

**TYPESCRIPT В СОВРЕМЕННЫХ ВЕБ-ПРИЛОЖЕНИЯХ**

Завадский Е.С.

Белорусский государственный университет информатики и радиоэлектроники,  
г. Минск, Республика Беларусь

Научный руководитель: Давыдович К.И. – магистр технических наук, ассистент кафедры ИПиЭ

**Аннотация.** В данной статье рассматривается *TypeScript* как один из основных инструментов для разработки современных интерфейсов. Автор подробно рассказывает о преимуществах *TypeScript*, особенностях его использования и сочетания с другими популярными и высокопроизводительными инструментами, которые применяются для создания веб-приложений, а также описаны возможные ошибки при использовании *TypeScript*.

**Ключевые слова:** *TypeScript*, разработка, типизация, веб-приложение, *GraphQL*

**Введение.** В современном мире веб-разработки, где динамичность, масштабируемость и надёжность являются ключевыми требованиями, язык программирования *TypeScript* становится всё более популярным среди разработчиков. Он является неотъемлемым инструментом в разработке, ускоряя и делая разработку приложения легче.

**Основная часть.** *TypeScript* является обратно совместимым с *JavaScript* и компилируется в последний. Фактически, после компиляции программу на *TypeScript* можно выполнять в любом современном браузере или использовать совместно с серверной платформой *Node.js*. Код экспериментального компилятора, транслирующего *TypeScript* в *JavaScript*, распространяется под лицензией *Apache*. Его разработка ведётся в публичном репозитории через сервис *GitHub*.

*TypeScript* отличается от *JavaScript* возможностью явного статического назначения типов, поддержкой использования полноценных классов (как в традиционных объектно-ориентированных языках), а также поддержкой подключения модулей, что призвано повысить скорость разработки, облегчить читаемость, рефакторинг и повторное использование кода, помочь осуществлять поиск ошибок на этапе разработки и компиляции, и, возможно, ускорить выполнение программ [1].

*TypeScript* можно использовать для разработки как клиентской стороны в связке с такими фреймворками как *React*, *Vue* или *Angular*, так и серверной используя *NestJS*, *Express* и другие популярные решения на сегодняшний день.

Можно подумать, что *TypeScript* является самостоятельным решением в разработке веб-приложений, однако это не так. Современный браузер «понимает» только *HTML*, *CSS*, *JavaScript*, поэтому в конечном итоге весь код, написанный на *TypeScript*, будет преобразован в *JavaScript*, также при использовании таких фреймворков как *React*, *Vue*, *Angular*, будет происходить парсинг в понятную для браузера структуру. Таким преобразованием занимаются так называемые сборщики, например, *webpack*, *gulp*, *babel* и другие.

*TypeScript* в современных веб-приложениях. *TypeScript* используется для строгой типизации кода, что необходимо для избежания ошибок при написании веб-приложений, также типизация помогает быстрее и легче масштабировать проект, улучшает читаемость кода и ускоряет разработку. Однако его использование требует определённой степени квалифицированности разработчика, так как данный инструмент может принести вред приложению при неумелом использовании, например, очень частой ошибкой является использование типа «*any*», его часто используют, когда затрудняются определить какой тип всё же нужен, однако таких ситуаций не должно происходить и их наличие сообщает о том, что в коде или архитектуре приложения существует ряд ошибок. Также написание повторяющихся типов, а не переиспользование их может привести к более долгой сборке

кода. Вместо написания подобных типов можно рассмотреть вариант написания одного общего типа и расширения его за счёт оператора «&» или ключевого слова «*extends*».

Так как типизация играет огромную роль в разработке, во многом из-за этого, один из самых популярных решений для взаимодействия с сервером и обмена данными с клиентской стороной является *GraphQL*.

*GraphQL* – язык запросов данных и язык манипулирования данными с открытым исходным кодом для построения веб ориентированных программных интерфейсов [2]. Одним из его главных преимуществ является обратная совместимость типов. Это значит, что типы написанные на *backend* стороне доступны к использованию на *frontend* стороне. Типы с серверной стороны приходят в формате *json* или *graphql* и преобразуются в файл формата *ts* с помощью пакета *@graphql-codegen/cli* или другого подобного инструмента. Таким образом осуществляется строгая типизация не только в коде *frontend* части, а и во взаимодействии с серверной стороной, что помогает избежать ошибок и упрощает разработку.

**Заключение.** В данный момент надёжность, удобство и оптимизированность являются основными критериями при выборе языка программирования, *TypeScript* считает в себе всё это и из-за чего становится всё более популярным среди разработчиков. По моему мнению *TypeScript* является обязательным к использованию инструментом при разработке веб-приложений.

### Список литературы

1. Документация TypeScript [Электронный ресурс] / Документация TypeScript – Дублин, 2012-2024. – Режим доступа: <https://www.typescriptlang.org> TypeScript: JavaScript With Syntax For Types – Дата доступа: 15.02.2024
2. Документация GraphQL [Электронный ресурс] / Документация GraphQL – Таллин, 2024. – Режим доступа: <https://graphql.org> GraphQL A query language for your API – Дата доступа: 15.02.2024

UDC 004.432

## TYPESCRIPT IN MODERN WEB APPLICATIONS

Zavadskij E.S.

*Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus*

*Davidovich K.I. – master of technical sciences, assistant of the department of EPE*

**Annotation.** This article discusses TypeScript as one of the main tools for developing modern interfaces. The author describes in detail the advantages of TypeScript, the peculiarities of its use and combination with other popular and high-performance tools that are used to create web applications, and also describes possible mistakes when using TypeScript.

**Keywords:** TypeScript, development, typing, web-applications, GraphQL