

20/01/2021-----

Array Concept in Java

Encapsulation

Wrapper Class

Abstract keyword

Interface

Array in Java:

Array is used to store multiple values in single variable.

```
int arr[ ]= new int [5];
```

```
array int [8]
```

10	20	30	40	50	60	70	80
----	----	----	----	----	----	----	----

0 1 2 3 4 5 6 7

// Addition of Array :

Array 1: {{10,20,30},

 {23,24,25}}

Array 2: { {20,30,40},

 {12,13,14} }

Encapsulation: To achieve data hiding, Control over variables.

Setter & Getter

---- class variable or attributes as private

Encapsulation:

Hide the sensitive data.

//Getter setter

Get(): returns the variable value

Set(): sets the value.

- If private variable: We can only access it within the class.
- Control over the class variables, methods.
- Flexible

Wrapper Class:

----- to convert primitive data type to object and objects to primitive data type.

Two Features:

- Autoboxing
- Unboxing

Serialization & Synchronization: To achieve these we require wrapper class.

To work with Collections concept, wrapper class is required.

java.util.*;

8 Wrapper Classes.

Autoboxing:

Automatic conversion of primitive data types into wrapper classes.

Example: int to Integer, char to Character

Unboxing: Reverse of autoboxing.

Conversion of wrapper class into its primitive types.

You can create your own wrapper class(custom wrapper class).

Wrapper Class

Collection

Primitive Data types	Wrapper Class
byte	Byte
short	Short
int	Integer
long	Long
float	Float
char	Character
double	Double
boolean	Boolean

---Many frameworks in java Hibernate, Collection API etc...use wrapper class.

---- used to achieve serializability.

--- autoboxing

---- unboxing

-----Aggregation in Java

---- Cosmic Class

ABSTRACT:

```
abstract class_name;
```

```
abstract protected fun1();
```

----To achieve the abstraction in java (it will not give u 100%)

-----once the class is made as abstract we can not instantiate it.

----- but you can make the reference of that class

----- It can have both abstract as well as non abstract method.

--- if your class is having one abstract method then that class should be abstract class.

```
Abstract_class a = new Abstract_class(); // Wrong
```

```
Abstract_class a = new referenced_class(); // Correct Way
```

Final Keyword?

If you want to give restrict the user.

- Final variable
- Final method
- Final class

```
Final int a;
```

The value will be initialized only in the constructors.

Once a method is declared as final, we can not inherit that method/ override that method.

Once a Class is declared as final, we can not inherit that class.