) |
| → cohen even we asie implementing Companiation Rotesface Compulsary | 9 |
| coe should parovide implementation for compared, and method | 3 |
| | Ü |
| · equalsc) implentation is optional, because it is already available for | 0 |
| Own class from Object class through Rohestania. | Ú |
| | U |

-b+ 100 - 16

15

→ Set is Child Interfood of Collection.

-> 28 we count to suppresent a good of objects where duplicates as not allowed & insention order is not pereserved. Then we should go for Set.



, → Set anterface does not Contain any method we have to use

Only Collection Rotes facimethod.

(i) HashSet (c):-

) - The underlying datastructure is Hashtable.

J → Desplicate Objects able not allowed.

U → 8F We ask toying to add duplicate objects we work to get to any C.5 on R. Estadd C) Simply metuons false, to hash code to the objects

The Installing conden is not preserved & an objects are inserted according

| -> Heterogeneous objects agre allowed. | |
|--|---------|
| The state of the s | n. E |
| -> hull insention is possible (only once) because diplicates are not allowed | |
| → HashSet Implements Sensationable & Clonable Interfores | 1 |
| | 13 |
| * Construction: | y k |
| 1) | , } |
| HashSet h = New HashSet(); | |
| → Coneates, an Empty HashSelt Object with different default initial | () |
| A | 1 |
| Capacity 16 & default fill Ratio 0.75 (75%). | .) |
| 2) HashSet h = new HashSet (int initialCapacity); . | .) |
| |) |
| → Coneates an Empty Hushselt object with the Specified Initial | .) |
| Capacity & default fill Ratio is 0.75. | J |
| 3) HashSet h = Dew HashSet (int institulapaeity, float fillmatio); | ூ: ் |
| O to 1 |) |
| 4) HoshSet b = new HashSet (Collection c); |) |
| Thanks I'm had harve (content) of, | •) |
| D | Э |
| Fine atio : - |) |
| -> After Completeing. The Specified states only a new HashSet |) |
| AND THE STATE OF T | .) |
| Object will be Cheated That pasticulasi Diatio is Called fillmatio of |) |
| load-factor. |) |
| -> The default finguatio is 0.75 but we can customized this value. |) |
| the depoir Allamio is only bor or any comment | Ü |
| | o o |
| | 9 |
| | 0 |
| | |

CHINE ON THE STREET

eg:imposit Java . Util - ": Class Iterator Demo public Static Void main (StaingE) args) Association 1 = new Association for (int i=0; i<=10; i++) 1. add (1); S.o. PIN(1); Lo, 1,2,3, - -. 10] Thereafor its = 1. iteratory; While (ita has Nexte (1) Integen I = (Integen) its. neat(); TF (IX2 ==0) Sopho(I); else Than . memove();) S.o.pln(1); [0,2,4,6,8,10] Limitations of ateration:-(1) In the Case of Itemation & Enumerication we can always move towards the footboard direction & we can't move backwood direction. O ive these Cuersons are Single directional Cuersons but not Biolitectional, (ii) while postermaning Iteration we can perform only mead sommer

| We Carit perform Steplacement & Addition of New Objects. | * |
|--|-----|
| → To Shesolve These penoblem Surv people Entenduced List-Iteration | |
| In 1-2 Vension | |
| | Ŧ |
| 1) List Iterator :- | 3 |
| -> List Eteration is the child Etheroface of Iteration. | 1 |
| → While Iterating Objects by ListInteration we can move either to | :1 |
| The forward on to the Backward direction in List Pterator | 3 |
| is a Bidirectional Curison. |) |
| Section and a contract section contract | .) |
| → While Iterating By ListIteration We Can perform theplacement & |) |
| addition of new objects also in addition to Akad & Remove operations. |) |
| -> We can Concate List Itexation object by using list-Itemation (1) at A | 9 |
| | 0 |
| List 20 texface. Small List object. | •) |
| Listaterator Liter = L. Listaterator(); | 0 |
| | |
| → List-Iteration anterface defines the following 9 methods | 0 |
| 1 (1) public boolean has Next (1) | O |
| and the state of t | 0 |
| South (ii) Public Object Dext(); | O |
| (iii) Public int nextIndex(); | 0 |
| 1 (1) Public booken has Panalous (). | 0 |
| 3) | 0 |
| (ii) public Object provious(); | o |
| (6) public int parevious Intent); | O |
| | 0 |
| | 6 |

```
1 public void 9 nemove();
       @ Public Void -Set (Object new); - Seplace on object with new Object

⑤ public void add (Object new); → add new obj.

€g:-
       imposit java-util. *;
       Class
              List Iterator Demo
        Public Static Void Main (Statingt) args)
            LinkedList 1 = new LinkedListo;
             ( add ( balationship);
              ( add( venky");
              l.add ( chiaid);
              Ladd (" rag");
              S. o. plo(e); [balakashna, verky, chionu, nag]
             Linhallist
             List Iteration An = 1. list Iteration().
             while (lfg. has Next ())
               Storing & = (Storing)(to. Deals()
               Pf (S. equals ('venki"))
                  Its, siemove U;
               if (S. equals (" chiau"))
                   ltm. sct ("chasan");
                if (s. equals (" nag"))
                  (ton. add (" chostu")
             1 12.0. HO(R);
                                             [Balaksusha, Charan, nag, Chally]
```

Moke.

- among 3 Cuersons Lest-Itematon is the most powerful Curson.

-But it is applicable only from List objects.

Compassion table of 3-Cussons :

| Potopesty | Enumeration (1-04) | 2 teenatoen (1.29) | List Plesatory (1.24) |
|------------------------|---------------------------------------|--------------------------------|----------------------------------|
| D Is 9t legacy | yes | No | No |
| It is applicable apply | only foon Legacyclasses | for any Collection objects | Only foo |
|) movement | Single dispections (only foscoord) | Single discolism (-famourd) | bi-discotional (forward Eback |
| D How to get it? | By using elemental) - method | By cessing Itexator() | By ceting |
| D Accessibility | only mead | Shead & Shemove | List Iterator() Bread Premove/ |
| 10 method | hasMoone Elements() | harnexto | 91epla@/add |
| | neat Elementu | Deato | 9 methods |

)

0

0

£9:impost java-util. *; Class TreeSetDemo3 Aublic Static void main (Staing[] args) Thee Zirteger IT = (Inlager) obj1; Integer 123/= (Integer ToneeSet t= new Traset (new my companion); ->0 t-add (20); t add (0); -> Compare (0, 20) -> +ve t-add (15); = Compare (15, 20) ->-ve Eadd (5); Compare (5,20) -> +ve Compare (5,0) -> +ve Eadd (10); Compare (5,10) -> -ve 9 Sopholes; Compare (10,0) -ve Compare (10,15) tue Compare (10, 5) -ve [20,15,10,5,0] class MyCompagnator Propriements Comparator Public Fot Companie (Object obj1, object obj2) Potegeon I, = (Integer) obil: anteger Iz = (Integer) obje; if- (I, < T2) return ((1,< 12) ?+1: (1,> 12 ?-1:0)); DETUDIO +100; else (P-(I+I2) elge deturno; Dietarin -1000;

)

-)

U

D

4

U

| → If we agre not passing Compagnation object at line 10 | |
|--|----------|
| Then Jum internally calls companietors which is ment for | 363 |
| default natural Scotling order In this case The opp is Co, 5, 10, 15, 20 | 3 |
| | J |
| → It we agre passing compagnation object at 10 then over own | 9. |
| Compasie method will be execuficited which is ment for Custom | izod) |
| Scorting condest. These are the op is [20, 15, 10, 5, 0] | |
| | , |
| Vasious alternatives of implementing compase(): |) |
| | .) |
| Class My Companiation implements Comparation |) |
| Direction to Company Comments and the state of the state | .) |
| Public int Companie (Object obj), object obj?) | .) |
| Integer I, = (Integer) obj ;; | 9 : |
| Integer Is = (Integer) Obje; |) |
| 1/ Deturn I, . composito (12); → [0,5,10,13,20] |) |
| And the state of t |) |
| Moletusin $-\mathbf{I}_1$. Composito (\mathbf{I}_2) ; \Rightarrow [20, 15, 10, 5, 0] |) |
| Dietusin Ig. Compareto (I); -1 (20, 15, 10, 5,0] | <u>ي</u> |
| // Stetuan - Iz. Compaseto (II); -> [0,5,10,15,20] | 5 |
| 1 Detunn -1; => [10, 5, 15, 0, 20] = Revenue of insection onder | • |
| / shelvan + + ; ⇒ [20, 0, 15, 5, 10] → insertion conden | 0 |
| |) |
| / stetusn o; = [20] |) |
| . j | 9 |
| } | 0 |
| | o |
| | 0 |
| | 0 |

Canada (1974) and the first of the first of

Southing conden is Dieverse of alphabetical conden. "imposit java. util. *; class Thee Set Demo2 Public Static . V. m(Staingel args) Theaset t = new Treaset (new my comparation ()): to add (A) f-ada (, z.); t-add ("K"); t-add ("B"); tadd ("a"); S. o. Pho(E);) 9

Class My Compasiation Traplements Comparation

Public int Compane (Object obj1, object obj2) Storing S = (Storing)obj1,

Strong Sp = Obje·tostoring (); ~

) Dietagin - SI. Compagneto (S2);

)

9

)

0

→ In Objects & StrungButter there is no Comparation, So we can convert 0 thto Stocks. O

20 object class companie

Note:

Method deap't Coolain Strings only contain assect the so

Objects Can be Consist into Storings by using typeasting

| * W.a.p to insert Strong & Strong Roffer objects into the Travel | |
|--|----------|
| Where the Southing outdoor increasing Length outdoor IP two objects | 11 |
| La cart 8 | |
| having the Same length then Consider their alphabetical conden | |
| A | 1.7 |
| 19 imposit java-util-+: | 1 |
| Class TreeSet Demois | - 3 |
| • | .) |
| PS V TO (Storing EI orgs) | 3 |
| , | .) |
| TreeSet t = Dew TreeSet (Dew My Comparator ()); | , |
| t. add ("A"); | •) |
| | -): |
| t.add (new Staring Burfor ("ABC")); | .) |
| t.add (new StrongBuffer ("AA")). |) |
| E.add (xx"); | .) |
| Production work the product | 9 ! |
| t.add ("ABCD"); | .) |
| tade ("A"), | • • |
| So.blu(F); [Y YY YX YE YEED] | .) .) |
| <i>y</i> , | 3 |
| class Mycomparator Proplements Comparaton | 3 |
| | 3 |
| Public int Compane Cobject obj1, object obj2) | 0 |
| · V | 0 |
| Storing S, = obj1. to Storing (); | 3 |
| Storing Sa = Obje . to Storing 12; | 3 |
| | J. |
| $SOE l_1 = S_1 \cdot length (s_1)$ | |
| The le = Sa. lengthon else setuan s. comparto(su) | 20 |
| if (1, <12) | ۵ |
| Dietron -1. | () |
| elec (le 710) Teleson to | U |
| The second secon | U |

to the second second second

CONTRACTOR

JP! [A, K, L, Z] NOK!. So SB Garbe Convert into Storing ~ 20 SP Garbe Converts mutton ~ 20 Storing Buffers there to Compares mutton ~ 28 we agre depending on default national Scorting condens Compulsary

U Objects should be thomogeneous & Companiable. Otherwise we writiged CCF.

1) > 84 we are depending on own own Sconting by Companion the Objects

Deed not be Composible fittomogeneous.

)

•

)

9

)

0

0

0

0

Companie Vs Componator :-

- Ton pose defined Companiable Classes default national Southing conder is almostly available of use able not Salistied with That we can define own own Customisted Southing By using Companiation

 Southing.
- Ton predefined Non-Companiable Classes default Datuma Souting of Order is not available Compulsary we should define Souting by using Companiation object only.
- 3 for own own Customized classes to define default natural Sorting ander we an go for Comparable & to define customized Sorting we should go for Comparator.
 - Sol- Employee, Student, Customen.

```
impost java. util. *;
     Class Employee Simplements Companiable
     d
        not eid;
        Employee (int eid)
          this eld - eld;
        Public Storing to Storing()
          neturns "E-" +eid;
        Public the CompasteTo (Object: obj)
          int eid = this eid;
         Employee e2 = (Employer)obj;
          ink eida = thakeg eid;
         TF (eids <eid2)
           9netuan -1;
         elsciteid, > eidz)
            aetuan +1;
         21Se
             return o:
       Class Comp Comp Demo
-)
0
         (spec 13 patrol8) m.v.aq
```

```
Employee e, = new Employee (200);
   Employee eg = New Employee (100);
    Employee & = new Employee(500):
   Employee ey = new Employee (500):
                                                               0.1
    Employee es = new Employee(700);
    Tocaset t, = new Treaset();
                                                               . .
     t. add (e);
    to add ( ca);
     ta add (e3);
      Ey add (ey);
      6, add (es);
                                                              0
                  [E-100, E-200, E-500, E-700]
    S.o.place,);
                                                              . )
                                                              .
   Theeset to = new Truset (new My Companation (1))
                                                              9:
                                                               )
    to add (e);
    to add ( e 2);
    to add (63);
                                                              .)
    ty: add (eq);
    to odd (es);
                                                              .)
   8.0. Pln (6); [E-700, E-500, E-800, E-100]
                                                              )
                                                              )
                                                              )
Class MyComposition Popplements Composition
1
                                                              0
  Public int Compane (Object obji, Object obje)
  f
                                                              O
       Employee e, = (Employee) Obj1;
                                                              O
       Employee ez = (Employee) obj1;
                                                              0
       Detasin ez. Composito(e); // setuan -e, composito(e);
```

| Naturnal Scotling condern is assistant the Same Salasiay the Same Salasiay the Theorem names. It was Compassation Class to alphabetical coordinated Employee no Same name then Consider desert | 0.00 |
|--|---|
| Companision blue Companio | Compositos |
| Define default national Scorting class implements Compositable character c | Dive Can use Comparation to define Customized Scotting conden. 3) This Potential present in journality Pockage. 3) defines two methods (i) comparator (ii) equals() 4) No Predefined Class implements Comparator Britishale. |

the first transfer of the contract transfer of

.

Compasisson table for Set amplemented Classes.?

| Proposty ' | Hashsent | Lin xed HashSelt | Touset |
|---------------------------|---------------|---------------------------|-----------------------|
| Underlying D.S | -tlashtable | Hashtable+ Linned lisk | Balanad Tree |
| insention anden | not-presented | preserved | not preserved |
| Soonling condean | N-A | N-A | posessored |
| Hefeorogeneous Objects | allowed | allowed | Not allowed |
| 5) Duplicate objects | not aslowed | notallowed | notallowed |
| null agptana | allowed (1) | allowed (1) | foor the empty |
| | 12 | *** ₁₂ | Thereset add The |
| | | | flish element |
| 10 | | | Dull Sosettion 95 |
| pe pe | | | Possible, in an other |
| | | | Gaes we will got NPE |
| | | | -9 |
| | | | - J |
| | | | ر |
| | | |) |
| | | | |
| | | | э |
| | | | Э |
| | | | ڼ |
| | | | θ |
| | | | o o |
| | | | O |
| | - | | 9 |
| | | | O |

Map (I) ?

- → 28 coe coant to shepphesent a group of objects as key-value points then coe should go from Map. both key & value one objects.
- → Both Key & Values asse Objects.
- → Duplicate keys ashe not allowed, But values can be duplicated.
- → Each Key-value passon 95 Called Entory.

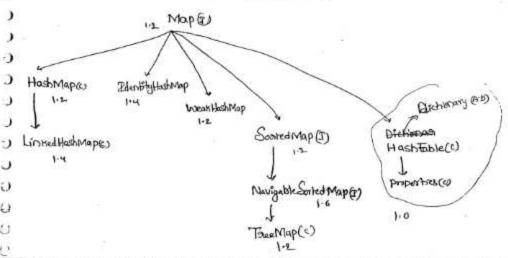
| Cyor. | Rollno | Dame | 7 |
|-------|--------|---------------|--------|
| | (01 | | Dentay |
| . 1 | 102 | duoga | Value |
| ney | 103 | Sainu Rau! | Valce |
| | 104 | Sambu | 1 |
| | los | Sinder | |

> Theore is no orelationship blu Collection & map.

) -- Collection ment foor a good of individual objects where as

·Map munt for a group of Key-value posting.

-> Map is not objid interface of Collection.



| Methods of Map 80/00face:- | |
|--|----|
| * O Object pur (Object key, Object value); | |
| - 11 4 | × |
| -> To add deby-value pain to the map | |
| → IP The Specified Key is almeady available than old value will be | 1 |
| Surfaced with | 1 |
| Steplaced with new value & old value will be stetusized. | 3 |
| (1) Vosa Put All (Map 10) | 7 |
| | |
| → To add a group of Key-value Pators for | , |
| 3 Obeck och Cale | , |
| 3 Object get Object Key) |) |
| -> Gretusins the value associated with Specified Key |) |
| |) |
| →28 The Key is not available then we will get Null | 9 |
| (1) Object Sternove(Object Key); |) |
| ACCEPTAGE AND AC |) |
| 6 boolean Containskey (Object Key) |) |
| 6 bootean Contains Value (Object value) | •) |
| 1 int size(); |) |
| | , |
| 3) boolean TEEmpty(): |) |
| 1 Void Clean () | 0 |
| Settlement of the Control of the Con | , |
| 1) Set KeySet(); | , |
| @ Evilection values: Collection views, of the Map. | 3 |
| (3) Set entogSet(); | ن |
| See dringsee(); | U |
| | O |
| j4 | 0 |
| | U |

Entory (Interfac):

-> Each key-value poson is Couled One Entony

- Without existing Map Object Those is no chang of Entony Object

thera, Interface Entry is define inside Map Britishas.

Gode: interface Map

Protesface Entary

0, Object get Key();
 0, Object getValue();

(a), Object setValue();

}

() Hashmap (

) -> The undealying dataStructure is HashTable

, → Heteonogeneous Objects aone allowed from both Kezys & values

) - duplicate Keys owne not allowed - from but the values Can be duplicated.

I Prosention conden 95 not possessed because it 95 based on Hash Code

U→ Dull Key is allowed (only on@)

U -> nall values asse allowed (any number of times).

10

- differences blw HashMap & HashTable :--HashMap +lashTable 10 No method 95 Synchronized O Every method is Synchroniad 1) multiple Threads Can Operates 3 At a time only one Thread is allowed Stinutoneously & Hena HashMap to Openate an Object. Hence et is Object is not Thread Safe Thread Safe. 3) Threads asie not siequioned to 3) It increases waiting those of the wait & hence Inelatively perstonnance Thread & Hoa performana is low. is High. D null is allowed for both DUM is not allowed for Both key & Key & value Values . Otherwise we will get NPE AM 3 anthoduced in 1-2 version & ⑤ Introduced, in 1.0 Version & it is legally 1 at is non-Legacy 2) How to get Synchronized Wasian of Hashmap? → Bydefault HashMap object is not Synchronized, but we can) get Synchronized version by using Synchronized Map () of Collections, Class. Map M = Collections. Synchronized Map (HushMap hm);

Constructor :-

- (i) HashMap m = new HashMapl);
- Coneates an Emply HashMap Object with clefault initial Capacity level is 16 & default fill Ratio 0-75 (78%).
- (II) HashMap m = new +MashMap (tot "nitrai Capacity)
- -HashMap Hashmap (int introdapacity, float fill Ratio) m = new
- Hashmap (4) no = new HashMap (Map m)
- Ex: imposit java. util. *
- Class HashMapDemo
- P.S.V.M (Stronges args)
- HashMap m = new HashMap();)
- m. put ("chiranjeevi", 700);)
- 10. put (balaiah , 800);)
- m. Put ("venkatesh", 1000);) 0
- m. put (* Dagasyjuna *, 500);)
- 8.0. pln(m); & wennates = 1000, balasah = 800, Chiannjewi=700,
- raganina = 500)
- S.o.pin (m. put ("chisanjeevi", 1000)); 700) 0
- Set s = m.keySek();)
- 8-0 PIN(s); [Warmatesh, balaiah, Chraanievi, nagarjuna] Collection c= m. values ();
- (002, 000), 000, 800 (Ca) (Ca)
 - Set S1 = m. entry Set(),
 - Itemator its = S, itemator();

| While (its. has Nezto) | |
|--------------------------------------|--|
|) | |
| Map. Entry m, = | (map. Entary) 945. neat(); |
| 1.34 Enring 1.11 = | () |
| 8.0.pln (m, · getkey | () + " " +m; -get values()) |
| ee (· · · · · · · · · · · · · · · · | Dagarina 500 |
| TP (TO, . get Key 1) . eq | vals (nogosjuna")) Verxula 1000 balash 800 |
| m, · Set Value (1000 | chinges 1000 |
| <u> j</u> | 1 |
| S o plo (m); } rago | onjuna =10000, Verikatesb=1000, balafosh=809, |
| , | Chiranjeevi=10002) |
| } | , |
| |) |
| 10 Linxed HashMap :- | 3 |
| | 2 |
| → It is the child class of . Ho | ashMap. |
| -> It is Exactly & same as H | ashmap except the following differences; |
| -Hashmap | Lioxed-HoshMap. |
| O The underlying D.S is HashTable | o the underlying D. S is Hashtable +) |
| 0 0 - 1 0 1111/100 | Linkedlist |
| @ ansertion Order & not posserved | the second secon |
| | (3) answitting orders is preserved |
| misitaly 3.1 ni bauboratas @ | 3 Enthoduced in 1-4 Version |
| | 3 |
| | |
| | able Steplacing HashMap with Linked) |
| Hashmap. The following is the a | φ. · |
| & chesaniesi = 700, balaich = 80 | 10 15 0 Katesh = 1000 magazina = 500 2 |
| | , 0 9 7 7 |
| i.e insention ander is poresen | |

U 100

→ The main application asses of LinkedHashSet & KinkedHashMap & athe cache applications implementation where duplication is not allowed & insortion ander must be pereserved. (iii) Identity Hash Map :-- It is exactly Same HashMap Exacept the following difference. -> En the Case of Hashmap to identify duplicate Keys JVM always uses equals(), which is mostly ment for Content Composition.) - If we want to use == operator instead of equalsu to identify duplicate keys we have to use IdentityHashMap (== openation always must ton neference Composition). (10) <u>eg</u>!-Hashmap m = new Hashmapu; 200 Integen 11 = new Integen(10) Thtegen 12 = New Integer (10); · equals() → Contact == -> reference m. put(11, " pavan'); I == I2 - falle m. Put(iz, " Kalyan"); I, equals (Ia) - True 8.0.pln (m); 10 = Kalyan) ightarrow 80 the above code 1, & i_a agre duplicate Keys because i, equality

Notes-

.)

)

)

Detuans take.

U → If we steplace HashMap with Identity HashMap Then The O/P is

(10 = pavan, 10 = Kalyan)

| Weak HashMap_s- | |
|---|------|
| -> It is exactly some as Hashmap exapt the following differen | DG. |
| → En The Case of HashMap Object is not alle eligible for g.c events | |
| It doesn't have any external references of it is associated with | יינ |
| Hashmap. i.e., Hashmap donsinates Garbage Collecton (g.c). | |
| -> But In the Case of weakflashmap Eventhough object associated | |
| With weakHashMap, it is eligible for g.c. if it doesnot have | |
| any external sufferences. i.e G.c dominates weakHashMays. | 0000 |
| ego - import java cutil+; | |
| Class Weak HoshMapDemo | 3 |
| P. S. v.m (Storing [] args) throws Intersupted Exception | - |
| P. S. V.M (Storing [] args) Himocos Interscripted Exception | |
| Hashmap m = new Hashmap U; | - |
| Temp t = new Temp(); | |
| | - |
| m. put (t, "duoiga"), | |
| S.o.pin(m); {temp = duage} | 3 |
| k = nun; | 0 |
| System ge (); | , |
| Theread . sleep (5000), | -) |
| 8.0.000 | -) |
| 8.0.phn(m); | J |
| (temp = dunga) | U |
| 1 | U |
| | |

W. S. 1. 1. M. S. 1.

Class Temp Public Staing to Staing() hetuan "temp"; public void finalized System.out.porintin("-finalize method called"); %! Hemp = duoigaz dtemp = duoyaj) If we treplace Hashmap with weak-lashmap then the opp is 9 t-temp = duaga} -Paralitie meltood Called

14)

9

) 0

| (3) Soonled Map (5): | |
|--|----------|
| | 45 |
| - If we want to Dieponesent a group of entoces according to | |
| Some Scotling borders then we should go for ScotledMap. The | ¥ V |
| Southing should be done based on the keys but not based on the Values. | ì |
| -> Scotted Map anterface of the child anterface of Map. | 1 |
| → Soute-Map Enterfale defines the following 6 Specific methods |) |
| O Object freshkey(); |) |
| (a) Object lastKey(); | 3 |
| 1 | 3 |
| 3 SoontedMap headMap (Object 18841); |) |
| (Soaled Map boil Map (object Key)) | ວ ອ : |
| © Sooted Map Silver Color & J. J. J. |) |
| Soonted Map SobMap (Object Keyl, Object Keyl) | • |
| 6 Companator Companator(); |) |
| | .) |
| (ii) TseeMap (ii) | 9 |
| -) -tl.: |) |
| -> The undealy D.S is RED-BLACK TOWN, |) |
| → anseation ondear is not pareserved & all-thronies agre inseated |) |
| according to some C |) |
| according to some Scottering Onder of Keys. | • |
| → 28 we ask depending on default national Souting conden Then |) |
| The Kox Charlel L. | 9 |
| The Keys Should be Homogeneous & Compactable. Otherwise coe coll | 0 |
| get Class Cash Exaption (CCE). | e e |
| | |
| -> 2f we agre defining our own Southing conden by Composition Then; | Oi |

The Keys need not be Homogeneous & Companiable.

- → There agre no greistauctions on Values, they can be Heterogeneous & non-Compagnable.
 - → duplicate keys agre not allowed but values Can be duplicated.

null acceptance:-

- 1 → foor the Empty FreeMap as the feet Entary is null key is
- , allowed but after insenting that Entry if we are trying to insent
-) any other Entary we will get NullpainterException (NDE).
-) foor the NON-Empty TolerMap of use able torying to inscort Entory
- Coith null key we will get Nullpointer Exaption (NDE)
-) These age no grestouctions on null values i.e, we can use
- , null any no of times any where for map values.
- , Constauctoons ?-
-) (1) Thermap t = new Thermap()
-) for default natural Souting conder,
- Theemap t = new TheeMap (Composition <)
- J foor Customized Scotling conders.
- , (it) The Map t = new Theemap (Map m)
- (v) Thermap t = new Thermap (Souted Map m)

```
Eg:
        Proposit Java util *;
        Class Theemap Demo3
           P·S·V·M (Starling[] angs)
             Thee Map in = new Thee Map();
                m. put (100, "zzz");
                m. put (103, "444");
                100 put (101 , "xxx");
                m. put (104 , 106);
                m. put (107, null);
                /m. put ("FFFF", "xxx"); // CCE
               / m. put (null , "xxx"); //NPE
                8-0-plo(m); 100 = zzz , 101 = xxx , 103=444 , 104 = 106 , 107=000)
   0/0-
   100 = ZZZ, 101 = XXX , 103 = YYY, 104 = 106, 107 = null }
                                                                      0
                                                                      0
                                                                      U
```

```
impoort java. util. *;
          Class TheeMapDemo
             P. S. V. m (Starting[] angs)
                 Theemap t = Dew TheeMap (new MyComparator());
                  t. put ("xxx", 10);
                  E-put ("AAA", 20);
                   E. put ("zzz", 80);
                   t. put ("LLL", 40);
                  8.0.pln(t);
            Class My Companation implements Composition
            1
              public int compose(Object obj1, object obj2)
               1
0
                    Strong S, = Obj1 - toStrongu;
0
                    Stating & = objectoStating();
)
)
                     neturn 82. composito (Si);
)
C
C
0
     0/01.
                         XXX = 10 , LLL = 40 , AAA = 20
J
J
O
```

U

Hashtable@: -> The Underlying datastructure is HashTable. -> Heterogeneous objects are allowed for both keys & values → Insertion condern is not presented & it is based on Hash Code of the Keys. → null is not allowed for both key & values otherwise one will get Numpointer Exception (NDE). → duplicate Keys one not allowed, but values can be duplicated. → All methods ashe Synchshonized & Hence Hashtable Object 95 Thoread Safe. Construction ! (1) Hashtable h = new Hashtableco -> Caeales an Empty Hashlable Object with default initial capacity, is 11 & default fellowahio 75% (0.75).

h = new Hashtable (int initial Capacity)

h = Dew Hashtable (int float firmatio)

Hashtable h = new Hashtable (map m):

(F)

(Ti)

(TU)

Hashtable

Hashtable

_eg!- imposit java-util.*; Class Hashtoble Demo P. S. v. m (Strong[] args) Hashtable h = new Hashtable(); 10 h - put (new Temp(s), "A"): 9 h-put (new Temp(2), "g"); 8 τ h put (new Temp (6), 40); 6=0 6 h put (new Temp(15), "D"); 5=A, 16=F 5 h. put (new Temp(23), " E"); 15=0 4 h put (new Temp(16), "F"); 3 2-3 2 1/h.pot("dwoga", null); //NPE 23 = E System out pountln(h); 9) {6=c,16=F,5=A,15=0,2=B,23=E}) C Class Temp -forom top to bottom & Right to Left -) C int it) Temp (int i))) thes. 7 = 1; 0 Public int bashCode()) Stelluan 1; 5) 0 Public String to String (0 U Detuan i+ " ", 0

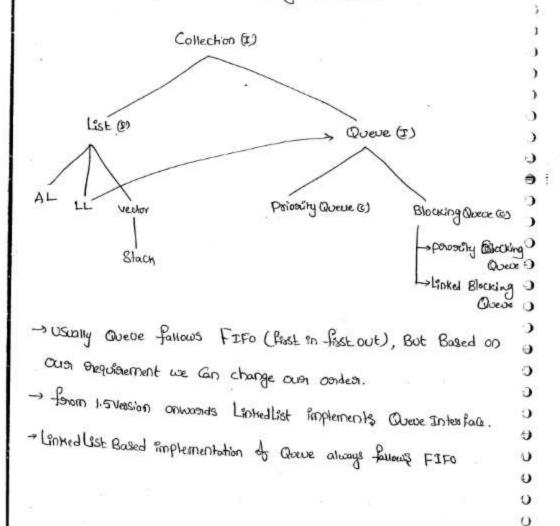
| 37) Pshoperhies (C):- | |
|--|-----------|
| → It is the child class of Hostitable | |
| → In Our perogram of any thing which changes frequently (like | |
| database usernames, passissands, woll never grelommended to | 1 |
| handlode the value in the Java program. Because for Every | 1 |
| Change, we have succomplex, shebuild, shedeploye the application & | 3 |
| Sometime even Seven nestant also negotimed. Which Cheates a | . } |
| big business impact to the client. |) |
| -> We have to Configurate those vasicables (parapeonies Positie | 1 |
| Posoperhies Pro C |) |
| Poroperties files & coe have to Gread Those values from javacode. |) |
| the main advantage of this approach is, If any change in | .) ∋ : |
| The peroperties for Till a lar | |
| The peroperties file Jost enedeployement is enough which is not |) |
| a business impact to the client. | ó |
| Construction: | -) |
| The state of the s |) |
| (i) Paropeaties p = new Paropeaties (); | 3 |
| - 1 - 1 - | • |
| → 8n the Case of Properties both key & value Should be Stock | jə o |
| Methods :- | , |
| x |) |
| *(1) Storing get Property (Storing Property Name) | 3 |
| - Stephane -+ | O |
| - Sieturns the value associated costs Specified powposty | O |
| (3) Stang Sekpenoposty (Stoung prome, Storing product); | O |
| | O |
| -> to Set a new peroperty. | £ E |

| | 16-33 |
|-----|--|
| | (1) Stocking Enumerication peroperty Names (1) |
| | * (b) Void load (Input Stoream is) |
| 9 | - To load the peroperties - From peroperties files into java properties- |
| 040 | Object. |
| 1 | (4) Void Stoone (Output Stoream as , Storing Comment) |
| 1 | > To Update peroperties from peroperties object finto propostics fre. |
| • | Eg:- Imposit java. ulil. *; [User = scott] |
| , | user = scott |
| 100 | impost Java 10. *; Venxi = 8888 |
| , | Class Peroperties Demo Pood = tiger |
| , |) |
|) | D. S. V. m. CC. |
|) | P. S. V m (Strange) angs) throws IDEXception |
| • | <u> </u> |
|) | Paopenties P = New Paropenties (): |
|) | File Input Stream Is = new File Input Stream ("abc. properties"). |
|) | P-load(Ris); |
|) | |
| 5 | System out powntln (p); |
| | Stagna a Dade Daniel (4 198) |
|) | Staring 8 = P. get Paro perky (" Venki"); |
|) | S.o.pln(s); |
|) | · · |
|) | P. Set Poroperty ("nag", "949999"); |
| 0 | I co O don't Co |
| 0 | Fix Output Stream for = new Fix Output Stream ('abc. proporties'); |
| U | P. Store (fos, "Updated by dunga for ScJP Demo class); |
| O |) Google for ScJP Demo class); |
| U | , J |
| 0 | 4 |
| 300 | |
| - | |

1.5 Vesision Enhancement:

Queue(I):

- It is the child anterface of Collection.
- -1 if we want to Represent a govern of individual Objects posson to possessoring. Then we should go for Queue.



Queve Interface methods 8-

- (1) boolean offer(Object obj)
 - → To add an object ento the Queve.
- (1) Object peek ();
 - -> To statusin head element of the Queve if Queve is Empty than
- This method shelmans null.
- 1 (167) Object element (1);
- To stetion head element of the Queve. If Queve is Empty
- Then we will get Runtime Exception Baying No Such Element Exception
-) (v) Object Policy;
- , -> TO Sternove & Stetusin head element of the Queoe. 28 Queue
-) Is Empty then this anothed meturns null.
-) (1) Object sumove();
- > To Diemove & Dietuan head element of the Queue, if Queue is
- , Empty then we will get stantime Exception Saying No Such Element Except

| Postoschy Queie (2):- | |
|---|-----|
| | |
| → This is the Data Stouchure to hold a good of individual Objects | 179 |
| perior to percassing According to Some periodity. | 1 |
| -> The parioacity can be either default natural Scotling andean con |) |
| Customized Souting conden. | 1 |
| → if we are depending on default natural Scorting Compulsary | .) |
| *1552.157 NO |) |
| Objects should be Homogeneous & Compostable otherwise we win get | .) |
| Class(ast-Exception- |) |
| + if we ask defining our own Customized Scorling by Comparation | •) |
| | 0 |
| Then the objects need not be Homogeneous & Companiable. | 3 |
| → duplicate objects asse not allowed. | .) |
| → Brisistion conden is not preserved |) |
| 200 Seekee Colon Lines |) |
| -> null insention is not possible even as first element also. |) |
| Constructions :- | • |
| Constructions :- |) |
| (i) poriosity Queve queve = new pareouty Queve (); | • |
| 20 20 20 | 0 |
| -> Creates an Empty Pourouty Queve with default thithau Capacity !! | • |
| & Paccaity andem is default naturnal Scotling condem. |) |
| · · · · · · · · · · · · · · · · · · · | 0 |
| (11) forthisothy Overe 9 = new poricothy Overe (10t inthe apacity) | Ċ |
| (6) Paingribe (a) | O |
| (16) Psilosity Queve 9 = new Psilosity Queve (into sither apacity, Compared | Ovo |
| (c) Parionity Queve 9 = New Parionity Queve (Connection c); | . O |
| - Landaces 1 - (sem Languard (Conscious) c) | U |

Prically Obere 9 = New Prically Obere (Scated Set S) **Ξ**g∶. imposit java. util *; Class Porostry Queue Demo P & v m (Stanger angs) Posiosofy Queue 9 = New Poslosofy Queue (), 8.0. plo (9. peeks); //nui 18.0. PIO(9. elementa); //NSE wo suchtograte x copiron for (Pot 1=0; 14=10; 14+) 9. offer (i); Soplo(3): // [0,1, 2, 3, 4, 5, --- 10] S.o.plo(q.poli()); //o Sopho(9); // [1,2,3,4,5 ---10] eg 2! -

U Class postority Queue Demos U (P. S. V. m (Storing C) args)

imposit java util. *;

)

0

)

C

)

)

O d

```
Poriosity Queve 9 = New Poriosity Queve (15, new My Comparator ()).
         9. offer ("A");
        9. offer( 'z');
         9. offer ("L");
         9- offer ( B)
         8-0-plo(9); / [z,L,B, A]
    Class My Comparator Emplements Comparator
      Public int Compare (Object obj!, Object obj.)
         Starg & = (Starg) obji;
         Storing Sz = Obje. to Storing ();
         netunn Sy. Componeto (Si);
of Di. [2, L, B, A]
                                                                    0
```

U

0.000 11 003

| 1.6 Version Enhancements :- | 161, |
|---|---------------------|
| (3) Navigable Set (3) :- | |
| → It is the child Enterface of SconledSet. | |
| > This Interface defines Several mathods to porous | le Support foo |
| navigation for the Toxeset object. | 100 S.A |
| → The -following List of various methods peresent | : in Navigable Set. |
| (i) Ceiling (e): | |
| → Stefuans the lowest element which is >= 0 | 0 |
| (1) higher (e) =- | - |
| -> 9 returns the lowest element which is >e | |
| (iti) Places (e): | |
| → stetusins highest element which is <=e | |
| (v) Lowence)! | |
| -) Stetumns the highest element which is | Ke. |
| (x) podlfrsk()! | |
| - Fremove & Freturns first element | |
| () pollast (): | |
| -> 9 remove & stetuans last element. | |
| (17) desending Set ()! | |
| W 33.54 | |
| - Steturns the navigable Set in Steverse condection | λ. |
| | |
| | |
| | |

```
imposit java util *:
Class Navigable Set Demo
  P.S. v. no (Storing [] angs)
    Thee Set < Integers t = new Thee Set < Integers ();
     t.add(1000):
     t. add (2000);
     L.ada (3000);
     t.add( 4000);
     t-add (5000);
     S.o.plo(t) $ [1000, 2000,
     S.o.pin(t. ceiling(2000)); [2000
     8.0.pln (t. higher(2000)); 3000
     S.o. Plo(t. Ploos (3000)); 3000
     8-0-pin(t. lower (3000)); 2000
     8.0.ph (t. palifixt()); 1000
      S.o.plo (t. poliLastes); 1000
      8.0.pm (t. desending Set 0), [$7000, 3000, 2000]
      S.o.phn(E); [2000, 8000, 4000]
```

| | 1005 |
|---|------------------|
| (Pr) Navigable Map (I):- | |
| → It is the child interface of ScothedMap to define | Several method |
| -foot Navigation Poorposes. | |
| -> The following is the list of methods present in 1 | lavigable Map |
| (1) Ceillingkey (e.) | |
| (I) higheakey (e) | |
| (5) Pleconkey (e) | |
| (N) locoenkey (e) | |
| (Polifiss Entay() | |
| (ii) pollLast Entry() | |
| (1911) cleseraling Maps | |
| Ego. impost java util.*; | |
| class NavgablemapDemo d p. S. v. m (Stoding [] angs) | |
| The Man Subar Stans L our The Man St | esta String > Os |
| The Map & Strong; Strong > t = new The Map < St | |
| t. put ("b", "banana"). | |
| t-put ("c1, "cat"); | |
| E-put ("a1, "apple"), | |
| t. pat ("d", "dog"); | |
| t-put ("g", gun"); S.o.pm(E); of a=apple, b=bonona, c=at,d=dog | , g-340) |

S.opin(t. ceiling key ("c"); S.o.pln(t. higherthey('e')); S.o.pin (E. Ploonkey ("e")); 8.0.pln(t. lowerkey ("e")); d 8.0-plo(t. pollfistEntry ()); a = apple 8. o.pln (t. poll Last Entry()); g=gun S-o-plo(t-desending Map()); of d=dog, c=cat, b=bonanay 8.0.pln(t); 1 b = banana, c= cat, d=dogg

Collections class

| Collections class:- |
|---|
| -> 2's an utility class possent in Java util package |
| → It defines Severial ultity methods for Collection implemented class object |
| Southing the elements of a list: |
| → Collections class defines the following methods to sout elements of last. |
| @ O Public Static void Sout(List ():- |
| → We can use these method to Scot according to natural Scot condern. |
| → 8n This Gase CompauSasy elements Should be Homogeonus & |
| Compasiable cities coise coe coill get Glassast Exception. |
| → List should not Contain null, otherwise we will get NullpointeGeop |
| @ Public Stake void Sout (List E, Companator) :- |
| → To Soort elements to a list according to Constornized Scorling condean |
| Seasching the elements of a list: |
| - Collections class defines the following method to Search elements of a list |
| 1 public static int binary Search (List 1, Object obj) |
| |

→ TF The List is Scated according to natural Scorting corden then we

have to use this method.

| @ public Static int binary Seasch (List 1, Object Key, Comparation C) | |
|---|----|
| → Zf the List is Souted according to Comparators than con house to co | ie |
| This method. | 1 |
| Conclusion 8- | 1 |
| 20.0001(8 | 3 |
| -> Anternally binasy Search method uses Binosy Seasich algorithm. | 1 |
| → Before Calling binosyseasich() method Compulsasy The List should be | 3 |
| Souted 4 | 3 |
| Sosted otherwise we will get unpredictable shesults | , |
| - Successful Search metuans index. |) |
| - unscricessfull Search Dietuous inseallion point | , |
| -> 8nSertion point is the Location cohere we can place element in the |) |
| Souted List. | • |
| |) |
| → 28 The List is Scotled according to Companator Then at the time | \$ |
| Search also coe should pass the Same Companation Otherwise we will g | |
| |) |
| Composedictable sussuits. | -) |
| En: To Search elements of list | -) |
| |) |
| Propose Jona. Off. 4; | , |
| class CollectionsSearchDemo | |
| 1 | 5 |
| P. s. v.m (Storing 17 angs) |) |
| 4 | Э |
| Associated to new Association; | J |
| 1. add ('z'); | O |
| 1. als ("A"); | O |
| 1. add ("m"); | O |
| | U |

東京 高 平東 等 一京 ト

1-add ('k')) 1. add ('a'); S-o-pln (L); [z, A, M, k, a] Cotlections. Soot (1); S. o. pln (1); S.O. pln (Collections. binasy Search (1, "z')); 3 S. o. pln (Collections. binary Seaton (1, "j")); -2 Ex2!-Propost Java. Util. *; Class Collections Seatish Dernol P. S. v. m (____) Appropriet (= new Appropriet(); (add (15); L. add (o); 1 . add (90); 0 Q. add (10); L - add (5); 15 0 20 10 5 8.0.pln(1);) Collections. sout ((, new MyComparator());) 20 15 10 50 S.o.pln(e);) 8.0. pln (Collections. binary Search (1,10, new My Comparators)); //2 J S.O. Pln (Collections - binary Search (1,13, new My Comparator ())); //-3 S. o. pln (collections, binary Search (1,17)); / -6 unposedictable 0 because it is not passing comparately

| Public for Compare (object obj.), Object obj.) Integer 1, = (Integer) obj.; Integer 12 - (Integer) obj.; Deturn 12 · Compase To (1); Note: | Class My Comparation Suprements Comparator of | |
|--|---|-----|
| Integer 1, = (Integer) obj 1; Integer 12 = (Integer) obj 2; Setuen 12 · Compase to (1); Note: | 23 | |
| Enleged is = (Integer) obje; Deturn is. Compose to (1); Note: | 1 | |
| Section 12. Compase to (11); Note: | Integer in = (Integer) Obj1: | 10 |
| Note: For the List Contains n elements Rapge of Successful Seasch Range of successful Seasch: 0 to n-1 (1) Range of unsuccessful Search: -(n+1) to -1 (2) total Range: -(n+1) to n-1 (3) total Range: -(n+1) to n-1 (4) Range of successful Search: -0 to 2 Range of successful Search: -4 to -1 Total Range: -4 to 2 | Zolegest 1/2 = (Integes) obja; | 70 |
| Note: For the List Contains n elements Rapge of Successful Seasch Range of successful Seasch: 0 to n-1 (1) Range of unsuccessful Search: -(n+1) to -1 (2) total Range: -(n+1) to n-1 (3) total Range: -(n+1) to n-1 (4) Range of successful Search: -0 to 2 Range of successful Search: -4 to -1 Total Range: -4 to 2 | Such an far Commeta (1). | 1 |
| Feo the List Contains n elements Rapge of Successful Seasich O Range of Successful Seasich: 0 to n-1 O Range of unsuccessful Search: -(n+1) to -1 O total Range: -(n+1) to n-1 Parge of Successful Search: 0 to 2 Range of successful Search: -4 to -1 Total Range: -4 to 2 | 5.5 | 2 |
| Range of successful Search: 0 to n-1 (3) Range: -(n+1) to n-1 (4) Range: -(n+1) to n-1 (5) Range of successful Search: -(n+1) to -1 (6) Range of successful Search: 0 to 2 Range of successful Search: -4 to -1 Total Range: -4 to 2 | Note :- | 3 |
| Range of successful Search: 0 to n-1 (3) Range: -(n+1) to n-1 (4) Range: -(n+1) to n-1 (5) Range of successful Search: -(n+1) to -1 (6) Range of successful Search: 0 to 2 Range of successful Search: -4 to -1 Total Range: -4 to 2 | Poor the List Contains a elements Range of Successive Seasoch |) |
| (a) total Range: -(n+1) to n-1 (b) total Range: -(n+1) to n-1 (c) 1 | | ; |
| (a) total Range: -(n+1) to n-1 (b) total Range: -(n+1) to n-1 (c) 1 | O Rarge of Successful Seasich: 0 to n-1 | , |
| (3) total Range: -(n+1) to n-1 En! 1 -2 -3 -4 10 20 30 0 1 1 Range of successful Search: 0 to 2 Range of unsuccessful Search: -4 to -1 Total Range: -4 to 2 | | 3 |
| Parge of Successful Search = 0 to 2 1 2 3 3 3 3 3 3 3 3 3 | e range of unsuccessfull Search: - (n+1) to -1 |) |
| Parge of Successful Search = 0 to 2 1 2 3 3 3 3 3 3 3 3 3 | 3 total Range: -(n+) to n-1 |) |
| Range of successful Search = 0 to 2 Range of unsuccessful Search: -4 to -1 Total Range: -4 to 2 | | ٠, |
| Range of successful Search: -4 to -1 Total Range: -4 to 2 | 9. | 9 : |
| Range of unsuccessful Scarch: -4 to -1 Total Range: -4 to 2 | 10 20 30 |) |
| Range of unsuccessful Scarch: -4 to -1 Total Range: -4 to 2 | 0 1 1 | , |
| Range of unsuccessful Scarch: -4 to -1 Total Range: -4 to 2 | Range of successful Search = 0 to 2 | 3 |
| Total Range: -4 to 2 | | Ĵ |
| | Kange of unsuccessful Search: |) |
| | Total Range: -4 to 2 |) |
| ວ ວ ວ ອ | | -) |
| ر ر ن | |) |
| 3 3 | | , |
| o o | | -3 |
| | | 3 |
| 3 | | O |

u

Reversing the elements of a list:

-> Collections class defines the following neverse method for this

15 0 20 10 5

Public Static void Sovense (List ();

Ed. To Revenue elements of List

Perpose Java-util- *;

Class Collections Reverse Demo

P- S- v-m (____)

1

AL (= new ALU);

1 . add (15);

1 · add (o);

1 . add (20);

1-add (10);

(2)bba. 1

S.o. pinces;

.

Collections. Dieveose (l);

S.o.pln(l); 5 10 20 0 15

) ,)

ာ ၅

9

)

)

Ð

| Shevense() Vs Shevense() !- | |
|---|------|
| → We Can use Shevease() method to Shevease the elements of a late | |
| and this method Contain List altroguement | Ŧ |
| this thenos Catholi List diseasement | - 3 |
| - Collections class defines shevense conden method also to shelusin |) |
| Companation object from Sherrenising Opiginal Souting conden | 1 |
| a confirm county concer | - 30 |
| Consider a Collection of the Control of | , |
| Compagnation C, = Collections. Dieverise Opiles (Compagnior C) | |
| | , |
| | , |
| decenting condean asenting condean | , |
| Section 1997 |) |
| - Dievesise Obdes () method Gan take Contains Companion assignement | ć |
| When as sevense() Contains List auguements. | 9 |
| |) |
| EN:- TO REVERSE ELEMENTS OF LIST |) |
| |) |
| Proposet Java · Util · *; | • |
| Class Collections Reverse Demo |) |
| (Constant) |) |
| Public Static Void main (Strange) orgs) of |) |
| Assaylist & = new Assaylist (); | • |
| l - add (15); |) |
| € ad(6); |) |
| £-add (20); |) |
| 1 - ad d (10) | 9 |
| 1.add (s); | J |
| 2.0.pln(1); 15 0 20 10 5 | O |
| | O |
| Collections. Devenue (U) | U |
| (if 9.0. bus (p) [21/0] = | 12 |

24.4

-Amonaya class

Abovays class :-

→ 8t is an Utility class possent in Util Package, to define Severnal Utility meltods flor Annays for both posinitive Amazys & Object-type Annays

Sociting the elements of Among:

- Asionally class defines the following methods for this.
- 1 public Static void Sout (pormitives) p);
- → To Sout elements of Annual According to Natural Souting conden
- 9 (3) public Static void Scat (Objects a)
-) TO SOAL elements of Object Asianay According to Natural Souting Obides.
-) In this Case Compulsary the elements should be Homogeneous & Compositable Otherwise we will get Class Caste Exception.
-) @ Public Static void Scot (Object () a, Comparation c) ...
-) → To scat elements of Object[] according to Customized Souting condex.
- · Note:-

0

- posimilare as object placeus can be Scated only by natural Scating conders
- where as Object Assays Can be Scated either by natural sooking.
 - Ooden on by Customized Sconting Conden.

| But To SORT elements of American | |
|--|------|
| Assays Sout Demo jova | |
| Persposit java util Assays; | |
| A STATE OF THE STA | 1.33 |
| imposet Java cutil Comparators | - 1 |
| Class AnnaysSontDemo | 1 |
| 7 | 1 |
| Public Static word main (Statige) age) | 1 |
| 1 | , |
| fotes a = \$10,5, 20,11,6}; | , |
| Sacket many . same hole sales. "1. | |
| 8-0-pln("portmitive Assoray before Sosting:"); | |
| -for (int at; a) | , |
| 8 1000 | , |
| S-o-pin(ai); |) |
| Ann C-1-C-2- | 9 : |
| Asistays, Sout (a); |) |
| 8.0. Pln C parmitive Assay After Souting: 1): |) |
| -form(sist as ; a) |) |
| 1 |) |
| 8.0. pln (ai); | .) |
| j | •) |
| State on a 14 AT 4 1 1077 | , |
| Stocky [7 5= 1" A", "2", "B" }; |) |
| Sopling Object Approx Before Scoting: 1): |) |
| -for (Stains az:5) | 3 |
| | ٠, |
| S.O.PIN (Q1); A | 3 |
| y | - 0 |
| Assays. Sost (S): | Ü |
| S.o. pln (* Objects Asianay After Bosting: "): | 9 |
| 9 | 363 |

THE TE

1

```
En! · imposit Java·ulil·*;
      imposit Static java util Assacs. *;
      Class AssayoSearch Demo
        P-8.V.m(---)
          intel a = $10, 5, 20, 11, 6};
          Approays. Sout (a); / Sout by notural order 5 6 10 5
         S.O. Pln (Abonays. Whoaby Search (a, 6)); 1/1
         S.o.pln (Asisays. binary Search (a, 141); 11-5
         Stanger s = / A", "z', 8',
          Assays. Sost (s);
                                                            AZB
         System.out. pountin (binosy Search (s, "z")); 1/2
          S.o. yn (binary Search (s, "s")); // -3
         Annays. Sout (s, new My Composatos (1);
          S.o. ph (binasy Search (s. "z", new My Comparator (1)); // o
          S.o.ph (binary Search (s, s, new My Composalor ()); 11-1
          S.o.pln(binasySearch(s, "N")); //unpredictable sesult
                                                                         .
   Class My Comparation finiplements Comparation
                                                                         0
       public mix Compare (Object or, Object or)
```

Storing S₁ = O₁ · to Storing ();

Storing S₂ = O₂ · to Storing ();

Storing S₃ . Composite (S₁);

Conventing Appage to List :-

6 Public Static List aslist(Object() a)

) → By USing this method we are not Caealing an independent List Object just we are Caealing list view for the existing Asianay Object.

) - By using List shefements if the perstrain any operation the Changes

Cutil be shefteded to the Asistay sheftering. Similarily, By cising Asistay shefterence if we perform any changes those changes will be shefted

to the list.

)

)

) — By Using List Dieference we can't perform any operation which works) — The State, Ci.e, and & Demove) otherwise we will get Shuntime Exaption

Saying "consupposited - Operation-Exception" (USOE)

) By using List preference use an persform preplacement operation

But Deplacement Should be could the Same-type of element only ottomise

we coin get Runtime Exception Saying "Array Stone Exception

U EN. TO VIEW ADONG IN LIST FORM.

Aonay Aslist Demo java

impost java. util.*; Class AssaysAslistDemo 2 Elpirel2 Public Static void main (Stranger args) Staing[] & = /'A', 'z', 'B'}; LEST L list 1 = Abovays.aslist(s); 8-0-plo(1); // [A, Z, B] KBB S[0] = 'K'; [A, z, 8]. [k, z, 8] So-pln(e); /[k,z,8] 1. Set (1, "L"); [k, £, 6] -for (Staing Si:s) S.o.pln(8); /[k, L, B] 1-add ('dooga'); & / USOE 1. Semove (2); R. 1/ USOE l. Set (1, "s'); [k, £, 8] . Ck, s, 8] 1 Set (1,10); ARE // About Stone Exception 4.

O