1. **Serializable**

**Định nghĩa:**

- Serialization là 1 cơ chế để truyền trạng thái của đối tượng qua mạng.

- Seriaalizable là một interface không có thuộc tính, không có method, được sử dụng để “ĐÁNH DẤU” các lớp java để các đối tượng của các lớp này có thể nhận được khả năng nhất định.

**Note:**

* Tất cả các lớp Wrapper mặc định implements interface Java.io.Serializable
* Một lớp implements Serializable thì tất cả các lớp con của nó cũng sẽ được serializable.
* Nếu một lớp có một tham chiếu của một lớp khác, tất cả các tham chiếu phải được implements giao tiếp Serializable nếu không quá trình serilization sẽ không được thực hiện.
* Nếu có bất kỳ thành viên dữ liệu static trong một lớp, nó sẽ không được serialized bởi vì static là một phần của lớp chứ không phải đối tượng.

**Từ khóa** transient

* Nếu không muốn serialize bất kỳ thuộc tính nào của một lớp, bạn có thể đánh dấu nó với từ khóa transient.

1. **Non-Blocking and Blocking IO. Advantages and disadvantage. Working with channel**

<http://tutorials.jenkov.com/java-nio/non-blocking-server.html>

1. **Networking – Client- Server architecture, basic protocols, their similarities and differences.**

**Dop:**

Table ( 2 4.3 мой столь )

Int xyz

double abc

String name

-84 -19 0 5 115 114 0 ???? (maybe something about serialization)

5(Length) 84(T) 97(a) 98(b) 108(l) 101(e) ClassName

-99 -18 109 -3 -30 126 -119 -111 SerialVersionUID

2 0 3 number of field

68 0 3 double

97 98 99 abc

73 0 3 int

120 121 122 xyz

76 0 4 String

110 97 109 101 name

116 0 18 76 106 97 118 97 47 108 97 110 103 47 83 116 114 105 110 103 59 120 112 ?????

64 17 51 51 51 51 51 51 4.3

0 0 0 2 2

116 0 35(length)

-61 -112 -62 -68 М

-61 -112 -62 -66 о

-61 -112 -62 -71 й

32 space

-61 -111 -17 -65 -67 с

-61 -111 -30 -128 -102 т

-61 -112 -62 -66 о

-61 -112 -62 -69 л

-61 -111 -59 -110] ь

Мой столь

If that is string : my table

116(string) 0 8(length) 77(m) 121(y) 32(space) 116(t) 97(a) 98(b) 108(l) 101(e)

Table ( 4 5.6 фывфыяч )

Int size

Double price

String asdf

[-84, -19, 0, 5, 115, 114, 0, 5, 84, 97, 98, 108, 101, 0, 0, 0, 0, 0, 0, 0, 123, 2, 0, 3,

68, 0,5,

112, 114, 105, 99, 101, price

73, 0, 4,

115, 105, 122, 101, name

76, 0, 4,

110, 97, 109, 101, asdf

116, 0, 18, 76, 106, 97, 118, 97, 47, 108, 97, 110, 103, 47, 83, 116, 114, 105, 110, 103, 59, 120, 112, 64, **22, 102, 102, 102, 102, 102, 102**, 5.6

**0, 0, 0, 4,** 4

**116, 0, 34, -61, -111, -30, -128, -98, -61, -111, -30, -128, -71, -61, -112, -62, -78, -61, -111, -30, -128, -98, -61, -111, -30, -128, -71, -61, -111, -17, -65, -67, -61, -111, -30, -128, -95] мой столь**

Table ( 4 5.6 фывфыяч )

Int xyz

Double abc

String name

-84 -19 0 5 115 114 0 5 84 97 98 108 101

0 0 0 0 0 0 0 123

2 0 3

68 0 3  **int 3**

97 98 99 **xyz**

73 0 3  **double 3**

120 121 122 **abc**

76 0 4 **String 4**

110 97 109 101 **name**

116 0 18 76 106 97 118 97 47 108 97 110 103 47 83 116 114 105 110 103 59 120 112 64

22 102 102 102 102 102 102 **5.6**

0 0 0 4 **4**

116 0 34 -61 -111 -30 -128 -98 -61 -111 -30 -128 -71 -61 -112 -62 -78 -61 -111 -30 -128 -98

-61 -111 -30 -128 -71 -61 -111 -17 -65 -67 -61 -111 -30 -128 -95 **мой столь**

Int double string

-39 58 -14 61 -57 65 52 -19

-39 58 -14 61 -57 65 52 -19

119 70 -13 84 73 106 -54 66

**Table**

**Int xyz**

**Double abc**

**String name**

[-84 -19 0 5 115 114 0 ???

5(length) 84(T) 97(a) 98(b) 108(l) 101(e) ClassName

37 21 -96 -77 -45 -82 44 19  **// modifers of field**

2 0 3 maybe 3 fields

68 0 3 field double 3

97 98 99 abc

73 0 3 field int 3

120 121 122 xyz

76 0 4 field String 4

110 97 109 101 name

116 0 18 76 106 97 118 97 47 108 97 110 103 47 83 116 114 105 110 103 59 120 112 64

22 102 102 102 102 102 102 5.6

0 0 0 4 4

116 0 34 -61 -111 -30 -128 -98 -61 -111 -30 -128 -71 -61 -112 -62 -78 -61 -111 -30 -128 -98 -61 -111 -30 -128 -71 -61 -111 -17 -65 -67 -61 -111 -30 -128 -95] мой столь

3.7

**OSI Model**

Has **7 layers,** used to **standardize communication** in a way that allowed multi vendor systems. For example Dell laptop connect with Huawei… etc.

The OSI model is a guide for developers and vendors to smooth the progress of developing communication products and software program that will work in cooperation with a commonly established model.

Application(7) : SMTP, FTP, Telnet

Presentation (6) : Format Data, Encryption

Session (5) : Start and Stop Sessions

Transport (4) : TCP, UDP, Port numbers

Network(3) : IP Address, Routers

Data Link (2) : MAC address, Switches

Physicial (1) : Cable, Network Interface Cards, Hubs.

1. The **physical** layer is the electrical/physical layer of the model. This layer encompasses the network cables, power plugs, …. Takes responsibility to carry the data across physical hardware such as Ethernet.
2. **Data Link layer :** The most complex of the layers. Divided into 2 sublayers : MAC ( Media Access Control ) and LLC (Logical Link Control). The layer sets up links across the physical network.When this layer receives data from physical layer, it checks for transmission errors then package the bits into data frames
3. **Network :** IP Addresses. Handle the routing of the data. The layer sends data to the correct destination on outgoing transmissions and receives incoming transmissions as well.
4. **Transport :** TCP , UDP . responsible for the transmission of data across network connections. This layer coordinates how much data to send, how fast, where it goes , and these sorts of things.
5. **Session** Connections between devices. This layer supports multiple types of connections. For example reconnection if a network interruption occurs.
6. **Presentation** Performs the uncomplicated task of syntax processing. Or convert data from one format to another. For example when ordering product from an online store, the transaction is handled in a secure transmission ( hashing here ).
7. **Application :** Provides network services to the end user. These services are protocols that work with the data the client is using. One of these protocols maybe HTTP ( Google Chrome, FireFox, Outlook, Skype, IE , Office ) . All of those interactive applications provide a set of services that allow the application send data to Presentation or receive data from Presentation layer.

Ж - 1046(10) – 0416(16) – 10000 010110(2)

2 bytes

110 + 5 bit + 10 + 6 bit

110 **10000** 10 **010110**

