**Enum Questions**

1.What is the output of below snippet?

public class Demo

{

public static void main(String[] args)

{

Map<Integer, Object> sampleMap = new TreeMap<Integer, Object>();

sampleMap.put(1, null);

sampleMap.put(5, null);

sampleMap.put(3, null);

sampleMap.put(2, null);

sampleMap.put(4, null);

System.out.println(sampleMap);

}

}

**Answer:**

{1=null, 2=null, 3=null, 4=null, 5=null}

2. Can we create instance of Enum outside of Enum itself?  
a) True  
b) False

**Answer**:

No, you can not create enum instances outside of Enum boundry, because Enum doesn't have any [public constructor](http://java67.blogspot.sg/2012/12/how-constructor-chaining-works-in-java.html), and compiler doesn't allow you to provide any public constructor in Enum. Since compiler generates lot of code in response to enum type declaration, it doesn’t allow public constructors inside Enum, which enforces declaring enum instances inside Enum itself.  
  
Read more: <http://www.java67.com/2013/07/15-java-enum-interview-questions-amswers-for-experienced-programmers.html#ixzz5NayiTAGJ>

3. enum Season {

WINTER, SPRING, SUMMER, FALL

};

System.out.println(Season.WINTER.ordinal());

**Answer**: a  
Explanation: ordinal() method provides number to the variables defined in Enum.

4. What is the output of below code snippet?

class A

{

}

enum Enums extends A

{

ABC, BCD, CDE, DEF;

}

**Answer**: b  
Explanation: Enum types cannot extend class.

5. What is the output of below code snippet?

enum Levels

{

private TOP,

public MEDIUM,

protected BOTTOM;

}

**Answer**: d  
Explanation: Enum cannot have any modifiers. They are public, static and final by default.

6. What is the output of below code snippet?

enum Enums

{

A, B, C;

private Enums()

{

System.out.println(10);

}

}

public class MainClass

{

public static void main(String[] args)

{

Enum en = Enums.B;

}

}

**Answer:**

10

10

10

7. . What is the output of below code snippet?

MyEnums

{

    FIRST, SECOND, THIRD, FOURTH;

}

public class Test

{

   public static void main(String[] args)

   {

       MyEnums[] myEnums = new MyEnums[4];

       for (int i = 0; i < myEnums.length; i++)

       {

           System.out.println(myEnums[i]);

       }

   }

}

**Answer**:

compilation error because enum keyword not present before MyEnums

8. . What is the output of below code snippet?

enum Levels

{

TOP, MEDIEUM(10), BOTTOM(20, 30);

int i, j;

private Levels()

{

}

private Levels(int i)

{

this.i = i;

}

private Levels(int i, int j)

{

this.i = i;

this.j = j;

}

} public class MainClass

{ public static void main(String[] args)

{ System.out.println(Levels.TOP.i);

System.out.println(Levels.TOP.j);

System.out.println(Levels.MEDIEUM.i);

System.out.println(Levels.MEDIEUM.j);

System.out.println(Levels.BOTTOM.i);

System.out.println(Levels.BOTTOM.j);

}

**Answer**:

0

0

10

0

20

30

}

**ArrayList Questions:**

1. Which of these standard collection classes implements a dynamic array?

a) AbstractList

b) LinkedList

c) ArrayList

d) AbstractSet

**Answer**: c  
Explanation: ArrayList class implements a dynamic array by extending AbstractList class.

2. Which of these class can generate an array which can increase and decrease in size automatically?

a) ArrayList()

b) DynamicList()

c) LinkedList()

d) MallocList()

**Answer**: a

3. Which of these method can be used to increase the capacity of ArrayList object manually?

a) Capacity()

b) increaseCapacity()

c) increasecapacity()

d) ensureCapacity()

**Answer**: d  
Explanation: When we add an element, the capacity of ArrayList object increases automatically, but we can increase it manually to specified length x by using function ensureCapacity(x);

4. Which of these method of ArrayList class is used to obtain present size of an object?

a) size()

b) length()

c) index()

d) capacity()

**Answer**: a

5. Which of these methods can be used to obtain a static array from an ArrayList object?

a) Array()

b) covertArray()

c) toArray()

d) covertoArray()

**Answer**: c

6. Which of these method is used to reduce the capacity of an ArrayList object?

a) trim()

b) trimSize()

c) trimTosize()

d) trimToSize()

**Answer**: d

7. What is the output of this program?

import java.util.\*;

class Arraylist

{

public static void main(String args[])

{

ArrayList obj = new ArrayList();

obj.add("A");

obj.add("B");

obj.add("C");

obj.add(1, "D");

System.out.println(obj);

}

}

a) [A, B, C, D].

b) [A, D, B, C].

c) [A, D, C].

d) [A, B, C].

**Answer**: b  
Explanation: obj is an object of class ArrayList hence it is an dynamic array which can increase and decrease its size. obj.add(“X”) adds to the array element X and obj.add(1,”X”) adds element x at index position 1 in the list, Hence obj.add(1,”D”) stores D at index position 1 of obj and shifts the previous value stored at that position by 1.

Output:

$ javac Arraylist.java

$ java Arraylist

[A, D, B, C].

8. What is the output of this program?

import java.util.\*;

class Output

{

public static void main(String args[])

{

ArrayList obj = new ArrayList();

obj.add("A");

obj.add(0, "B");

System.out.println(obj.size());

}

}

a) 0

b) 1

c) 2

d) Any Garbage Value

**Answer**: c

9. What is the output of this program?

import java.util.\*;

class Output

{

public static void main(String args[])

{

ArrayList obj = new ArrayList();

obj.add("A");

obj.ensureCapacity(3);

System.out.println(obj.size());

}

}

a) 1

b) 2

c) 3

d) 4

**Answer**: a  
Explanation: Although obj.ensureCapacity(3); has manually increased the capacity of obj to 3 but the value is stored only at index 0, therefore obj.size() returns the total number of elements stored in the obj i:e 1, it has nothing to do with ensureCapacity().

Output:

$ javac Output.java

$ java Output

1

10. What is the output of this program?

class Output

{

public static void main(String args[])

{

ArrayList obj = new ArrayList();

obj.add("A");

obj.add("D");

obj.ensureCapacity(3);

obj.trimToSize();

System.out.println(obj.size());

}

}

a) 1

b) 2

c) 3

d) 4

**Answer**: b  
Explanation: trimTosize() is used to reduce the size of the array that underlines an ArrayList object.  
Output:

$ javac Output.java

$ java Output

2