A. Which of the following is NOT an improved feature of AXIS over its predecessor?

1. AXIS is faster as compared with the previous version of Apache SOAP.
2. 
   AXIS introduces a deployment procedure to allow services to be registered into an AXIS engine.
3. AXIS provides the flexibility of introducing different combination of handlers to achieve a task.
4. AXIS provides tools to enable developers to generate WSDL document of a service.

B. Why do we need to introduce serializer and/or deserializer handlers to an AXIS engine?

1. Complex data types such as array or structure are so complicated that requires the corresponding serializer and/or deserializer to extract their contents.
2. Serializer and/or deserializer is required to convert a big object into many smaller ones to improve the network transmission efficiency of an AXIS engine.
3. Serializer and/or deserializer is required to translate between the XML representation of some data types and the format native to the programming language they are defined.
4. Serializer and/or deserializer is required to provide the encryption and/or decryption service to some data types to enhance the service security.

C. What do we mean if the “scope” of a service is set to “application”?

1. The service will be available until a client stops it.
2. The service will be available until the client who starts the service finishes its operation.
3. If one client modifies the content of an object created by this service, the other clients can also see this modification when they retrieve this object.
4. The object created by this service can be used across different requests of a client of this service.

D. The following segment of client program is used to invoke a service. Which of the following statements is correct about the program?

```java
String endpoint = "http://158.132.208.18:8080/axis/services/CountNumber";
Service service = new Service();
int number = 1;
Call call = (Call) service.createCall();
call.setTargetEndpointAddress(new java.net.URL(endpoint));
call.setOperationName(new QName("http://158.132.208.99/server","count"));
String ret = (String)call.invoke(new Object[]{number});
```

1. The remote method CountNumber() of server with IP address **158.132.208.18** will be called by the client.
2. The remote service requires an integer array as the input parameter.
3. The service to be invoked is “count” located at **http://158.132.208.99/server**.
4. The “invoke” method of the “Call” class by default will return an Object class object.