**National University of Computer & Emerging Sciences, Karachi**

**Computer Science Department**

**Spring 2023, Lab Manual – 11**

|  |  |
| --- | --- |
| **Course Code: CL-217** | **Course : Object Oriented Programming Lab** |
| **Instructor(s) :** | **Abeer Gauher, Hajra Ahmed, Shafique Rehman** |

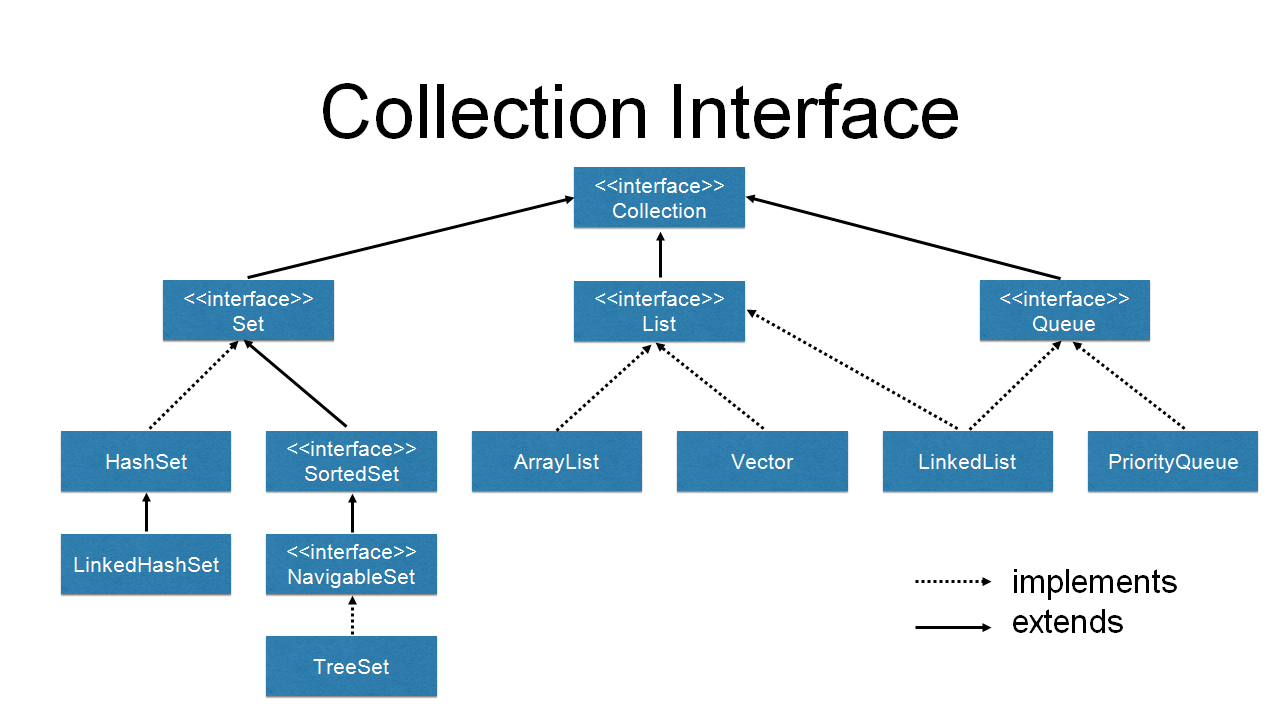
**LAB - 11**

Generics in Java

File Handling in Java

**Collection Framework:**

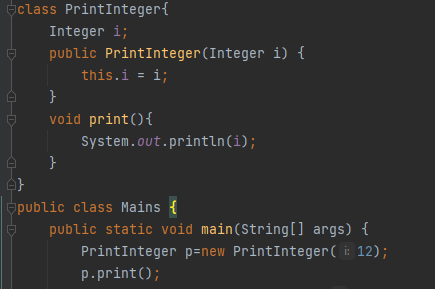
The Java collections framework provides a set of interfaces and classes to implement various data structures and algorithms like Stacks, queues, arrays, linked list etc.



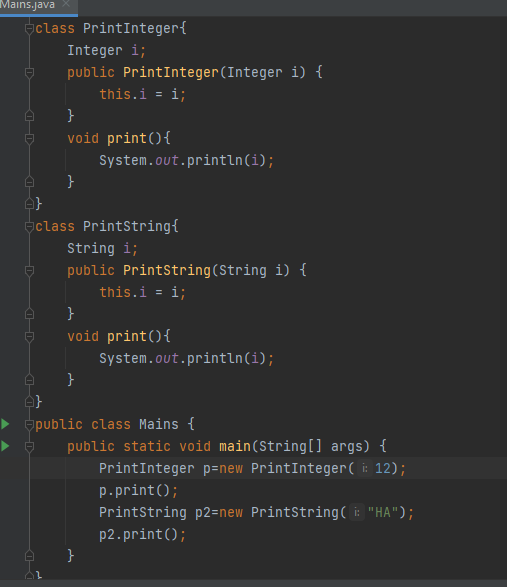
**Generics:**

**Generic Classes:**

Consider an example, let’s say I want a class that prints an integer value. So, I write a piece of code like this.

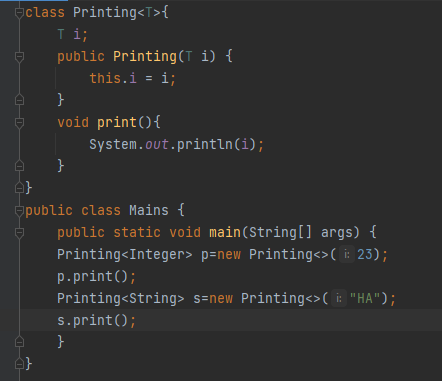


Now, for printing a String object we have to replicate the code for String class and after that for other data types as well.



So, it’s better to make use of Generics concept in Java. Here we will make a generic class and use it for printing objects of different data types.

* **The class name is followed by a type parameter section.**
* **The type parameter section of a generic class can have one or more type parameters separated by commas.**

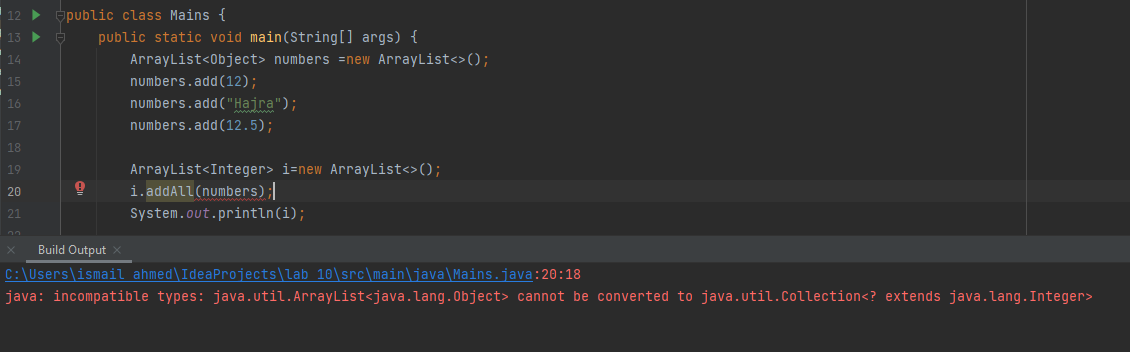


Generics are frequently used with collection frameworks in Java. For example,



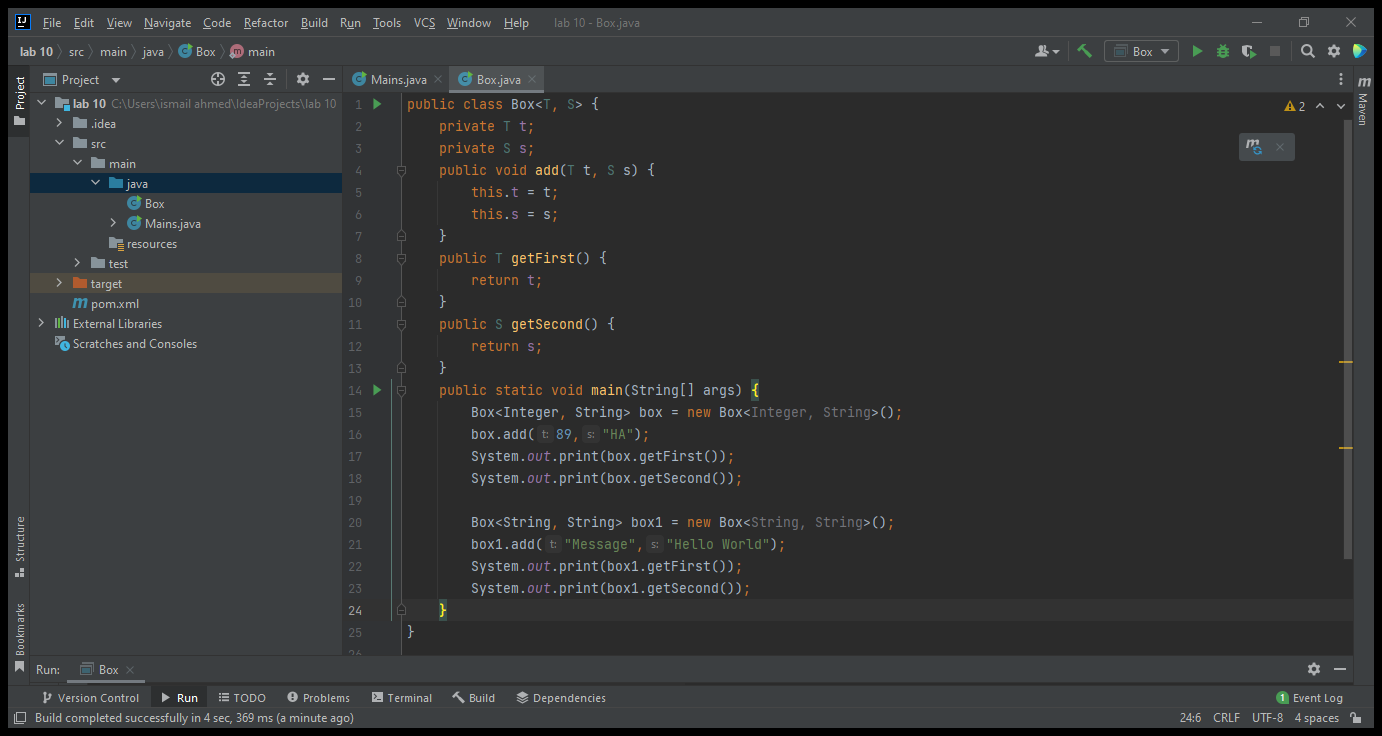
Generics avoid the type Cast Exceptions as they are type safe.

For example, you might think that lets create an array list of Object type and add any type of Item to it. But then type casting error might occur. Shown in example below;



So, generics ensure type safety as well.

A generic class may have multiple parameters.



**Type Parameters**

**The common type parameters are as follows:**

T - Type

E - Element

K - Key

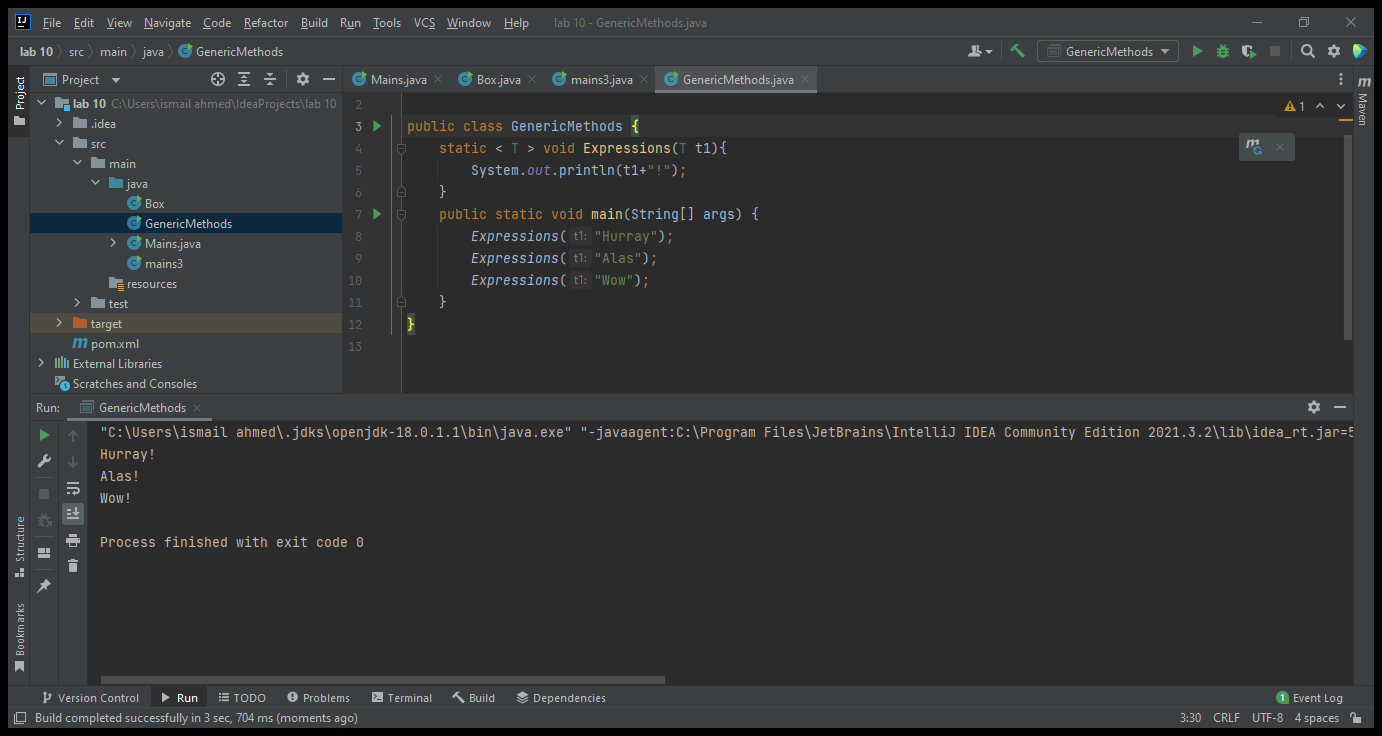
N - Number

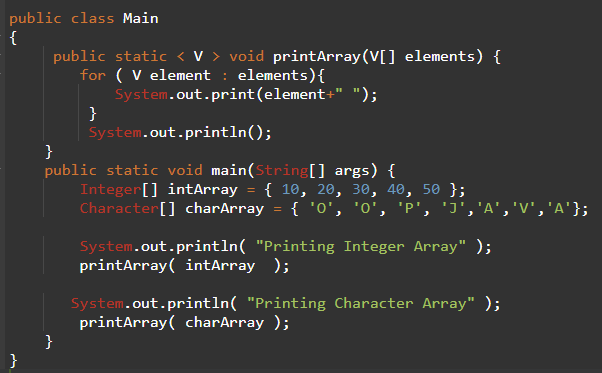
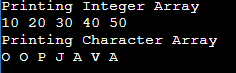
V - Value

**Generic Methods:**

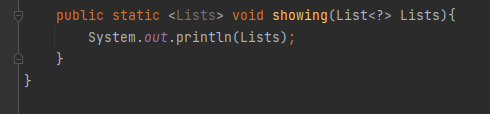
We can also write generic functions that can be called with different types of arguments based on the type of arguments passed to the generic method. The compiler handles each method.  Following are the rules to define Generic Methods :

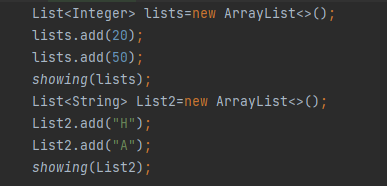
* All generic method declarations have a type parameter section delimited by angle brackets (< and >) that precedes the method's return type ( < T> in the example shown below).
* Each type parameter section contains one or more type parameters separated by commas. A type parameter, also known as a type variable, is an identifier that specifies a generic type name.
* The type parameters can be used to declare the return type and act as placeholders for the types of the arguments passed to the generic method, which are known as actual type arguments.
* Note that type parameters can represent only reference types, not primitive types





We can also use wildcards when we don’t know about the parameter being passed to the methods like shown below. Here its not known whether it’ll be an integer or string or whatever:





Don’ts for generics:

* Generics don’t work with primitive data types.
* Don’t make type parameters as static in a generic class, as compiler cannot determine which type to use for different objects.
* Don’t instantiate a type parameter with in a class.
* Overloading is not supported in generic class

**File Handling:**

File handling in java comes under IO operations. Java IO package java.io classes are specially provided for file handling in java. We will discuss some operations on filing as:

* Create file
* Delete file
* Read file
* Write file
* Change file permissions

Different streams are used for file IO operations. let’s discuss them first then we will proceed to IO operations.

**Streams in Java:**

In Java, a sequence of data is known as a stream.This concept is used to perform I/O operations on a file.There are two types of streams namely input Streams & output streams

***Input Streams:***

The Java InputStream class is the superclass of all input streams. The input stream is used to read data from numerous input devices like the keyboard, network, etc. We will use different methods of InputStream like read and close.



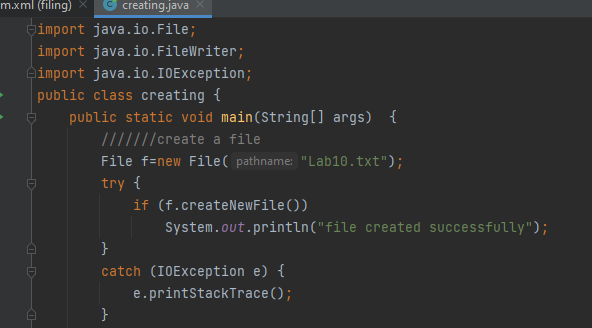
***Output Streams:***

The output stream is used to write data to numerous output devices like the monitor, file, etc. We will use different methods of OutputStream like Write and close.



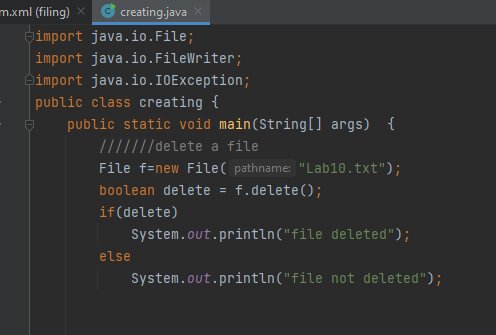
**Create a file:**

We can use File class createNewFile() method to create new file. This method returns true if file is successfully created, otherwise it returns false.



**Delete a file:**

We can use File class delete() method to remove file.

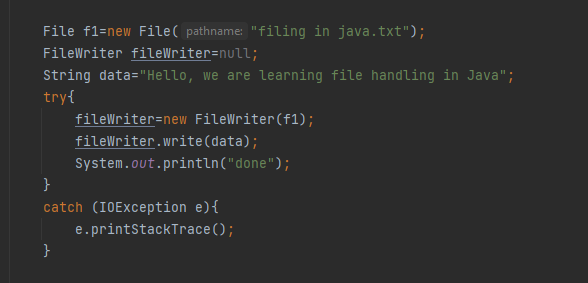


**Write a file:**

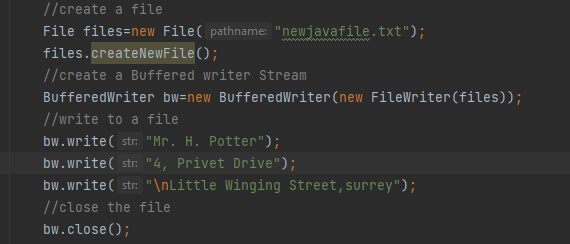
We have four options for writing to a file in Java.

* FileWriter
* BufferedWriter (uses internal buffer to write data, used when you have more write operations)
* FileOutputStream (use for writing raw stream data to be written into file)
* Files (Internally, it’s using OutputStream to write byte array into file).

We will uae FileWriter as FileWriter is the simplest way to write a file in Java. It provides overloaded write method to write int, byte array, and String to the File. You can also write part of the String or byte array using FileWriter. FileWriter writes directly into Files and should be used only when the number of writes is less.

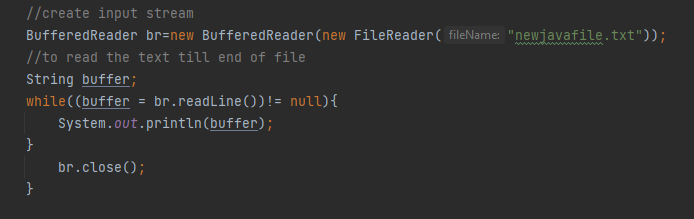


Writing using BufferedWriter



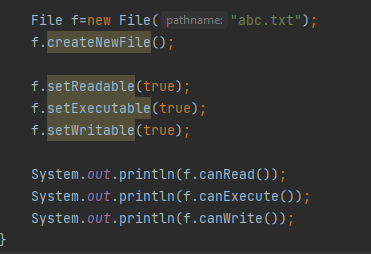
**Read a file:**

We use BufferedWriter to read file and print to console.



**Changing Permissions of a file:**

To set and check the status for the Read Write & eXecutable permissions we can do:



**Lab Tasks**

**Task 1:**

Create an arraylist of 5 elements added by user. Then write this list to the text file present on the Desktop of your PC using BufferedWriter. After writing, delete the file.

**Task 2:**

Create a text file whose name is your name. Write your full name and qualifications in separate lines and then read it to show the text present in file to console window.

**Task 3:**

Write a generic method that swaps the array elements of the specified two indices. All inputs should be taken by user.

Method signature should be:

**public static <T> void Swap(T[] array, int index1,int index2)**

**Task 4:**

Create a generic class named as Stack having an object type Array list as an attribute.

Now create following functions:

* Push: to add an element in the list
* Pop: to delete last element from the list
* Peek: to get the first element from the list
* Print: to print whole list
* isEmpty: to check whether the list is empty or filled.

Create an appropriate main method to execute your program and show all outputs

**Task 5:**

Create a text file named as “Confidential.txt”. Now, take input of designation from user. If the entered input is “Faculty”, then give the RWX permissions to user. If the designation is student then give only the read permissions.

After assigning the permissions, show the Access Rights’ status.