



**Microsoft** Partner  
Silver Learning



# TypeScript

Вступ. Змінні та функції

# TypeScript

## Introduction



Igor Михайличенко  
Frontend Engineer

 iMykhailchenko

 Ihor Mykhailchenko



# TypeScript

## План курсу

1. Вступ. Змінні та функції
2. Класи та інтерфейси
3. Generics
4. Модулі та простір імен



# TypeScript

## Тема уроку

### Вступ. Змінні та функції



# TypeScript

## План уроку

1. TypeScript, основні можливості
2. Необхідні інструменти
3. Компіляція коду TS. Tsconfig.json
4. Змінні та типи даних
5. Функції



# TypeScript

## TypeScript

TypeScript is **JavaScript**  
**with syntax for types.**

TypeScript is a strongly typed programming language that builds on JavaScript, giving you better tooling at any scale.

**TypeScript** – це строго типізована мова програмування, яка базується на JavaScript.

<http://www.typescriptlang.org/>



# TypeScript

## Чому потрібно вивчати TypeScript?

Дослідження DOU:

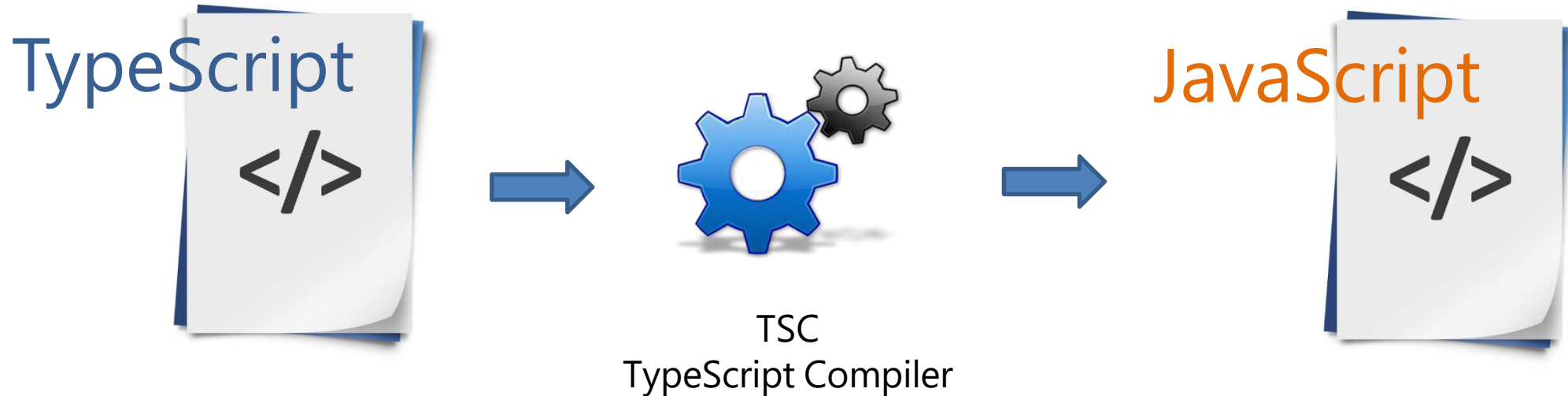
- Кількість проектів написаних на TypeScript збільшилась на 29.9% у 2023 році
- Лідер "Особистих вподобань" для розробників
- 5 місце у загальному рейтингу мов

<https://dou.ua/lenta/articles/language-rating-2023/>



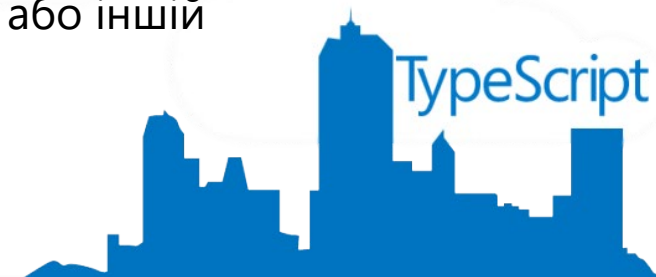
# TypeScript

## Транспілятор



**Транскомпілятор** або **транспілятор** - програма, яка приймає вихідний код, написаний на одній мові програмування і повертає еквівалентний вихідний код на тій самій або іншій мові програмування

<https://uk.wikipedia.org/wiki/>





# TypeScript

## Установка та інструменти

1. Встановити **Node.js** на ваш комп'ютер
2. Відкрити проект у редакторі коду
3. Ініціалізувати новий проект - **npm init -y**
4. Встановити npm пакет TypeScript -  
**npm i -D typescript**
5. Транспіляція ваших файлів -  
**npx tsc helloworld.ts**

### Visual Studio



Visual Studio



Visual Studio Code



Visual Studio

### And More...



Sublime Text



Emacs



Atom



WebStorm



Eclipse



Vim

<https://nodejs.org/en>



# TypeScript

## Playground

### TypeScript

```
1 class Greeter {
2     greeting: string;
3
4     constructor(message: string) {
5         this.greeting = message;
6     }
7
8     greet() {
9         return "Hello, " + this.greeting;
10    }
11 }
12
13 const greeter = new Greeter("world");
14
15 const button = document.createElement('button');
16 button.textContent = "Say Hello";
17 button.onclick = function () {
18     alert(greeter.greet());
19 }
20
21 document.body.appendChild(button);
```

### JavaScript

```
"use strict";
var Greeter = /** @class */ (function () {
    function Greeter(message) {
        this.greeting = message;
    }
    Greeter.prototype.greet = function () {
        return "Hello, " + this.greeting;
    };
    return Greeter;
})();
var greeter = new Greeter("world");
var button = document.createElement('button');
button.textContent = "Say Hello";
button.onclick = function () {
    alert(greeter.greet());
};
document.body.appendChild(button);
```

<http://www.typescriptlang.org/play/index.html>



# TypeScript

## Важливі ключові слова

Ключове слово	Опис
class	Контейнер для властивостей та методів
constructor	Ініціалізація класу
exports	Експорт із модуля
extends	Наслідування класу
implements	Реалізація інтерфейсу
import	Імпорт модуля
interface	Контракт для реалізації типом даних
private/public	Модифікатори доступу



# TypeScript

## tsconfig

**tsconfig.json** – файл, вказує на те, що поточна директорія є коренем TypeScript проєкту.

При запуску tsc без вказівки файлу для компіляції використовуватимуться налаштування з файлу tsconfig.json

```
{  
  "compilerOptions": {  
    "module": "commonjs",  
    "noImplicitAny": true,  
    "removeComments": true,  
    "sourceMap": true  
  },  
  "files": [  
    "folder-1/file.ts",  
    "folder-2/file.ts"  
  ]  
}
```

Як будуть генеруватись модулі

Неможна використовувати any

Видалити коментарі

Створити map файли для відладки

Список файлів для компіляції

<https://www.typescriptlang.org/docs/handbook/compiler-options.html>



# TypeScript

## tsconfig

```
"include": [  
  "src/**/*"  
],  
"exclude": [  
  "node_modules",  
  "**/*.spec.ts"  
]
```

Компілювати всі файли у всіх директорія в директорії src

Не компілювати вміст директорії node\_modules  
Не компілювати файли з розширенням spec.ts



# Інформаційний відеосервіс для розробників програмного забезпечення



TypeScript

# Перевірка знань

TestProvider.com



Перевірте, як ви засвоїли даний матеріал на [TestProvider.com](https://testprovider.com)

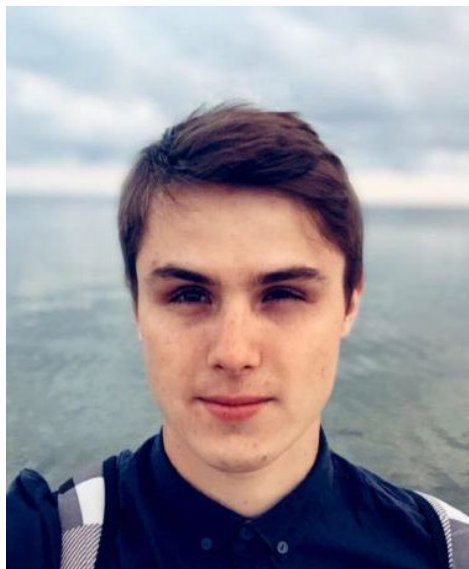
TestProvider – це online-сервіс перевірки знань з інформаційних технологій. За його допомогою ви можете оцінити свій рівень та виявити слабкі місця. Він буде корисним як у процесі вивчення технології, так і для загальної оцінки знань IT-спеціаліста.

Успішне проходження фінального тестування дозволить отримати відповідний Сертифікат.



# TypeScript

Дякую за увагу! До нової зустрічі!



Ігор Михайличенко  
Frontend Engineer

