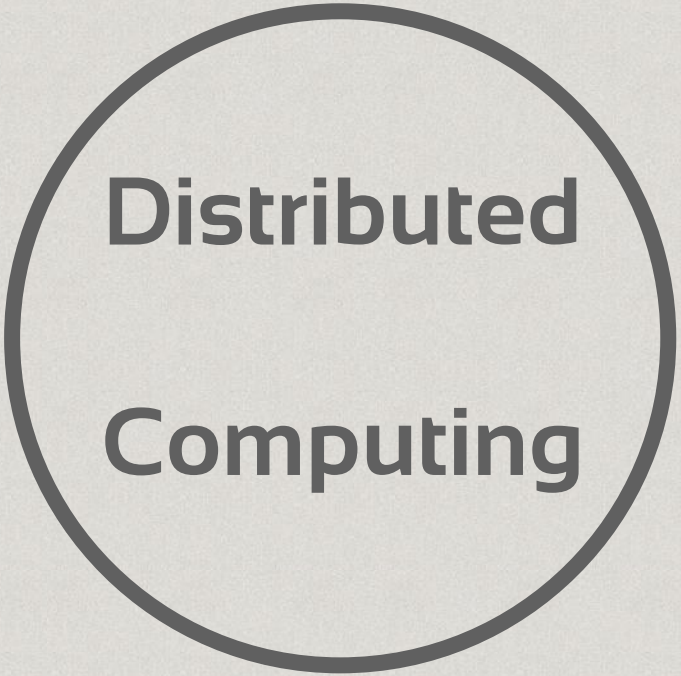


Serialization & Synchronization

Mahmoud Badry, mma18@fayoum.edu.eg

AGENDA

1. Serialization
2. Synchronization
3. Sleeping Barber problem
4. End
   Questions

Mahmoud Badry, mma18@fayoum.edu.eg

..

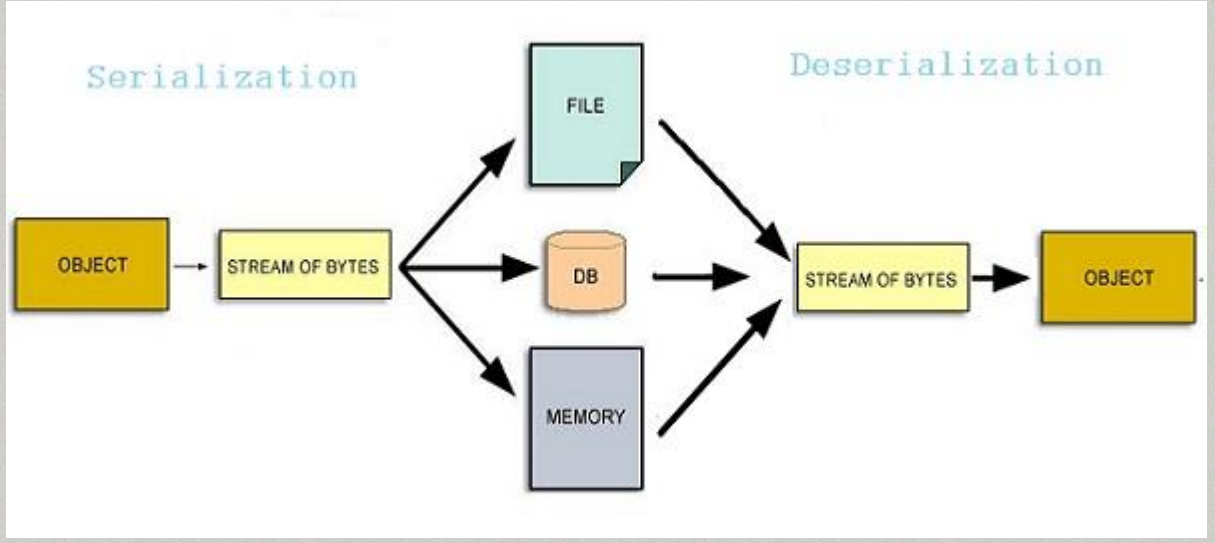
Mahmoud Badry, mma18@fayoum.edu.eg

Definition

- In the context of data storage, **serialization** is the process of **translating** data structures or object state into a **format** that can be stored (for example, in a **file** or **memory** buffer, or transmitted across a **network** connection link) and **reconstructed** later in the same or another computer environment.
- This process of **serializing** an object is also called **marshalling** an object. The opposite operation, extracting a data structure from a series of bytes, is **deserialization** (which is also called **unmarshalling**).

Mahmoud Badry, mma18@fayoum.edu.eg

Serialization explianed



Many programming languages supports Serialization like JAVA, c#, c/c++ and PHP

Mahmoud Badry, mma18@fayoum.edu.eg

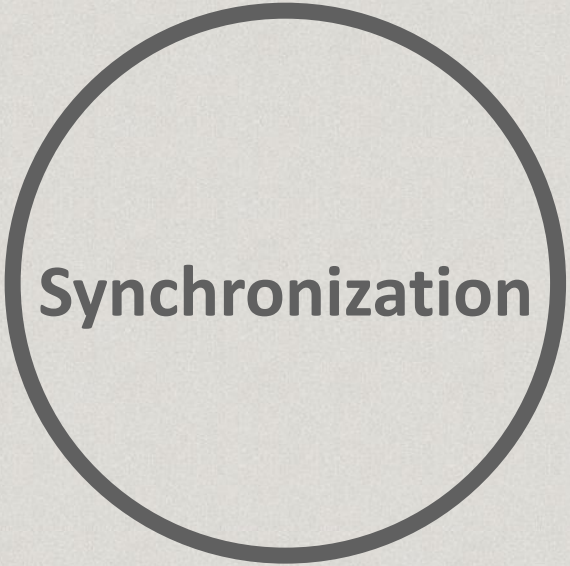
Java - Serialization

- Serialization in Java is JVM independent, meaning an object can be serialized on one platform and deserialized on an entirely different platform.
- Classes ObjectInputStream and ObjectOutputStream are high-level streams that contain the methods for serializing and deserializing an object.

Mahmoud Badry, mma18@fayoum.edu.eg

Writing and reading object from file

Mahmoud Badry, mma18@fayoum.edu.eg



..

Mahmoud Badry, mma18@fayoum.edu.eg

Synchronization

- **Synchronization** refers to one of two distinct but related concepts:
  - Synchronization of processes.
  - Synchronization of data.
- **Process synchronization** refers to the idea that **multiple processes** are to join up or **handshake** at a certain point, in order to **reach** an **agreement** or **commit** to a certain sequence of **action**.
- **Data Synchronization** refers to the idea of keeping **multiple copies** of a dataset in coherence with one another, or to maintain **data integrity**.

Mahmoud Badry, mma18@fayoum.edu.eg

Java - Thread Synchronization

- When we start **two or more threads** within a program, there may be a situation when multiple threads try to access the **same resource** and finally they can produce **unforeseen** result due to **concurrency** issue.
- For example if **multiple threads** try to write within a same **file** then they may **corrupt** the **data** because one of the **threads** can **override** data .
- So there is a need to **synchronize** the action of **multiple threads** and make sure that only **one thread** can access the **resource** at a given **point** in **time**. This is implemented using a concept called **monitors**. Each **object** in Java is associated with a **monitor**, which a thread can **lock** or **unlock**. Only **one** thread at a time may hold a **lock** on a monitor.

Mahmoud Badry, mma18@fayoum.edu.eg

Java - Thread Synchronization Cont.

- Java programming language provides a very handy way of creating threads and synchronizing their task by using **synchronized** blocks. You keep **shared resources** within this block. Following is the general form of the synchronized statement:

```
synchronized(objectidentifier) {
   // Access shared variables and other shared resources
}
```


Mahmoud Badry, mma18@fayoum.edu.eg

Multithreading with/without synchronization

Mahmoud Badry, mma18@fayoum.edu.eg

Sleeping Barber problem

Mahmoud Badry, mma18@fayoum.edu.eg

QUESTIONS

Mahmoud Badry, mma18@fayoum.edu.eg

THANK YOU!

Mahmoud Badry, mma18@fayoum.edu.eg