

# Вступ. Змінні та функції

№ уроку: 1 Курс: TypeScript Fundamentals

Засоби навчання: Visual Studio, Visual Studio Code, NotePad++

## Огляд, мета та призначення уроку

Мета уроку – ознайомлення з мовою **TypeScript**, її основними можливостями. Завдання – ознайомити студентів із компілятором **TS** і конфігураційним файлом **tsconfig.json**. Вивчення типізації змінних, а також стрілкових функцій.

## Вивчивши матеріал цього заняття, учень зможе:

- Встановлювати компілятор **Typescript** на комп'ютер.
- Користуватись файлом **tsconfig.json**.
- Працювати з мовою **TS**, використовуючи різні **IDE**.
- Розуміти та використовувати типізацію змінних.
- Працювати зі стрілковими функціями.

## Зміст уроку

1. Вступ до **TypeScript**.
2. Розгляд основних можливостей **TypeScript**.
3. Компіляція коду.
4. Змінні та типи даних.
5. Функції.

## Резюме

- **TypeScript** – типізована надмножина мови **JavaScript**, яка компілюється у чистий **JavaScript**. Основні можливості:
  - підтримує стандартний **JavaScript**-код;
  - статична типізація;
  - інкапсуляція;
  - класи та інтерфейси;
  - підтримує можливості **ECMAScript 6**;
  - **intellisense** та перевірка синтаксису.
- **Компілятор** – програма, що перетворює текст, написаний алгоритмічною мовою, на програму, що складається з машинних команд. Компілятор створює завершений варіант програми машинною мовою.
- **Npm install typescript – g** – глобальна установка компілятора **TypeScript**.
- **Tsc – v** – перевірка версій компілятора.
- **Tsc app.ts** – компіляція файлу.
- **tsconfig.json** – файл, який вказує на те, що поточна директорія є коренем **TypeScript**-проєкту.
- **Number, string, Boolean, any, enum, tuples, void** – додані типи даних.
- **let** та **const** – нові типи оголошення змінних у **TypeScript**. **let** дає користувачам змогу уникнути деяких із загальних помилок, з якими стикаються в **JavaScript**. **Const** – ключове слово, яке запобігає перевизначення змінних.
- Під час визначення функції в **TypeScript** необхідно вказати тип вхідних аргументів, а також тип значення, яке повертається.

## Закріплення матеріалу

- Що означає ключове слово **let**?
- Для чого застосовується файл **tsconfig.json**?
- Яка команда дає змогу скомпілювати файл без використання налаштування конфігурації?
- Для чого використовуються **map**-файли?
- Для чого використовуються методи **call**, **apply**?

## Додаткове завдання

Використовуючи отримані знання, налаштуйте процес компіляції так, щоб під час перетворення:

- видалялися **map**-файли;
- видалялися коментарі;
- була певна директорія для розміщення файлів;
- цільова версія під час компіляції – **ES5**.

Використовуйте 2 підходи – **tsconfig.json** і командний рядок.

## Самостійна діяльність учня

### Завдання 1

Вивчити основні поняття, розглянуті під час уроку.

### Завдання 2

Використовуючи TS, створіть функцію, яка розраховує вартість морозива.

Магазин пропонує 2 розміри морозива:

- маленький стакан (10 грн);
- великий стакан (25 грн).

Морозиво доповнюється однією або декількома начинками (мінімум – однією):

шоколад (+5 грн);

карамель (+ 6 грн);

ягоди (+10 грн).

Додатково морозиво (необов'язково) можна посипати маршмелоу (+5 грн).

Вхідні параметри користувач вводить через prompt.

## Рекомендовані ресурси

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

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

<https://www.typescriptlang.org/docs/handbook/tsconfig-json.html>

<https://github.com/Microsoft/TypeScript>