1.12 Generics

This section will guide you to:

* Create and run TypeScript code in your terminal
* Define generic classes and functions in TypeScript

This lab has four subsections, namely:

1.12.1 Writing a TypeScript program that defines generic functions

1.12.2 Executing the program and verifying how generic functions work

1.12.3 Writing a TypeScript program that defines generic classes

1.12.4 Executing the program and verifying how generic classes work

**Step 1.12.1:** Writing a TypeScript program that defines generic functions

Open your code editor and create a new file to type the code in

* *[Right click]* on the **File menu** of the code editor -> Select *New File*
* Enter the below code resolving the warning and errors due to compatibility-related issues
* *[Right click]* on the **File menu** of the code editor -> Select *Save as* -> Enter the filename (with**.ts** file extension) -> Click *Save*

class Queue<T> {

data: T[] = [];

push = (item: T) => this.data.push(item);

pop = (): T => this.data.shift();

}

let n = new Queue<number>();

n.push(10);

n.push(20);

n.push(30);

console.log(n.data);

console.log(n.pop());

console.log(n.data);

let s = new Queue<string>();

s.push('a');

s.push('b');

s.push('c');

console.log(s.data);

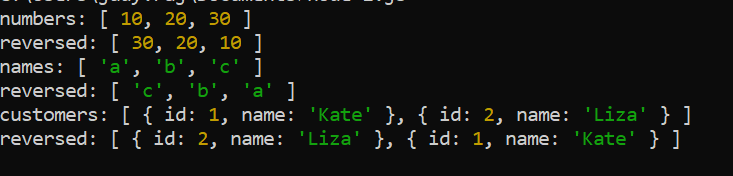
console.log(s.pop());

console.log(s.data);

**Step 1.12.2:**  Executing the program and verifying how generic functions work

If TypeScript is not already installed, you’ll need to install it by running the command, ***npm install –g typescript*** from the terminal. Before you execute the program, check for syntactical corrections. If no errors are found, follow the steps mentioned below:

* Open the terminal
* Navigate to the directory where the code is stored
* Type the command *tsc [fileName.ts]* and press Enter
* The code will compile
* Type the command node [fileName.js] and press Enter
* The code will produce the following output



**Step 1.12.3 :** Writing a TypeScript program that defines generic classes

Open your code editor and create a new file to type the code in

* *[Right click]* on the **File menu** of the code editor -> Select *New File*
* Enter the below code resolving the warning and errors due to compatibility-related issues
* *[Right click]* on the **File menu** of the code editor -> Select *Save as* -> Enter the filename (with**.ts** file extension) -> Click *Save*

class Queue<T> {

data: T[] = [];

push = (item: T) => this.data.push(item);

pop = (): T => this.data.shift();

}

let n = new Queue<number>();

n.push(10);

n.push(20);

n.push(30);

console.log(n.data);

console.log(n.pop());

console.log(n.data);

let s = new Queue<string>();

s.push('a');

s.push('b');

s.push('c');

console.log(s.data);

console.log(s.pop());

console.log(s.data);

**Step 1.12.4:**  Executing the program and verifying how generic classes work

If TypeScript is not already installed, you’ll need to install it by running the command, ***npm install –g typescript*** from the terminal. Before you execute the program, check for syntactical corrections. If no errors are found, follow the steps mentioned below:

* Open the terminal
* Navigate to the directory where the code is stored
* Type the command *tsc [fileName.ts]* and press Enter
* The code will compile
* Type the command node [fileName.js] and press Enter
* The code will produce the following output

