




# Xây dựng Class TypeScript




# Phân tích về Class “Student”


- Thuộc tính: studentId, lastName, firstName, grade, isEnrollment, ...
- Constructor
- Getter/Setter



```
✓ class Student {  
    // Khai báo các thuộc tính  
    studentId: number;  
    lastName: string;  
    firstName: string;  
    grade: number;  
    isEnrollment: boolean;
```



```
// Các phương thức khởi tạo
constructor(studentId: number, lastName: string, firstName: string, grade: number, isEnrollment:
  this.studentId = studentId;
  this.lastName = lastName;
  this.firstName = firstName;
  this.grade = grade;
  this.isEnrollment = isEnrollment;
}
```



```
// Các phương thức khác của lớp Student
// Ví dụ: phương thức lấy thông tin học sinh
getStudentInfo(): string {
    return `Student ID: ${this.studentId}, Name: ${this.firstName} ${this.lastName}, Grade: ${this.grade},
    Enrollment: ${this.isEnrollment ? 'Yes' : 'No'}`;
}
}
```



```
// Sử dụng lớp Student
```

```
const student1 = new Student(1, 'Doe', 'John', 95, true);
```

```
const student2 = new Student(2, 'Smith', 'Jane', 88, false);
```

```
console.log(student1.getStudentInfo()); // In thông tin của học sinh 1
```

```
console.log(student2.getStudentInfo()); // In thông tin của học sinh 2
```

Access Modifier	Ý Nghĩa
<code>`public`</code>	Có thể truy cập từ bất kỳ đâu, không có sự giới hạn.
<code>`private`</code>	Chỉ có thể truy cập từ bên trong lớp đó.
<code>`protected`</code>	Chỉ có thể truy cập từ bên trong lớp đó và các lớp con của nó.
<code>`readonly`</code>	Giá trị chỉ đọc, không thể thay đổi sau khi được thiết lập.

# Lỗi khi biên dịch

```
class Student {  
  private studentId: number;  
  lastName: string;  
  firstName: string;  
  grade: number;  
  isEnrollment: boolean;
```

```
// Lỗi biên dịch (compile error) khi cố gắng truy cập thuộc tính private từ bên ngoài lớp  
console.log(student1.studentId);  
// Lỗi: Property 'studentId' is private and only accessible within class 'Student'.
```





## --noEmitOnError

- Khi bạn chạy tsc với tùy chọn này, TypeScript Compiler sẽ dừng lại sau khi phát hiện lỗi biên dịch đầu tiên và không tiến hành việc biên dịch các phần còn lại của mã nguồn.

```
tsc --noEmitOnError app.ts
```