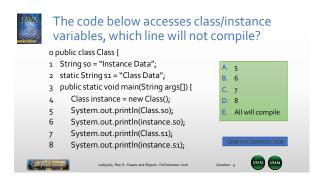


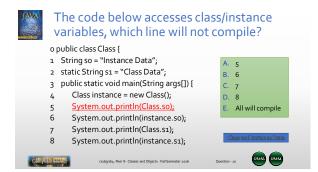
<ol> <li>In Java, code and data can only exist in a class.</li> <li>Instantiation does not require memory allocation.</li> <li>Instantiation makes a class from an object.</li> <li>Many objects can be made from a single class.</li> <li>Only a single object can be made from a class.</li> <li>A. 1) and 3) C. 2) and 3) E. 1) and 5)</li> <li>B. 1) and 4) D. 2) and 4)</li> </ol>	
3) Instantiation makes a class from an object. 4) Many objects can be made from a single class. 5) Only a single object can be made from a class.  A. 1) and 3) C. 2) and 3) E. 1) and 5)	
4) Many objects can be made from a single class.  5) Only a single object can be made from a class.  A. 1) and 3) C. 2) and 3) E. 1) and 5)	
5) Only a single object can be made from a class.  A. 1) and 3) C. 2) and 3) E. 1) and 5)  B. 1) and 4) D. 2) and 4)	
A. 1) and 3) C. 2) and 3) E. 1) and 5)	
B 1) and (1) D 2) and (1)	
B. 1) and 4) D. 2) and 4)	
CsuSybSq. Peer 8 - Classes and Objects - Fall Semester 2006 Question - 3	

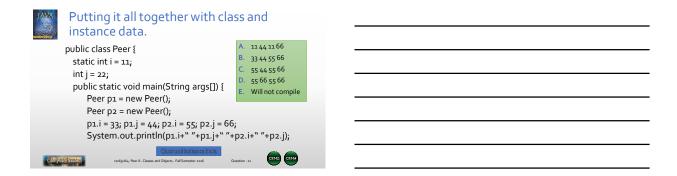
JAVA	Which of the following statements about objects and classes are correct?	_	
	objects and classes are correct:		
1)	In Java, code and data can only exist in a class.	_	
2)	Instantiation does not require memory allocation.		
3)	Instantiation makes a class from an object.		
4)	Many objects can be made from a single class.		
5)	Only a single object can be made from a class.		
	A. 1) and 3) C. 2) and 3) E. 1) and 5)		
	B. <u>1) and 4)</u> D. 2) and 4)		
Compil	cosignisis, Peer 8 - Classes and Dejects - Fall Semester Joses Ouestron - 4 (S164)		

Which of the following statements about	
public versus private are correct?	
A. Public variables and methods cannot be accessed outside the class in which they are defined.     B. Priyate variables can be accessed outside the class.	
only by writing "getter" or "setter" methods.  C. Private methods cannot be non-static, but public	
methods can be, and both can be static.  D. Private methods comprise the 'interface' provided	
to users of the class.  E. If you instantiate a class from outside the class you instantiate a class from outside the class you will provide the class you will be compared to the class you will be class y	
can access both private and public variables.  cusjudi, Peter 8- Clause and Objects - Pal Semester assis  Cusjudic - Semester - Clause and Objects - Pal Semester - Clause - C	

Which of the following public versus private  A. Public variables and methods outside the class in which they.  B. Private variables can be access only by writing "getter" or "se conduction of the conduction of the class.  C. Private methods cannot be comprise the to users of the class.  E. If you instantiate a class from can access both private and process of the class.	are correct?  cannot be accessed y are defined. sed outside the class tter" methods. on-static, but public a be static. b'interface' provided  outside the class you united the class you united the class you united the class	
Which of the following static and non-static  A. Static data is also called instant data is called class data.  B. Only one copy of instance (not C. There is a separate copy of instant is instantiated.  D. Accessing class data using the object name is not a good practice. Accessing instance data does name, if done from within the	are correct?  nce data, and non-static  n-static) exists.  stance data for every object  e class name instead of the ctice.  not require use of the class same class.	
Which of the following static and non-static  A. Static data is also called instandata is called class data.  B. Only one copy of instance (notential composition of the copy of instance).  D. Accessing class data using the object name is not a good pract. Accessing instance data does name, if done from within the	are correct?  nce data, and non-static  n-static) exists.  stance data for every object  e class name instead of the ctice.  not require use of the class	







Putting it all together with instance data.	h class and
<pre>public class Peer {     static int i = 11;    </pre>	A. 11 44 11 66 B. 33 44 55 66 C. 55 44 55 66
int j = 22; public static void main(String args Peer p1 = new Peer();	D cc 66 cc 66
Peer p2 = new Peer();	:
p1.i = 33; p1.j = 44; p2.i = 55; p2 System.out.println(p1.i+" "+p1	•
Chas and instance Co	Question - 12 (S16)