

Experiment – 1 b: TypeScript

Name of Student	Sneha Patra
Class Roll No	D15A_40
D.O.P.	<u>30/01/2025</u>
D.O.S.	<u>06/02/2025</u>
Sign and Grade	

Aim: To study Basic constructs in TypeScript.

Overview of Task Performed:

In this experiment, we explored the fundamental concepts of TypeScript, including class-based inheritance, composition, and interfaces. We implemented a **Student** class with a subclass **GraduateStudent**, demonstrating method overriding. A separate **LibraryAccount** class was created to showcase composition over inheritance. Additionally, an **Employee Management System** was designed using an **Employee** interface, with **Manager** and **Developer** classes implementing it to enforce structure and type safety.

GitHub Link: https://github.com/Sneha0321/WebX_Exp1

Output:

a) Student and GraduateStudent with Composition

```
PS C:\Users\Govind\Desktop\webx> tsc sm.ts
PS C:\Users\Govind\Desktop\webx> node sm.js
Student Name: Sneha, ID: 101, Grade: A
Student Name: Bob, ID: 102, Grade: A+, Thesis Topic: AI Research
Thesis Topic: AI Research
Library Account ID: 5001, Books Issued: 3
Student Name: Sneha, ID: 101, Grade: A
Library Account ID: 5001, Books Issued: 3
PS C:\Users\Govind\Desktop\webx> 
```

The output displayed the details of a Student and a GraduateStudent, including their names, IDs, grades, and thesis topics (for graduate students). The LibraryAccount class was associated with a Student using composition, and the combined details were printed. The results confirmed that method overriding in GraduateStudent worked correctly and that composition was successfully implemented.

b) Employee Management System

```
PS C:\Users\Govind\Desktop\webx> tsc em.ts
PS C:\Users\Govind\Desktop\webx> node em.js
Manager Name: Alice, ID: 101, Role: Manager, Department: HR
Developer Name: Sneha, ID: 102, Role: Developer, Programming Languages: TypeScript, JavaScript, Python
PS C:\Users\Govind\Desktop\webx>
```

The output displayed details of employees based on the Employee interface. The Manager class showed employee name, ID, role, and department, while the Developer class displayed name, ID, role, and programming languages.

Conclusion:

In this experiment, we studied the basic constructs of TypeScript, including inheritance, composition, and interfaces. We implemented class-based inheritance using Student and GraduateStudent, demonstrating method overriding. Composition was used to associate a LibraryAccount with a Student. Additionally, we designed an Employee Management System using interfaces, which ensured structure and type safety. This experiment effectively illustrated the advantages of TypeScript in terms of code readability, maintainability, and type checking.